Team Number:	apmcm2211488
Problem Chosen:	C

Global Warming Trends Forecasting and Causes Analysising

Summary

Global warming has widely concerned by people around the world in recent years, and people hope to find suitable ways to mitigate the global warming problem. In this paper, we forecast the trend of global temperature and analysis the causes.

For problem 1,we establish **Mann-Kendall Test** model to detect temperature change anomalies and find that the temperature growth in March 2022 is not the fastest in any decade, but the fastest growth moment is in **November 2014**.We build three time series forecasting models, **Prophet, SARIMA and exponential smoothing**, and use the data for training to forecast a number of years in the future, and obtain the forecasting results as shown in **Fig.(3)(7)(8)**.We find that the global average temperature will not reach 20°C in 2050 and 2100 by the prediction results, and we obtain the time when the model prediction reaches 20°C by solving the intersection of the prediction curve with 20°C in **2173, 2196, and 2230 years**, respectively. Then we evaluate the models using two indicators, **MAPE and MSE**, and the evaluation results are shown in the **table(1)**, and the best-performing model is **Prophet**.

For problem 2,We find through our analysis that the temperature is only related to latitude, but not longitude. Also we analyze the effect of sunlight irradiation on temperature and find that temperature shows a quadratic function with the sine of latitude.Considering temperature with time and location at the same time, we establish a **quadratic regression** model and obtain the functional relationship as shown in **euqation(16)**,goodness of fit$\boldsymbol{R^2 = 0.713}$.We collect data related to natural disasters, including volcanic eruptions, forest fires, floods, droughts, and COVID-19, and we count the number of natural disasters occurring in each year, and we find that the trend is not related to the trend of temperature change, while we confirm this by calculating the **Pearson coefficient**, and the correlation coefficient is shown in **Fig(15)**.We look up the information and find that the main cause of global warming is **greenhouse gases**. We collect data on the concentration of three greenhouse gases, CO2, CH4 and N2O, and find that the trends are very similar to those of temperature change, and calculate Pearson coefficients of 0.93, 0.86 and 0.92 respectively, as shown in the **Fig(16)**.Finally, based on the results of the analysis and the collected information, we propose measures to mitigate the global warming problem for **individual, enterprise or factory, and government** respectively.

For problem 3,We summarize the results of the analysis and discussion, combine the information we have found, translate the technical terms into easy-to-understand phrases, and write a non-technical article to the APMCM organizing committee, proposing some reasonable suggestions and measures.

Keywords: Mann-Kendall Test Prophet SARIMA Exponential Smoothing Quadratic Regression Pearson Coeffience Greenhouse Gases

Contents

I. Introduction

1.1 Background

Global warming is the most concerned global environmental issue in today's human society,which ranks first among three categories of global environmental problems (global warming, acid rain and the hole in the ozone layer).The Earth's climate change as a whole is characterized by fluctuations,but there has never been a time in Earth's history when climate change has been as rapid as it is now.Since the Industrial Revolution, the rate of global temperature increase has been very fast,for nearly 100 years,the global average temperature has risen by about 0.6°C[1].

Many climate scientists agree that if the global average temperature rises by more than 2°C in a short period of time,it will lead to serious ecological damage.This damage will exacerbate the extinction of many plant and animal species.And it will change existing agricultural patterns,leading to a series of social and economic problems.The Intergovernmental Panel on Climate Change(IPCC) noted that human beings and their activities have been responsible for a worldwide average temperature increase between 0.8 and 1.2°C (1.4 and 2.2°F) since preindustrial times, and most of the warming over the second half of the 20th century could be attributed to human activities in its Sixth Assessment Report (AR6) published in 2021 [2].

So,it's obvious that we urgently need a tool that can predict future temperature changes,and determine which factor, natural or man-made, has more to do with global temperature change.That's what our research is based on.

1.2 Our Work

We built multiple time series models to predict long-term future temperature changes by regression.And obtain data on natural disasters and greenhouse gases from various databases on the Internet (Natural Disaster Information Database, Our World in Data) to analyze which factors have more influence on temperature change.

1.3 Significance

Through this study, we can roughly judge the future temperature change trend, as well as the main factors leading to this change.In the future, corresponding measures can be taken to mitigate such a rapid trend of global warming.

II. The Description of the Problem

2.1 Problem Restatement

- **Problem 1(a)**:Using data to analyze whether the increase of global temperature in March 2022 is greater than the increase of any previous decade.
- **Problem 1(b)**:Based on historical data, build two or more models to evaluate past temperatures and predict future temperatures.

- **Problem 1(c)**:Use the model we built to predict when the global average temperature will reach 20 degrees Celsius, in 2050 or 2100?
- **Problem 1(d)**:Evaluate models and choose the best one.
- **Problem 2(a)**:Based on the data, build a model to analyze the relationship among global temperature, time and location.
- **Problem 2(b)**:Analyze the impact of natural disasters on global temperature.
- **Problem 2(c)**:Analyze the main causes of global temperature change.
- **Problem 2(d)**:Propose measures to slow global warming.
- **Problem 3**:Write a non-technical article to present the results and give recommendations.

2.2 Problem Analysis

- **Problem 1(a)**:Considering that the global temperature has been increasing, we study whether the temperature growth rate in March 2022 is the fastest in the last decade.This question is suitable for using anomaly detection model, if a point is detected as an anomaly point, it means that the change rate of this point is significantly different from other points on the sequence, so as to determine whether the growth rate is the fastest.
- **Problem1 (b)**:The global temperature data is a time series data, and the time series data is autocorrelated with each other, so it is not appropriate to use regression model for prediction, so we use time series model for this question.Commonly used time series models include Autoregressive model (AR), Moving Average model (MA), Autoregressive Moving Average model (ARMA), Autoregressive Integrated Moving Average model (ARIMA), Seasonal Autoregressive Integrated Moving Average (SARIMA), exponential smoothing, LSTM, and Prophet.
- **Problem 1(c)**:After we obtain the time series model, we predict the future temperature based on the autoregressive characteristic and quantify when the model will reach 20 $°C$.
- **Problem 1(d)**:For the evaluation of time series models, we generally have Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Symmetric Mean Absolute Percentage Error (SMAPE), etc. Here we use MAPE and MSE to evaluate the model. We select a truncation point from the sequence, the one before the truncation point is the training set and the rest is the testing set.
- **Problem 2(a)**:According to common sense, we know that there is a relationship between geographical location and temperature: the greater the latitude, the lower the temperature. And due to the greenhouse effect, the global temperature is now continuously increasing. So there is also a positive relationship between time and global temperature.Based on the available data, we wanted to obtain an explicit, quantified model of the relationship among global temperature, time and location. Considering that location and global temperature are not a linear relationship, we used a quadratic regression method to build the model.
- **Problem 2(b)**:The impact of natural disasters often lasts for a period of time, so the occurrence of natural disasters can also be considered as a time series, so that our goal is to analyze the correlation between the two time series of temperature and natural disasters.
- **Problem 2(c)**:The greenhouse effect means that greenhouse gases form an insulating film on the surface of the earth, and the heat emitted from the earth to the outside is less than the heat absorbed by the earth from the outside, resulting in the accumulation of heat on the earth and

an increase in temperature. Therefore, we can analyze that the main cause of global change should be the rise of greenhouse gas content, and find the relationship between the two with correlation.

- **Problem 2(d)**:By looking at the data and combining the results we have obtained, we propose strategies to slow down global warming.
- **Problem 3**： We need to translate the results of the problem into easy-to-understand words and present them to the APMCM organizing committee with feasible recommendations.

III. Model Assumption

- Assuming that global temperature changes are only related to human activities and natural laws.
- It is assumed that the global temperature varies cyclically during the year, i.e., the temperature is seasonal.
- Supposed that the global average temperature will not reach saturation within a period of 100 to 200 years.
- Assuming that the temperature data errors obey a normal distribution.

IV. Symbol description

Symbol	Description	Unit
n	Sample size	-
t	Time	month/year
UF_k	Rank series of sequential time series	-
UB_k	Rank series of inverse time series	-
ϵ_t	t moment noise	-
$S_t^{(k)}$	k times exponential smoothing value at t moment	-
φ	Latitude	°
θ	Angle of parallel light to the Earth's surface	°
$T(\varphi, t)$	Temperature with latitude φ and time t	°C
ρ_{XY}	Correlation coefficient of X, Y	-

V. Model Building and Solution of Problem 1

The first problem is divided into four subproblems. In the first subproblem we use the Mann-Kendall anomaly test to determine whether the temperature growth rate is abruptly changing. The last three subproblems can be distilled into building a time series forecasting model, forecasting and testing the accuracy of the model. For the time series forecasting model, we build three models, Prophet, ARIMA, and exponential smoothing, and for the model testing metrics, we choose MAPE, MSE.

5.1 Model Building

5.1.1 *Mann-Kendall Test*

The Mann-Kendall method is a nonparametric statistical test for testing outliers, which has the following steps:

For a time series x with n sample sizes, construct a rank series:

$$s_k = \sum_{i=1}^{k} r_i \qquad (k = 2, 3, ..., n) \tag{1}$$

where

$$r_i = \begin{cases} +1 & if\ x_i > x_j \\ 0 & else \end{cases} \quad (j = 1, 2, ..., i) \tag{2}$$

Under the assumption of random independence of the time series, define the statistics:

$$UF_k = \frac{[s_k - E(s_k)]}{\sqrt{Var(s_k)}} \qquad (k = 1, 2, ..., n) \tag{3}$$

where $UF_1 = 0$, $E(s_k)$, $Var(s_k)$ are the mean and variance of the cumulative s_k in $x_1, x_2, ..., x_n$ when they are independent of each other and have the same continuous distribution, they can be calculated by the following equation:

$$\begin{aligned} E(s_k) &= \frac{n(n+1)}{4} \\ Var(s_k) &= \frac{n(n-1)(2n+5)}{72} \end{aligned} \tag{4}$$

UF_i is a standard normal distribution, which is a sequence of statistics in the order of time series x $x_1, x_2, ..., x_n$ computed as a sequence of statistics, and given a significance level α, if $UF_i > U_\alpha$, it indicates that there is a significant trend change in the sequence.

Inverse order by time series x $x_n, x_{n-1}, ..., x_1$, then repeat the above process while making $UB_k = -UF_k, k = n, n-1, ..., 1, UB = 0$, and the intersection of UF_k and UB_k is the anomaly point.

5.1.2 *Prophet*

Prophet algorithm is a machine learning algorithm based on time series decomposition, which adds seasonal trends and holiday trends compared to traditional time series analysis, allowing the model to have stronger explanations while achieving better fits and more accurate predictions of long term trends.

The Prophet model is shown below:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \tag{5}$$

where $g(t)$ represents the growth function for fitting acyclic transformations, $s(t)$ represents periodic transformations, and $h(t)$ represents holidays, festivals.The ϵ_t is the noise term. The growth term is fitted using a logistic Stiffness function, the seasonal term uses a Fourier series, and the holidays are represented by a specific indicator function with the following expression:

$$\begin{aligned} g(t) &= \frac{C(t)}{1+e^{-(k+a(t)^T\delta)(t-(m+a(t)^T\gamma))}} \\ s(t) &= \sum_{n=1}^{N}\left(a_n\cos(\frac{2\pi nt}{P})+b_n\sin(\frac{2\pi nt}{P})\right) \\ h(t) &= \sum_{i=1}^{L}\kappa_i\cdot I_{\in D_i} \end{aligned} \tag{6}$$

For this problem, we assume that the rising trend of temperature will not slow down in the short term, so the growth term is fitted with a linear function instead of a logistic Stiff function when considering the growth term, so that the modified $g(t)$ is as follows:

$$g(t) = (r + a(t)^T\delta)t + (d + a(t)^T\gamma) \tag{7}$$

For the holiday term, we consider the effect of holidays on temperature to be negligible, so the term is 0.

5.1.3 *SARIMA*

The SARIMA model is based on the ARIMA model taking into account the seasonality factor. Autoregressive Integrated Moving Average model (ARIMA) is a commonly used time series analysis model, which consists of three steps: Autoregressive (AR), Integrated (I), and Moving Average (MA), and is capable of fitting and forecasting a smooth time series, and has a wide range of applications in time series analysis. the ARIMA model is shown below:

$$y(t) = \mu + \sum_{i=1}^{p}\gamma_i y(t-i) + \epsilon_t + \sum_{i=1}^{q}\theta_i\epsilon_{t-i} \tag{8}$$

where μ is the constant term, ϵ_t is the noise term, γ_i and θ_i are the coefficients, and three parameters, p, d, and q, are the parameters of the AR, I, and MA models, respectively.

The steps of the ARIMA model are as follows:

Step 1: The series is tested for smoothness, and if the series is not smooth, it is differenced until it becomes a smooth series after d-order differencing.

Step 2: The maximum lag points of p and q by partial autocorrelation coefficient (PACF) and autocorrelation coefficient (ACF)

Step 3: Test the significance of the model parameters, the validity of the model itself, and test whether the residual series is a white noise series.

Step 4: Use the established model to make predictions for future data.

SARIMA introduces seasonal factors on the basis of ARIMA, and considers the seasonal factor S as an ARIMA model as well, which together with the original ARIMA model constitutes the SARIMA model.

5.1.4 *Exponential Smoothing*

For time series forecasting problems, we usually consider that the more recent term has more influence on the future forecast, so we obtain the exponential smoothing model by weighting the sum of the current term and the past term. Commonly used exponential smoothing models are simple exponential smoothing, Holt's linear trend method, and Holt-Winters' seasonal method.

Simple exponential smoothing formula is as follows:

$$y_{t+1}^{'} = ay_t + (1-a)y_t^{'} \tag{9}$$

where $y_{t+1}^{'}$ is the predicted value at $t+1$ and a is the weight.

Holt's linear trend method formula is as follows:

$$\begin{aligned} & S_t^{(2)} = \alpha S_t^{(1)} + (1-\alpha) S_{t-1}^{(2)} \\ & \hat{Y}_{t+T} = a_t + b_t \cdot T \\ & \begin{cases} a_t = 2S_t^{(1)} - S_t^{(2)} \\ b_t = \frac{a}{1-a}\left(S_t^{(1)} - S_t^{(2)}\right) \end{cases} \end{aligned} \tag{10}$$

$S_t^{(2)}$ is the quadratic exponential smoothing value at t, $S_t^{(1)}$ is the primary exponential smoothing value at t, α is the weight, and $\hat{Y}_{t+T}$ is the predicted value at $t+T$.

Holt-Winters' seasonal method formula is as follows:

$$\begin{aligned} & S_t^{(3)} = \alpha S_t^{(2)} + (1-\alpha) S_{t-1}^{(3)} \\ & \hat{y}_{t+T} = a_t + b_t \cdot T + c_t \cdot T^2 \\ & \begin{cases} a_t = 3S_t^{(1)} - 3S_t^{(2)} + S_t^{(3)} \\ b_t = \frac{\alpha}{2(1-\alpha)^2}\left[(6-5\alpha)S_t^{(1)} - 2(5-4\alpha)S_t^{(2)} + (4-3\alpha)S_t^{(3)}\right] \\ c_t = \frac{\alpha^2}{2(1-\alpha)^2}\left[S_t^{(1)} - 2S_t^{(2)} + S_t^{(3)}\right] \end{cases} \end{aligned} \tag{11}$$

$S_t^{(3)}$ is the quadratic exponential smoothing value at t and $\hat{y}_{t+T}$ is the predicted value at $t+T$.

5.1.5 *MAPE & MSE*

MAPE is a metric for judging the difference between the predicted and true values of the model with the following equation:

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{\hat{y}_i - y_i}{y_i} \right| \tag{12}$$

where n is the number of samples, $\hat{y}_i$ is the predicted value, and y_i is the true value.

MSE is also a metric to assess the difference between the model output and the true value with the following equation:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2 \tag{13}$$

where n is the number of samples, $\hat{y}_i$ is the predicted value, and y_i is the true value.

MAPE is an error relative to the sample value and MSE is an absolute error. We want good models to achieve smaller values on both.

5.2 Problem solving

5.2.1 *Problem (a)*

Since the global temperature has been increasing, we consider the trend from 2012 to 2022 and find the point from which the fastest growth rate is observed. We use the Mann-Kendall test to test the data for outliers and the results are shown below:

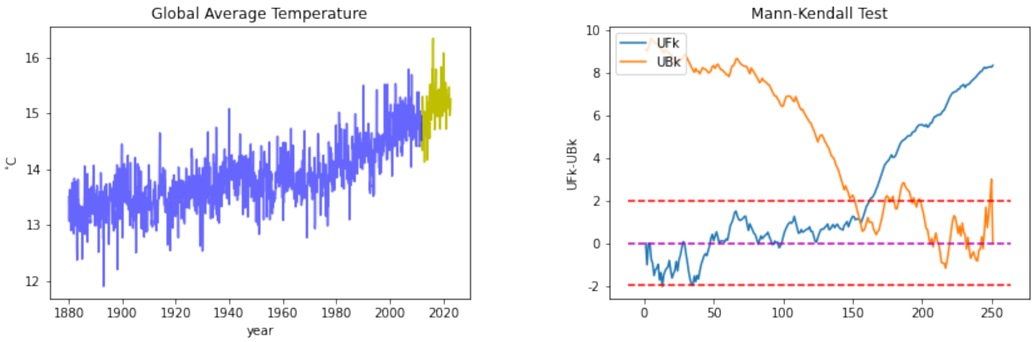

Figure 1 Mann-Kendall Test

The left graph marks the data from 2012 to 2022, while the intersection of the two curves in the right graph is November 2014, where the temperature growth rate is the highest, indicating that March 2022 is not the fastest, thus denying the statement in the question.

5.2.2 *Problem (b)*

1.Prophet

We represent the temperature and time data over the years with a line chart to analyze the possible relationship between temperature and time.

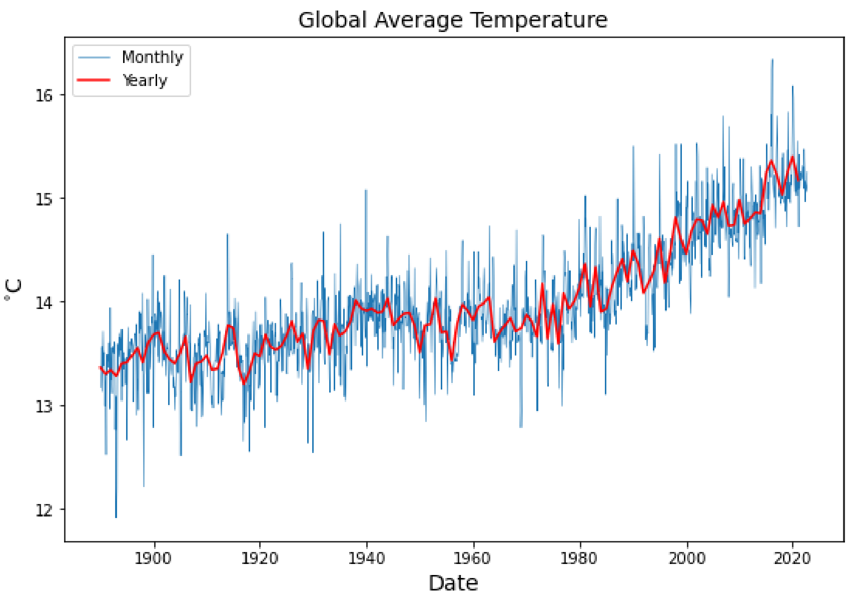

Figure 2 Temperature-Year

It can be seen that the global average temperature will fluctuate locally, but on the whole it has risen steadily, and the margin of rise has increased since 1970. The average monthly temperature is also cyclical throughout the year.

Therefore, we use the trend term and seasonal term in the model to fit. Since the temperature change is not sensitive to holidays, the holiday term can be discarded.

We input the time series into the Prophet model with the month as the time unit and the global average temperature as the prediction target.The result of the model fitting is as follows:

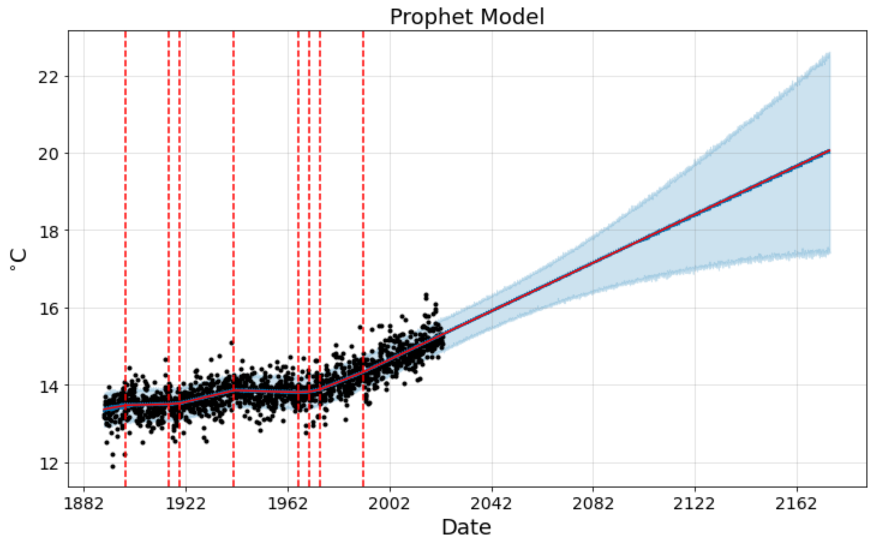

Figure 3 Prophet

From the results, we can see that the model believes that the trend has undergone a total of 8 changes, among which the largest change occurred around 1970, which is consistent with our previous analysis.The blue filled area represents the 80% confidence interval.

2.SARIMA

First, we make first-order difference for the original time series, and the result is as follows:

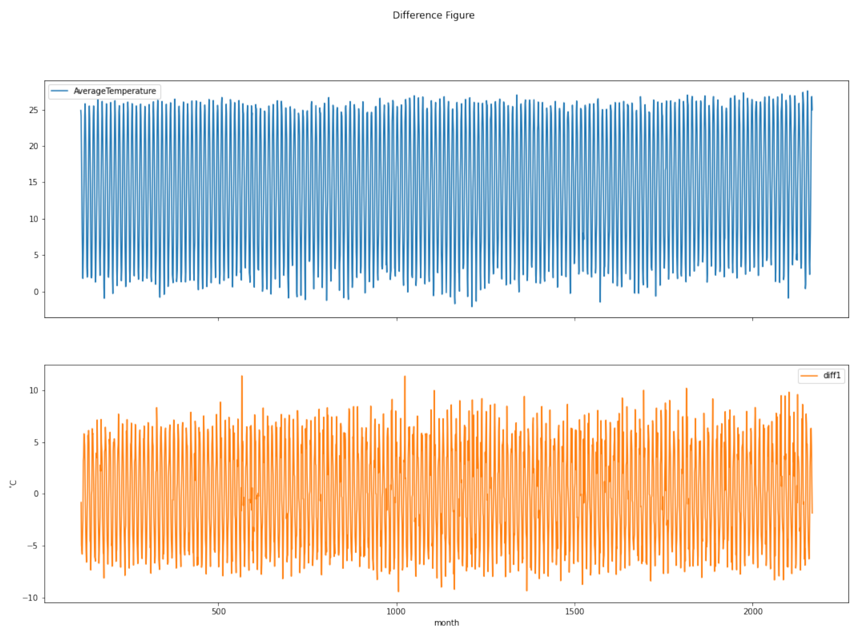

Figure 4 Difference Result

We can see that the differential sequence is much like a stationary time series. Next, the sequence is tested for stationarity, using the root of unity test and the white noise test results as follows:

```
   Augmented Dickey-Fuller Results
=====================================
Test Statistic                -13.177
P-value                         0.000
Lags                               26
-------------------------------------

Trend: Constant
Critical Values: -3.43 (1%), -2.86 (5%), -2.57 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

	lb_stat	lb_pvalue	bp_stat	bp_pvalue
1	1151.239986	2.412093e-252	1149.559344	5.593165e-252
2	1537.735524	0.000000e+00	1535.302578	0.000000e+00
3	1537.751904	0.000000e+00	1535.318919	0.000000e+00
4	1958.456134	0.000000e+00	1954.794815	0.000000e+00
5	3194.402447	0.000000e+00	3186.531092	0.000000e+00
6	4837.456068	0.000000e+00	4823.188397	0.000000e+00
7	6080.704201	0.000000e+00	6060.991648	0.000000e+00
8	6496.443791	0.000000e+00	6474.708175	0.000000e+00
9	6496.574843	0.000000e+00	6474.838525	0.000000e+00
10	6901.578099	0.000000e+00	6877.476799	0.000000e+00
11	8143.433159	0.000000e+00	8111.475841	0.000000e+00

Figure 5 Stability test

It can be seen from the test results that $p-value$of both tests is close to 0 which means rejecting the null hypothesis, we can consider the sequence to be stationary.Therefore, we determine that the order of the difference is of the first order.

Then, we conducted autocorrelation coefficient test and partial autocorrelation coefficient test for the sequence, and the results are as follows:

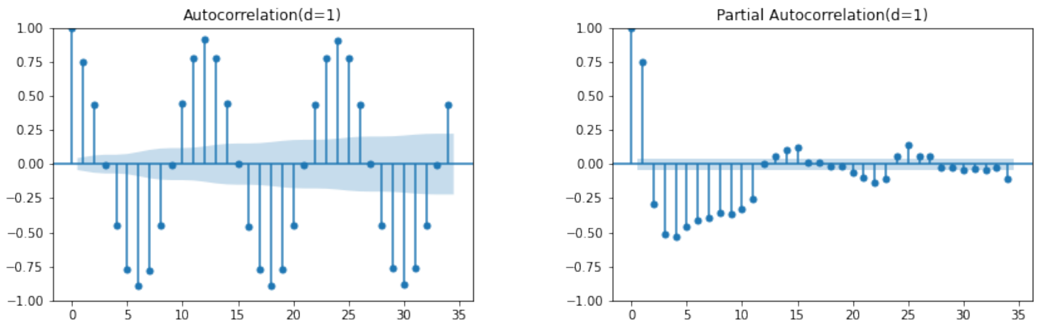

Figure 6 ACF and PACF Test

We can see that PACF shows a first-order truncation, while ACF shows periodic oscillation with a period of 12 orders. This is because the temperature shows periodic changes in 12 months per year, so it is necessary to model the seasonal factor S.

By adjusting the parameters, we determined the SARIMA model as ARIMA(1,1,1)xS(0,1,1), and used this model to predict the global average temperature. The results are as follows:

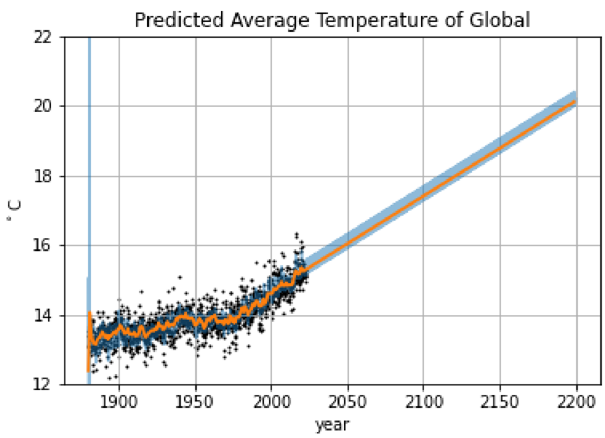

Figure 7 SARIMA Result

According to the forecast results, the future temperature will show a straight line rise, which is also in line with the trend of recent decades of temperature rise. You can also roughly see the slope of the line, which rises about 3°C every 100 years.

3.Exponential Smoothing

Since exponential smoothing is classified into first-order, second-order and third-order models, and the relationship between temperature and time has an obvious trend and seasonality, here we use a third-order exponential smoothing model, i.e. Holt Winters model, to fit the data. We adjust the hyperparameter smoothing_level, smoothing_slope and smoothing_seasonal continuously, and the final model results are as follows:

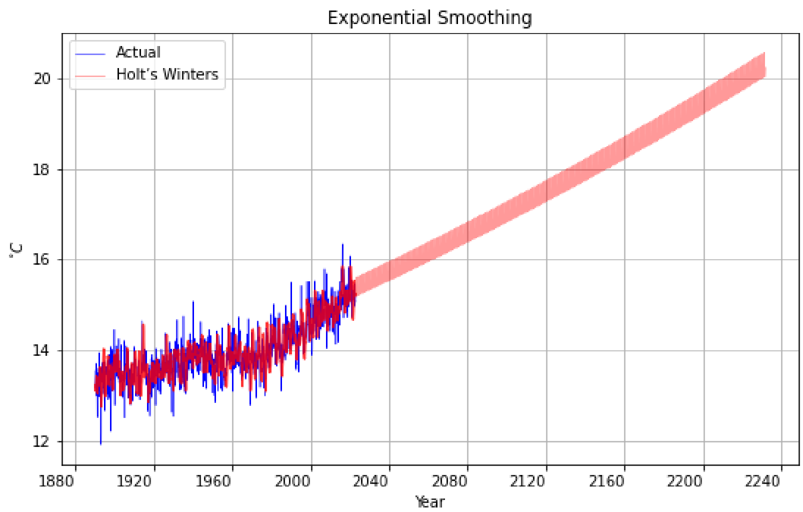

Figure 8 Holt Winters Result

The model shows a linear increase in future temperature predictions and a cyclical pattern within each year, indicating that the model learns the trend and seasonality.

5.2.3 *Problem (c)*

Using the prediction curves plotted in problem (b), we can find that the global average temperature does not reach 20°C in 2050 and 2100.By finding the intersection of the three models with 20°C, we can obtain the years when the predicted values of the three models reach 20°C. The years when the predicted values of the Prophet, SARIMA, and exponential smoothing models reach 20°C are are 2173, 2196, and 2230 years.

5.2.4 *Problem (d)*

We evaluate each of the three models built with the metrics MAPE and MSE. The performance of the models on the test set is shown in Fig:

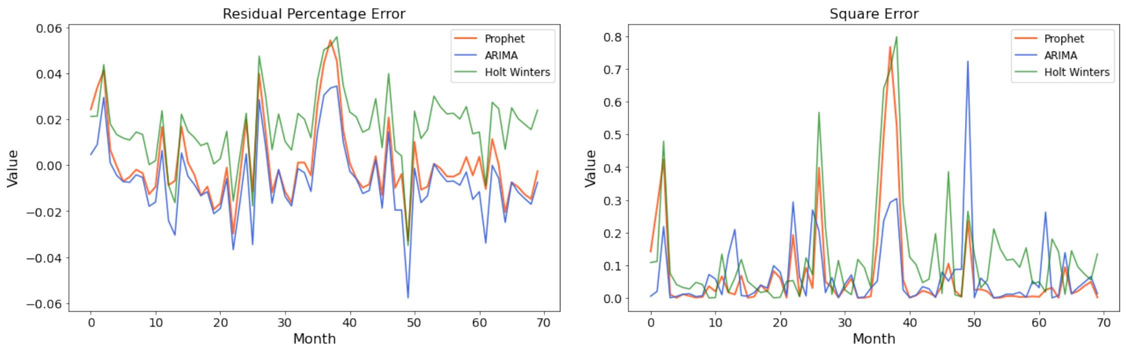

Figure 9 Evaluation on test set

Averaging the performance on the test set we can obtain the MAPE and MSE for each model, as shown in the following table:

Models	MAPE	MSE
Prophet	0.01244	0.07188
ARIMA	0.01358	0.07029
Holt-Winters	0.01951	0.12660

Table 1 Evaluation of each model

Our ideal model should have low MAPE and MSE values, the Holt-Winters model is higher in both, while Prophet outperforms ARIMA in MAPE and similarly in MSE. We can conclude that the Prophet model is the most accurate model.

VI. Model building and solution of problem 2

The second question is divided into four small questions, Q1 asks us to analyze the relationship between temperature, time, and position, and we use the quadratic regression model for analysis. Q23 asks for an analysis of natural disasters or other factors that contribute to global warming, and we use the Pearson correlation coefficient for correlation analysis. Q4, we propose relevant measures based on conclusions.

6.1 Model Building

6.1.1 *Quadratic Regression*

We need to explore the relationship between global temperature and time and location. First of all, we notice that the temperature of a certain region is actually independent of longitude, because the Earth's rotation is in a daily cycle, so we only need to use latitude to represent location information. In the first question, we have explored the relationship between temperature and time, which is monotonically increasing overall. Temperature is related to sun exposure, according to common sense, the lower the latitude, the higher the temperature, the higher the latitude, the lower the temperature, which is due to the different angles formed by sunlight on the earth, resulting in different degrees of heat dispersion.

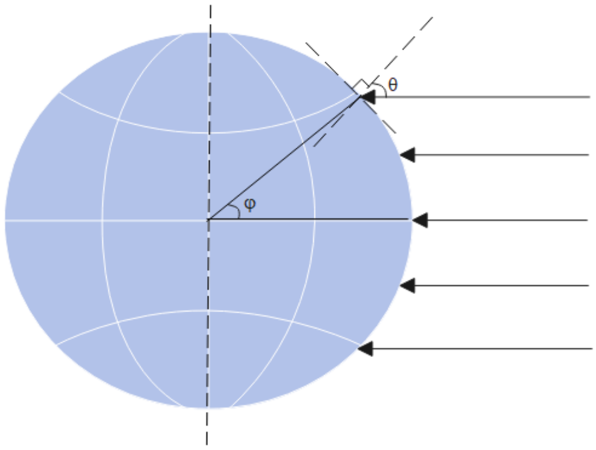

Figure 10 Diagram of solar irradiation

If the latitude is represented by φ, then we can consider that at the same time, the temperature of each region of the Earth shows roughly a functional relationship with $\sin\varphi$.

We use the data to plot the relationship between temperature, time, and latitude as shown below:

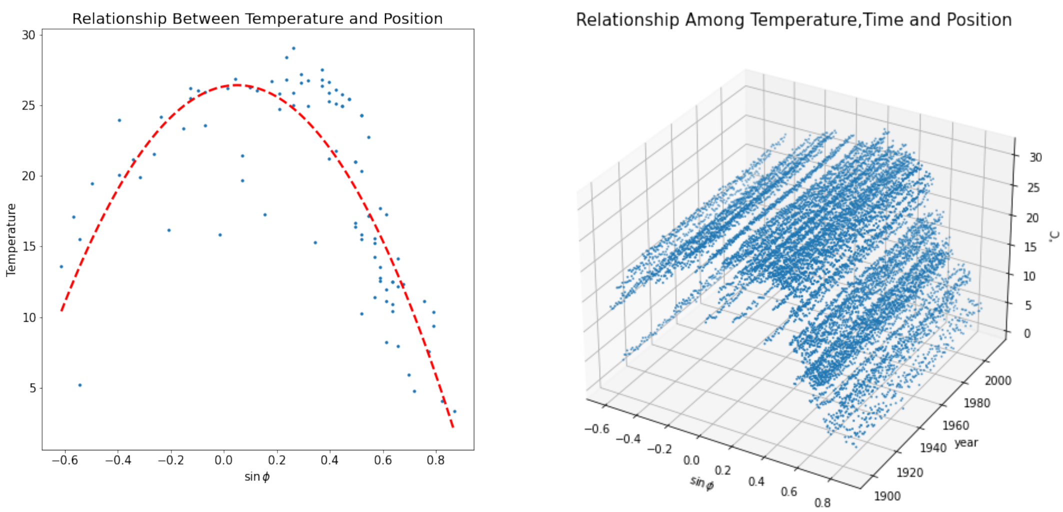

Figure 11 Relationship

From the figure we can see that temperature and latitude roughly present a quadratic relationship and approximate a quadratic surface relationship in the 3D image, we can use quadratic regression to fit the relationship between the three, and the expression is as follows:

$$T(\varphi, t) = p_0 + p_1 \sin\varphi + p_2 t + p_3 t \sin\varphi + p_4 \sin^2\varphi + p_5 t^2 \tag{14}$$

where T is the temperature, φ is the latitude, t is the time, and $p_i(i = 0, 1, 2, 3, 4, 5)$ is the coefficient.

6.1.2 *Pearson Coeffience*

Pearson's coefficient can be used to express the correlation between two time series, the closer its absolute value is to 1 the stronger the correlation and the closer it is to 0 the weaker the correlation. The Pearson coefficient formula is as follows:

$$\rho_{XY} = \frac{Cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}} \tag{15}$$

where $Cov(X,Y)$ represents the covariance of X and Y and $D(X)$ represents the variance of X.

6.2 Model Solving

6.2.1 *Problem (a)*

We fit the curves using the established quadratic regression model using Matlab's cftool and the results are as follows:

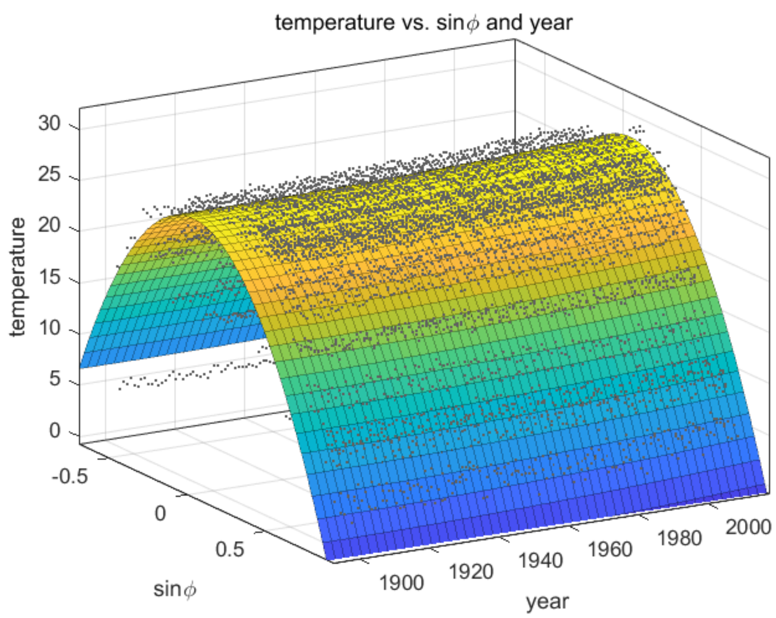

Figure 12 Quadratic Regression

$$T(\varphi,t) = 335+0.4908\sin\varphi-0.3246t+1.48\times10^{-3}t\sin\varphi-37.34\sin^2\varphi+8.539\times10^{-5}t^2 \tag{16}$$

The goodness-of-fit $R^2 = 0.713$ indicates a good fit.

6.2.2 *Problem (b)*

To analyze whether natural disasters affect global warming, we collected data and counted the number of major natural disasters occurring in each year from 2012 to 2022. There are four kinds of natural disasters: volcanic eruption, forest fire, flood and drought. The data are shown as follows:

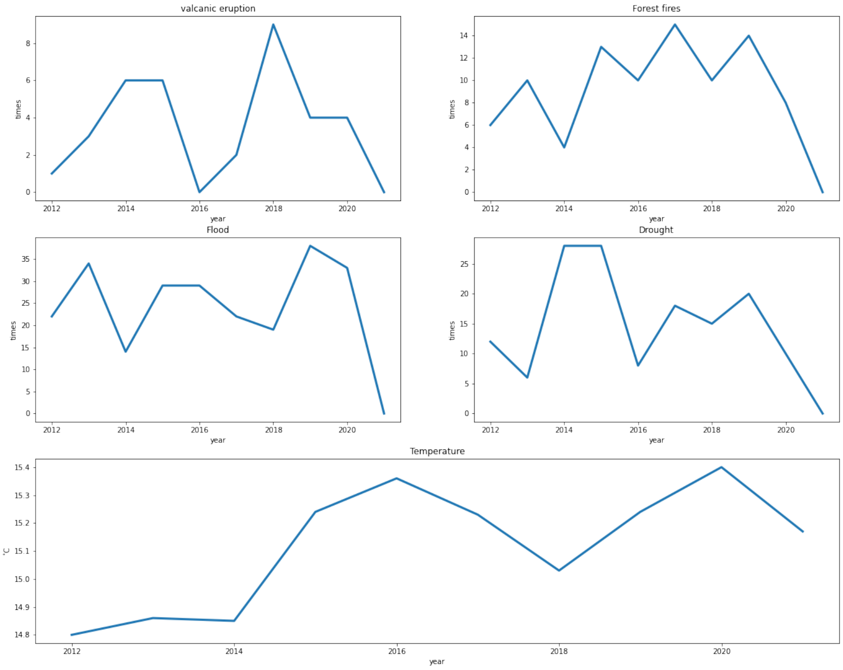

Figure 13 Disaster and Temperature Trends

As can be seen from the figure, there is no obvious correlation between the occurrence of natural disasters and the change of global temperature. The frequency of natural disasters shows an oscillating trend, while the global temperature shows an increasing trend. At the same time, we know that the outbreak of COVID-19 occurred in 2020, and the global temperature even decreased from 2020 to 2021, so we can judge that there is no obvious relationship between COVID-19 and global warming.

We use Pearson correlation coefficient for further analysis, and the results are as follows:

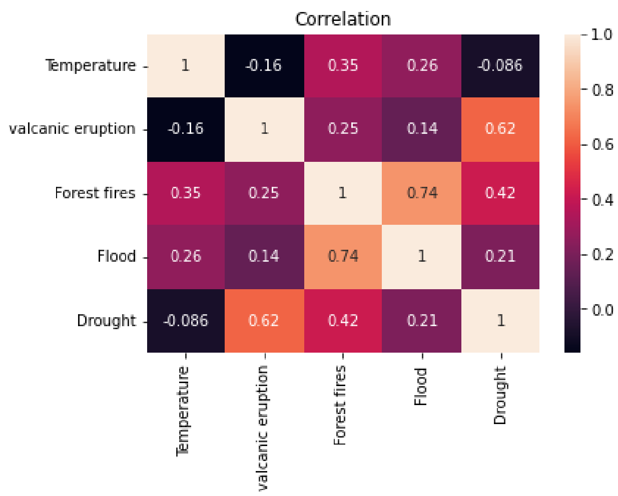

Figure 14 Pearson Coeffience Among Disasters and Temperature

As can be seen from the figure, the correlation coefficients between the four kinds of natural disasters and temperature are not high, which are -0.16, 0.35, 0.26 and -0.086 respectively. It can be concluded that there is no obvious relationship between natural disasters and global warming.

6.2.3 *Problem (c)*

According to the data, the main factor of global warming is greenhouse gas emissions[3].So we collected changes in greenhouse gas concentrations from 1978 to 2021.There are three types of greenhouse gases we collected:CO_2、CH_4、N_2O,The data is shown as follows：

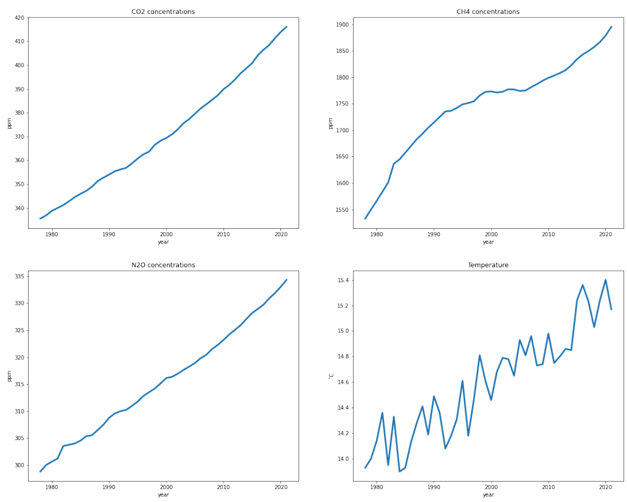

Figure 15 Greenhouse Gases and Temperature Trends

It is clear that all three greenhouse gases and global temperatures are on the rise, and a strong correlation can be judged between them.We used the Pearson coefficient for further analysis and the results are as follows:

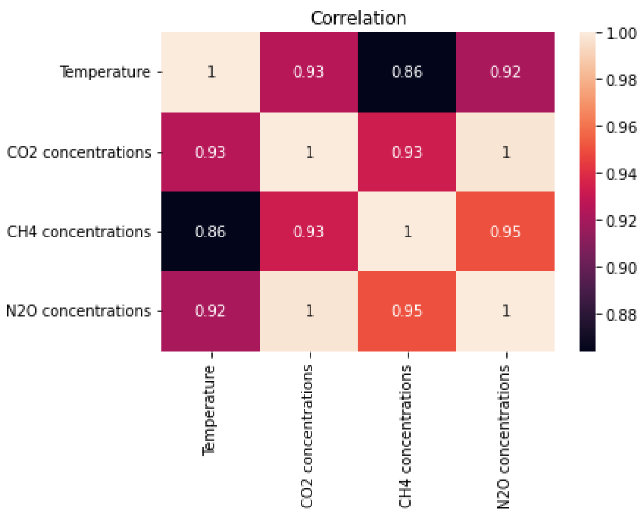

Figure 16 Pearson Coeffience Among Greenhouse Gases and Temperature

As can be seen from the figure, the correlation coefficients between the three greenhouse gases and temperature are very high, 0.93, 0.86, 0.92, respectively, and it can be concluded that greenhouse gases are the main cause of global warming.

6.2.4 *Problem (d)*

After analysis, we found that the most important factor in global warming is human factors.Among all the human factors, the most important factor is the excessive emission of greenhouse gases.Curbing global warming completely is a daunting challenge, but there are ways to mitigate it.

- **For each individual:** Human beings should change their consumption habits.Eat less meat because it takes a lot of energy to make and transport and eat more vegetables which takes less energy.;Drive less and cycle more.The most important source of greenhouse gases is the exhaust gas from cars, and trying not to drive is a very effective way to mitigate global warming.If you can't drive less, then you need to keep your car in good condition, because a poorly maintained car emits more greenhouse gases than a well-maintained car.
- **For businesses and factories:** Optimize production methods, adopt advanced production technologies (such as low-carbon energy technology, carbon sink technology, emission reduction technology), and reduce carbon emissions in the production process;Promote the use of carbon dioxide recycling technology, using carbon dioxide gas to produce chemical products or recycle and compress the carbon dioxide [4].
- **For government organizations:** Introduce agreements to limit carbon emissions.Just like the Kyoto Protocol, it binds individuals and businesses in the form of legal provisions;Promote transportation projects.Reduce traffic jams and reduce greenhouse gas emissions due to traffic jams;Adjusting the energy structure.Increase the use of new energy (such as natural gas, solar energy, etc.) and increase the proportion of new energy in the total energy;Shut down illegal enterprises.Providing financial support to clean energy companies to help them optimize their production methods;Expanding forest area.The growth of trees can absorb carbon dioxide through photosynthesis and convert it into oxygen, avoiding the continuous accumulation of carbon dioxide in the atmosphere, which is essential for slowing global warming.

Of course, the causes of global warming are not only greenhouse gases, but also other factors.For example, the destruction of the atmosphere leads to a hole in the ozone layer.The cushioning of sunlight from the Sun to the Earth decreases, and the surface receives more ultraviolet light from the sun.This is another major cause of global warming.The main cause of the ozone hole is the massive industrial use of freon gas, as well as human use of air conditioning containing melamine, fluoride emissions, resulting in the destruction of the ozone layer.In order to solve this global warming caused by the destruction of the ozone layer, humans should reduce the emission of fluoride-containing gases, or reduce the use of potentially polluting fluoride at the root.

VII. Evaluation of Strengths and Weaknesses

7.1 Strengths

- Our models have high accuracy, and can precisely forecast future temperature.
- We use many test methods to guarantee the stability of models.
- Our models take seasonal factors into account.

7.2 Weaknesses

- The quadratic regression model does not take the autocorrelation of the time series into account.
- We don't consider the effect of hidden factors.

Our Findings and Future Work in Global Warming

To:APMCM organizing committee
From: Team 2211488

Through this study, we found that global warming is a very serious problem and will become more serious with the passage of time.In 2173, our global average temperature could reach 20°C, which is a very scary concept.If the average temperature is 20 degrees throughout the year, given Antarctica's huge ice sheet, we have reason to worry about whether we will be flooded by the ocean.

In addition to this, we also found the strang relationship between natural disasters and the global average temperature – –natural disasters do not directly affect the global temperature, but the resulting greenhouse gas release, ozone layer destruction and other problems will indirectly lead to global temperature increase, which in turn will lead to more natural disasters.Therefore, they are indirectly positively correlated.If it continues to evolve and nothing is done to stop it, the number of future natural disasters will increase and the growth rate will also increase.

And, of course, we found the main contributor to global warming – –greenhouse gases. In recent years, we can see that greenhouse gas emissions have continued to increase and have not even gone down at all. This has also led to an increase in the global average temperature. It is almost certain that greenhouse gases are a major contributor to global warming.

We suggest that there are three ways to solve the problem of global warming in the future.The first is to reduce greenhouse gas emissions.By combining individual green travel, factory emission reduction and national control, we can reduce greenhouse gas emissions through individuals, factories and countries.The second aspect, from the scientific and technological aspect.If possible, new technologies that are more advanced and more environmentally friendly can be developed in the future, rather than older ones that are less environmentally friendly at the present stage.The third aspect is to start with environmental protection.Human bings should respect nature and adhere to the concept of ”Clear waters and green mountains are as good as mountains of gold and silver.” and give the top priority to ecological and environmental issues.

We believe that if we have the joint efforts of all mankind, the problem of global warming can be solved.

VIII. References

[1] https://www.lwchenxin.com/30058.html

[2] https://www.britannica.com/science/global-warming

[3] Lashof, D. A. , and D. R. Ahuja . ”Relative contributions of greenhouse gas emissions to global warming.” Nature 344.6266(1990):529-531.

[4] https://new.qq.com/omn/20220226/20220226A076PU00.html

[5] http://berkeleyearth.org/data/

[6] Cox, et al. ”Acceleration of global warming due to carbon-cycle feedbacks in a coupled climate model. ” Nature (2000).

[7] Lynch, A. H. , and W. Wu . ”Impacts of Fire and Warming on Ecosystem Uptake in the Boreal Forest.” Journal of Climate 13.13(2000):2334-2338.

IX. Appendix

```
[ ]: ###### MK Test ######
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
[ ]: global_data = pd.read_csv('./GlobalAverageTemperature-M.csv')
global_data['Temperature'] += 13.83
index = (global_data['Year'].values - global_data['Year'].values % 100) / 100 +␣
↪(global_data['Year'].values % 100) / 12
index1 = index >= 2012
index2 = (2002 <= index) & ~index1
```

```
[ ]: plt.plot(index[index1], global_data.loc[index1, 'Temperature'], c='y')
plt.plot(index[~index1], global_data.loc[~index1, 'Temperature'], c='b',␣
↪alpha=0.6)
plt.title('Global Average Temperature');
plt.xlabel('year')
plt.ylabel('$^{\circ}$C')
plt.savefig('10')
```

```
[ ]: k = MK(global_data.loc[index1 | index2, 'Temperature'])
index[index1 | index2][k]
```

```
[ ]: def MK(data):
    data = np.array(data)
    n = data.shape[0]
    Sk = [0]
    UFk = [0]
    s = 0
    Exp_value = [0]
    Var_value = [0]
    for i in range(1,n):
        for j in range(i):
            if data[i] > data[j]:
                s = s + 1
            else:
                s = s+0
        Sk.append(s)
        Exp_value.append((i+1)*(i+2)/4 )
        Var_value.append((i+1)*i*(2*(i+1)+5)/72 )
        UFk.append((Sk[i]-Exp_value[i])/np.sqrt(Var_value[i]))
    Sk2 = [0]
    UBk = [0]
    UBk2 = [0]
    s2 =  0
    Exp_value2 = [0]
    Var_value2 = [0]
```

```
    dataT = list(reversed(data))
    for i in range(1,n):
        for j in range(i):
            if dataT[i] > dataT[j]:
                s2 = s2 + 1
            else:
                s2 = s2 + 0
        Sk2.append(s2)
        Exp_value2.append((i+1)*(i+2)/4 )
        Var_value2.append((i+1)*i*(2*(i+1)+5)/72 )
        UBk.append((Sk2[i]-Exp_value2[i])/np.sqrt(Var_value2[i]))
        UBk2.append(-UBk[i])
    UBkT = list(reversed(UBk2))
    diff = np.array(UFk) - np.array(UBkT)
    K = list()
    for k in range(1,n):
        if diff[k-1]*diff[k]<0:
            K.append(k)
    plt.plot(range(1,n+1) ,UFk  ,label='UFk')
    plt.plot(range(1,n+1) ,UBkT ,label='UBk')
    plt.ylabel('UFk-UBk')
    x_lim = plt.xlim()
    plt.plot(x_lim,[-1.96,-1.96],'m--',color='r')
    plt.plot(x_lim,[  0  ,   0  ],'m--')
    plt.plot(x_lim,[+1.96,+1.96],'m--',color='r')
    plt.legend(loc=2)
    plt.title('Mann-Kendall Test')
    plt.savefig('mk test')
    plt.show()
    return K
```

[ ]:
```
###### Prophet ######
from prophet import Prophet
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
```

[ ]:
```
data=pd.read_csv('GlobalAverageTemperature-M.csv')
data=data[['Date','Temperature']]
data['Date']=pd.to_datetime(data['Date']).dt.date
data['Temperature']=data['Temperature']+13.83
data=data.rename(columns={'Date':'ds','Temperature':'y'})
data=data.iloc[120:,:]
```

[ ]:
```
data2=pd.read_csv('GlobalAverageTemperature-Y.csv')
data2=data2[['Date','Temperature']]
```

```
data2['Date']=pd.to_datetime(data2['Date']).dt.date
data2['Temperature']=data2['Temperature']+13.83
data2=data2.rename(columns={'Date':'ds','Temperature':'y'})
data2=data2.iloc[10:]
```

[ ]:
```
plt.figure(figsize=(9,6))
plt.title("Global Average Temperature",fontsize=14)
plt.xlabel("Date",fontsize=14)
plt.ylabel("$^{\circ}$C",fontsize=14)
sns.lineplot(x="ds",y="y",data=data,linewidth=0.5,label='Monthly')
sns.lineplot(x="ds",y="y",data=data2,linewidth=1.5,label='Yearly',color='red')
plt.legend()
plt.savefig("Tem.png",bbox_inches='tight')
plt.show()
```

[ ]:
```
model=Prophet(growth = "linear",
                yearly_seasonality = True,
                weekly_seasonality = False,
                daily_seasonality = False,
                seasonality_mode='additive')
model.fit(data)
future = model.make_future_dataframe(periods=1826, freq='MS')
forecast = model.predict(future)
plt.figure()
model.plot(forecast)
plt.title("Prophet Model",fontsize=18)
plt.xlabel("Date",fontsize=18)
plt.ylabel("$^{\circ}$C",fontsize=18)
plt.xticks(size=14)
plt.yticks(size=14)
plt.savefig("prophet.png", bbox_inches='tight')
plt.show()
```

[ ]:
```
y_hat=np.array(forecast['yhat'])
year_average=[]
for i in range(int(len(y_hat)/12+0.5)):
    if i!=int(len(y_hat)/12):
        year_average.append(y_hat[12*i:12*(i+1)].mean())
    else:
        year_average.append(y_hat[12*i:].mean())
minlen = len(y_hat)/12
index = np.linspace(1890,2200,len(y_hat))
year_index=[*range(int(min(index)),int(min(index)+len(year_average)))]
plt.plot(year_index, year_average)
plt.xlabel('year')
plt.ylabel('$^{\circ}$C')
plt.title('Global Temperature')
```

```
print('Global temperature will reach 20 °C on {}!'.format(year_index[(np.
  ↪array(year_average) > 20).tolist().index(True)]))
```

[ ]:
```
###### ARIMA ######
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
from arch.unitroot import ADF
from statsmodels.stats.diagnostic import acorr_ljungbox
import statsmodels.api as sm
```

[ ]:
```
data = pd.read_csv('./2022_APMCM_C_Data.csv', encoding='gbk')
```

[ ]:
```
kabul = data.loc[data['City'].isin(['Kabul']), ['AverageTemperature']]
null_index = np.array(kabul.index)[kabul.isnull().values.ravel()]
kabul_data = kabul.iloc[null_index[-2]+1:null_index[-1]]
```

[ ]:
```
kabul_data['diff1'] = kabul_data['AverageTemperature'].diff(1)
kabul_data['diff2'] = kabul_data['diff1'].diff(1)
kabul_data[['AverageTemperature','diff1']].plot(subplots=True,␣
  ↪figsize=(18,12),title='Difference Figure')
plt.xlabel('month')
plt.ylabel('$^{\circ}$C');
plt.savefig('diff_fig')
```

[ ]:
```
print(ADF(kabul_data.diff1[1:]))
```

[ ]:
```
print(acorr_ljungbox(kabul_data.diff1[1:], lags = [i for i in␣
  ↪range(1,12)],boxpierce=True))
```

[ ]:
```
sm.graphics.tsa.plot_acf(kabul_data.AverageTemperature)
plt.title('Autocorrelation(d=0)');
```

[ ]:
```
sm.graphics.tsa.plot_pacf(kabul_data.AverageTemperature)
plt.title('Partial Autocorrelation(d=0)');
```

[ ]:
```
sm.graphics.tsa.plot_acf(kabul_data.diff1[1:])
plt.title('Autocorrelation(d=1)');
plt.savefig('diff_acf')
```

[ ]:
```
sm.graphics.tsa.plot_pacf(kabul_data.diff1[1:])
plt.title('Partial Autocorrelation(d=1)');
plt.savefig('diff_pacf')
```

[ ]:
```
import itertools
import warnings
p = q = range(0, 2)
```

```
d = range(1,2)
pdq = list(itertools.product(p, d, q))
seasonal_pdq = [(x[0], x[1], x[2], 12) for x in list(itertools.product(p, d,␣
 ↪q))]
warnings.filterwarnings("ignore")
for param in pdq:
    for param_seasonal in seasonal_pdq:
        try:
            mod = sm.tsa.statespace.SARIMAX(kabul_data["AverageTemperature"],
                                            order=param,
                                            seasonal_order=param_seasonal,
                                            enforce_stationarity=False,
                                            enforce_invertibility=False)
            results = mod.fit()
            print('ARIMA{}x{} - AIC:{}'.format(param, param_seasonal, results.
 ↪aic))
        except:
            continue
```

[ ]:

```
mean_list = []
for item in data['City'].unique():
    region = data.loc[data['City'].isin([item]), ['AverageTemperature']]
    null_index = np.array(region.index)[region.isnull().values.ravel()]
    if len(null_index) >= 2:
        region_data = data.loc[null_index[-2]+1:null_index[-1]-1,␣
 ↪['AverageTemperature']]
    else:
        region_data = data.loc[region.index[0]:region.index[-1]-1,␣
 ↪['AverageTemperature']]
    mod = sm.tsa.statespace.SARIMAX(region_data["AverageTemperature"],
                                    order=(1,1,1),
                                    seasonal_order=(0,1,1,12),
                                    enforce_stationarity=False,
                                    enforce_invertibility=False).
 ↪fit()
    pred = mod.predict(0, len(region_data)+2248)
    date = data.loc[pred.index[0],'dt']
    if '-' in date:
        char = '-'
    elif '/' in date:
        char = '/'
    mean=[]
    for i in range(int(len(pred)/12+0.5)):
        if i !=int(len(pred)/12):
            mean.append(pred[12*i:12*(i+1)].mean())
        else:
            mean.append(pred[12*i:].mean())
```

```
    mean_list.append(mean)
    if  item == 'Kabul':
        index = np.array([*range(len(pred))])/12+int(date.
 ↪split(char)[0])+int(date.split(char)[1])/12
        year_index=[*range(int(min(index)),int(min(index)+len(mean)))]
        plt.plot(index,pred,alpha=0.5)
        plt.scatter(index[:len(region_data)], region_data['AverageTemperature'].
 ↪values, c='k', s=1)
        plt.plot(year_index,mean,linewidth=3)
        plt.title('Predicted Average Temperature of '+item)
        plt.xlabel('year')
        plt.ylabel('$^{\circ}$ C');
        plt.show()
```

```
[ ]: global_data = pd.read_csv('./GlobalAverageTemperature-M.csv')
global_data['Temperature'] += 13.83
mod = sm.tsa.statespace.SARIMAX(global_data["Temperature"],
                                            order=(1,1,1),
                                            seasonal_order=(0,1,1,12),
                                            enforce_stationarity=False,
                                            enforce_invertibility=False).fit()
pred = mod.predict(0,len(global_data) + 2126)
```

```
[ ]: mean = []
for i in range(int(len(pred)/12+0.5)):
        if i !=int(len(pred)/12):
            mean.append(pred[12*i:12*(i+1)].mean())
        else:
            mean.append(pred[12*i:].mean())
minlen = len(pred)/12
index = np.linspace(1880,2200,len(pred))
year_index=[*range(int(min(index)),int(min(index)+len(mean)))]
plt.plot(year_index, mean)
plt.xlabel('year')
plt.ylabel('$^{\circ}$C')
plt.title('Global Temperature')
print('Global temperature will reach 20 °C on {}!'.format(year_index[(np.
 ↪array(mean) > 20).tolist().index(True)]))
plt.savefig('ARIMA mean')
```

```
[ ]: plt.plot(index,pred,alpha=0.5)
plt.scatter(index[:len(global_data)], global_data['Temperature'].values, c='k',␣
 ↪s=1)
plt.plot(year_index,mean,linewidth=2)
plt.title('Predicted Average Temperature of Global')
plt.xlabel('year')
```

```
plt.ylim([12,22])
plt.ylabel('$^{\circ}$ C');
plt.grid()
plt.savefig('pred_fig')
plt.show()
```

[ ]:
```
###### SES ######
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.holtwinters import (ExponentialSmoothing,
                                         SimpleExpSmoothing,
                                         Holt)
%matplotlib inline
```

[ ]:
```
data=pd.read_csv('GlobalAverageTemperature-M.csv')
data=data[['Date','Temperature']]
data['Date']=pd.to_datetime(data['Date']).dt.date
data['Temperature']=data['Temperature']+13.83
data=data.rename(columns={'Date':'ds','Temperature':'y'})
data=data.iloc[120:,:]
data_train=data.loc[:1644]
data_test=data.loc[1644:]
# data_train=data.loc[:120]
# data_test=data.loc[120:]
data=pd.Series(data=np.array(data['y']),index=data['ds'])
data_train=data_train.reset_index(drop=True)
data_test=data_test.reset_index(drop=True)
data_train=pd.Series(data=np.array(data_train['y']),index=data_train['ds'])
data_test=pd.Series(data=np.array(data_test['y']),index=data_test['ds'])
```

[ ]:
```
fit2 = ExponentialSmoothing(
    data,
    seasonal_periods=12,
    trend="mul",
    seasonal="mul",
    use_boxcox=False,
    initialization_method="estimated",
).fit(smoothing_level=0.1,
      smoothing_slope=0.5)
fcast2=fit2.forecast(2510)
data.plot(color='b',
          title='Exponential Smoothing',
          label='Actual',
          legend=True,
          figsize=(9,6),
          xlabel="Year",
```

```
          ylabel='$^{\circ}C$',
          linewidth=0.5)
fcast2.plot(color='r', legend=True, label="Holt’s Winters",linewidth=0.
 ↪8,alpha=0.4)
fit2.fittedvalues.plot(color='r',xlabel="Year",alpha=0.8)
plt.grid()
plt.savefig('Holt Winters.png',bbox_inches='tight')
```

[ ]:
```
###### Validate ######
from prophet import Prophet
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

[ ]:
```
data=pd.read_csv('GlobalAverageTemperature-M.csv')
data=data[['Date','Temperature']]
data['Date']=pd.to_datetime(data['Date']).dt.date
data['Temperature']=data['Temperature']+13.83
data=data.rename(columns={'Date':'ds','Temperature':'y'})
data=data.iloc[120:,:]

data_train=data.loc[:1644]
data_test=data.loc[1644:]
data_train=data_train.reset_index(drop=True)
data_test=data_test.reset_index(drop=True)
```

[ ]:
```
model=Prophet(growth = "linear",
                yearly_seasonality = True,
                weekly_seasonality = False,
                daily_seasonality = False,
                seasonality_mode='additive')
model.fit(data_train)
yhat=model.predict(data_test)
yhat=np.array(yhat['yhat'])
ytrue=data_test['y']
mape1=0
mse1=0
#calculate MAPE
for i in range(len(yhat)):
    mape1+=abs((ytrue[i]-yhat[i])/ytrue[i])
mape1=mape1/(len(yhat))
for i in range(len(yhat)):
    mse1+=(ytrue[i]-yhat[i])**2
mse1=mse1/(len(yhat))
```

[ ]:
```
import statsmodels.api as sm
mod = sm.tsa.statespace.SARIMAX(data_train['y'],
```

```
                                            order=(1,1,1),
                                            seasonal_order=(0,1,1,12),
                                            enforce_stationarity=False,
                                            enforce_invertibility=False).fit()
pred = mod.predict(len(data_train),len(data_train)+len(data_test)-1)
pred=np.array(pred)
mape2=0
mse2=0
for i in range(len(pred)):
    mape2+=abs((ytrue[i]-pred[i])/ytrue[i])
mape2=mape2/(len(pred))
for i in range(len(yhat)):
    mse2+=(ytrue[i]-pred[i])**2
mse2=mse2/(len(yhat))
```

[ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.holtwinters import (ExponentialSmoothing,
                                         SimpleExpSmoothing,
                                         Holt)
data=pd.read_csv('GlobalAverageTemperature-M.csv')
data=data[['Date','Temperature']]
data['Date']=pd.to_datetime(data['Date']).dt.date
data['Temperature']=data['Temperature']+13.83
data=data.rename(columns={'Date':'ds','Temperature':'y'})
data=data.iloc[120:,:]
data_train=data.loc[:1644]
data_test=data.loc[1644:]
data=pd.Series(data=np.array(data['y']),index=data['ds'])
data_train=data_train.reset_index(drop=True)
data_test=data_test.reset_index(drop=True)
data_train=pd.Series(data=np.array(data_train['y']),index=data_train['ds'])
data_test=pd.Series(data=np.array(data_test['y']),index=data_test['ds'])
fit2 = ExponentialSmoothing(
    data_train,
    seasonal_periods=12,
    trend="add",
    seasonal="add",
    use_boxcox=False,
    initialization_method="estimated",
).fit(smoothing_level=0.3,
      smoothing_slope=0.42)
fcast2=fit2.forecast(len(data_test))
fcast2=np.array(fcast2)
mape3=0
mse3=0
```

```
for i in range(len(fcast2)):
    mape3+=abs((ytrue[i]-fcast2[i])/ytrue[i])
mape3=mape3/(len(fcast2))
for i in range(len(yhat)):
    mse3+=(ytrue[i]-fcast2[i])**2
mse3=mse3/(len(yhat))
```

[ ]:
```
ape1=[]
ape2=[]
ape3=[]
se1=[]
se2=[]
se3=[]
for i in range(len(yhat)):
    ape1.append((ytrue[i]-yhat[i])/ytrue[i])
    se1.append((ytrue[i]-yhat[i])**2)
    ape2.append((ytrue[i]-pred[i])/ytrue[i])
    se2.append((ytrue[i]-pred[i])**2)
    ape3.append((ytrue[i]-fcast2[i])/ytrue[i])
    se3.append((ytrue[i]-fcast2[i])**2)
plt.figure(figsize=(10,6))
plt.title("Residual Percentage Error",fontsize=16)
plt.xlabel("Month",fontsize=16)
plt.ylabel("Value",fontsize=16)
plt.xticks(size=14)
plt.yticks(size=14)
plt.plot(range(len(ytrue)),ape1,label="Prophet",c='orangered',alpha=0.
 ↪8,linewidth=2)
plt.plot(range(len(ytrue)),ape2,label="ARIMA",c='royalblue',alpha=1)
plt.plot(range(len(ytrue)),ape3,label="Holt Winters",c='g',alpha=0.7)
plt.legend(fontsize=12)
plt.savefig("Residual Percentage Error.png",bbox_inches='tight')
plt.show()
```

[ ]:
```
plt.figure(figsize=(10,6))
plt.title("Square Error",fontsize=16)
plt.xlabel("Month",fontsize=16)
plt.ylabel("Value",fontsize=16)
plt.xticks(size=14)
plt.yticks(size=14)
plt.plot(range(len(ytrue)),se1,label="Prophet",c='orangered',alpha=0.
 ↪8,linewidth=2)
plt.plot(range(len(ytrue)),se2,label="ARIMA",c='royalblue',alpha=1)
plt.plot(range(len(ytrue)),se3,label="Holt Winters",c='g',alpha=0.7)
plt.legend(fontsize=12)
plt.savefig("Square Error.png",bbox_inches='tight')
plt.show()
```

[ ]:
```
###### Regression ######
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.api as sm
```

[ ]:
```
data = pd.read_csv('./2022_APMCM_C_Data.csv', encoding='gbk')
```

[ ]:
```
mean_list = []
angle = []
minlen = 10000
for item in data['City'].unique():
    region = data[data['City'].isin([item])]
    region_temp = region['AverageTemperature']
    null_index = np.array(region_temp.index)[region_temp.isnull().values.
 ↪ravel()]
    if len(null_index) >= 2:
        region_data = data.loc[null_index[-2]+1:null_index[-1]-1,␣
 ↪['AverageTemperature']]
    else:
        region_data = data.loc[region_temp.index[0]:region_temp.index[-1]-1,␣
 ↪['AverageTemperature']]
    if len(region_data) == 0:
        continue
    date = data.loc[region_data.index[0],'dt']
    if '-' in date:
        char = '-'
    elif '/' in date:
        char = '/'
    mean=[]
    for i in range(int(len(region_data)/12+0.5)):
        if i !=int(len(region_data)/12):
            mean.append(region_data[12*i:12*(i+1)].mean())
        else:
            mean.append(region_data[12*i:].mean())
    mean_list.append(mean)
    if len(mean) < minlen :
        minlen = len(mean)
    if mean[-2].values != 0:
        string = region.loc[region.index[0], 'Latitude']
        theta = float(string[:-1]) * np.pi / 180
        if string[-1] == 'N':
            angle.append(np.sin(theta))
        elif string[-1] == 'S':
            angle.append(-1 * np.sin(theta))
```

```
[ ]: time = [*range(2013 - minlen, 2013)]
     temp = np.zeros([len(angle), minlen])
     for i in range(len(angle)):
         temp[i, :] = mean_list[i][-int(minlen):]
```

```
[ ]: plt.figure(figsize=(10,10))
     plt.scatter(angle, temp[:, 0], s=10)
     fit = np.polyfit(angle, temp[:, 0], 2)
     xx = np.linspace(min(angle),max(angle),100)
     yy = np.polyval(fit, xx)
     plt.plot(xx, yy, 'r--', linewidth=3)
     plt.xlabel('$\sin\phi$',size=15)
     plt.ylabel('Temperature',size=15)
     plt.xticks(size=15)
     plt.yticks(size=15)
     plt.title('Relationship Between Temperature and Position',size=20);
     plt.savefig('rela')
```

```
[ ]: from mpl_toolkits.mplot3d import Axes3D
     fig = plt.figure(figsize=(15, 8))
     ax = fig.gca(projection='3d')
     axisx, axisy = np.meshgrid(angle, time)
     ax.scatter(axisx, axisy, temp.T, s=1)
     ax.set_xlabel('$\sin\phi$')
     ax.set_ylabel('year')
     ax.set_zlabel('$^{\circ}$C');
     ax.set_title('Relationship Among Temperature,Time and Position', size=15)
     plt.savefig('three',bbox_inches='tight',pad_inches=0.0)
     from scipy.io import savemat
     savemat('a_t_t.mat', {'axisx':axisx,'axisy':axisy,'temp':temp.T})
```

```
[ ]: ###### Correlation ######
     import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import seaborn as sns
```

```
[ ]: global_temp = pd.read_csv('./GlobalAverageTemperature-Y.csv')
     global_temp['Temperature'] += 13.83
```

```
[ ]: index = (global_temp['Year'] >= 2012)
     temp = global_temp.loc[index, 'Temperature']
     disaster_df = pd.DataFrame(temp)
     disaster = pd.read_excel('./disaster.xlsx')
     for dis in disaster['disasters'].unique():
         array = np.zeros(10)
         index = disaster['disasters'] == dis
```

```
    date = disaster.loc[index, 'start date']
    for dt in date:
        year = int(str(dt).split('-')[0])
        array[year - 2012] += 1
    disaster_df[dis] = array
disaster_df.index = range(2012, 2022)
corr_mat = disaster_df.corr()
sns.heatmap(corr_mat, annot=True)
plt.title('Correlation');
plt.savefig('dis_corr', bbox_inches='tight')
```

[ ]:
```
plt.figure(figsize=(20,16))
ax1 = plt.subplot(321)
ax1.plot(disaster_df.index, disaster_df['valcanic eruption'], linewidth=3)
ax1.set_xlabel('year')
ax1.set_ylabel('times')
ax1.set_title('valcanic eruption')
ax2 = plt.subplot(322)
ax2.plot(disaster_df.index, disaster_df['Forest fires'], linewidth=3)
ax2.set_xlabel('year')
ax2.set_ylabel('times')
ax2.set_title('Forest fires')
ax3 = plt.subplot(323)
ax3.plot(disaster_df.index, disaster_df['Flood'], linewidth=3)
ax3.set_xlabel('year')
ax3.set_ylabel('times')
ax3.set_title('Flood')
ax4 = plt.subplot(324)
ax4.plot(disaster_df.index, disaster_df['Drought'], linewidth=3)
ax4.set_xlabel('year')
ax4.set_ylabel('times')
ax4.set_title('Drought')
ax5 = plt.subplot(313)
ax5.plot(disaster_df.index, disaster_df['Temperature'], linewidth=3)
ax5.set_xlabel('year')
ax5.set_ylabel('$^{\circ}$C')
ax5.set_title('Temperature');
plt.savefig('dis_t')
```

[ ]:
```
index = (global_temp['Year'] >= 1978)
temp = global_temp.loc[index, 'Temperature']
green_df = pd.DataFrame(temp)
green = pd.read_excel('./greeenhouse gases.xlsx')
cls = ['CO2 concentrations', 'CH4 concentrations', 'N2O concentrations']
for item in cls:
    index = green['Year'].values >= 1978
    index[-1] = False
```

```
    green_df[item] = green.loc[index, item].values
green_df.index = range(1978, 2022)
corr_mat = green_df.corr()
sns.heatmap(corr_mat, annot=True)
plt.title('Correlation');
plt.savefig('gre_corr', bbox_inches='tight')
```

[ ]:
```
plt.figure(figsize=(20,16))
ax1 = plt.subplot(221)
ax1.plot(green_df.index, green_df['CO2 concentrations'], linewidth=3)
ax1.set_xlabel('year')
ax1.set_ylabel('ppm')
ax1.set_title('CO2 concentrations')
ax2 = plt.subplot(222)
ax2.plot(green_df.index, green_df['CH4 concentrations'], linewidth=3)
ax2.set_xlabel('year')
ax2.set_ylabel('ppm')
ax2.set_title('CH4 concentrations')
ax3 = plt.subplot(223)
ax3.plot(green_df.index, green_df['N2O concentrations'], linewidth=3)
ax3.set_xlabel('year')
ax3.set_ylabel('ppm')
ax3.set_title('N2O concentrations')
ax4 = plt.subplot(224)
ax4.plot(green_df.index, green_df['Temperature'], linewidth=3)
ax4.set_xlabel('year')
ax4.set_ylabel('$^{\circ}$C')
ax4.set_title('Temperature');
plt.savefig('gre_t')
```