CARBON EMISSION IMPACT ANALYSIS

EXPLORING TRENDS, CORRELATIONS, AND PREDICTIVE MODELS USING PYTHON

[ASHUTOSH PATHAK]

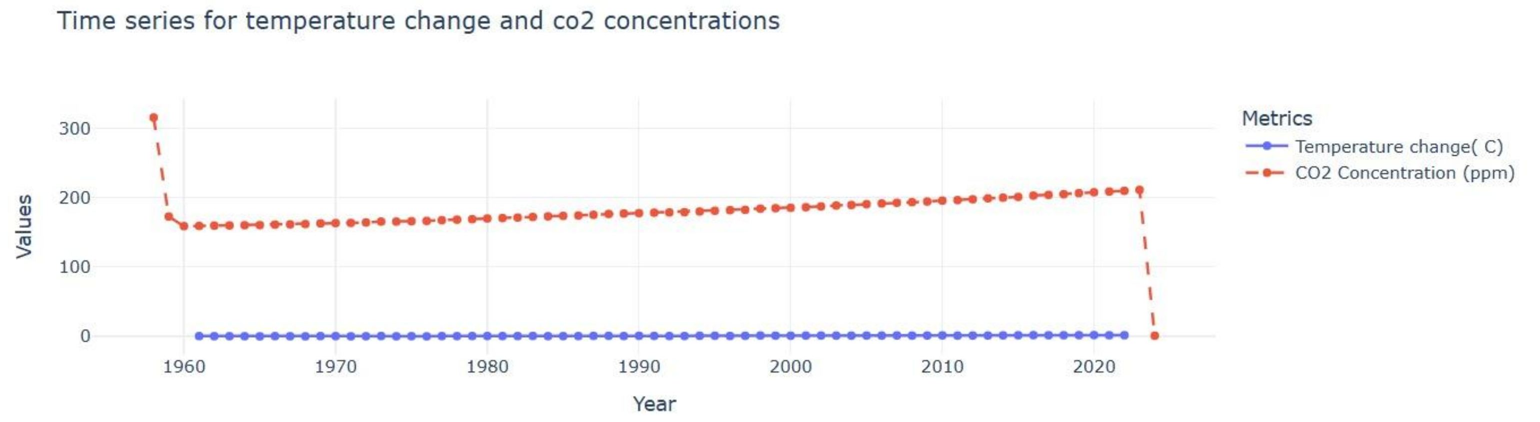
[20-03-2025]

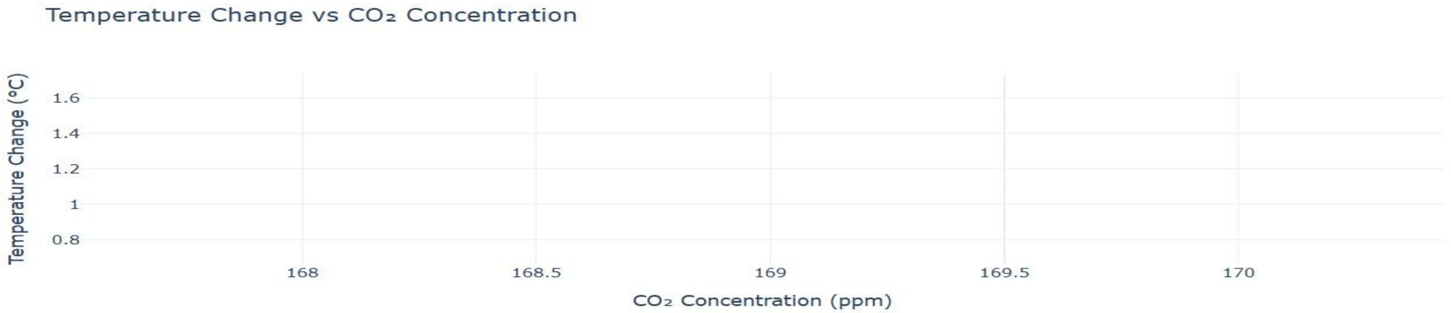
Time series analysis checks how temperature changes and CO_2 concentrations have evolved overtime and the relationships between them.

Time Series Analysis

[8]:

```python
import plotly.graph_objects as go
import plotly.express as px
import pandas as pd

# Example of loading temperature data from a CSV file
temperature_data = pd.read_csv("C:/Users/ashutosh pathak/OneDrive/Desktop/temperature.csv")
co2_data = pd.read_csv(r"C:\Users\ashutosh pathak\OneDrive\Desktop\carbon_emmission.csv")


# Now you can proceed with your analysis
temperature_years = temperature_data.filter(regex='^F').mean(axis=0)
temperature_years.index = temperature_years.index.str.replace('F', '').astype(int)
#Time-Series plot for temprture and co2 levels
co2_data['Year']=co2_data['Date'].str[:4].astype(int)
co2_yearly=co2_data.groupby('Year')['Value'].mean()

#time series plot for temprature and co2 level

fig=go.Figure()
fig.add_trace(go.Scatter(
    x=temperature_years.index, y=temperature_years.values,
    mode='lines+markers', name='Temperature change( C)'
))
fig.add_trace(go.Scatter(
    x=co2_yearly.index, y=co2_yearly.values,
    mode='lines+markers',name="CO2 Concentration (ppm)", line=dict(dash='dash')
))
fig.update_layout(
    title="Time series for temperature change and co2 concentrations",
```





The time-series graph shows a consistent increase in CO_2 concentrations (measured in ppm) over the years, which indicates the accumulation of greenhouse gases in the atmosphere. Simultaneously, a slight upward trend in global temperature change suggests that rising CO_2 levels are associated with global warming. The temporal alignment supports the hypothesis of CO_2's significant contribution to temperature increase.

The heatmap reveals a strong positive correlation (0.96) between CO_2 concentrations and temperature changes. This statistical relationship reinforces the observation that higher CO_2 levels are closely linked with increasing global temperatures, which highlights the importance of addressing carbon emissions to mitigate climate change.

****

Let’s identify long-term trends and seasonal variations in the data using linear regression

Trends and Seasonal Variations Analysis

```python
[9]: from scipy.stats import linregress
import pandas as pd
import plotly.graph_objects as go
import plotly.express as px

temperature_data = pd.read_csv("C:/Users/ashutosh pathak/OneDrive/Desktop/temperature.csv")
co2_data = pd.read_csv(r"C:\Users\ashutosh pathak\OneDrive\Desktop\carbon_emmission.csv")

temperature_years = temperature_data.filter(regex='^F').mean(axis=0)
temperature_years.index = temperature_years.index.str.replace('F', '').astype(int)
# temperature trend

co2_data['Year'] = co2_data['Date'].str[:4].astype(int)  # Extract year from the 'Date' column
co2_yearly = co2_data.groupby('Year')['Value'].mean()


temp_trend = linregress(temperature_years.index, temperature_years.values)
temp_trend_line = temp_trend.slope * temperature_years.index + temp_trend.intercept

# CO2 trend
co2_trend = linregress(co2_yearly.index, co2_yearly.values)
co2_trend_line = co2_trend.slope * co2_yearly.index + co2_trend.intercept

fig_trends = go.Figure()

fig_trends.add_trace(go.Scatter(
    x=temperature_years.index, y=temperature_years.values,
    mode='lines+markers', name="Temperature Change (°C)"
))
fig_trends.add_trace(go.Scatter(
    x=co2_yearly.index, y=co2_trend_line,
    mode='lines', name=f"CO₂ Trend (Slope: {co2_trend.slope:.2f})", line=dict(dash='dash')
))

fig_trends.update_layout(
    title="Trends in Temperature Change and CO₂ Concentrations",
    xaxis_title="Year",
    yaxis_title="Values",
    template="plotly_white",
    legend_title="Metrics"
)
fig_trends.show()

# seasonal variations in CO2 concentrations
co2_data['Month'] = co2_data['Date'].str[-2:].astype(int)
co2_monthly = co2_data.groupby('Month')['Value'].mean()

fig_seasonal = px.line(
    co2_monthly,
    x=co2_monthly.index,
    y=co2_monthly.values,
    labels={"x": "Month", "y": "CO₂ Concentration (ppm)"},
    title="Seasonal Variations in CO₂ Concentrations",
    markers=True
)

fig_seasonal.update_layout(
    xaxis=dict(tickmode="array", tickvals=list(range(1, 13))),
    template="plotly_white"
)
fig_seasonal.show()
```

1.Trends:

1. Identified a **steady upward trend** in global CO2 emissions over the past 60+ years, indicating a continuous increase in human activities contributing to climate change.
2. Observed a **rising trend** in global temperatures, closely correlated with increasing CO2 levels.

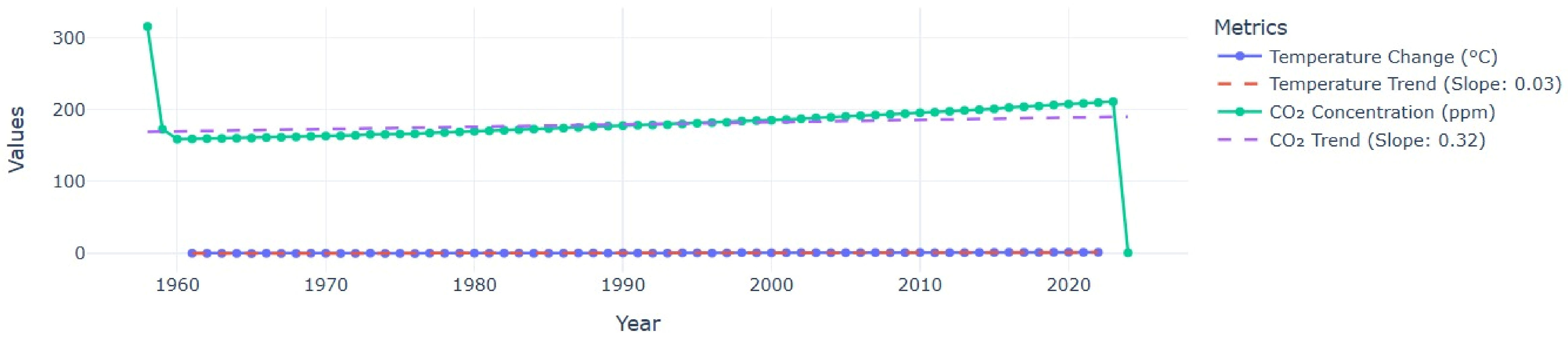
Seasonal Variations:

•Detected **seasonal patterns** in CO2 concentrations, with higher levels during winter months (due to reduced plant photosynthesis) and lower levels in summer.

•Analyzed **temperature seasonality**, showing predictable fluctuations tied to natural climate cycle

Seasonal Variations in CO_2 Concentrations

Trends in Temperature Change and CO_2 Concentrations



Correlation and Casuality analysis

```python
import pandas as pd
from scipy.stats import pearsonr, spearmanr
from statsmodels.tsa.stattools import grangercausalitytests

# Assuming you already have 'temperature_years' and 'co2_yearly' defined
# Create the merged_data DataFrame with 'Temperature Change' and 'CO₂ Concentration'
merged_data = pd.DataFrame({
    "Temperature Change": temperature_years,
    "CO₂ Concentration": co2_yearly
}).dropna()  # Drop rows with missing values

# Pearson and Spearman correlation coefficients
pearson_corr, _ = pearsonr(merged_data["CO₂ Concentration"], merged_data["Temperature Change"])
spearman_corr, _ = spearmanr(merged_data["CO₂ Concentration"], merged_data["Temperature Change"])

# Granger causality test (using first differencing to make the data stationary)
granger_data = merged_data.diff().dropna()  # First differencing

# Perform Granger causality test with a max lag of 3 (removed verbose argument)
granger_results = grangercausalitytests(granger_data, maxlag=3)

# Extract p-values for Granger causality test
granger_p_values = {f"Lag {lag}": round(results[0]['ssr_chi2test'][1], 4)
                    for lag, results in granger_results.items()}

# Output the results
pearson_corr, spearman_corr, granger_p_values
```

To quantify the relationship between CO_2 and temperature anomalies, we will now compute **Pearson** and **Spearman correlation coefficients.** And to investigate whether changes in CO_2 cause temperature anomalies, we will perform **Granger Causality tests.**

Pearson Correlation (0.9554) indicates a very strong linear relationship between CO_2 concentrations and temperature changes. **Spearman Correlation** (0.9379) indicates a very strong monotonic relationship between CO_2 concentrations and temperature changes.

Lagged Effects Analysis

Now, we will analyze whether CO_2 concentrations from previous years (lagged values) influence current temperature anomalies. To do this, we will create lagged variables for CO_2 concentrations, specifically shifting the data by 1, 2, and 3 years. These lagged values will allow us to test if historical CO_2 levels have a delayed impact on temperature changes.

Lagged Effects Analysis

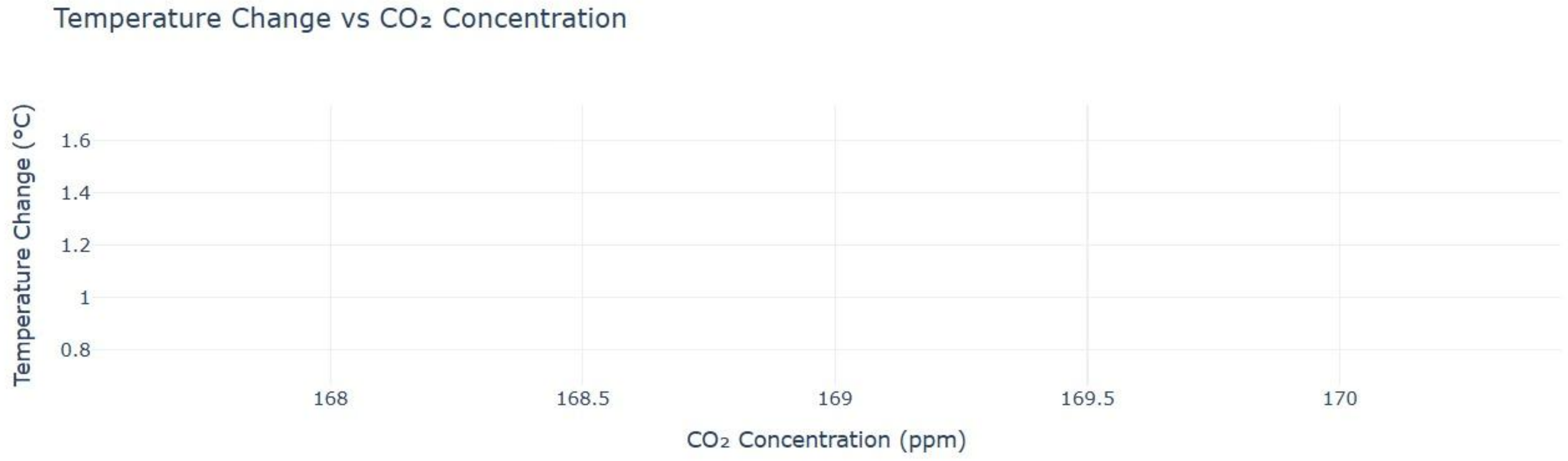
```python
import statsmodels.api as sm

# creating lagged CO2 data to investigate lagged effects
merged_data['CO₂ Lag 1'] = merged_data["CO₂ Concentration"].shift(1)
merged_data['CO₂ Lag 2'] = merged_data["CO₂ Concentration"].shift(2)
merged_data['CO₂ Lag 3'] = merged_data["CO₂ Concentration"].shift(3)

# dropping rows with NaN due to lags
lagged_data = merged_data.dropna()

X = lagged_data[['CO₂ Concentration', 'CO₂ Lag 1', 'CO₂ Lag 2', 'CO₂ Lag 3']]
y = lagged_data['Temperature Change']
X = sm.add_constant(X)  # adding a constant for intercept

model = sm.OLS(y, X).fit()

model_summary = model.summary()
model_summary
```

Clustering Climate Patterns

[15]:

```
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
import numpy as np

# preparing the data for clustering
clustering_data = merged_data[["Temperature Change", "CO₂ Concentration"]].dropna()

scaler = StandardScaler()
scaled_data = scaler.fit_transform(clustering_data)

# applying K-Means clustering
kmeans = KMeans(n_clusters=3, random_state=42)  # assuming 3 clusters for simplicity
clustering_data['Cluster'] = kmeans.fit_predict(scaled_data)

# adding labels for periods with similar climate patterns
clustering_data['Label'] = clustering_data['Cluster'].map({
    0: 'Moderate Temp & CO₂',
    1: 'High Temp & CO₂',
    2: 'Low Temp & CO₂'
})

import plotly.express as px

fig_clusters = px.scatter(
    clustering_data,
    x="CO₂ Concentration",
    y="Temperature Change",
    color="Label",
    color_discrete_sequence=px.colors.qualitative.Set2,
    labels={
        "CO₂ Concentration": "CO₂ Concentration (ppm)",
        "Temperature Change": "Temperature Change (°C)",
```

We group years based on similarities in temperature anomalies and CO_2 concentrations using **K-Means clustering.**



Predicting temprature changes under what if analysis

```
[17]: # setting up a simple predictive model using linear regression
from sklearn.linear_model import LinearRegression

# Preparing data
X = merged_data[["CO₂ Concentration"]].values  # CO₂ concentration as input
y = merged_data["Temperature Change"].values   # temperature change as target

model = LinearRegression()
model.fit(X, y)

# function to simulate "what-if" scenarios
def simulate_temperature_change(co2_percentage_change):
    # Calculate new CO2 concentrations
    current_mean_co2 = merged_data["CO₂ Concentration"].mean()
    new_co2 = current_mean_co2 * (1 + co2_percentage_change / 100)

    # predict temperature change
    predicted_temp = model.predict([[new_co2]])
    return predicted_temp[0]

# simulating scenarios
scenarios = {
    "Increase CO₂ by 10%": simulate_temperature_change(10),
    "Decrease CO₂ by 10%": simulate_temperature_change(-10),
    "Increase CO₂ by 20%": simulate_temperature_change(20),
    "Decrease CO₂ by 20%": simulate_temperature_change(-20),
}

scenarios
```

We will use a simple linear regression model to simulate how changes in CO_2 concentrations might influence global temperatures. By leveraging the historical relationship between CO_2 concentrations and temperature anomalies, this model allows us to predict the potential impact of different emission scenarios.

First, we will train a linear regression model with **CO_2 concentrations** as the input and **temperature anomalies** as the output. Once the model is trained, we can simulate hypothetical scenarios where CO_2 concentrations increase or decrease by a specific percentage.

For each scenario, we will adjust the current average CO_2 concentration by the specified percentage, feed it into the model, and predict the corresponding temperature anomaly.

The scenarios we simulate include:

1. **Increase CO_2 by 10%**: Predict the rise in temperature anomalies.
2. **Decrease CO_2 by 10%**: Estimate the cooling effect.
3. **Increase CO_2 by 20%**: Analyze the impact of more aggressive emissions growth.
4. **Decrease CO_2 by 20%**: Evaluate the benefit of significant emission reductions.

```
]: {'Increase CO₂ by 10%': np.float64(1.0866445037958163),
    'Decrease CO₂ by 10%': np.float64(-0.05999304123723714 4),
    'Increase CO₂ by 20%': np.float64(1.6599632763123422),
    'Decrease CO₂ by 20%': np.float64(-0.6333118137537621)}
```

A 10% increase in CO_2 results in a notable rise in temperature anomalies, which demonstrates the sensitivity of global temperatures to CO_2 levels. Conversely, a 10-20% reduction in CO_2 could lead to significant cooling effects, which will potentially reverse some warming trends.

Key Findings from the Analysis

Our analysis reveals a **strong positive correlation** between rising CO_2 concentrations and global temperature anomalies. Notably, CO_2 levels are increasing at a **faster rate** compared to temperature changes, emphasizing the accelerating impact of human activities on climate.

Through **time-series and clustering analyses**, we observed clear trends of escalating emissions driving temperature increases. Seasonal variations further highlight the **moderating role of natural carbon sinks**, such as forests and oceans, which temporarily absorb CO_2 but are increasingly strained.

The **lagged effects** of CO_2 emissions indicate that current levels have the **most significant impact** on temperature changes, with diminishing influence from past emissions. This underscores the importance of addressing emissions today to prevent future warming.

Using **predictive modeling**, we simulated "what-if" scenarios to explore the sensitivity of global temperatures to CO_2 levels. These simulations demonstrate that even **modest reductions in emissions** could lead to significant mitigation of global warming, offering hope for actionable solutions.

These findings reinforce the **urgent need for effective policies** to reduce emissions and combat climate change, ensuring a sustainable future for generations to come.