Iteration 3 OSAS (Steps 1 – 8)

1. Situation Understanding

There is overwhelming scientific consensus that the Earth's climate is changing, primarily due to human activities that increase the concentration of greenhouse gases (GHGs) in the atmosphere. GHGs such as carbon dioxide (CO2), methane (CH4), and nitrous oxide (N2O) trap heat in the Earth's atmosphere, leading to a warming effect known as the greenhouse effect. Global temperature records show that the Earth's average surface temperature has been rising steadily over the past century, with the most pronounced increases observed in recent decades. The burning of fossil fuels for energy production, transportation, and industrial processes is the largest source of anthropogenic CO2 emissions, contributing significantly to global warming. Deforestation and land-use changes release stored carbon into the atmosphere and reduce the Earth's capacity to absorb CO2 through photosynthesis, exacerbating the greenhouse effect. Industrial activities such as cement production, chemical manufacturing, and waste management release CO2 and other GHGs, further intensifying global warming. Agriculture, including livestock production and rice cultivation, generates methane and nitrous oxide emissions, contributing to climate change. Rising temperatures lead to shifts in weather patterns, including more frequent and severe heatwaves, droughts, floods, and storms, affecting ecosystems, agriculture, and human health. Warming temperatures cause the melting of polar ice caps, glaciers, and ice sheets, leading to sea-level rise, coastal erosion, and loss of habitat for polar wildlife. Increased atmospheric CO2 levels lead to ocean acidification, threatening marine ecosystems, coral reefs, and seafood resources. Climate change exacerbates habitat loss, fragmentation, and degradation, leading to shifts in species distributions, extinction risks, and loss of ecosystem services.

Addressing the problem of climate change and rising global temperatures requires concerted efforts at the global, national, and local levels, with a focus on mitigation, adaptation, resilience-building, and sustainable development. To address this situation some of the objectives would include:

- Analyze global temperature records to understand long-term trends, variability, and recent changes in average surface temperatures.
- Evaluate climate model projections and scenarios to forecast future changes in temperature, precipitation, sea-level rise, and extreme weather events.

1.2 Assessment of the Situation

While it is relatively easy to understand and assess climate change, it remains crucial to take action and tackle the situation effectively.

Resources

The situation will be assessed by compiling data collected from various global and national climate organization repositories into a single dataset. Machine learning models like Linear Regression will play a crucial role in handling predictions and providing insights into the situation. Basic programming languages like Python will be used as a framework to implement the model and perform exploratory data analysis with visual representations as well.

Requirements

The basic requirement would include data with some historical trends and relevant facts that could be helpful in extrapolation and impact temperature changes over the years.

Potential risks and contingencies

Predicting future global temperatures and climate change based on historical data comes with several risks and uncertainties:

1. Data Limitations: Using historical data up to 2013 may not capture all relevant factors influencing climate change. There may be new variables or trends emerging after 2013 that could significantly impact temperature patterns.

2. Model Uncertainty: Forecasting future temperatures relies on the assumption that historical patterns will continue. However, climate is a complex system influenced by numerous factors, and future trends may not follow past patterns.

3. External Factors: External events such as volcanic eruptions, changes in solar radiation, or policy decisions affecting greenhouse gas emissions can have unforeseen impacts on global temperatures.

4. Feedback Loops: Climate change can trigger feedback loops that amplify warming, such as melting ice leading to reduced reflectivity (albedo effect) or thawing permafrost releasing additional greenhouse gases. Predicting the timing and magnitude of these feedback loops is challenging.

Contingency Measures:

1. Scenario Analysis: Consider multiple scenarios with different assumptions about future trends and external factors. This can help assess the range of possible outcomes and their associated uncertainties.

2. Sensitivity Analysis: Test the sensitivity of your model to changes in key variables or assumptions. Identify which factors have the greatest impact on your predictions and evaluate their uncertainty.

3. Expert Consultation: Consult with climate scientists or domain experts to gain insights into emerging trends, potential feedback mechanisms, and areas of uncertainty in climate models.

4. Communication of Uncertainty: Clearly communicate the limitations and uncertainties associated with your predictions. Provide stakeholders with a nuanced understanding of the potential risks and the range of possible outcomes.

5. Adaptation Strategies: Develop adaptive strategies that can be implemented in response to different climate change scenarios. This may include infrastructure improvements, land-use planning, and policy interventions to mitigate the impacts of climate change.

Cost/Benefit Analysis

Certain data might be sensitive or necessitate special access, potentially incurring high costs depending on its quality. While implementing machine learning models using Python frameworks on a small scale could be practically free, scaling up could prove computationally and financially expensive, especially if sincere accurate predictions are needed.

1.3 Data Mining Goals

1. Pattern Recognition: Identifying recurring patterns or associations within the data that can provide valuable insights or predictive capabilities.

2. Anomaly Detection: Detecting unusual or unexpected patterns in the data that may indicate errors, anomalies, or other significant events.

3. Prediction and Forecasting: Developing models to predict future trends or outcomes based on historical data, such as forecasting future temperatures or climate change patterns.

1.4 Project Plan

Project Objective: To analyze historical climate data and identify trends and patterns to understand the impact of climate change.

Duration: 1 week (Friday, April 26, 2024 - Friday, 3 May, 2024)

Phase 1: Project Setup and Preparation

Day 1

- Activities:
  - Gather relevant information, determine situation for the project, understand the situation.
  - Define project scope, objectives, and success criteria.
  - Revise lab exercises and course material. Refine Python skills
  - Develop initial project timeline and milestones.

Phase 2: Data Collection and Exploration

Day 2

  - Identify data sources for climate data (e.g., NOAA, NASA, Berkley Earth).
  - Collect historical climate data for analysis.
  - Explore the collected data to understand its structure, format, and quality.
  - Clean the data by addressing missing values, outliers, and inconsistencies.
  - Perform preliminary data visualization to identify patterns and trends.

Phase 3: Data Preparation and Feature Engineering

Day 3

-Select relevant features for analysis based on project objectives.

- Engineer new features if necessary (e.g., calculate monthly or yearly averages).
- Integrate data from different sources if applicable.
- Normalize or standardize the data to ensure consistency.
- Split the data into training, validation, and test sets.

Phase 4: Model Development and Evaluation

Day 4

Activities:

- Select appropriate data mining algorithms for analysis (e.g., linear regression, time series analysis).
- Develop predictive models using the training data.
- Evaluate model performance using validation data and adjust parameters as needed.
- Fine-tune models to improve accuracy and generalization.
- Validate models using the test data set and assess their performance.

Phase 5: Results Interpretation and Documentation

Day 5

-Activities:

- Interpret the results of the data mining analysis.
- Document key findings, insights, and recommendations.
- Prepare visualizations and reports to communicate the results to stakeholders.
- Review and finalize the project documentation.

Phase 6: Project Review and Iteration

Day 6

Activities:
- Conduct a project review meeting to assess the outcomes and lessons learned.
- Identify areas for improvement and potential future iterations.
- Update project documentation and knowledge repository.
- Plan for future iterations or follow-up projects based on the findings.

2.1 Collecting Data

Initial raw data was collected from non – profit Berkely Earth with reports on reports on how land and ocean temperature vary by location (https://berkeleyearth.org/data/). The high-resolution Berkeley Earth data set has been used to construct new country, regional and local summaries. However, for this Iteration only temperatures with respect to the countries have been chosen.

The Berkeley Earth Surface Temperature Study combines 1.6 billion temperature reports from 16 pre-existing archives. It is nicely packaged and allows for slicing into interesting subsets (for example by country). They publish the source data and the code for the transformations they applied. They also use methods that allow weather observations from shorter time series to be included, meaning fewer observations need to be thrown away.

However, the Final Dataset used is a compilation of three Datasets of global temperatures over sea and land and temperatures by regions and cities. **(https://berkeleyearth.org/temperature-region/southern-hemisphere)**

(https://berkeleyearth.org/temperature-region/northern-hemisphere)

2.2 Data Description

The dataset includes four main features:

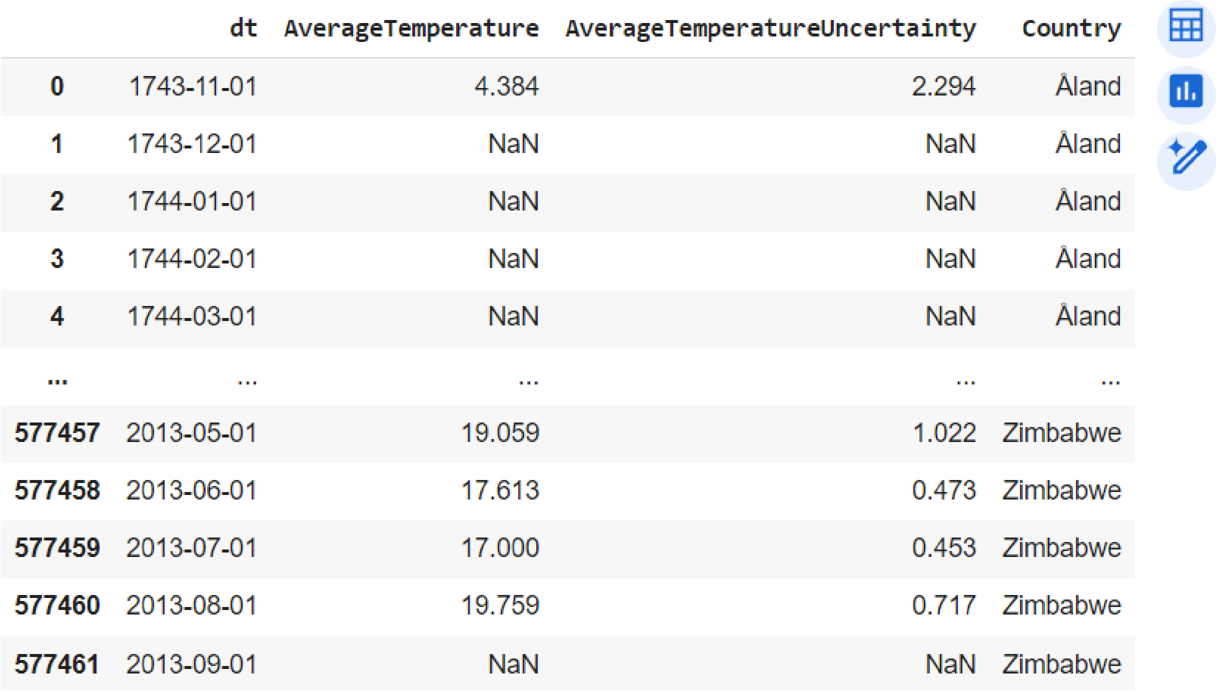
Date: Range of date from 1743 to 2013 in YY-MM-DD format

LandAverageTemperature: global average land temperature in Celsius (Continuous Data)

LandAverageTemperatureUncertainty: the 95% confidence interval around the average (Continuous Data)

Country: List of over 243 countries from Åland to Zimbabwe (Categorical Data)

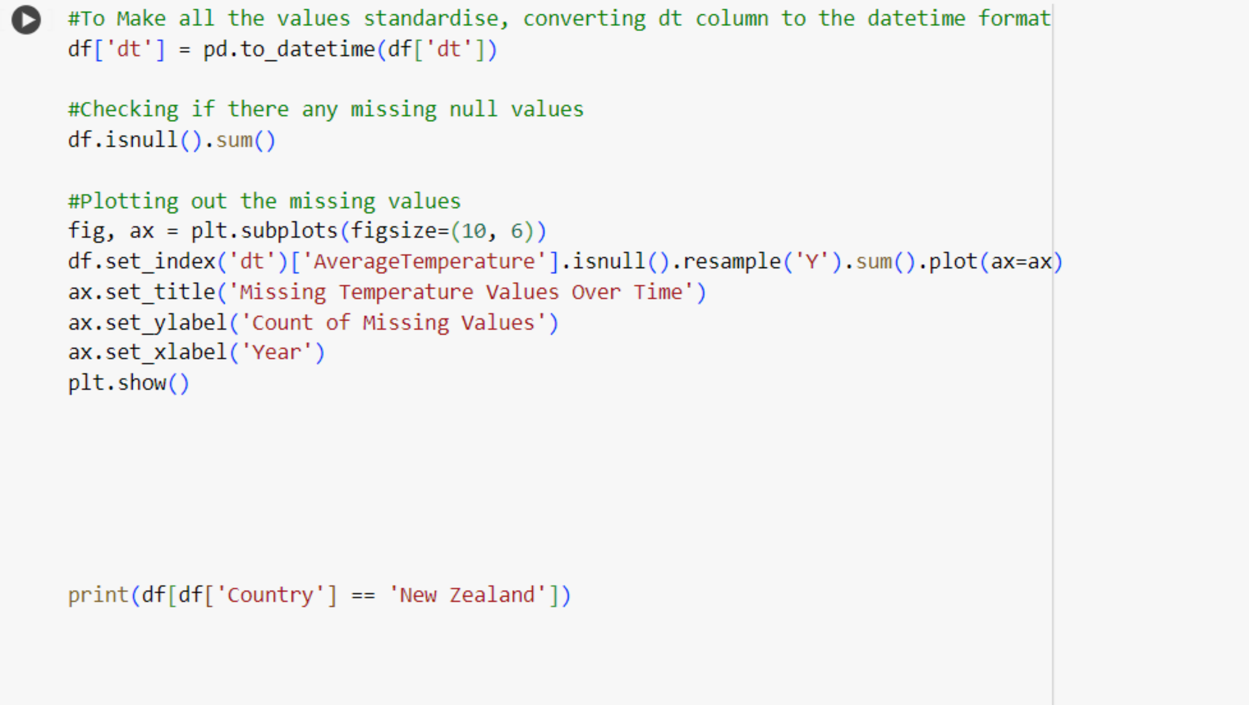
2.3 Data Exploration:



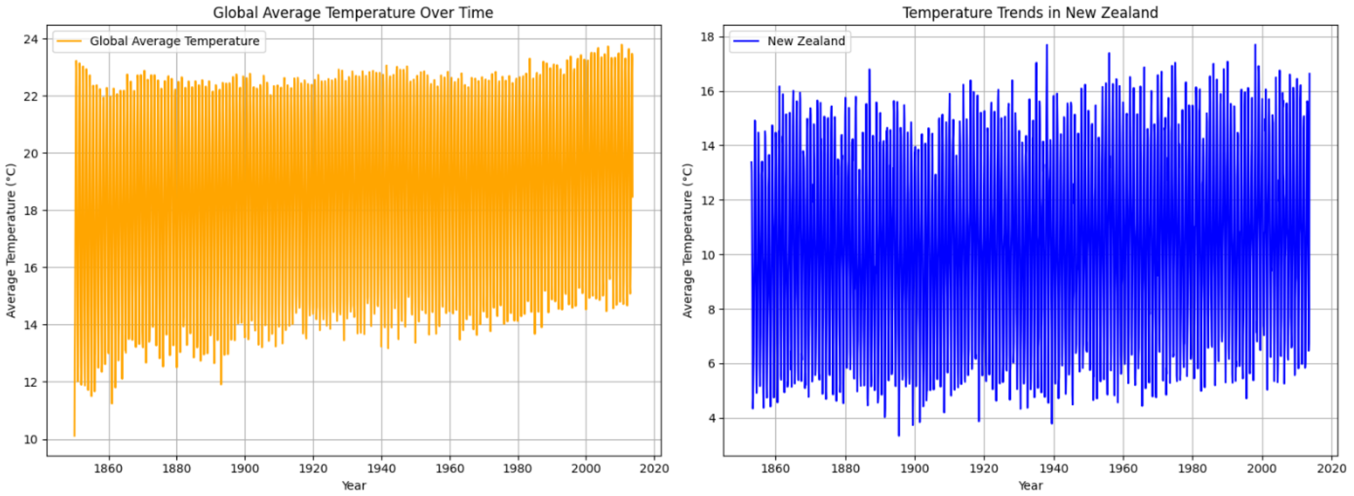
	dt	AverageTemperature	AverageTemperatureUncertainty	Country
0	1743-11-01	4.384	2.294	Åland
1	1743-12-01	NaN	NaN	Åland
2	1744-01-01	NaN	NaN	Åland
3	1744-02-01	NaN	NaN	Åland
4	1744-03-01	NaN	NaN	Åland
...	...	...	...	...
577457	2013-05-01	19.059	1.022	Zimbabwe
577458	2013-06-01	17.613	0.473	Zimbabwe
577459	2013-07-01	17.000	0.453	Zimbabwe
577460	2013-08-01	19.759	0.717	Zimbabwe
577461	2013-09-01	NaN	NaN	Zimbabwe

The Data is visually very large to show in a tabular form. Thus, the first and the last 5 rows have been presented. The ‘dt’ column representing the date feature ranges from 1743 -11-01 to 2013-09-01. The data has been marked for the 1st day of every month of every year after 1743 up to the year 2013.

In order to gain deeper insights into temperature trends over time, a line plot was created to compare global average temperature trends with those of New Zealand. This involved grouping the variable "global_avg_temp" by the "AverageTemperature" category, and a new variable "nz_avg_temperature" was similarly grouped by the category "New Zealand." To facilitate side-by-side comparison of the results, a tight layout was generated.

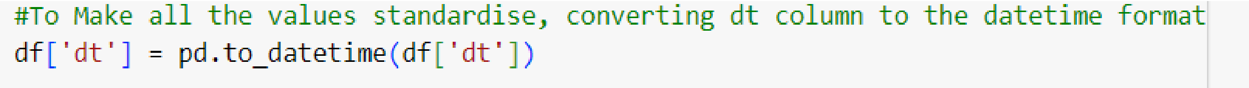


```
#To Make all the values standardise, converting dt column to the datetime format
df['dt'] = pd.to_datetime(df['dt'])

#Checking if there any missing null values
df.isnull().sum()

#Plotting out the missing values
fig, ax = plt.subplots(figsize=(10, 6))
df.set_index('dt')['AverageTemperature'].isnull().resample('Y').sum().plot(ax=ax)
ax.set_title('Missing Temperature Values Over Time')
ax.set_ylabel('Count of Missing Values')
ax.set_xlabel('Year')
plt.show()





print(df[df['Country'] == 'New Zealand'])
```



3. Data Preparation

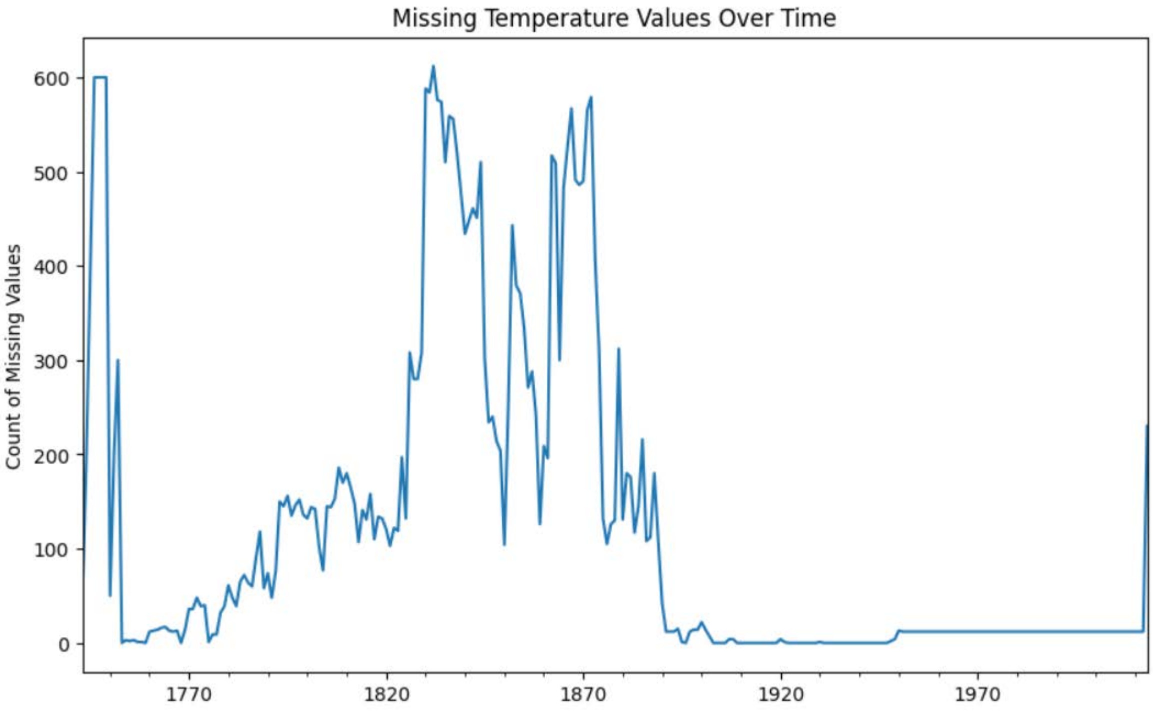
The values in dt column were converted to the generalized datetime format in order make all the values standardize.



```
#To Make all the values standardise, converting dt column to the datetime format
df['dt'] = pd.to_datetime(df['dt'])
```

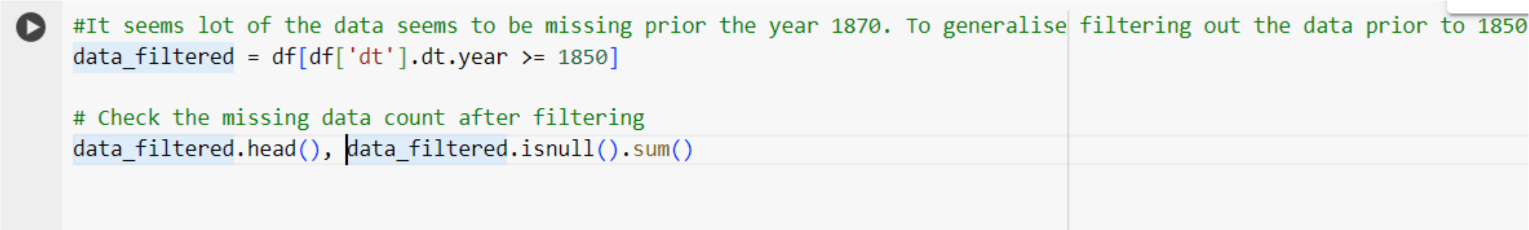
There might be some null values in early years for the columns or features AverageTemperature and AverageTemperatureUncertainity. Inconsistencies in the values vary up to the year 1850.

The missing values were sought out through a visual representation using a subplot by plotting count of missing values against the years.

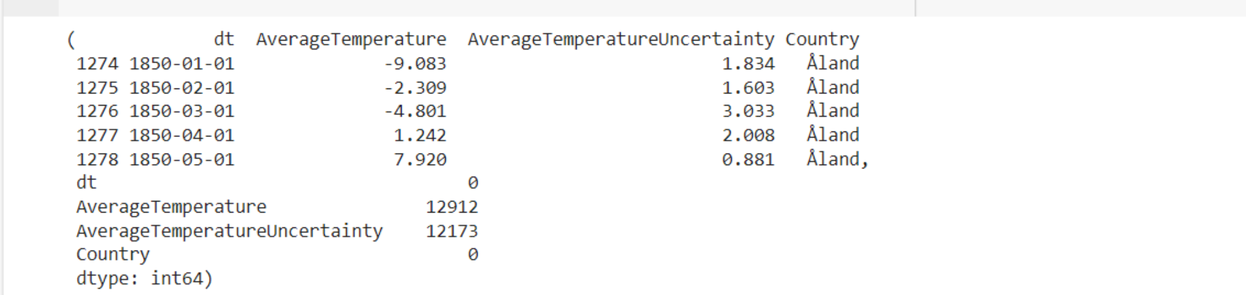


Upon examining the Figure, it becomes evident that the highest level of inconsistencies occurs within the time span from 1770 to 1870, leading to the drawn conclusion. Additionally, missing values have been identified and documented using numerical representation, with a total of 32,651 missing values for the Average Temperature feature and 31,912 for the Average Temperature Uncertainty feature.

Based on the observations from the graph, a conclusion was reached indicating that a significant number of missing values persist prior to the year 1850. Consequently, the data was filtered accordingly, resulting in a reduction of nearly 7,000 value counts. Subsequently, a reassessment was conducted to verify the presence of any remaining inconsistencies.



```
#It seems lot of the data seems to be missing prior the year 1870. To generalise filtering out the data prior to 1850
data_filtered = df[df['dt'].dt.year >= 1850]

# Check the missing data count after filtering
data_filtered.head(), data_filtered.isnull().sum()
```

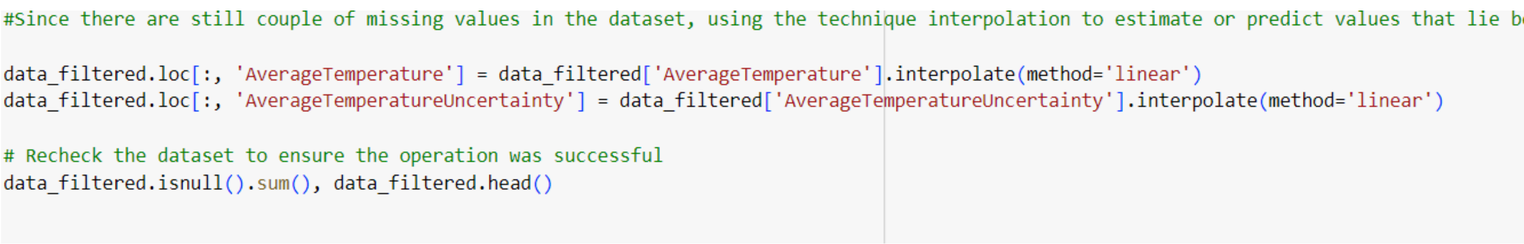


```
(             dt  AverageTemperature  AverageTemperatureUncertainty Country
 1274 1850-01-01              -9.083                          1.834   Åland
 1275 1850-02-01              -2.309                          1.603   Åland
 1276 1850-03-01              -4.801                          3.033   Åland
 1277 1850-04-01               1.242                          2.008   Åland
 1278 1850-05-01               7.920                          0.881   Åland,
 dt                                   0
 AverageTemperature               12912
 AverageTemperatureUncertainty    12173
 Country                              0
 dtype: int64)
```

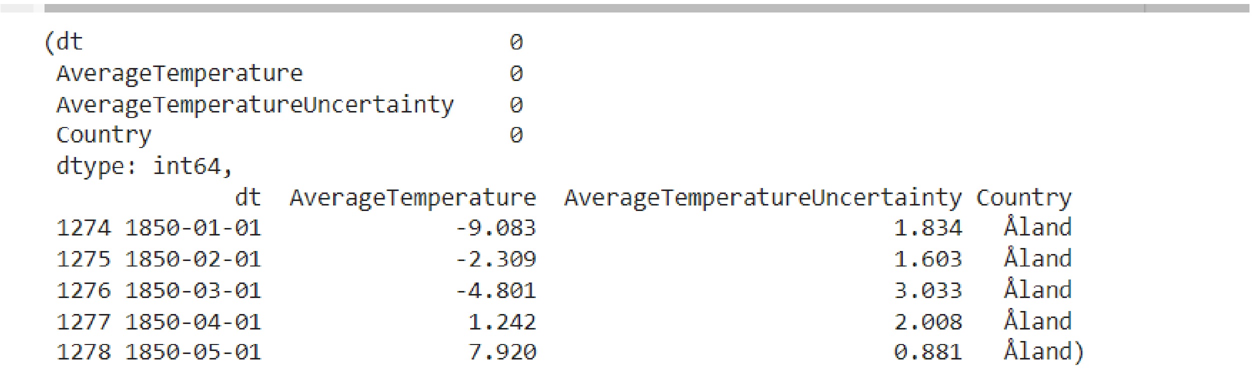
The count of missing values was reduced by almost 2500.

4. Data Transformation

To further mitigate the process Interpolation method was used which is the process of filling in missing values in a DataFrame or Series by estimating values based on the surrounding data points. The inbuilt library of python called pandas provides several methods for interpolation, which allows to choose the most appropriate method based on the data and requirements. For this dataset linear method was used.



```
#Since there are still couple of missing values in the dataset, using the technique interpolation to estimate or predict values that lie b

data_filtered.loc[:, 'AverageTemperature'] = data_filtered['AverageTemperature'].interpolate(method='linear')
data_filtered.loc[:, 'AverageTemperatureUncertainty'] = data_filtered['AverageTemperatureUncertainty'].interpolate(method='linear')

# Recheck the dataset to ensure the operation was successful
data_filtered.isnull().sum(), data_filtered.head()
```



```
(dt                               0
 AverageTemperature               0
 AverageTemperatureUncertainty    0
 Country                          0
 dtype: int64,
              dt  AverageTemperature  AverageTemperatureUncertainty Country
 1274 1850-01-01              -9.083                          1.834   Åland
 1275 1850-02-01              -2.309                          1.603   Åland
 1276 1850-03-01              -4.801                          3.033   Åland
 1277 1850-04-01               1.242                          2.008   Åland
 1278 1850-05-01               7.920                          0.881   Åland)
```

5. Data Mining Method:

The feature AverageTemperature has continuous values that refer to numerical variables that can take on an infinite number of possible values within a certain range. Regression is a supervised learning technique used to predict continuous numerical values based on input features. It aims to model the relationship between independent variables and a dependent variable. Thus, for this particular dataset Linear Regression has been chosen to identify recurring patterns or associations within the data over the years that can provide valuable insights or predictive capabilities.

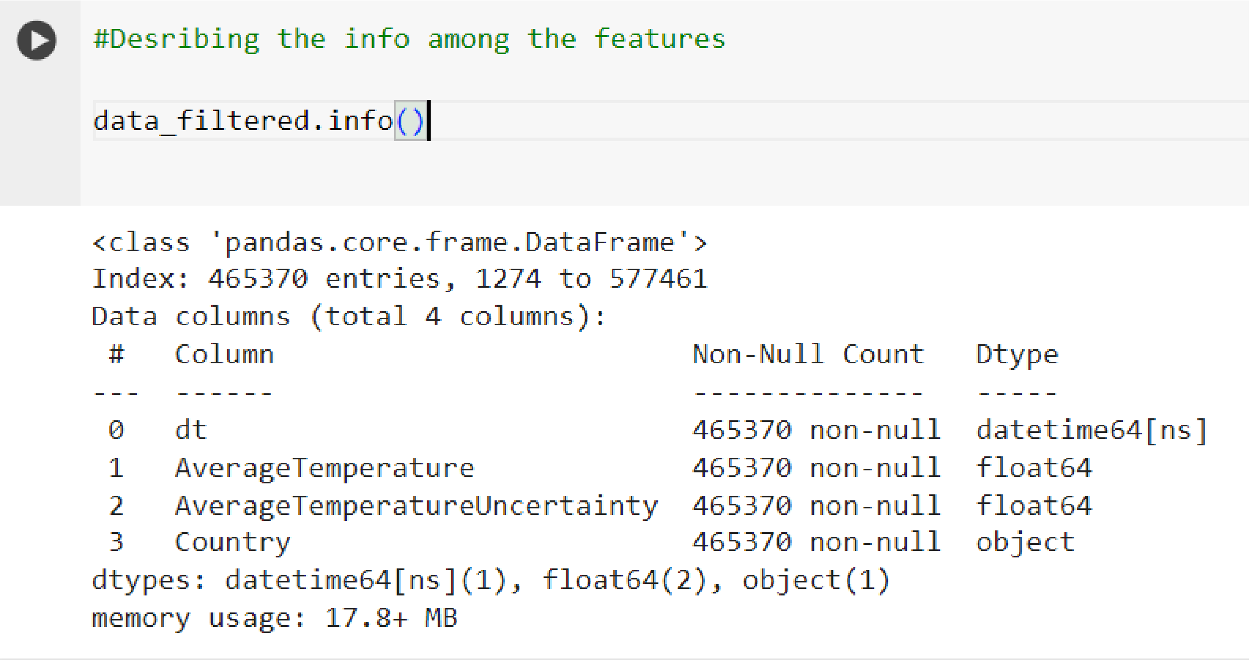
The commonly used metrics such as Absolute Mean Square Error (MSE), Root Mean Square Error (RMSE), and R2 Score are used for evaluating the performance of regression models.

R2 Score represents the proportion of the variance in the dependent variable that is explained by the independent variables (features) in the model.

It will be used to measure the fitness of our model will indicate how well the model will predict the variability in our data which ultimately will prove helpful towards detecting unusual or unexpected patterns in the data that may indicate errors, anomalies or another significant event.

6. DM Algorithm and Selection

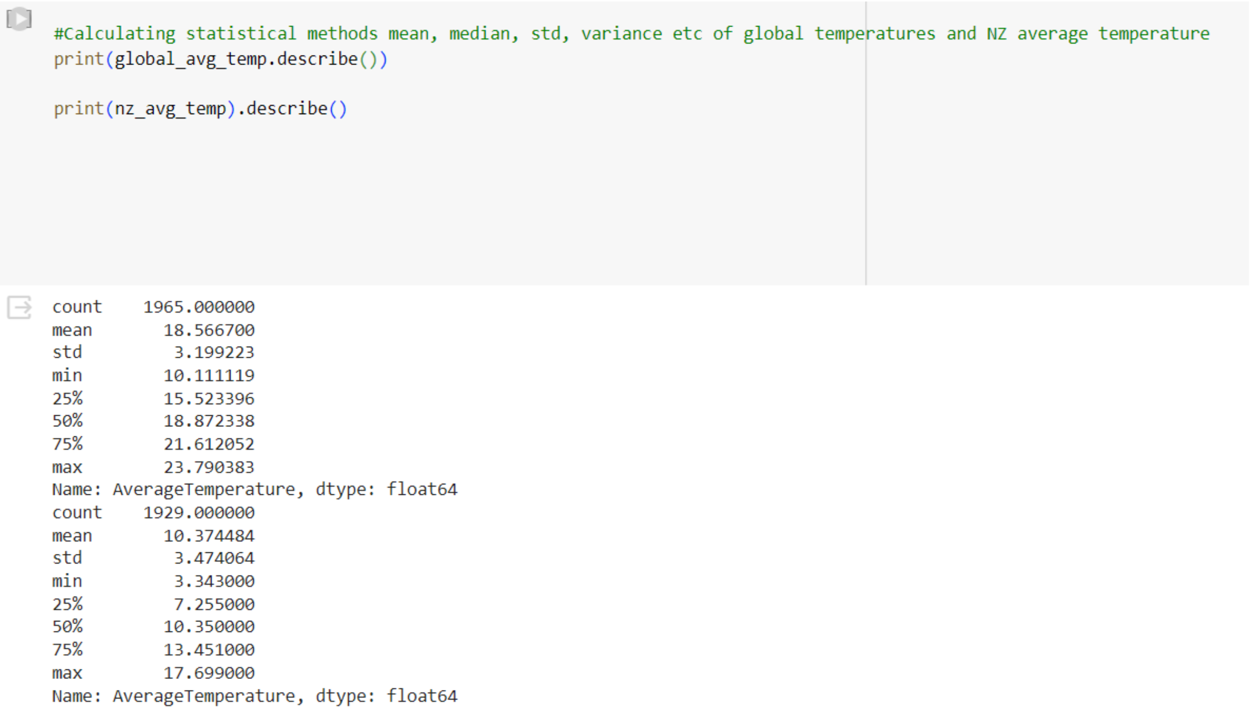
A basic representation of the information of the dataset



```
#Desribing the info among the features

data_filtered.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 465370 entries, 1274 to 577461
Data columns (total 4 columns):
 #   Column                         Non-Null Count   Dtype
---  ------                         --------------   -----
 0   dt                             465370 non-null  datetime64[ns]
 1   AverageTemperature             465370 non-null  float64
 2   AverageTemperatureUncertainty  465370 non-null  float64
 3   Country                        465370 non-null  object
dtypes: datetime64[ns](1), float64(2), object(1)
memory usage: 17.8+ MB
```

To get a better understanding of the data statistically basic computations were calculated.



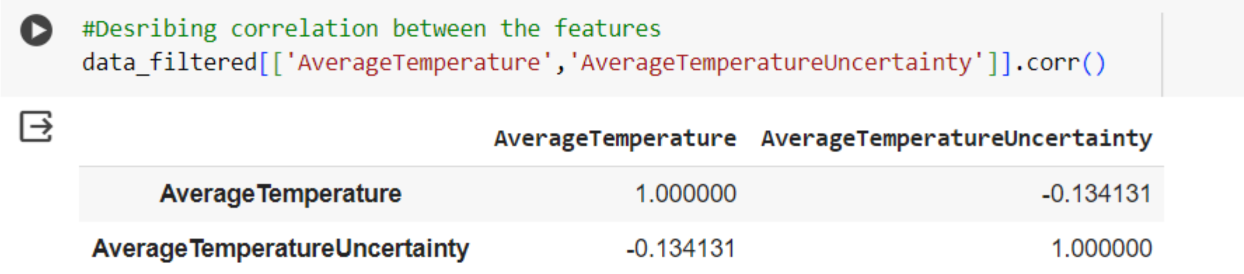
```
#Calculating statistical methods mean, median, std, variance etc of global temperatures and NZ average temperature
print(global_avg_temp.describe())

print(nz_avg_temp).describe()
```

```
count    1965.000000
mean       18.566700
std         3.199223
min        10.111119
25%        15.523396
50%        18.872338
75%        21.612052
max        23.790383
Name: AverageTemperature, dtype: float64
count    1929.000000
mean       10.374484
std         3.474064
min         3.343000
25%         7.255000
50%        10.350000
75%        13.451000
max        17.699000
Name: AverageTemperature, dtype: float64
```

The global temperature has been varied immensely over the years with minimum of 10 degree Celsius to almost 24 degree Celsius maximum with a standard deviation of 3.2.

New Zealand’s temperature has also varied significantly over the years ranging from min 3.34 degree Celsius to a maximum of 17.7 degree Celsius having a standard deviation of 3.5.

The accuracy of the data is validated by computing the correlation between the features AverageTemperature and AverageTemperature Uncertainty.

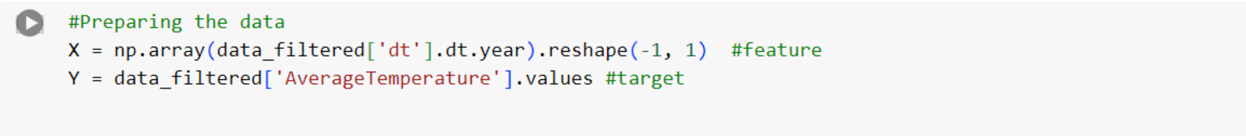


```
#Desribing correlation between the features
data_filtered[['AverageTemperature','AverageTemperatureUncertainty']].corr()
```

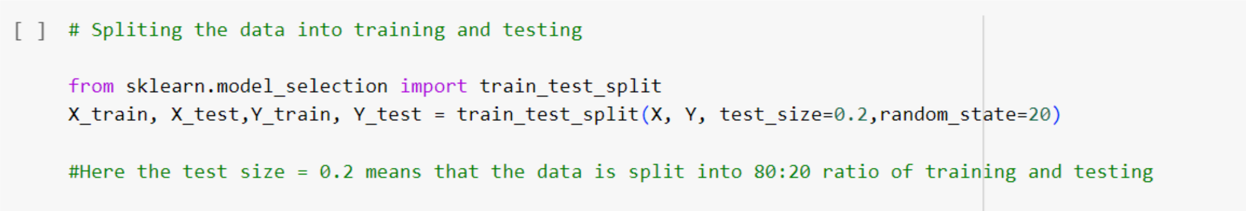
	AverageTemperature	AverageTemperatureUncertainty
AverageTemperature	1.000000	-0.134131
AverageTemperatureUncertainty	-0.134131	1.000000

There is a negative correlation of **- 0.134** between the features which signifies that lower levels of uncertainty are typically associated with higher accurate temperature measurement devices or values.

The objective is to predict the varying temperature trends over the years. This will be implemented by first setting the target variable on which the DM model will predict the values of. In correspondence to this the data is split into training testing data where the X variable (the independent variable) is set as Date feature while the Y variable is set to be AverageTemperatue (the target variable).



```
#Preparing the data
X = np.array(data_filtered['dt'].dt.year).reshape(-1, 1)  #feature
Y = data_filtered['AverageTemperature'].values #target
```



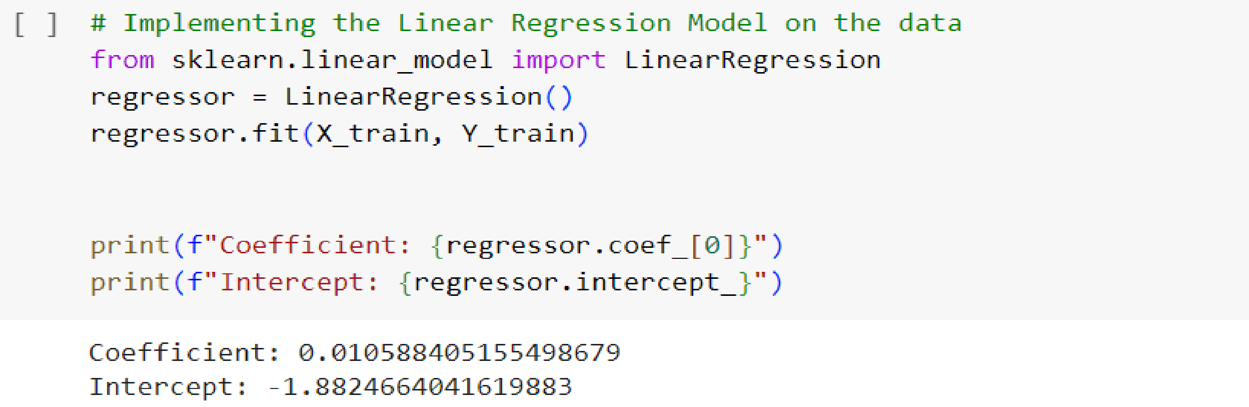
```
[ ]  # Spliting the data into training and testing

     from sklearn.model_selection import train_test_split
     X_train, X_test,Y_train, Y_test = train_test_split(X, Y, test_size=0.2,random_state=20)

     #Here the test size = 0.2 means that the data is split into 80:20 ratio of training and testing
```

Using the inbuilt library of python sklearn we import the module train_test_split and split the dataset into train test data with split ratio of 80:20. The 80:20 split strikes a balance between bias and variance in the model. With a larger training set (80%), the model can capture more complex patterns in the data, potentially reducing bias. Meanwhile, the smaller testing set (20%) helps prevent overfitting by providing a sufficient number of data points to estimate the model's performance without excessively increasing variance.

Here the random state is just a random number that is generally selected to make our data orientation static and not change after every iteration.

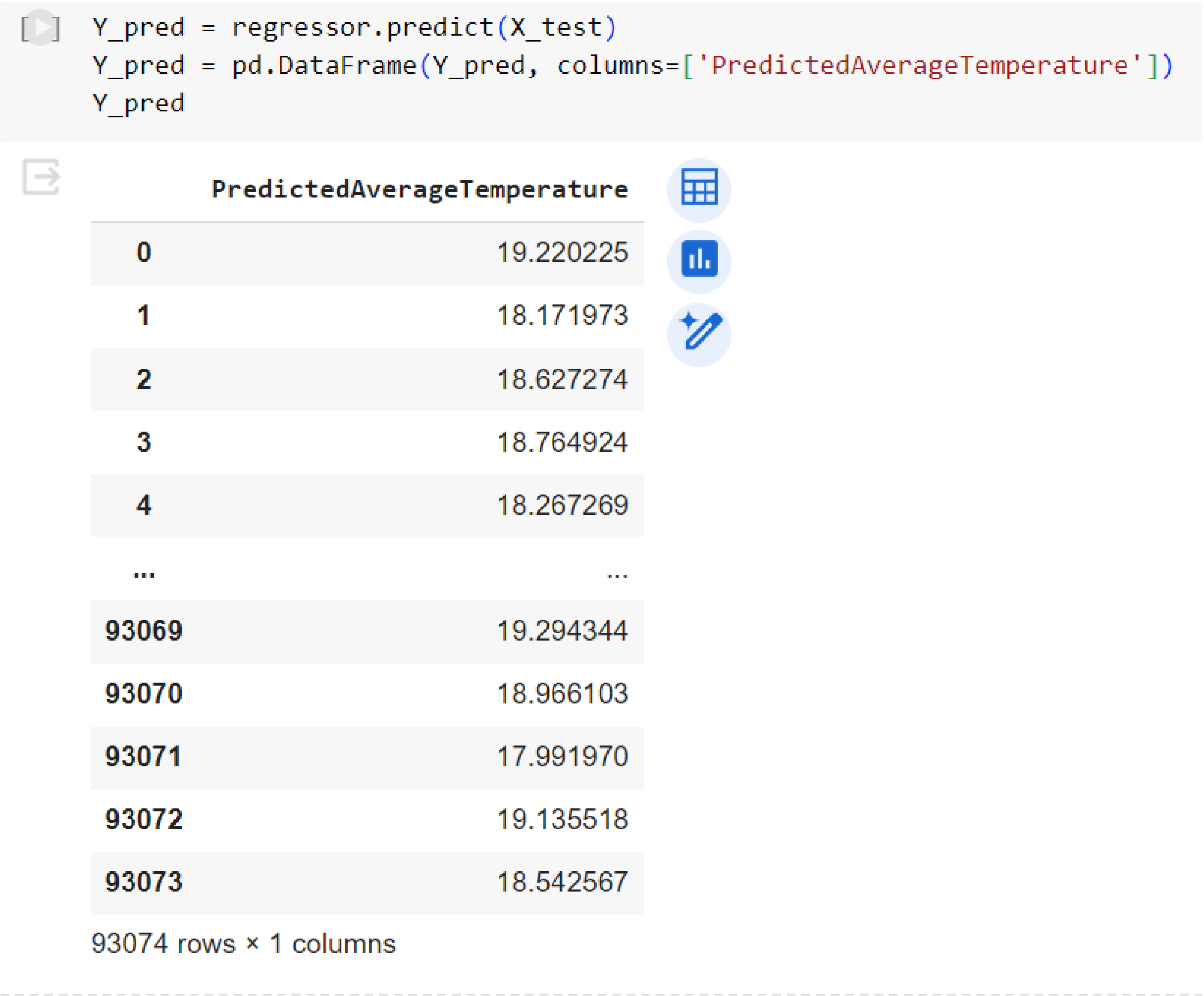
7. Data Mining

From the inbuilt python library sklearn we import the LinearRegression model module and implement it on the data. The implementation will include Regression model fitting the X_train and Y_train data and predicting the required values with X_test and Y_test data.



```
[ ]  # Implementing the Linear Regression Model on the data
     from sklearn.linear_model import LinearRegression
     regressor = LinearRegression()
     regressor.fit(X_train, Y_train)


     print(f"Coefficient: {regressor.coef_[0]}")
     print(f"Intercept: {regressor.intercept_}")
```

```
Coefficient: 0.010588405155498679
Intercept: -1.8824664041619883
```

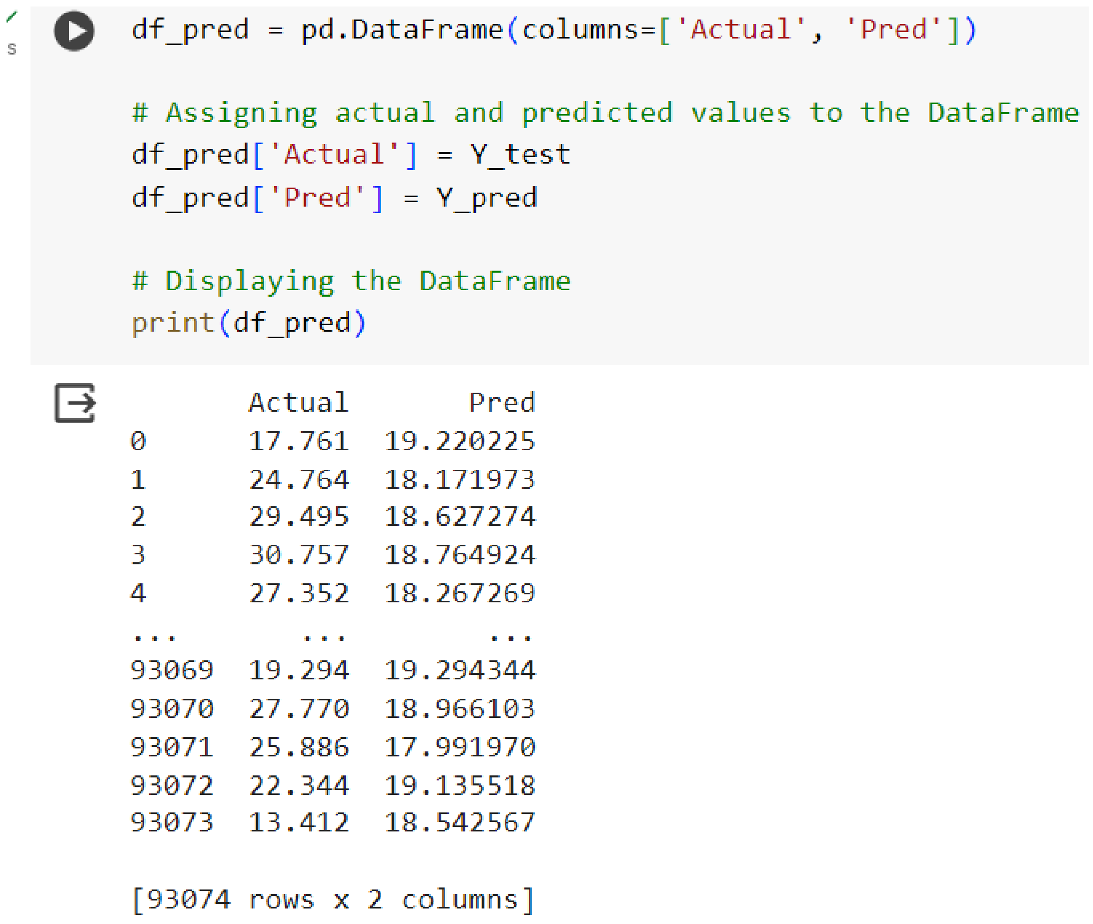


```
Y_pred = regressor.predict(X_test)
Y_pred = pd.DataFrame(Y_pred, columns=['PredictedAverageTemperature'])
Y_pred
```

	PredictedAverageTemperature
0	19.220225
1	18.171973
2	18.627274
3	18.764924
4	18.267269
...	...
93069	19.294344
93070	18.966103
93071	17.991970
93072	19.135518
93073	18.542567

93074 rows × 1 columns

8. Interpretation



```python
df_pred = pd.DataFrame(columns=['Actual', 'Pred'])

# Assigning actual and predicted values to the DataFrame
df_pred['Actual'] = Y_test
df_pred['Pred'] = Y_pred

# Displaying the DataFrame
print(df_pred)
```

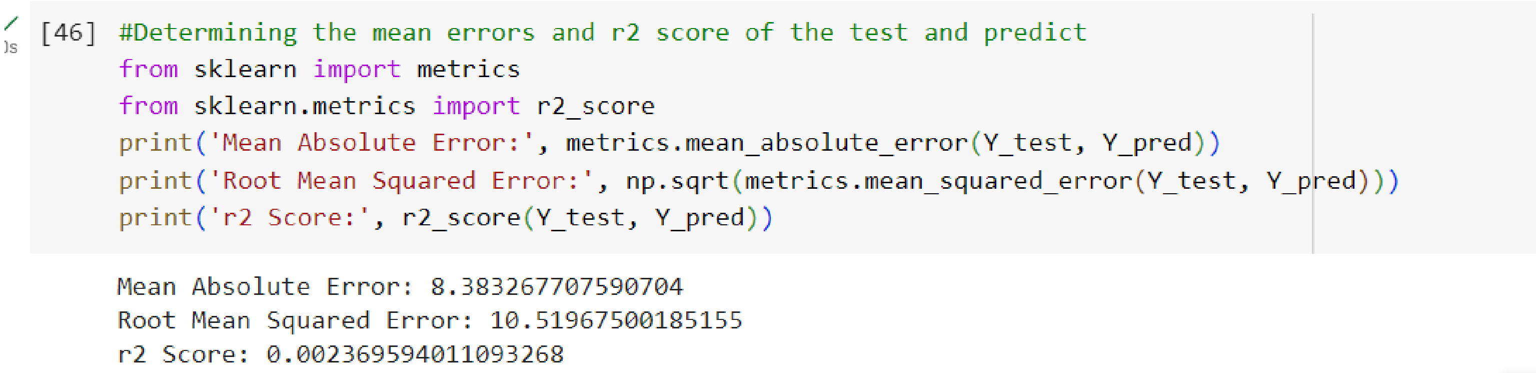
```
       Actual       Pred
0      17.761  19.220225
1      24.764  18.171973
2      29.495  18.627274
3      30.757  18.764924
4      27.352  18.267269
...       ...        ...
93069  19.294  19.294344
93070  27.770  18.966103
93071  25.886  17.991970
93072  22.344  19.135518
93073  13.412  18.542567

[93074 rows x 2 columns]
```

Coefficient: 0.01066. This numerical value denotes the estimated alteration in average temperature (in degrees Celsius) for each additional year.

Intercept: -2.021. This point indicates the intersection of the regression line with the y-axis of the graph. In the context of the model, it signifies the anticipated value of the dependent variable (in this instance, average temperature) when all independent variables (in this case, time as the year) are set to zero.

The positive coefficient implies a long - term pattern of increasing temperatures, aligning with the prevailing understanding of global warming. According to the model, for each passing year, an approximate rise of 0.01066°C in the global average temperature is projected.

Additionally, again from the inbuilt library of Python sklearn we import sklearn metric to compute the errors in the model predictions.



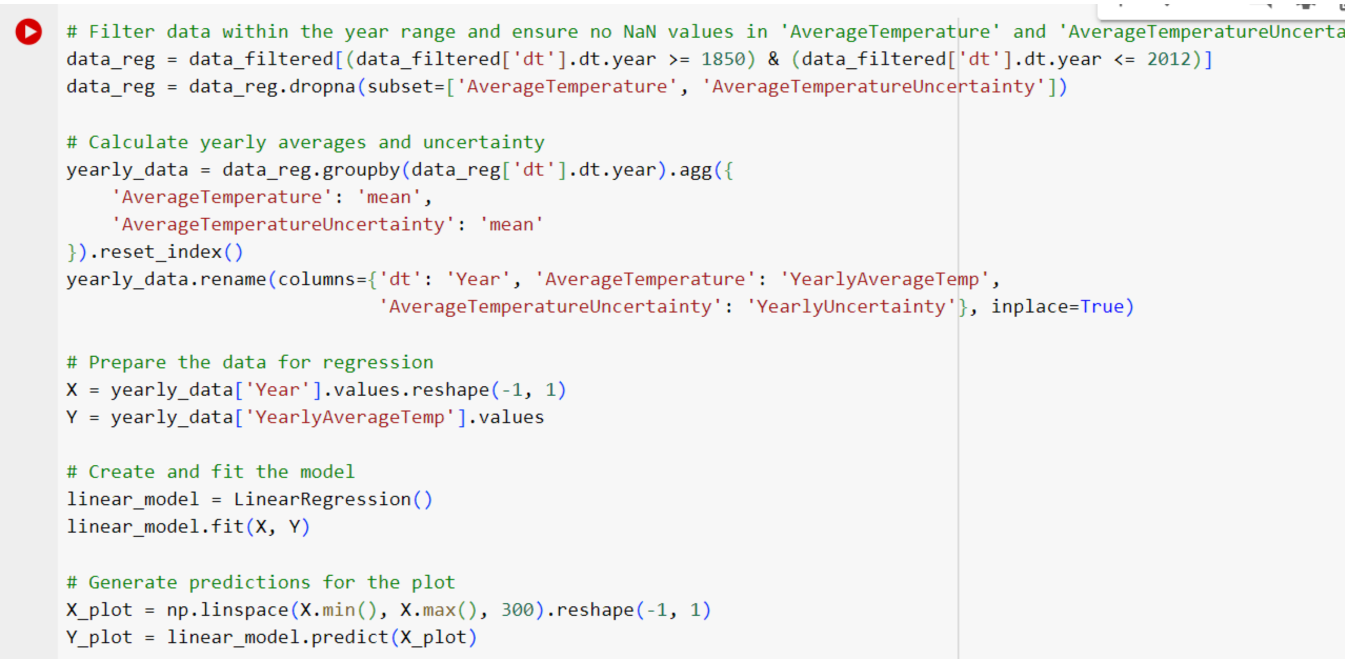
```
#Determining the mean errors and r2 score of the test and predict
from sklearn import metrics
from sklearn.metrics import r2_score
print('Mean Absolute Error:', metrics.mean_absolute_error(Y_test, Y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(Y_test, Y_pred)))
print('r2 Score:', r2_score(Y_test, Y_pred))
```

```
Mean Absolute Error: 8.383267707590704
Root Mean Squared Error: 10.51967500185155
r2 Score: 0.002369594011093268
```

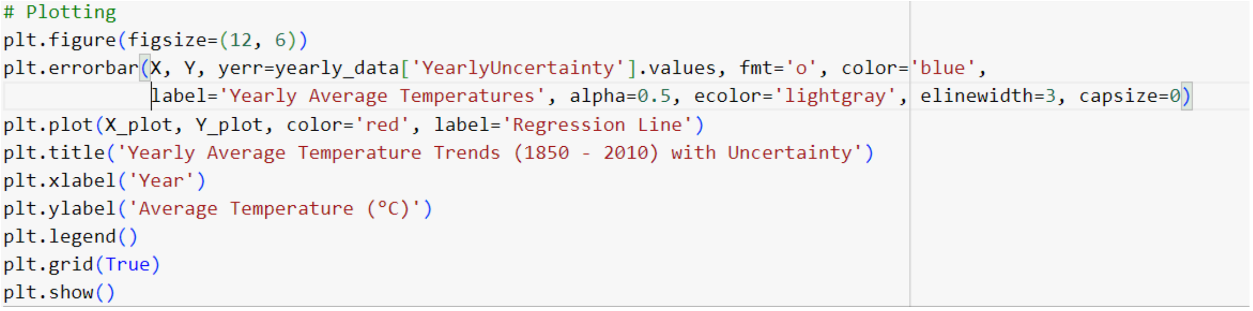
An R2 score of 0.02 indicates that the regression model explains only a small proportion of the variance in the dependent variable. Specifically, it means that approximately 2% of the variability in the observed data is accounted for by the independent variables included in the model.

Although this signifies that the model's ability to predict the variation in the dependent variable is very limited, and the majority of the variability remains unexplained. This low R2 score suggests that the model may not be effectively capturing the underlying relationships between the independent and dependent variables, and its predictive power is poor. But this was expected given the size of the data. This can be avoided by performing multiple iterations with less data size.

We can perform another iteration by specifying a specific range for the training data X reducing the training and testing size and predicting required values which can be visually represented.

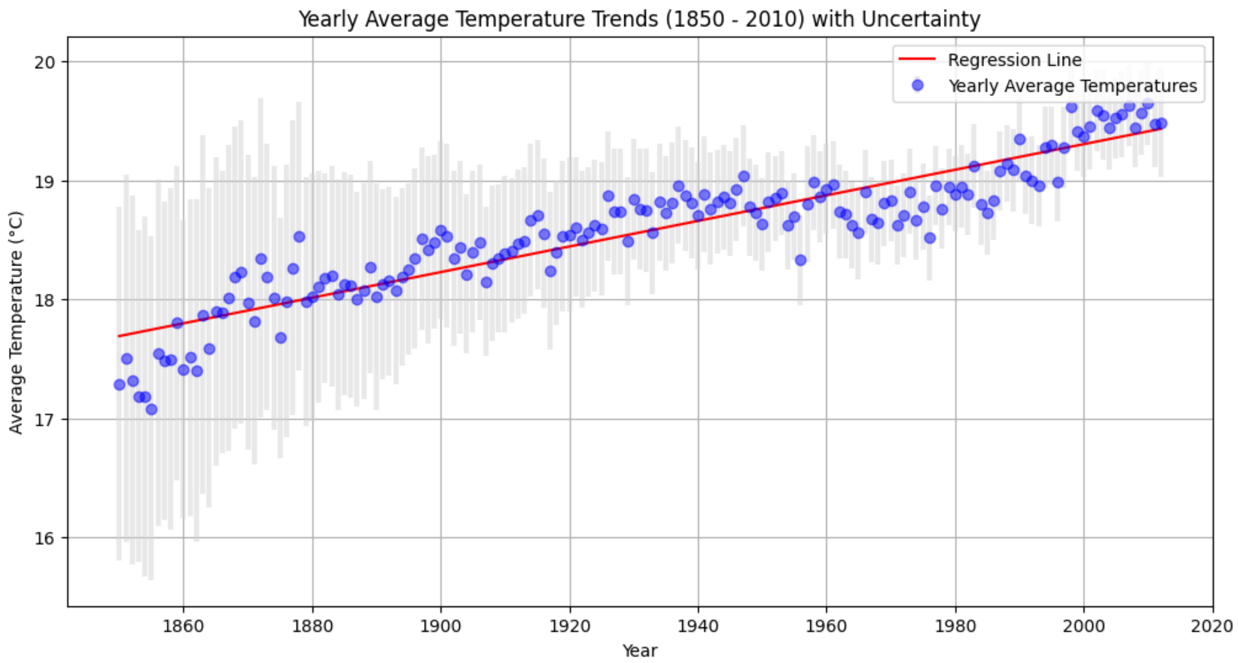


```python
# Filter data within the year range and ensure no NaN values in 'AverageTemperature' and 'AverageTemperatureUncerta
data_reg = data_filtered[(data_filtered['dt'].dt.year >= 1850) & (data_filtered['dt'].dt.year <= 2012)]
data_reg = data_reg.dropna(subset=['AverageTemperature', 'AverageTemperatureUncertainty'])

# Calculate yearly averages and uncertainty
yearly_data = data_reg.groupby(data_reg['dt'].dt.year).agg({
    'AverageTemperature': 'mean',
    'AverageTemperatureUncertainty': 'mean'
}).reset_index()
yearly_data.rename(columns={'dt': 'Year', 'AverageTemperature': 'YearlyAverageTemp',
                            'AverageTemperatureUncertainty': 'YearlyUncertainty'}, inplace=True)

# Prepare the data for regression
X = yearly_data['Year'].values.reshape(-1, 1)
Y = yearly_data['YearlyAverageTemp'].values

# Create and fit the model
linear_model = LinearRegression()
linear_model.fit(X, Y)

# Generate predictions for the plot
X_plot = np.linspace(X.min(), X.max(), 300).reshape(-1, 1)
Y_plot = linear_model.predict(X_plot)
```

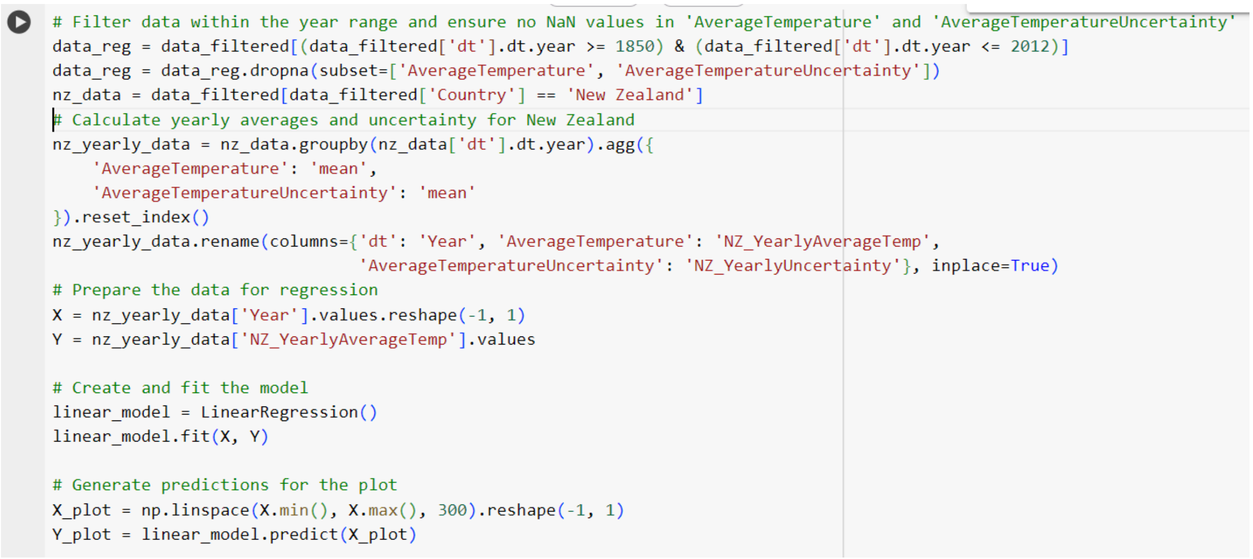


```python
# Plotting
plt.figure(figsize=(12, 6))
plt.errorbar(X, Y, yerr=yearly_data['YearlyUncertainty'].values, fmt='o', color='blue',
             label='Yearly Average Temperatures', alpha=0.5, ecolor='lightgray', elinewidth=3, capsize=0)
plt.plot(X_plot, Y_plot, color='red', label='Regression Line')
plt.title('Yearly Average Temperature Trends (1850 - 2010) with Uncertainty')
plt.xlabel('Year')
plt.ylabel('Average Temperature (°C)')
plt.legend()
plt.grid(True)
plt.show()
```

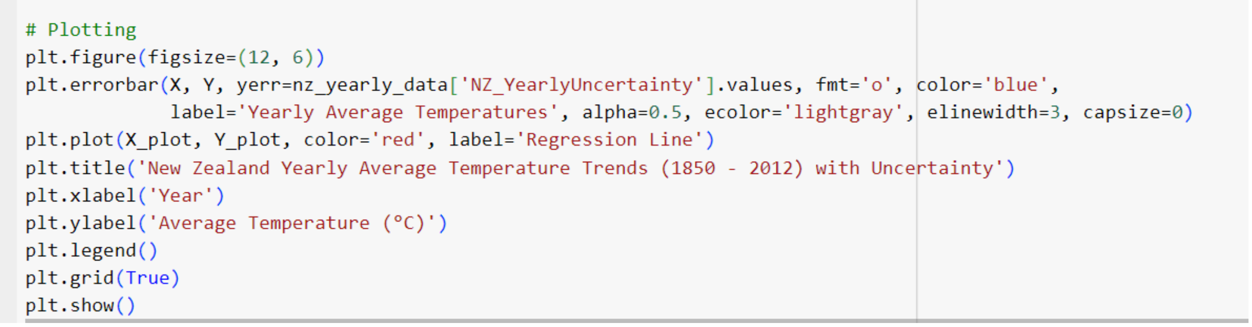
Here the target variable unlike before is set as the aggregate mean of Average Temperature and Average Temperature Uncertainty while the X variable remains the same. This approach reduces the dimensionality of the training dataset and as a result gives more information on the model dataset. The prediction is performed by plotting the Y_pred values over a linespace graph. The names of the features have been changed a bit to bring more clarification such as column name ‘dt’ to ‘year’, ‘AverageTemperature’ to ‘YearlyAverageTemp’ and ‘AverageTemperatureUncertainity’ to ‘YeralyUncertainity’.



The model can also be used to predict the temperatures trends of a Country.

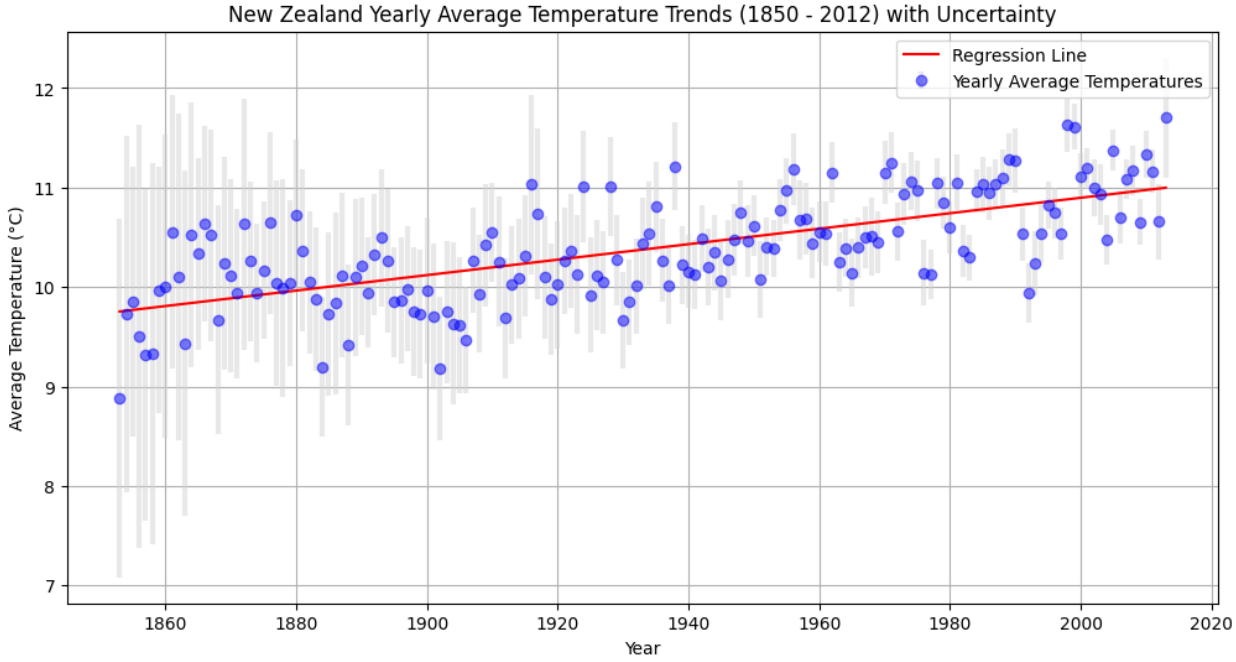


```
# Filter data within the year range and ensure no NaN values in 'AverageTemperature' and 'AverageTemperatureUncertainty'
data_reg = data_filtered[(data_filtered['dt'].dt.year >= 1850) & (data_filtered['dt'].dt.year <= 2012)]
data_reg = data_reg.dropna(subset=['AverageTemperature', 'AverageTemperatureUncertainty'])
nz_data = data_filtered[data_filtered['Country'] == 'New Zealand']
# Calculate yearly averages and uncertainty for New Zealand
nz_yearly_data = nz_data.groupby(nz_data['dt'].dt.year).agg({
    'AverageTemperature': 'mean',
    'AverageTemperatureUncertainty': 'mean'
}).reset_index()
nz_yearly_data.rename(columns={'dt': 'Year', 'AverageTemperature': 'NZ_YearlyAverageTemp',
                               'AverageTemperatureUncertainty': 'NZ_YearlyUncertainty'}, inplace=True)
# Prepare the data for regression
X = nz_yearly_data['Year'].values.reshape(-1, 1)
Y = nz_yearly_data['NZ_YearlyAverageTemp'].values

# Create and fit the model
linear_model = LinearRegression()
linear_model.fit(X, Y)

# Generate predictions for the plot
X_plot = np.linspace(X.min(), X.max(), 300).reshape(-1, 1)
Y_plot = linear_model.predict(X_plot)
```



```
# Plotting
plt.figure(figsize=(12, 6))
plt.errorbar(X, Y, yerr=nz_yearly_data['NZ_YearlyUncertainty'].values, fmt='o', color='blue',
             label='Yearly Average Temperatures', alpha=0.5, ecolor='lightgray', elinewidth=3, capsize=0)
plt.plot(X_plot, Y_plot, color='red', label='Regression Line')
plt.title('New Zealand Yearly Average Temperature Trends (1850 - 2012) with Uncertainty')
plt.xlabel('Year')
plt.ylabel('Average Temperature (°C)')
plt.legend()
plt.grid(True)
plt.show()
```

Here we are filtering out the data by the country New Zealand and grouping it with AverageTemperature and AverageTemperatureUncertainity from the year range 1850 to 2012.



9.1 Applying the Knowledge and Deploying the Implementation:

- Data Collection and Preprocessing: Gather historical climate data from reliable sources and clean the data to ensure accuracy and consistency.

- Feature Engineering: Extract relevant features such as average temperature, temperature anomalies, and geographic information.

- Model Development: Utilize data mining techniques such as regression analysis, time series forecasting, or machine learning algorithms to analyze temperature trends by country.

Visualization: Create visualizations such as line plots, heatmaps, or geographical maps to illustrate temperature trends and climate change impacts.

-Interpretation and Reporting: Interpret the results and findings from the analysis and generate reports to communicate insights to stakeholders.

9.2 Monitoring the Implementation:

- Regular Data Updates: Monitor data sources for updates and ensure that the analysis incorporates the most recent data.

- Model Performance Monitoring: Continuously evaluate the performance of the models and algorithms used for analysis.

- Quality Assurance: Implement checks to ensure data quality and identify any anomalies or inconsistencies.

- Feedback Collection: Solicit feedback from stakeholders to assess the relevance and effectiveness of the analysis.

9.3 Maintaining the Implementation:

- Documentation: Maintain documentation of data sources, preprocessing steps, model configurations, and analysis results for reproducibility and future reference.

- Software Maintenance: Keep software libraries and tools used for analysis up to date to ensure compatibility and performance.

- Data Governance: Establish data governance practices to manage data access, security, and privacy.

- Team Collaboration: Foster collaboration among team members to share knowledge, address challenges, and optimize analysis workflows.

9.4 Enhancing the Solution in the Future:

- Advanced Modeling Techniques: Explore advanced modeling techniques such as deep learning or ensemble methods to improve prediction accuracy.

- Integration of Additional Data Sources: Incorporate additional data sources such as satellite imagery, ocean temperature data, or socioeconomic indicators to enrich the analysis.

- Predictive Analytics: Develop predictive models to forecast future temperature trends and assess the potential impacts of climate change.

- Real-time Monitoring: Implement real-time monitoring systems to track temperature variations and climate events as they occur.

- Automation: Automate data collection, preprocessing, and analysis processes to streamline workflow efficiency.

- Collaborative Research: Collaborate with other researchers and organizations to leverage expertise and resources for broader insights into climate change dynamics.

By applying these strategies, the data mining project can effectively analyze global temperature trends, monitor implementation progress, maintain data quality, and model performance, and enhance the solution to address evolving challenges and opportunities in climate change research.

"I acknowledge that the submitted work is my own original work in accordance with the University of Auckland guidelines and policies on academic integrity and copyright.

I also acknowledge that I have appropriate permission to use the data that I have utilized in this project. (For example, if the data belongs to an organization and the data has not been published in the public domain, then the data must be approved by the rights holder.) This includes permission to upload the data file to Canvas. The University of Auckland bears no responsibility for the student's misuse of data."