Final Project Report
Capstone 3 Time Series Analysis: When Will The Average Temperature Exceed The 1.5°C Threshold

Table of Contents

Introduction

You may be wondering what the 1.5°C threshold is exactly. Essentially, the IPCC (Intergovernmental Panel on Climate Change) has stated that a 1.5°C increase in global average temperature from pre-industrial times will lead to various different climate catastrophes such as extreme heatwaves and rising ocean levels that would threaten the extinction of many different species such as the coral reef population. As such, the 1.5°C threshold is like a danger limit that is set on the average temperature such that efforts need to be made to control the increase in average temperature to be under 1.5°C from pre-industrial times.

The problem that our world is encountering is that due to global warming and other human activities (i.e. gas emissions), the increase in average temperature is approaching the 1.5°C danger line. As such, the 2015 Paris Agreement was signed to urge countries to contribute in controlling and limiting the global average temperature rise to under 1.5°C through human intervention.

Thus, it is only natural to wonder that without any human interference, when exactly will the 1.5°C threshold be reached under current circumstances in order to comprehend how urgent the situation really is. As such, this capstone project aims to perform a time series analysis on average temperature data and forecast exactly when the 1.5°C threshold is reached and consistently exceeded.

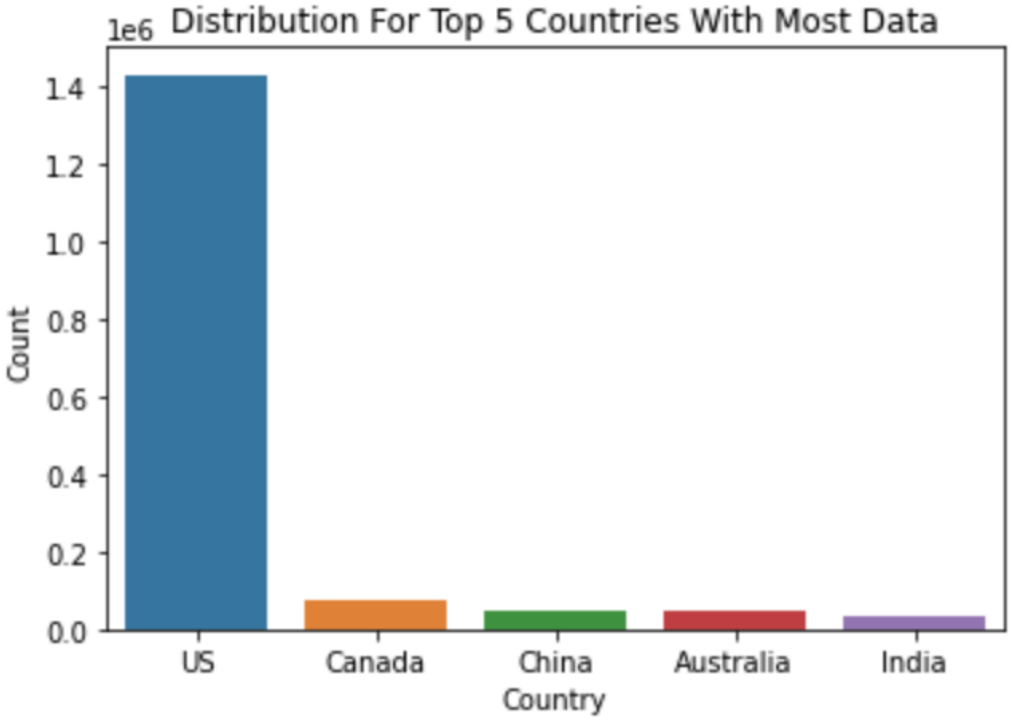
Dataset Information

The data used for this capstone project is from the *Daily Temperature of Major Cities* from Kaggle (https://www.kaggle.com/sudalairajkumar/daily-temperature-of-major-cities) provided by SRK. This dataset contains information regarding the daily average temperatures from 321 major cities from around 125 countries. There are approximately 2.9 million rows of data in this dataset.

The dataset's 8 features include the following:

- Region (continents)
- Country
- State (for United States)
- City
- Month
- Day
- Year
- AvgTemperature

The data distribution is skewed towards the US as it is comprised of approximately 1.4 million rows from the dataset, which is a lot more compared to other countries as shown:



However, there are 1.5 millions rows for non-US average temperature data that would help counterbalance the weight of the US such that the dataset can still be a representation of global average temperature.

Data Wrangling

State

From performing data wrangling on the dataset, there appears to be many null values for the *State* feature, and since it is not a necessary feature for calculating the global temperature, we remove it from the dataset.

AvgTemperature

The data in the AvgTemperature feature was originally stored as Fahrenheit. It had 79672 rows with an average temperature of -99 degrees Fahrenheit which is unrealistic, so we assume it is equivalent to a placeholder for null values and remove them from the dataset accordingly. We then convert the temperature data from Fahrenheit to Celsius since we want to match the unit with the temperature threshold of 1.5 degree Celsius.

Year

Counting the number of existing rows for every year from 1995 to 2020, there are only ~38,000 rows for the year 2020 while every other year has over 100,000 rows respectively. Due to this imbalance in distribution, we decide to exclude the data involving 2020 from the dataset as it has too much missing data compared to the other years.

Season

Since average temperatures tend to vary during each particular season, seasons may be a useful feature for EDA, and so we can create a season column. For simplicity, instead of using the exact dates as the reference point, we label the rows with months for January – March as Winter, April – June as Spring, July – September as Summer, and October – December as Fall.

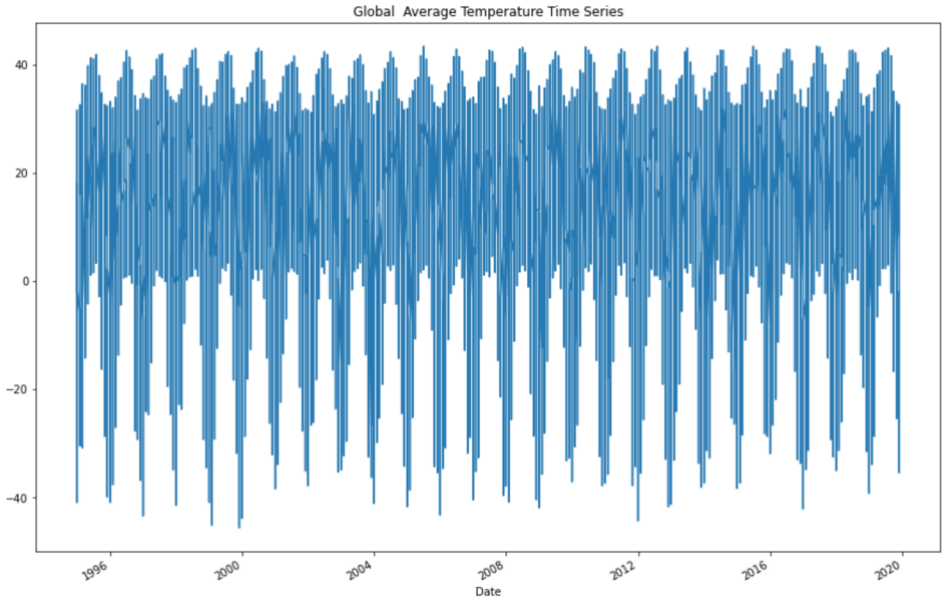
Latitude & Longitude

Additionally, average temperatures can vary depending on the location of a city as well, and so the latitude and longitude could also be useful features for EDA. As such, latitude and longitude values are added to the dataset using geocode based on the city.

Exploratory Data Analysis

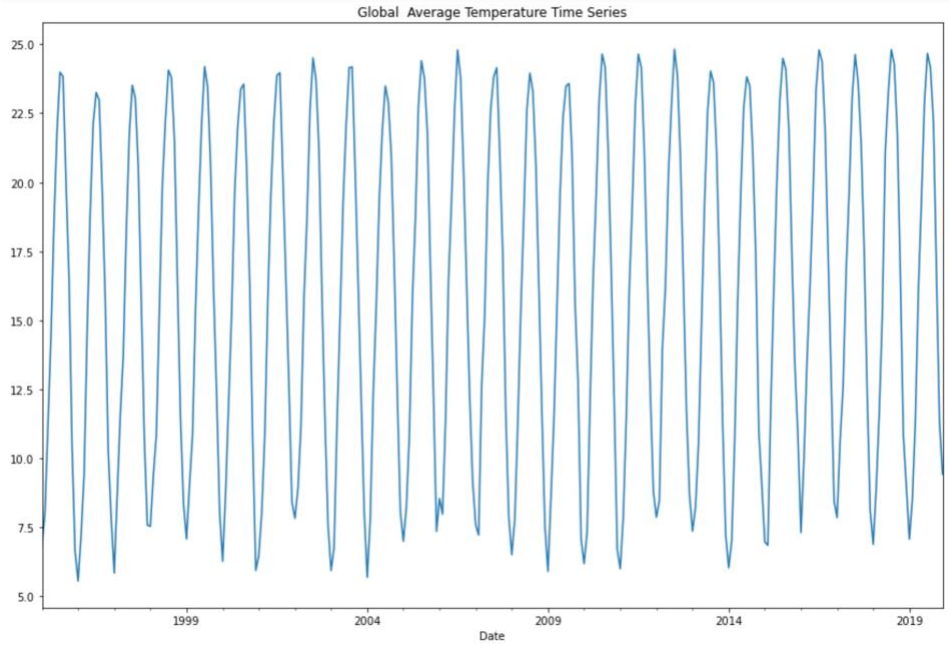
Initial Data Visualization:

Plotting the global daily average temperature (of all locations) as a time series looks like the following:



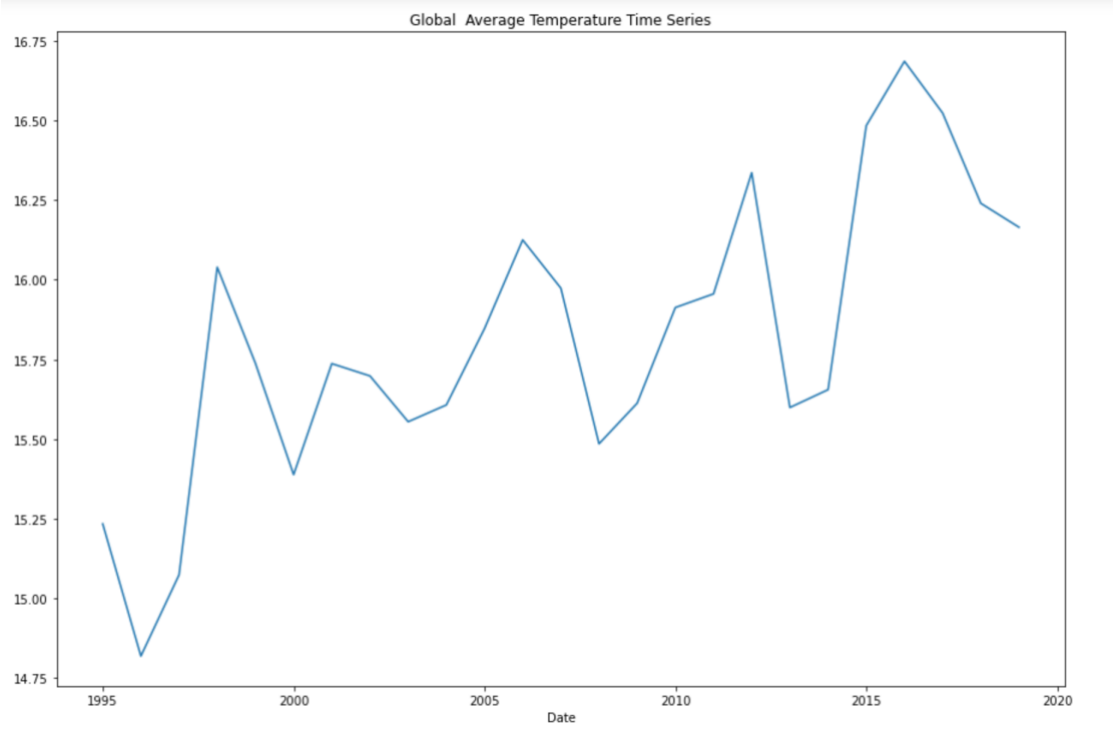
This distribution appears to be very cluttered and difficult to notice a trend or pattern. As such, we can try to group the average temperature data by month instead of daily.

Plotting the global monthly average temperature as a time series looks like the following:



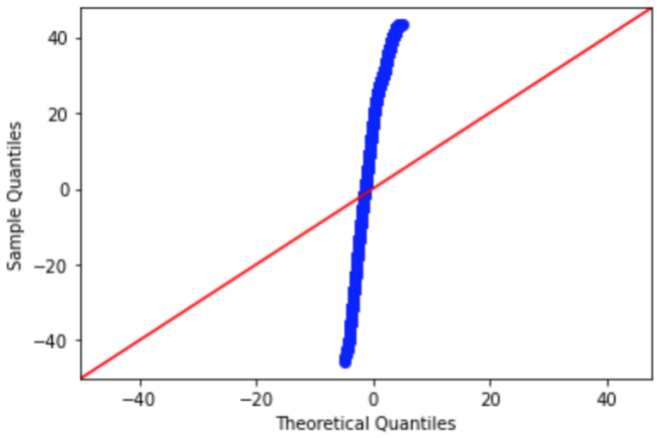
This distribution appears to be less cluttered, but it's still difficult to notice a trend or pattern.

We can now try to plotting the global yearly average temperature as a time series, which looks like the following:



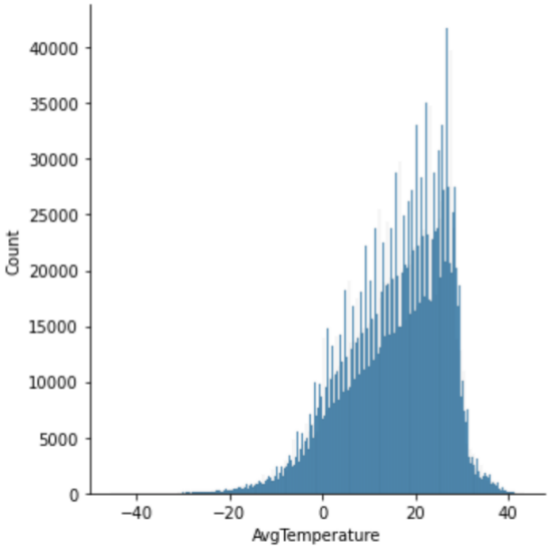
From this distribution, we can see that the global average temperature does indeed have an increasing trend throughout the years, which matches the real life scenarios of global warming steadily increasing the global temperature. However, in this case, we are taking the average of an average, which can distort the results of the original data.

We can try to check to see if the yearly global average temperature data is normally distributed using a qq-plot:



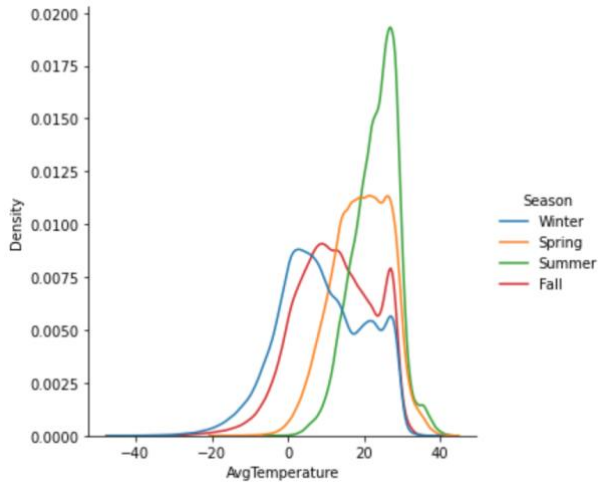
From the qq-plot, we can see that the yearly average temperature data is not normally distributed, and so c. As such, we need to find other ways to represent the data.

Looking at a depiction of the original average temperature data distribution, we have the following:



We can see that the data distribution is rightly skewed, which visualizes how our data is not normally distributed.

Filtering the data by seasons, we can see that the right skew is explained by different seasons each having their own different temperature ranges:



As such, we want to represent the increasing nature of the average temperature data by taking seasonality into account while also normalizing the distribution for an easier time in the feature engineering and modelling processes.

Temperature Anomaly Method:

One simple way to calculate and measure the increase in average temperature is the temperature anomaly method. The temperature anomaly essentially represents a change in temperature between the current temperature and the temperature from a particular reference point of time.

The general formula for calculating the temperature anomaly would be the following:
$Anomaly\ (Current\ Date) = Temperature\ (Current\ Date) -\ Reference\ Temperature$

For our dataset, since we want to best represent the average temperature from pre-industrial times, we will set the earliest existing year from the data as the reference point for our anomaly calculations, which is 1995.

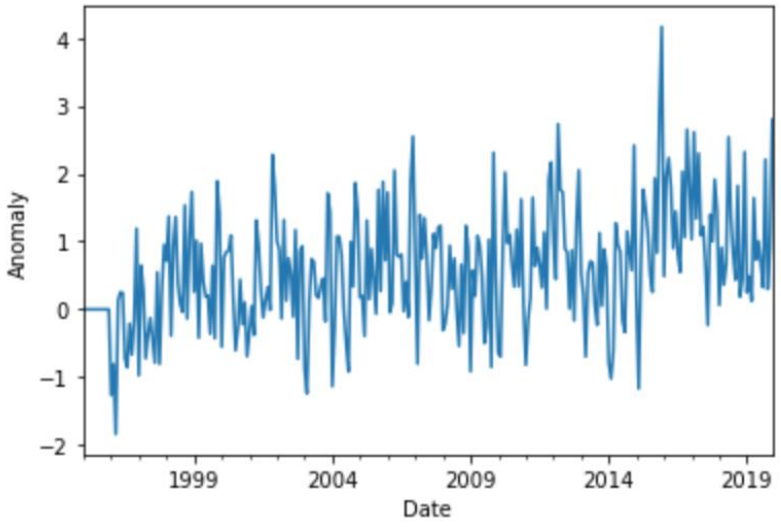
We also have two choices of choosing between grouping the temperatures by month or by year, and, ultimately, I decided that it would be better to group by month as we would end up with more remaining data. This method would also resolve the issue of seasonality skewing the data distribution as the temperatures of a particular season would only be compared with temperatures in 1995 from that same particular season.

An example of calculating the monthly temperature anomaly for January 2016 would be like the following:

	Date_By_Month	Month	Day	Year	AvgTemperature	RefTemperature	Temp_Anomaly
252	2016-01-01	1	1	2016	7.311212	6.822478	0.488734

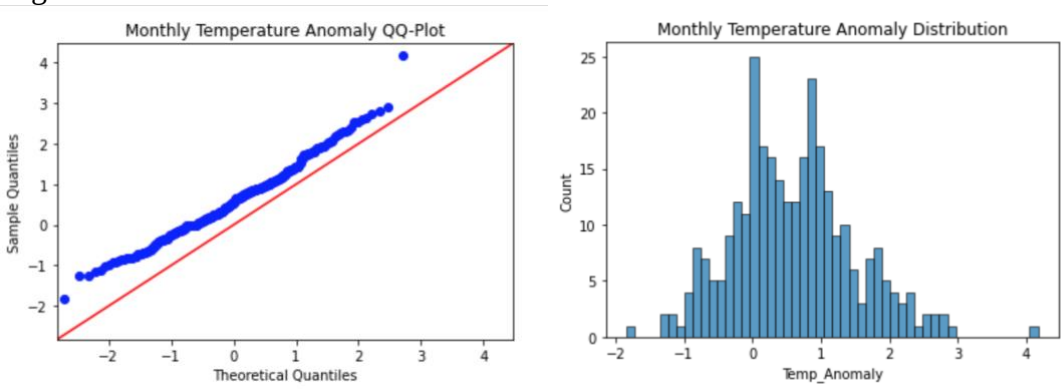
Anomaly = Actual temperature – reference temperature
Anomaly (Jan 2016) = Jan 2016 Temp – Jan 1995 Temp
Anomaly (Jan 2016) = 7.311212 – 6.822478 = 0.488734

Visualizing the monthly temperature anomaly data as a time series, we get the following:



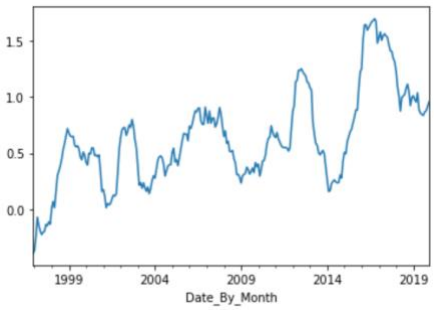
We can see an increasing trend in the anomaly (change in temperature), which effectively represents the real life scenarios of global warming steadily increasing the global temperature.

Checking for normality of the monthly anomaly data using qq-plots and histograms, we have the following:



From these plots, we can see that the monthly temperature anomaly data is normally distributed, and so this data would be good to use for feature engineering and modelling purposes.

However, we can still smooth out the data to be less volatile by using a rolling mean of 1 year, which we then get the following:



Modelling

ARIMA Testing:

Since the ultimate goal is to select a model to forecast data for a time series, the ARIMA model would best suit this purpose.

For the order (p, d, q) of the ARIMA(p, d, q) model, I have decided to set the range of each input parameter as 0 to 2 where 0 <= p, d, q <= 2, and so we would be testing from 27 different ARIMA models to select the best one.

In order to determine which of the 27 ARIMA models would be the best one for our dataset, I have decided to use RMSE (Root Mean Squared Error), MAPE (Mean Absolute Percentage Error), and MAE (Mean Absolute Error) as the evaluation metrics since these three evaluation metrics are the most commonly used ones for time series analysis.

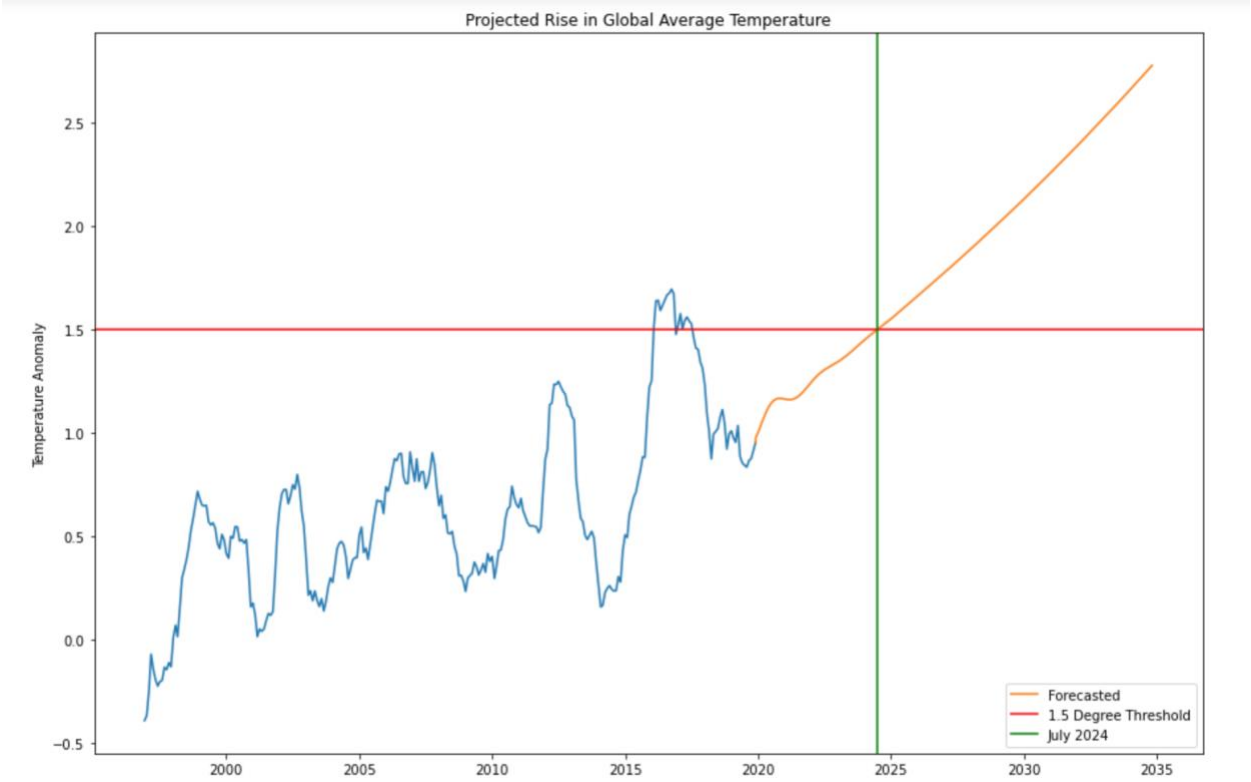
From evaluating through all 27 iterations of the ARIMA models, we obtain the following results:

```
ARIMA(0, 0, 0) RMSE=0.24948 MAPE=0.18982 MAE=0.22693
ARIMA(0, 0, 1) RMSE=0.12862 MAPE=0.09700 MAE=0.11591
ARIMA(0, 1, 0) RMSE=0.02510 MAPE=0.01590 MAE=0.01819
ARIMA(0, 1, 1) RMSE=0.02439 MAPE=0.01511 MAE=0.01729
ARIMA(0, 1, 2) RMSE=0.02440 MAPE=0.01508 MAE=0.01731
ARIMA(0, 2, 0) RMSE=0.03050 MAPE=0.01873 MAE=0.02160
ARIMA(0, 2, 1) RMSE=0.02407 MAPE=0.01557 MAE=0.01799
ARIMA(0, 2, 2) RMSE=0.02606 MAPE=0.01655 MAE=0.01908
ARIMA(1, 0, 0) RMSE=0.02542 MAPE=0.01645 MAE=0.01881
ARIMA(1, 0, 1) RMSE=0.02480 MAPE=0.01535 MAE=0.01755
ARIMA(1, 0, 2) RMSE=0.02486 MAPE=0.01543 MAE=0.01772
ARIMA(1, 1, 0) RMSE=0.02426 MAPE=0.01496 MAE=0.01716
ARIMA(1, 1, 1) RMSE=0.02369 MAPE=0.01465 MAE=0.01686
ARIMA(1, 1, 2) RMSE=0.02371 MAPE=0.01453 MAE=0.01672
ARIMA(1, 2, 0) RMSE=0.02739 MAPE=0.01699 MAE=0.01964
ARIMA(1, 2, 1) RMSE=0.02466 MAPE=0.01521 MAE=0.01742
ARIMA(1, 2, 2) RMSE=0.02425 MAPE=0.01550 MAE=0.01780
ARIMA(2, 0, 0) RMSE=0.02475 MAPE=0.01521 MAE=0.01744
ARIMA(2, 0, 1) RMSE=0.02441 MAPE=0.01573 MAE=0.01815
ARIMA(2, 0, 2) RMSE=0.02449 MAPE=0.01566 MAE=0.01807
ARIMA(2, 1, 0) RMSE=0.02411 MAPE=0.01499 MAE=0.01725
ARIMA(2, 1, 1) RMSE=0.02370 MAPE=0.01456 MAE=0.01676
ARIMA(2, 1, 2) RMSE=0.02301 MAPE=0.01443 MAE=0.01665
ARIMA(2, 2, 0) RMSE=0.02473 MAPE=0.01504 MAE=0.01731
ARIMA(2, 2, 1) RMSE=0.02431 MAPE=0.01524 MAE=0.01752
ARIMA(2, 2, 2) RMSE=0.02461 MAPE=0.01546 MAE=0.01777
Best ARIMA (by RMSE) (2, 1, 2) RMSE=0.02301
Best ARIMA (by MAPE) (2, 1, 2) MAPE=0.01443
Best ARIMA (by MAE) (2, 1, 2) MAE=0.01665
```

We can see that the ARIMA(2,1,2) model is the best for all three evaluation metrics, and so that will be the model that we will use to forecast the anomaly.

Forecasting With Best ARIMA Model:

Using the ARIMA(2,1,2) model to forecast the anomaly data, we get the following results:



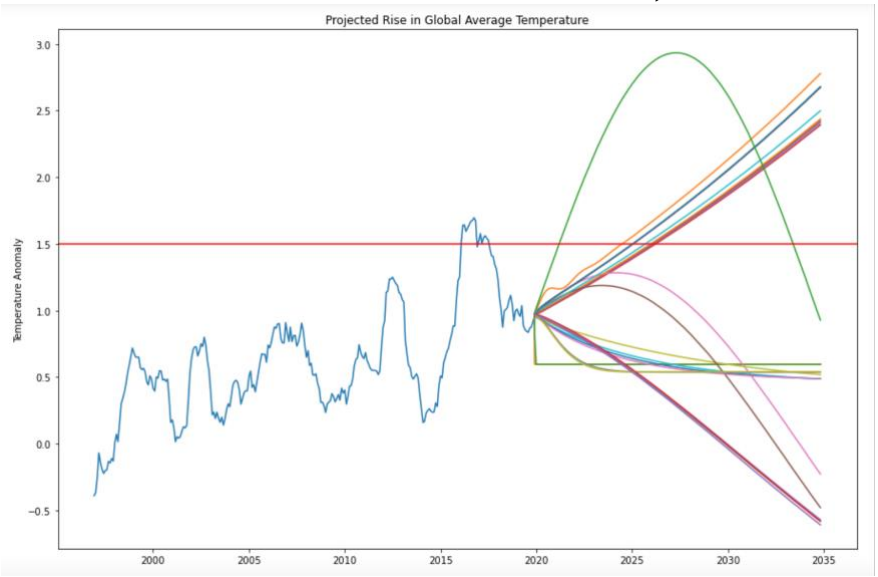
From this graph, we can see that the forecasted date for when the average temperature would consistently reach and exceed the 1.5 degree threshold is in July 2024.

However, a problem that we have is that the orange forecasted curve appears to be continuously increasing. This means that according to this model, the average temperature would be infinitely increasing. Although this trend makes sense short term in real life due to projected increases in average temperature from global warming, this trend does not make sense in real life from a long term perspective as an ice age is projected to happen eventually (in around 1500 years), causing the average temperature to drop, which would not be represented by a continuously increasing curve.

As such, we should look at the forecasted results from the other ARIMA models to see if any of them represent real life more accurately.

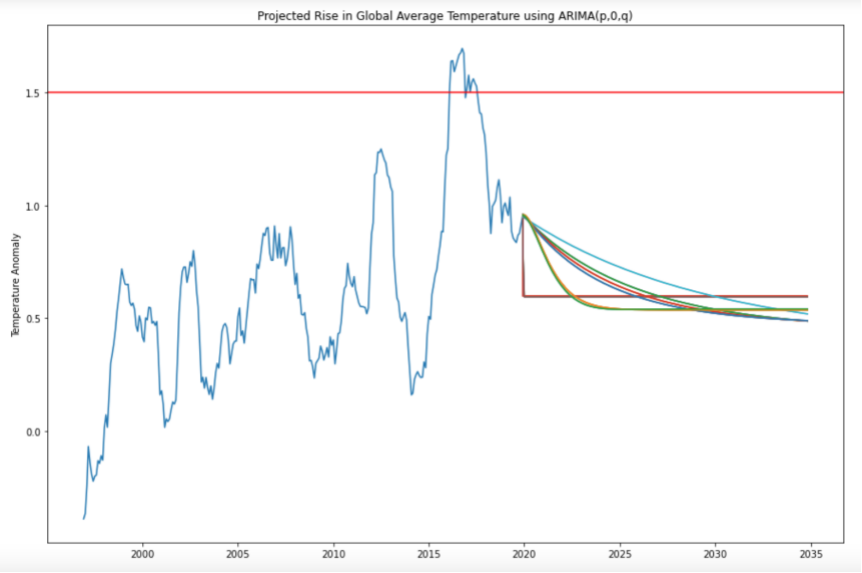
Comparing All Possible ARIMA Forecasting Results:

From plotting all the forecasts of the 27 different ARIMA models, we have:



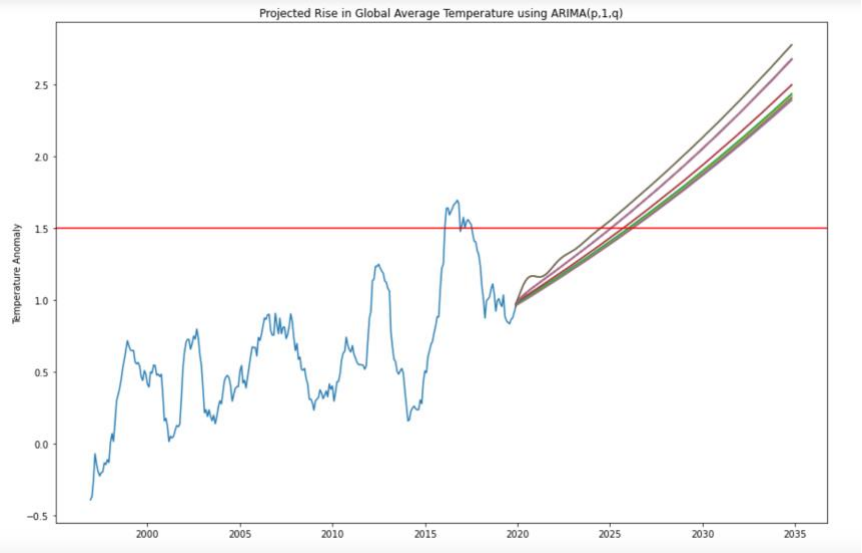
As we can see, there are three distinct patterns for the forecasted curves, which is caused by the degree of differencing parameter of ARIMA (specifically the d parameter in (p, d, q)). Let us visualize each of the three different values separately.

ARIMA(p, 0, d):



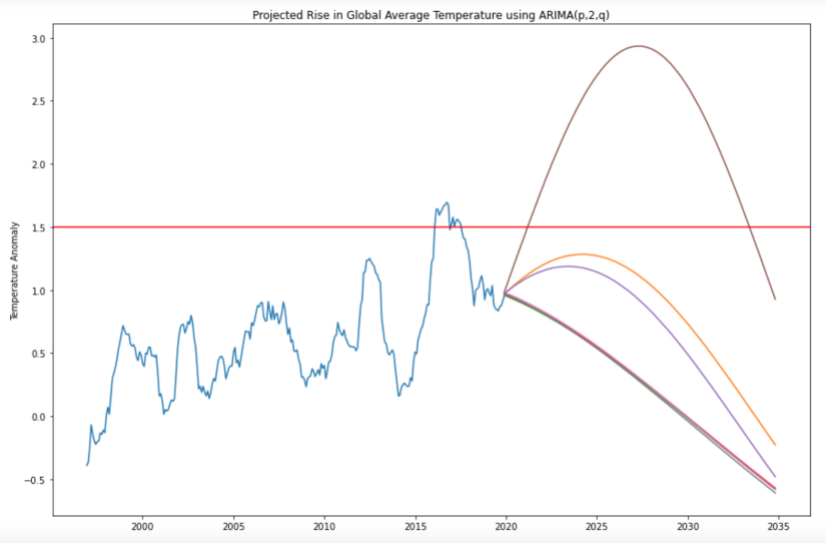
We can see that for ARIMA(p, d, q) models of d=0, the forecasted values follow a trend of decreasing until they converge around 0.5 Temperature Anomaly, which does not make sense given real life context that the average temperature is projected to increase given global warming. As such, we can eliminate all ARIMA(p, d, q) models with d=0 from being viable models to be selected for our situation.

ARIMA(p, 1, d):



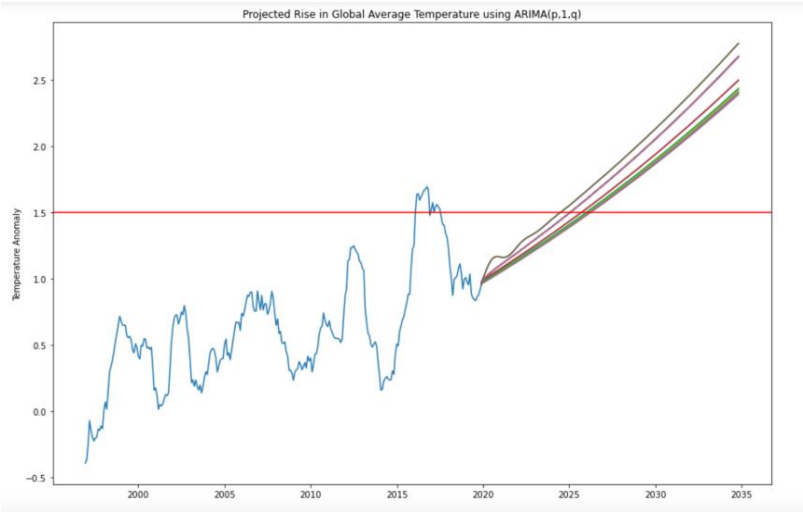
As ARIMA(p, d, q) models of d=1 contains ARIMA(2,1,2), the best model according to evaluation metrics, these models share the same problem of an exponential increasing trend making sense in the short term but not in the long term for real life.

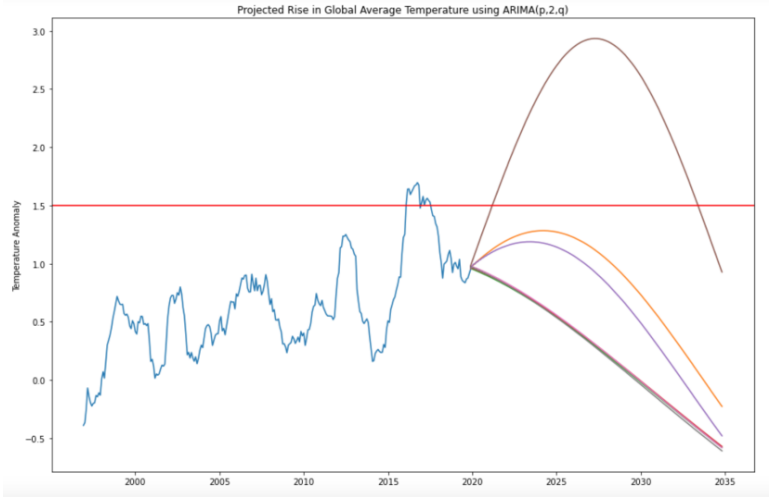
ARIMA(p, 2, d):



We can see that for ARIMA(p, d, q) models of d=2, the forecasted values follow a trend of a parabola, reaching a peak and then start decreasing which resembles the ice age scenario, and so the parabola trend makes sense in real life in the long term. However, according to the models, we will be reaching the ice age around the years between 2020 and 2030, which does not make sense given that the next ice age is projected to occur around 1,500 years from now. As such, these trends do not make sense in the short term given real life context.

We have both a forecasting trend that makes sense in the short term but not the long term (ARIMA(p, 1, q)) and a forecasting trend that makes in the long term but not the short term (ARIMA(p, 2, q)). As such, we would ideally want to combine the concepts of both trends together to mimic the real life scenario of experiencing rises in average temperature for the short term and the ice age scenario of the average temperature reaching a peak and then decreasing back down for the long term (around 1,500 years later).





From comparing the two plots, we can argue that the trends for the ARIMA(p,1,q) models can appear to be the beginning of a very long parabola trend (like the ARIMA(p,2,q) models) that peaks at around the year 3500 (1,500 years later for the projected ice age year), but since the scope of our project (predicting when the 1.5 degree threshold will be reached) does not encompass a time period that far away, then a continuously increasing trend from ARIMA(p,1,q) models would make sense to represent the real life scenario of increasing average temperatures from global warming within the next few decades.

As such, the forecasting trend from the ARIMA(2,1,2) model does represent a realistic real life scenario of climate change for the next few decades.

Main Takeaways and Future Improvements

Overall, the best ARIMA model forecasts that the average temperature will reach and exceed the 1.5°C threshold by July 2024. This result is earlier than the IPCC's predictions (early 2030s), which can be explained by our dataset missing a lot of locations (i.e. cities, oceans, etc.) for a complete representation of global average temperature. As such, an improvement for this project would be to gather more data on locations not yet in the original dataset.

Furthermore, another potential improvement for this project would be to consider other ways of measuring the increase in global average temperature, such as with NASA's weighted mean surface area anomaly method which weighs the temperature data of certain locations more heavily than others. Our current dataset does not have the existing data to attempt this method, and so more data gathering would also be required for considering this improvement.

Ultimately, the main takeaway from this project is that global warming is no longer a distant threat. The 1.5°C threshold will inevitably be exceeded within the next few decades if no intervention is done by us as a society.