Analyzing the Impact of Climate Change on Global Temperature and Sea Level Rise

Introduction:

The aim of this project is to analyze the impact of climate change on global temperature and sea level rise over time. This includes analyzing ancient statistics to understand traits and patterns in temperature and sea level measurements. To complete this study, I collected data from two different source.

Motivation

After complete this study I hope I will get the answer of my research questions which are How has global temperature changed over the past century, and what are the underlying trends? What is the relationship between global temperature rise and sea level rise? Which regions are most affected by sea level rise, and what are the implications for coastal communities? Can we predict future trends in global temperature and sea level rise using historical data?

Data Source

I choose two datasets for a comprehensive and focused study on the effects of temperature changes on sea level rise. These datasets provide the necessary information to analysis and understand the intricate dynamics between climate change and its real-world impact's and making them ideal for my project.

1. Datasource1: Global Temperature
   Metadata URL:https://data.giss.nasa.gov/gistemp/
   Data URL: https://data.giss.nasa.gov/gistemp/tabledata_v4/GLB.Ts+dSST.csv
   Data Type: CSV
   Description: This dataset provide the annual global temperature from 1980 to 2024
   Data Quality: The Accuracy Consistency and Relevancy of Data are good also Timeliness is well check
.

2. Data Source 2: Global Average Absolute Sea Level Change
   Metadata URL: https://tidesandcurrents.noaa.gov/
   Data URL: https://www.epa.gov/system/files/other-files/2022-07/sea-level_fig-1.csv
   Data Type: CSV
   Description: This dataset provide two different source data in one dataset which are CSIRO - Adjusted sea level (inches) , NOAA - Adjusted sea level (inches)
   Data Quality: The Accuracy ,Consistency and Relevancy of Data are good also Timeliness is well check
.

Methodology This project flows ETL(Extract, Transform and Load) pipeline structure

Functions

1. **dataFetch(url, skip)**:
   - Reads CSV data from the specified URL.
   - Skips rows as defined in the **skip** list.
   - Returns the data as a pandas DataFrame.
2. **columnSelector(data, selection)**:
   - Selects specific columns from the DataFrame based on the **selection** list.
   - Returns the filtered DataFrame.
3. **dropNull(data)**:
   - Drops rows with any null values.
   - Returns the cleaned DataFrame.

```python
import pandas as pd
import sqlite3

def dataFetch(url, skip):
    data = pd.read_csv(url, skiprows=skip)
    return pd.DataFrame(data)

def columnSelector(data, selection):
    return data[selection]

def dropNull(data):
    return data.dropna(how = "any", axis = 0)

def yearSelector(data, f, t):
    return data[(data["Year"] >= f) & (data["Year"] <= t)]

def dataMerge(t1, t2, key, t1S, t2S):
    return pd.merge(t1, t2, how ='left', on = key, suffixes=(t1S, t2S))

def tempAverage(data):
    subset = data[["Jan",   "Feb",   "Mar",   "Apr",   "May",   "Jun",   "Jul",   "Aug",   "Sep",   "Oct",   "Nov",   "Dec"]]
    subset = subset.apply(pd.to_numeric)
    data["Average"] = subset.mean(axis = 1, skipna = True)
    return data

def save(data, path, file, table):
    try:
        conn = sqlite3.connect(path+file)
        data.to_sql(table, conn, if_exists='replace', index=False)
        conn.close()

    except:
        conn = sqlite3.connect("../data/"+file)
        data.to_sql(table, conn, if_exists='replace', index=False)
        conn.close()

```

4. **yearSelector(data, f, t)**:
    - Filters the DataFrame to include only rows where the "Year" column value is between **f** and **t**.
    - Returns the filtered DataFrame.
5. **dataMerge(t1, t2, key, t1S, t2S)**:
    - Merges two DataFrames (**t1** and **t2**) on the specified **key**.
    - Adds suffixes to the columns to differentiate between the two tables.
    - Returns the merged DataFrame.
6. **tempAverage(data)**:
    - Computes the average temperature across all months for each year.
    - Adds this average as a new column "Average" in the DataFrame.
    - Returns the DataFrame with the new column.
7. **save(data, path, file, table)**:
    - Saves the DataFrame to a SQLite database.
    - Tries to save to the specified path; if it fails, saves to a default "../data/" path.

Main Function

1. **Load and Process Temperature Data:**
    - Fetches data from the NASA URL.
    - Selects relevant columns.
    - Drops rows with null values.
    - Filters rows for the years 1995-2010.
    - Computes the average monthly temperature**.**
2. **Load and Process Sea Level Data:**
    - Fetches data from the EPA URL.
    - Selects relevant columns.
    - Drops rows with null values.
    - Filters rows for the years 1995-2010.

```
def main():
    url1 = "https://data.giss.nasa.gov/gistemp/tabledata_v4/GLB.Ts+dSST.csv"
    data1 = dataFetch(url1, [0])
    data1Header = [ "Year",   "Jan",   "Feb",   "Mar",   "Apr",   "May",   "Jun",   "Jul",   "Aug",   "Sep",   "Oct",   "Nov",   "Dec",]
    data1 = columnSelector(data1, data1Header)
    data1 = dropNull(data1)
    data1 = yearSelector(data1, 1995, 2010)
    data1 = tempAverage(data1)

    url2 = "https://www.epa.gov/system/files/other-files/2022-07/sea-level_fig-1.csv"
    data2 = dataFetch(url2, [0,1,2,3,4,5])
    data2 = columnSelector(data2, ["Year", "CSIRO - Adjusted sea level (inches)", "NOAA - Adjusted sea level (inches)"])
    data2 = dropNull(data2)
    data2 = yearSelector(data2, 1995, 2010)

    finalData = dropNull(dataMerge(data1, data2, ["Year"], "T", "S"))

    targetedPath = "./data/"
    fileName = "Data.sqlite"
    dbName = "TempSeaLevel"
    save(finalData, targetedPath, fileName, dbName)

if __name__ == "__main__":
    main()
```

3. **Merge Data and Save**:
    - Merges the processed temperature and sea level data on the "Year" column.
    - Drops any rows with null values in the merged data.
    - Saves the final data to a SQLite database.

The final dataset includes average annual temperatures and adjusted sea level measurements from 1995 to 2010. The data is stored in an SQLite database named Data.sqlite with the table name TempSeaLevel.

	Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Average	CSIRO - Adjusted sea level (inches)	NOAA - Adjusted sea level (inches)
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1995	0.52	0.79	0.47	0.47	.27	.43	.45	.45	.33	.47	.44	.26	0.445833333333333	6.622047237	6.57674495
2	1996	0.24	0.46	0.33	0.33	.27	.29	.37	.48	.25	.24	.38	.37	0.334166666666667	6.78346456	6.723710173
3	1997	0.33	0.41	0.52	0.34	.35	.54	.34	.41	.52	.61	.64	.59	0.466666666666667	7.059055111	6.779514525
4	1998	0.58	0.88	0.63	0.64	.68	.77	.66	.65	.42	.41	.43	.55	0.608333333333333	6.669291332	6.770265291
5	1999	0.48	0.64	0.32	0.32	.26	.36	.38	.31	.38	.34	.37	.40	0.38	7.003937001	6.840497137
6	2000	0.25	0.56	0.55	0.56	.36	.40	.39	.42	.38	.26	.30	.28	0.3925	7.055118103	6.919122465
7	2001	0.45	0.44	0.55	0.5	.58	.52	.59	.49	.52	.50	.72	.56	0.535	7.271653536	7.116325549
8	2002	0.77	0.79	0.88	0.58	.64	.53	.62	.53	.63	.54	.59	.44	0.628333333333333	7.366141725	7.258223788
9	2003	0.75	0.58	0.6	0.55	.61	.48	.58	.65	.62	.73	.53	.75	0.619166666666667	7.728346449	7.410511627
10	2004	0.58	0.73	0.63	0.61	.37	.44	.26	.46	.50	.61	.73	.51	0.535833333333333	7.712598417	7.465565214
11	2005	0.75	0.6	0.74	0.67	.63	.65	.61	.60	.71	.75	.73	.68	0.676666666666667	7.716535425	7.566645782
12	2006	0.56	0.73	0.63	0.47	.48	.66	.54	.70	.65	.70	.74	.79	0.6375	7.885826764	7.678404507
13	2007	1.02	0.7	0.72	0.76	.69	.61	.59	.60	.60	.59	.59	.50	0.664166666666667	7.960629913	7.674412818
14	2008	0.3	0.38	0.75	0.54	.49	.49	.60	.46	.61	.67	.69	.54	0.543333333333333	8.303149598	7.828383728
15	2009	0.65	0.53	0.53	0.61	.65	.64	.73	.69	.71	.66	.79	.67	0.655	8.531496054	7.973212968
16	2010	0.75	0.83	0.92	0.85	.76	.68	.63	.67	.64	.71	.81	.46	0.725833333333333	8.83464566	8.03087412

Conclusion: The summery of this report is successfully processed and combined historical temperature and sea level data to analyze the impact of climate change. The cleaned and merged data can be used for further analysis, such as exploring trends, correlations, and predictive modeling to understand the relationship between global temperature rise and sea level rise