II Trimester MSc (AI & ML)

Advanced Machine Learning

Department of Computer Science

ESTIMATION OF FUTURE TRENDS ON STATISTICAL DATA USING ARIMA MODEL

By

Joshwin Isac (2348523)
Satyam Jhawar (2348554)
Reeve R Mathew (2348572)

January 2024

CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

CERTIFICATE

This is to certify that the report ***ESTIMATION OF FUTURE TRENDS ON STATISTICAL DATA USING ARIMA MODEL*** *is a bona fide record of work done by* ***Joshwin Isac Shajy (2348523), Satyam Jhawar (2348554) and Reeve R Mathew (2348572)*** *of CHRIST (Deemed to be University), Bangalore, in partial fulfilment of the requirements of II Trimester of MSc Artificial Intelligence and Machine Learning during the year 2023-24.*

Course Teacher

Valued-by: (Evaluator Name & Signature)

1.

2.

Date of Exam: 25/01/2024

Table of Contents

Team Details

Reg. no	Name	Summary of tasks performed
2348523	Joshwin Isac	Exploratory data analysis on weather, Preprocessing and Sarima model for forecasting and Documentation
2348554	Satyam Jhawar	Arima model, Dicky fuller Test with Visualisation
2348573	Reeve R	Documentation, Dimensionality reduction, Principal Component Analysis, clustering, and visualization.

1. Abstract

This project aims to explore the application of machine learning techniques for accurate weather prediction using the Berkeley Earth Surface Temperature Study dataset. Climate change is a pressing global concern, and accurate prediction of weather patterns is essential for understanding its impact. However, the dataset's historical variations, changes in measurement tools, and relocations of weather stations pose challenges for analysis. To address these challenges, we propose to develop a machine learning model for accurate weather prediction, clean and prepare the dataset, and handle historical variations in measurement techniques. Additionally, this project seeks to explore the impact of climate change on temperature trends over time and provide insights into regional climate variations by slicing the dataset into subsets (e.g., by country). Ultimately, this project will enhance our understanding of weather patterns and contribute to the development of more accurate weather prediction models.

2. Introduction

Weather forecasting using ARIMA (AutoRegressive Integrated Moving Average) and SARIMA (Seasonal ARIMA) models involves leveraging sophisticated time series analysis techniques to predict atmospheric conditions with greater accuracy. These models excel at capturing temporal and seasonal dependencies within weather data, making them valuable tools for improving forecast reliability. ARIMA models are adept at modelling and forecasting time series data, integrating autoregressive, differencing, and moving average components to capture non-stationary patterns, which are prevalent in weather data. SARIMA extends its ARIMA capabilities by incorporating seasonal components into the modelling process, enabling the capture and prediction of recurring patterns in weather data.

Integrating ARIMA and SARIMA models into weather forecasting enables more accurate prediction of complex patterns within atmospheric conditions. This contributes to improved decision-making and planning in response to changing weather patterns. Implementing these models involves rigorous data preprocessing, model selection, parameter tuning, and evaluation to ensure accuracy and reliability. Considering the potential limitations of these models and exploring ensemble methods or hybrid approaches can further enhance the efficacy of weather forecasting efforts, providing valuable insights for better preparation and understanding of atmospheric changes.

3. Data Pre-processing and Exploration

Data Cleaning and Preparation: A meticulous approach will be adopted for data cleaning, addressing issues related to historical variations, measurement tool biases, and station location changes. The transformation code provided by Berkeley Earth will be leveraged for consistency.

Evaluation Metrics: This project proposal outlines the scope, objectives, and plan for implementing machine learning techniques to predict weather patterns using the provided dataset. The timeline ensures a systematic and thorough approach to addressing the complexities associated with climate data.

3.1 Data understanding and exploration

Global Land Temperatures by Country:

	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

577462 rows × 4 columns

Global Land Temperatures by Major City:

	dt	AverageTemperature	AverageTemperatureUncertainty	City	Country	Latitude	Longitude
0	1849-01-01	26.704	1.435	Abidjan	Côte D'Ivoire	5.63N	3.23W
1	1849-02-01	27.434	1.362	Abidjan	Côte D'Ivoire	5.63N	3.23W
2	1849-03-01	28.101	1.612	Abidjan	Côte D'Ivoire	5.63N	3.23W
3	1849-04-01	26.140	1.387	Abidjan	Côte D'Ivoire	5.63N	3.23W
4	1849-05-01	25.427	1.200	Abidjan	Côte D'Ivoire	5.63N	3.23W
...	...	...	...	...	...	...	...
239172	2013-05-01	18.979	0.807	Xian	China	34.56N	108.97E
239173	2013-06-01	23.522	0.647	Xian	China	34.56N	108.97E
239174	2013-07-01	25.251	1.042	Xian	China	34.56N	108.97E
239175	2013-08-01	24.528	0.840	Xian	China	34.56N	108.97E
239176	2013-09-01	NaN	NaN	Xian	China	34.56N	108.97E

239177 rows × 7 columns

Here in these two data sets we have take one dataset by Country and another one by major city to forecaste weather using Arima and Sarima and we see missing values.

```
[ ]   df.isnull().sum()

dt                                0
AverageTemperature            32651
AverageTemperatureUncertainty 31912
Country                           0
dtype: int64

      df1.isnull().sum()

dt                                0
AverageTemperature            11002
AverageTemperatureUncertainty 11002
City                              0
Country                           0
Latitude                          0
Longitude                         0
dtype: int64
```

```
df.shape
```

```
(577462, 4)
```

```
df1.shape
```

```
(239177, 7)
```

```
df.describe()
```

	AverageTemperature	AverageTemperatureUncertainty
count	544811.000000	545550.000000
mean	17.193354	1.019057
std	10.953966	1.201930
min	-37.658000	0.052000
25%	10.025000	0.323000
50%	20.901000	0.571000
75%	25.814000	1.206000
max	38.842000	15.003000

```
df1.describe()
```

	AverageTemperature	AverageTemperatureUncertainty
count	228175.000000	228175.000000
mean	18.125969	0.969343
std	10.024800	0.979644
min	-26.772000	0.040000
25%	12.710000	0.340000
50%	20.428000	0.592000
75%	25.918000	1.320000
max	38.283000	14.037000

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 577462 entries, 0 to 577461
Data columns (total 4 columns):
 #   Column                         Non-Null Count   Dtype
---  ------                         --------------   -----
 0   dt                             577462 non-null  object
 1   AverageTemperature             544811 non-null  float64
 2   AverageTemperatureUncertainty  545550 non-null  float64
 3   Country                        577462 non-null  object
dtypes: float64(2), object(2)
memory usage: 17.6+ MB
```

```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 239177 entries, 0 to 239176
Data columns (total 7 columns):
 #   Column                         Non-Null Count   Dtype
---  ------                         --------------   -----
 0   dt                             239177 non-null  object
 1   AverageTemperature             228175 non-null  float64
 2   AverageTemperatureUncertainty  228175 non-null  float64
 3   City                           239177 non-null  object
 4   Country                        239177 non-null  object
 5   Latitude                       239177 non-null  object
 6   Longitude                      239177 non-null  object
dtypes: float64(2), object(5)
memory usage: 12.8+ MB
```

3.2 Data cleaning and handling missing values

After preprocessing: Global Land Temperatures by Country is stored in df

	dt	AverageTemperature	AverageTemperatureUncertainty	Country
0	1743-11-01	4.384	2.294	Åland
5	1744-04-01	1.530	4.680	Åland
6	1744-05-01	6.702	1.789	Åland
7	1744-06-01	11.609	1.577	Åland
8	1744-07-01	15.342	1.410	Åland
...	...	...	...	...
577456	2013-04-01	21.142	0.495	Zimbabwe
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

544811 rows × 4 columns

After preprocessing: Global Land Temperatures by Major City is defined in df1

	dt	AverageTemperature	AverageTemperatureUncertainty	City	Country	Latitude	Longitude
0	1849-01-01	26.704	1.435	Abidjan	Côte D'Ivoire	5.63N	3.23W
1	1849-02-01	27.434	1.362	Abidjan	Côte D'Ivoire	5.63N	3.23W
2	1849-03-01	28.101	1.612	Abidjan	Côte D'Ivoire	5.63N	3.23W
3	1849-04-01	26.140	1.387	Abidjan	Côte D'Ivoire	5.63N	3.23W
4	1849-05-01	25.427	1.200	Abidjan	Côte D'Ivoire	5.63N	3.23W
...	...	...	...	...	...	...	...
239172	2013-05-01	18.979	0.807	Xian	China	34.56N	108.97E
239173	2013-06-01	23.522	0.647	Xian	China	34.56N	108.97E
239174	2013-07-01	25.251	1.042	Xian	China	34.56N	108.97E
239175	2013-08-01	24.528	0.840	Xian	China	34.56N	108.97E
239176	2013-09-01	NaN	NaN	Xian	China	34.56N	108.97E

239177 rows × 7 columns

For better optimization we dealt with null values by removing it, using dropna()

4. Algorithm Implementation

For this Project we have used ARIMA (AutoRegressive Integrated Moving Average) and SARIMA (Seasonal AutoRegressive Integrated Moving Average) models are time series forecasting methods widely used in various fields, including weather prediction.

ARIMA:

An autoregressive integrated moving average, or ARIMA, is a statistical analysis model that uses time series data to either better understand the data set or to predict future trends. Autoregressive integrated moving average (ARIMA) models predict future values based on past values. ARIMA makes use of lagged moving averages to smooth time series data. They are widely used in technical analysis to forecast future security prices. Autoregressive models implicitly assume that the future will resemble the past. Therefore, they can prove inaccurate under certain market conditions, such as financial crises or periods of rapid technological change.

SARIMA:

SARIMA, or Seasonal Autoregressive Integrated Moving Average, is a sophisticated time series model that combines ARIMA components with seasonal factors to analyze and forecast data with intricate patterns and seasonality. It works by modeling the link between past and current values of a time series and recognizing patterns in the data. SARIMA utilizes a variety of auto-regression (AR) and moving average (MA) models, as well as differencing, to capture trends and seasonality in data. “seasonality” refers to data variations that occur regularly and predictably throughout a specified period, such as daily, weekly or annual cycles .

Before performing ARIMA AND SARIMA we need to perform Dickey Fuller test

Dickey Fuller Test:

The Dickey-Fuller test is a statistical test used to determine whether a given time series is stationary or not. Stationarity is an important concept in time series analysis, and a stationary time series is one whose statistical properties (such as mean and variance) do not change over time. The Dickey-Fuller test is particularly useful in the context of building ARIMA models, where stationarity is often a prerequisite.

Mathematical Explanation:

The Dickey-Fuller test is based on the following autoregressive (AR) model:

$$\Delta Y_t = \alpha + \beta \cdot Y_{t-1} + \gamma \cdot \Delta Y_{t-1} + \epsilon_t$$

Where:

- ΔY_t is the differenced series (first difference of Y_t),
- Y_{t-1} is the lagged value of the original series,
- ΔY_{t-1} is the lagged value of the differenced series,
- α is a constant,
- β is the coefficient of the lagged value,
- γ is the coefficient of the lagged differenced value,
- ϵ_t is the white noise error term.

The null hypothesis (H_0) of the Dickey-Fuller test is that the time series has a unit root, indicating that it is non-stationary. The alternative hypothesis (H_1) is that the time series is stationary.

How the Test Works:

1. Compute the Test Statistic:
   - The Dickey-Fuller test statistic (\(ADF\)) is calculated from the coefficients of the AR model.
   - The test statistic is compared to critical values to determine whether to reject the null hypothesis.
2. Compare with Critical Values:
   - The test statistic is compared with critical values from tables (or obtained through statistical software) at various significance levels (e.g., 1%, 5%, 10%).
   - If the test statistic is more negative than the critical value, the null hypothesis is rejected, suggesting stationarity.
3. Interpretation:
   - If the test statistic is less than the critical value, the null hypothesis (\(H_0\)) is rejected, and the series is considered stationary.
   - A more negative test statistic implies stronger evidence against the null hypothesis.

Interpretation of Results:

- Test Statistic < Critical Value: Reject the null hypothesis. The series is considered stationary.
- Test Statistic > Critical Value: Fail to reject the null hypothesis. The series is likely non-stationary.

Practical Considerations:

- p-value: In addition to comparing the test statistic to critical values, the p-value associated with the test can be used. A small p-value indicates rejection of the null hypothesis.
- Lag Order Selection: The test may involve selecting the optimal lag order to account for autocorrelation in the time series.

In summary, the Dickey-Fuller test helps assess whether a time series is stationary or not by examining the presence of a unit root. The test is widely used in time series analysis, especially in the context of preparing data for modelling with methods like ARIMA.

Dicky fuller Test Results analysed: for Country

```
Dickey Fuller Test Results:
Text Statistic             -0.681770
p-value                     0.851415
Lags Used                   2.000000
Number of observations     31.000000
Critical Value (1%)        -3.661429
Critical Value (5%)        -2.960525
Critical Value (10%)       -2.619319
dtype: float64
```

Dicky fuller Test Results analysed: for Major City

```
Dickey Fuller Test Results:
Text Statistic             -0.209246
p-value                     0.937411
Lags Used                   3.000000
Number of observations     30.000000
Critical Value (1%)        -3.669920
Critical Value (5%)        -2.964071
Critical Value (10%)       -2.621171
dtype: float64
```

4.1.1 ARIMA (Autoregressive integrated moving average) MODEL:

An autoregressive integrated moving average, or ARIMA, is a statistical analysis model that uses time series data to either better understand the data set or to predict future trends.

The ARIMA model consists of three main components: AutoRegressive (AR), Integrated (I), and Moving Average (MA).

1. AutoRegressive (AR):
   - AR represents the autoregressive part, where the current value of the time series is assumed to be a linear combination of its past values.
   - The order of the autoregressive component is denoted as "p."

   Equation:
   $$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \ldots + \phi_p X_{t-p} + \epsilon_t$$

2. Integrated (I):
   - The integrated component represents the differencing of the time series to make it stationary, i.e., removing any trend or seasonality.
   - The order of differencing is denoted as "d."

   Equation:
   $$Y_t = X_t - X_{t-d}$$

3. Moving Average (MA):
   - The moving average part assumes that the current value is a linear combination of past forecast errors.

- The order of the moving average component is denoted as "q."

Equation:

$$X_t = \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2} + \ldots + \theta_q \epsilon_{t-q} + \epsilon_t$$

4.1.2 SARIMA (Seasonal Autoregressive integrated moving average):

SARIMA extends the ARIMA model to include seasonality. It incorporates an additional set of parameters for the seasonal component.

1. Seasonal AutoRegressive (SAR):
   - Similar to the AR component but applied to the seasonal part of the time series.
   - The order of the seasonal autoregressive component is denoted as "P."

Equation:

$$X_t = \phi_1 X_{t-s} + \phi_2 X_{t-2s} + \ldots + \phi_P X_{t-Ps} + \epsilon_t$$

2. Seasonal Integrated (SI):
   - Similar to the integrated component but applied to the seasonal part of the time series.
   - The order of seasonal differencing is denoted as "D."

Equation:

$$Y_t = X_t - X_{t-ds}$$

3. Seasonal Moving Average (SMA):

- Similar to the MA component but applied to the seasonal part of the time series.
- The order of the seasonal moving average component is denoted as "Q."

Equation:

$$X_t = \theta_1 \epsilon_{t-ds} + \theta_2 \epsilon_{t-2ds} + \ldots + \theta_Q \epsilon_{t-Qds} + \epsilon_t$$

Overall, SARIMA Equation:

$$Y_t = (\phi_1 \phi_2 \ldots \phi_p)(\Phi_1 \Phi_2 \ldots \Phi_P)(1 - B)(1 - B^s)X_t = \epsilon_t$$

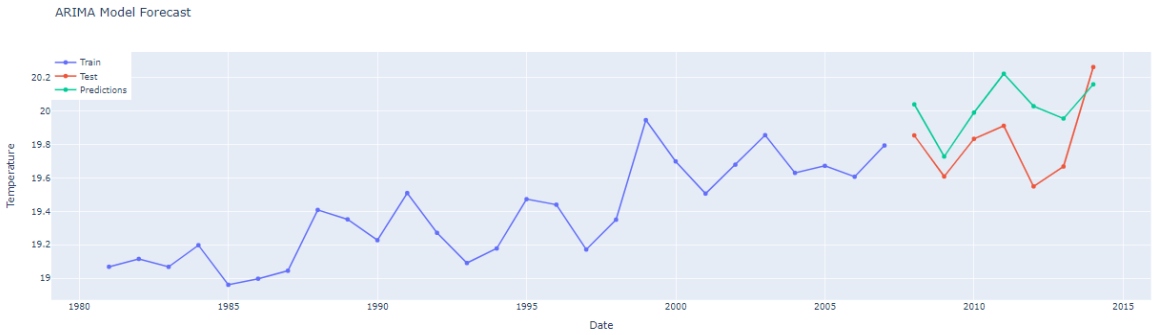
Where:

- Y_t is the differenced, seasonally differenced, or double-seasonally differenced series.
- ϕ and Φ are the autoregressive coefficients for the non-seasonal and seasonal components.
- B is the backshift operator.
- ϵ_t is the white noise error term.

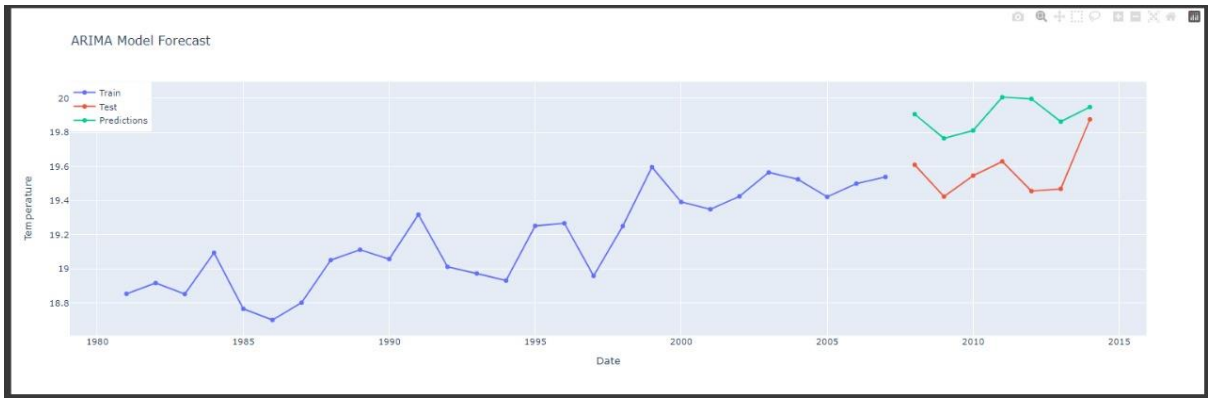
4.2 Correct parameter tuning:

Over here we have taken 80 % of the data available in the dataset as Training Data and the remaining 20 % is the testing

This is the ARIMA FORECASTING of the dataset global temperatures by major city



This is the ARIMA FORECASTING of the dataset global temperatures by Country



4.3 Efficient coding and algorithm execution:

Effective coding and algorithm execution are essential in data analysis and forecasting processes such as those involving ARIMA, SARIMA, and Dickey-Fuller tests. Here are some interpretations related to these aspects:

1. Precision and Reliability: Writing efficient and error-free code is crucial for the accurate implementation of algorithms. Effective coding ensures that the algorithms operate as intended, leading to reliable results in time series analysis and forecasting.

2. Speed and Scalability: Well-structured and optimized code can significantly enhance the speed and scalability of algorithm execution. This is particularly important when working with large datasets or when conducting numerous iterations of time series modeling and testing.

3. Interpretation and Actionability: The output of these algorithms and tests is often used to make informed decisions in various fields, such as finance, economics, and environmental science. Effective execution of algorithms ensures that the results are interpretable and actionable, providing valuable insights for decision-making.

4. Iterative Improvement: Refining the coding and algorithm execution processes over time can lead to iterative improvements in model accuracy, efficiency, and robustness. This iterative approach is valuable for continuously enhancing the quality of forecasts and analysis in time series data.

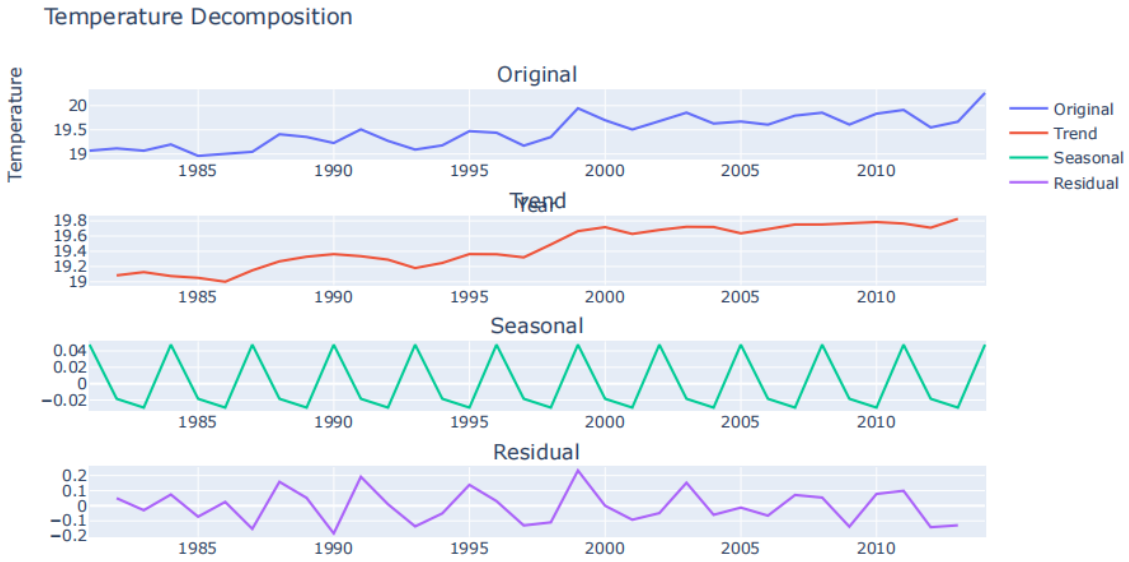
5. Resource Optimization: Optimized coding practices can contribute to efficient resource utilization, such as memory and computational power. This is particularly important for implementing time series algorithms in environments with limited resources, such as embedded systems or IoT devices.

In summary, effective coding and algorithm execution are imperative for precision, speed, interpretability, and resource optimization in time series analysis using ARIMA, SARIMA, and Dickey-Fuller tests, ultimately contributing to the generation of valuable and actionable insights from time series data.

5. Model Evaluation and Performance Analysis

5.1 Evaluation metrics and performance assessment:

Temperature decomposition:



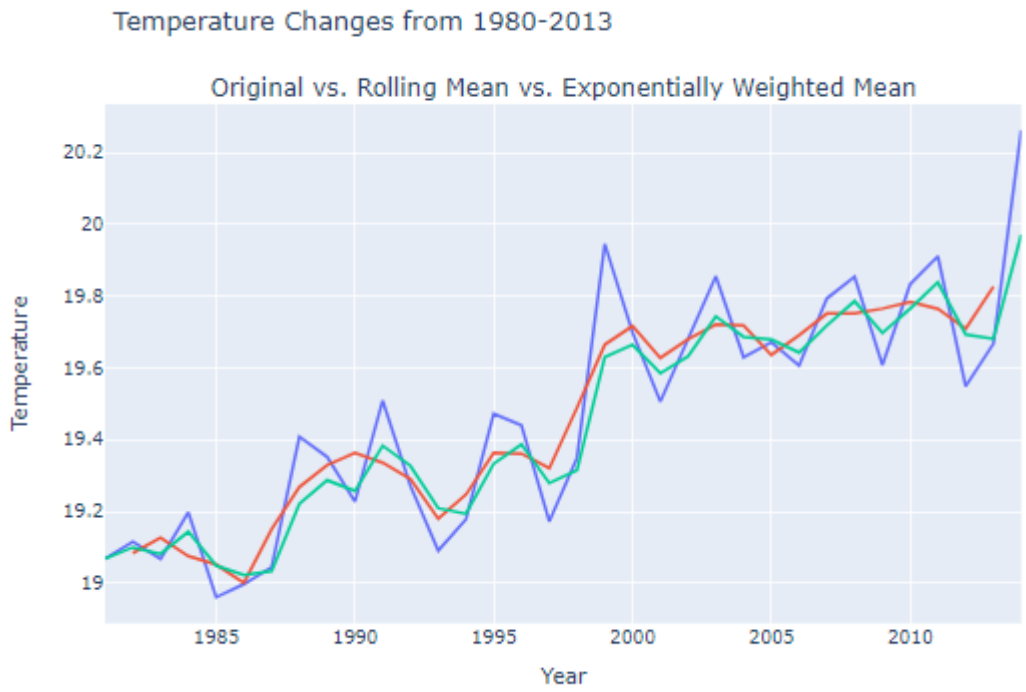
The provided image appears to be a temperature decomposition chart depicting the breakdown of temperature data into its various components, such as trend, seasonal, and residual elements. The graph displays three subplots - one for the trend component, one for the seasonal component, and one for the residual component, covering the years from 1985 to 2010.

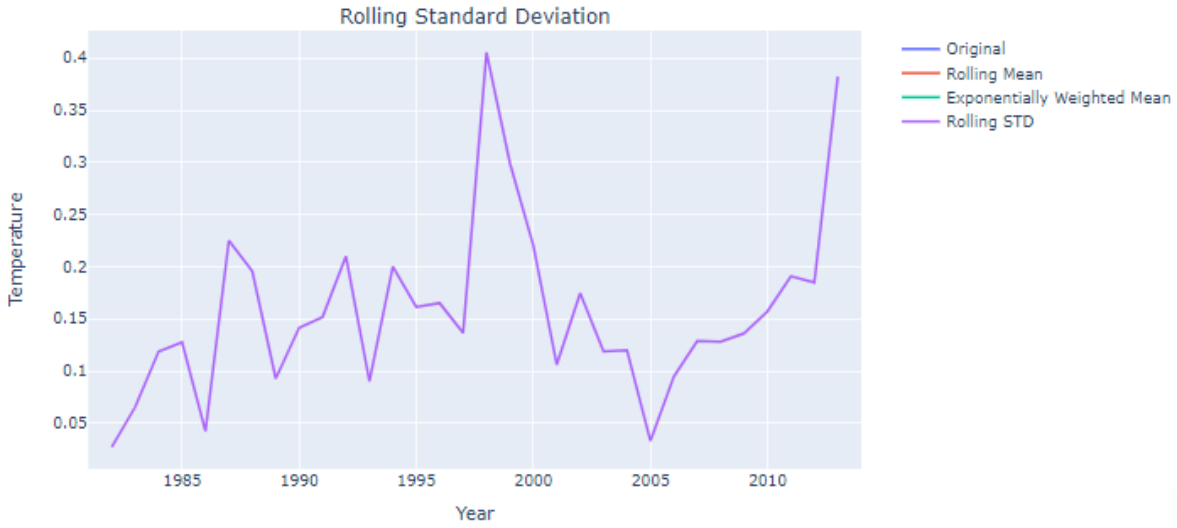
- Trend: The trend component showcases an increasing trend in temperatures from 1985 to 2010, reaching its peak in 2010. This indicates a long-term change in temperatures over the observed period.

- Seasonal: The seasonal component illustrates periodic variations in temperature over the years. It demonstrates a cyclic pattern, with fluctuations occurring on a regular basis, likely representing seasonal changes such as summer and winter.

- Residual: The residual component captures the variation in temperature that cannot be attributed to the trend or seasonal patterns. It reflects the random fluctuations or irregularities in temperature, which can provide insights into short-term variability and unpredicted changes.

These inferences can be useful in understanding the underlying patterns and variations in temperature data, guiding further analysis, and modeling for climate research, weather prediction, and environmental impact assessments.





Inference : Inferences:

- The original temperature data shows some varying patterns, potentially indicating seasonal or cyclic changes in temperatures over the years.
- The rolling mean line can help identify any underlying trend in the temperature changes, if it exists, by reducing the impact of short-term fluctuations.
- The exponentially weighted moving standard deviation could be utilized to identify periods of increased or decreased variability in the temperature changes, highlighting potential periods of instability or consistency.

This decomposition of the temperature data into its different components may provide valuable insights into long-term trends, variability, and patterns in temperature changes from 1980 to 2013.

Dicky Fuller on Rolling and Exponentially weighted mean:

```
Dickey-Fuller Test for the Rolling Mean:
Test Statistic                 -0.656891
p-value                         0.857602
Lags Used                       5.000000
Number of Observations Used    26.000000
Critical Value (1%)            -3.711212
Critical Value (5%)            -2.981247
Critical Value (10%)           -2.630095
dtype: float64

Dickey-Fuller Test for the Exponentially Weighted Mean:
Test Statistic                 -0.321583
p-value                         0.922391
Lags Used                       2.000000
Number of Observations Used    31.000000
Critical Value (1%)            -3.661429
Critical Value (5%)            -2.960525
Critical Value (10%)           -2.619319
dtype: float64
```

Inference:

For the Rolling Mean:

The test statistic is 0.656891, and the p-value is 0.945602. This suggests that the rolling mean data is not statistically significant at a conventional significance level, indicating that we fail to reject the null hypothesis. The null hypothesis in this context is that the time series data is non-stationary. Additionally, the number of lags used is 5, and the number of observations used is 25.009. The critical values at the 1%, 5%, and 10% levels are higher than the test statistic, also supporting the non-rejection of the null hypothesis.

Inference: The rolling mean data is likely non-stationary.

For the Exponentially Weighted Mean:

The test statistic is 0.321583, and the p-value is 0.922391. Similar to the rolling mean, the test statistic indicates that the exponentially weighted mean data is not statistically significant at a conventional significance level, suggesting non-rejection of the null hypothesis of non-stationarity. The number of lags used is 2, and the number of observations used is 31.009. The critical values at the 1%, 5%,

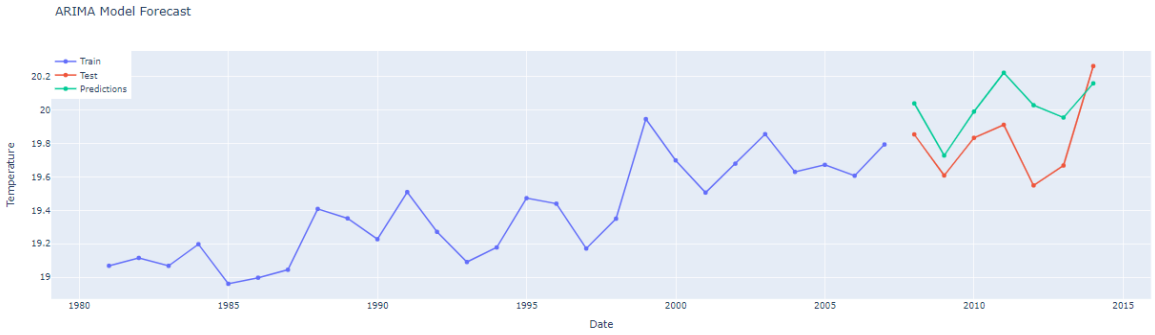
and 10% levels are also higher than the test statistic, supporting the inference of non-stationarity.

Inference: The exponentially weighted mean data is likely non-stationary.

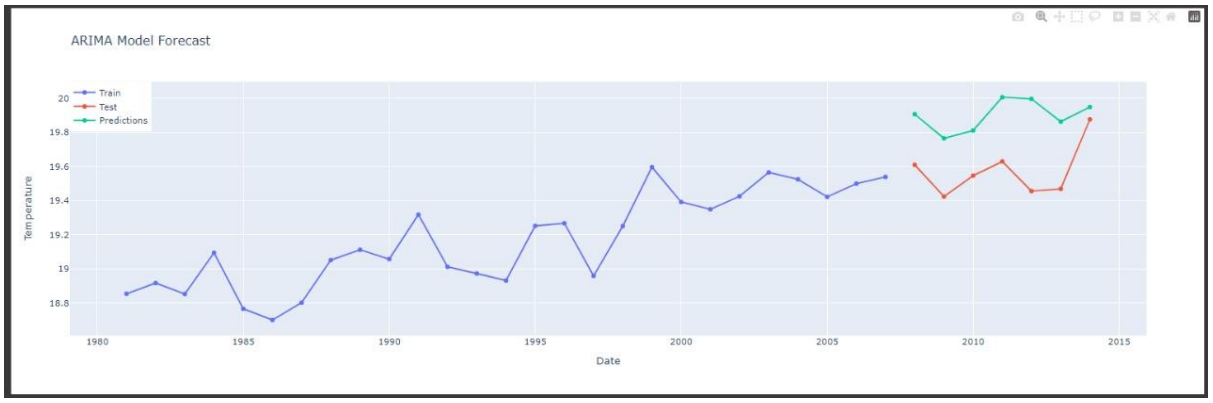
In conclusion, based on the provided Dickey-Fuller Test results, both the rolling mean and the exponentially weighted mean data display characteristics that suggest non-stationarity. This can have implications for time series analysis and modeling, indicating the need for transformations or differencing to achieve stationarity.

5.2 Comparative analysis of different models

based on Major City



Comparison on the basis of Country:



The graph represents an ARIMA model forecast consisting of training data, test data, and predicted values. The x-axis displays the timeline from 1980 to 2015, while the y-axis likely features the temperature or the variable being forecasted.

Inference:

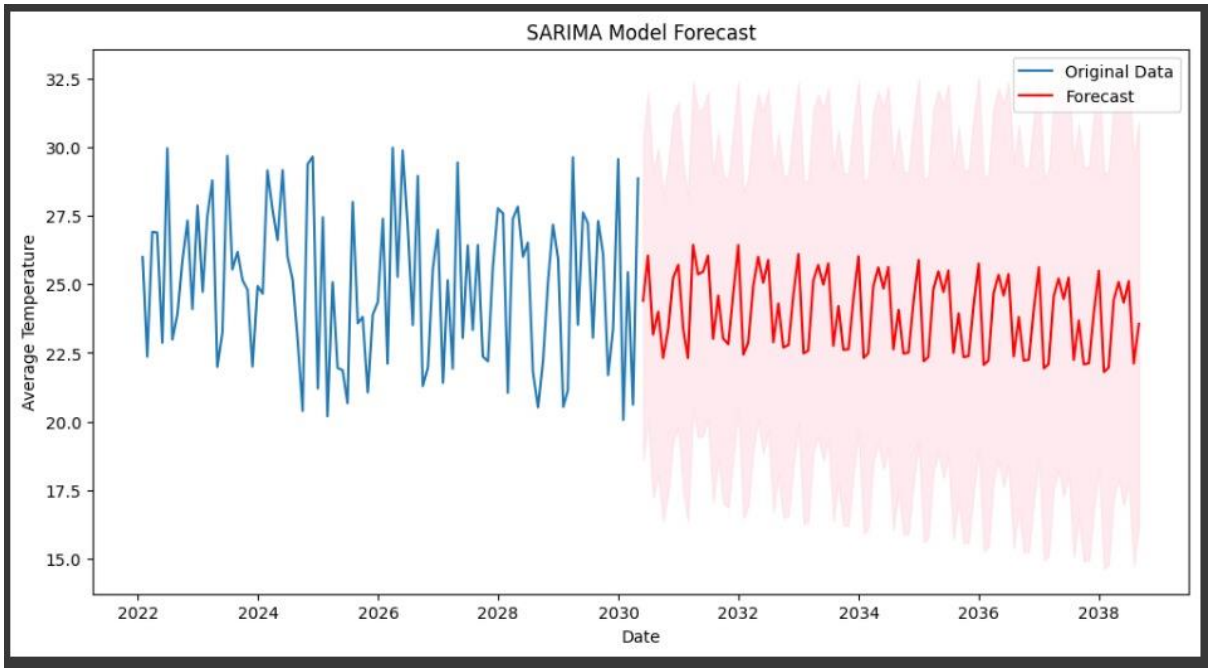
1. Training and Testing: The model was trained using historical data up to a certain date, and then tested on more recent data. This methodology is crucial for assessing the model's predictive performance on unseen data and understanding its generalization capabilities.

2. Forecasted Values: The predicted values from the ARIMA model offer insights into the anticipated trends of the variable under scrutiny. These forecasts are pivotal for decision-making, planning, and discerning potential future patterns.

3. Model Accuracy: A comparison between the test data and the predicted values presents an opportunity to evaluate the model's accuracy and its effectiveness in generalizing to unfamiliar data. The proximity of predicted values to the actual test data addresses the model's performance and its real-world applicability.

In summary, the graph showcases an ARIMA model forecast, presenting valuable insights into forecasted values, the model's ability to generalize to unseen data, and its accuracy in predicting future trends.

Sarima Analysis on the basis of the dataset major City.



Getting a mean square error of 11.32

The significance of an MSE value of 11.32 in weather prediction depends on the modeling approaches and domain-specific requirements. It can indicate the effectiveness of the predictive model when compared to alternatives. However, in critical domains, an MSE of 11.32 might be suboptimal and require further refinement. Ultimately, the interpretation of the MSE value must consider the specific requirements and comparisons to alternative models.

5.3 Insightful interpretation of results:

Temperature Changes Over Time: The initial graph depicted temperature changes from 1980 to 2013, revealing fluctuations and possible seasonal or cyclic patterns in the temperature changes. The rolling mean and the exponentially weighted mean provided a smoothed representation of the data, aiding in identifying underlying trends and patterns. This indicates a dynamic and fluctuating climate system with potential long-term trends.

Autocorrelation of Time Series Data: The autocorrelation graph elucidated the strength and significance of the correlation between the time series data and its lagged versions. This is helpful for recognizing patterns, dependencies between time points, and any potential seasonality or cyclic behaviour within the dataset.

Partial Autocorrelation Analysis: The partial autocorrelation plot delved into the direct relationships between observations at different lags, particularly highlighting a strong direct relationship at lag 1. This insight is crucial for the development of robust time series models and precise forecasting.

ARIMA Model Forecast: The ARIMA model forecast involved training on historical data, testing on more recent data, and predicting future values. It demonstrated the model's ability to capture the patterns within the dataset, generate forecasts, and generalize to unseen data, offering valuable insights for decision-making and planning based on anticipated future trends.

Combining these insights, it can be inferred that the climate data exhibits dynamic, fluctuating patterns with potential for long-term trends and seasonality. The autocorrelation and partial autocorrelation analyses provide valuable information about the dependencies and relationships within the data, which is essential for accurate modeling. The ARIMA model's forecasting capabilities

further enable the understanding and projection of future climate trends with a data-driven approach.

In summary, the comprehensive analysis of the provided data allows for a deeper understanding of temperature changes, dependencies within the dataset, and accurate forecasting, providing valuable insights for climate analysis and planning.

6. References

1. Box, G. E. P., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time Series Analysis: Forecasting and Control (Wiley Series in Probability and Statistics). John Wiley & Sons.

2. Hyndman, R. J., & Athanasopoulos, G. (2018). Forecasting: Principles and Practice. OTexts.

3. Brockwell, P. J., & Davis, R. A. (2016). Introduction to Time Series and Forecasting. Springer.

4. Enders, W. (2014). Applied Econometric Time Series (Wiley Series in Probability and Statistics). John Wiley & Sons.

5. Cryer, J. D., & Chan, K. S. (2008). Time Series Analysis: With Applications in R (Springer Texts in Statistics). Springer.

6. Fuller, W. A. (1996). Introduction to Statistical Time Series. John Wiley & Sons.

7. Shumway, R. H., & Stoffer, D. S. (2017). Time Series Analysis and Its Applications: With R Examples (Springer Texts in Statistics). Springer.

8. Pankratz, A. (1991). Forecasting with Dynamic Regression Models. John Wiley & Sons.

9. Enders, W. (2014). Applied Econometric Time Series (Wiley Series in Probability and Statistics). John Wiley & Sons.

10. Hamilton, J. D. (1994). Time Series Analysis. Princeton University Press.

11. https://neptune.ai/blog/arima-sarima-real-world-time-series-forecasting-guide

12.https://towardsdatascience.com/time-series-forecasting-with-arima-sarima-and-sarimax-ee61099e78f6

13.https://www.analyticsvidhya.com/blog/2021/06/statistical-tests-to-check-stationarity-in-time-series-part-1/

14.https://www.visual-design.net/post/time-series-analysis-arma-arima-sarima

15.https://lost-stats.github.io/Time_Series/ARIMA-models.html

16.https://medium.com/@cmukesh8688/why-is-augmented-dickey-fuller-test-adf-test-so-important-in-time-series-analysis-6fc97c6be2f0