



BMMS2074 Statistics for Data Science

Assignment

Semester 202305

Programme (Year & Group)	:	RDS2S1G2
Tutorial Group	:	2
Date Submitted	:	1/10/2023

Team members:

No	Name (Block Letters)	Registration No.	Signature	Contribution (%)
1	RYAN KHO YUEN THIAN	22WMR04097	<i>RyanK</i>	33.33%
2	ONG WENG KAI	22WMR03309	<i>WengKai</i>	33.33%
3	THONG CHENG HOW	22WMR03154	<i>ThongCH</i>	33.33%
4				

PART-II: Depth of Knowledge Assessment Rubrics

Program Learning Outcomes	Evaluation Criteria	Competency Levels				Score
		0-3 Unsatisfactory	4-7 Fair	8-11 Good	12-15 Outstanding	
Critical Thinking and Problem Solving (75%)	Written Communication (15%)	Attempts to use a consistent system for basic organization; minimal attempts to use sources to support ideas in the writing and these sources may not be correctly documented using an appropriate referencing style and/or may not be fully relevant to the task at hand.	Follows expectations appropriate to a specific discipline and/or writing task for basic organization, and content; use credible and/or relevant sources to support ideas and to document these sources properly using APA or Harvard referencing style.	Demonstrates consistent use of important conventions particular to a specific discipline and/or writing task; consistently use credible, relevant sources appropriate to the discipline and genre to support ideas and documents sources with few errors or exceptions using APA or Harvard referencing style.	Demonstrates detailed attention to and successful execution of a wide range of conventions particular to a specific discipline and/or writing task (including organization, content, formatting, and stylistic choices); synthesize a range of high-quality, credible, relevant sources that are appropriate for the discipline and genre to develop ideas and fully documents these sources using APA or Harvard referencing style.	
	Problem Solving Strategy and Approaches (15%)	Unable to identify an approach to possible solution.	Identifies a possible but very general approach to a solution without a clear sense of the steps to solve the problem.	Identifies a reasonable and problem specific possible approach to a solution with some sense of steps to be undertaken to reach a solution.	Identifies at least one reasonable and problem specific possible approach to a solution. Outlines several steps in detail and/or identifies another reasonable and problem specific possible approach.	
	Analysis (15%)	Demonstrates emerging understanding of the data analysis without showing evident e.g. diagrams, models, timelines, illustrations, explanations, or a series of steps in the report. The report fails to tie into basic concepts and build on prior knowledge.	Demonstrates moderate understanding of the data analysis that are somewhat evident e.g. diagrams, models, timelines, illustrations, explanations, or a series of steps within the report. The report may fail to tie into basic concepts and build on prior knowledge.	Demonstrates considerable understanding of the data analysis and are evident e.g. diagrams, models, timelines, illustrations, explanations, or a series of steps within the report. The report ties into basic concepts and builds on prior knowledge.	Demonstrates in-depth/ thorough understanding of the data analysis and are clearly evident e.g. diagrams, models, timelines, illustrations, explanations, or a series of steps throughout the report. The report ties into basic concepts and builds on prior knowledge.	
	Critical Thinking and Perspective-Taking (15%)	Specific position is stated but is simplistic and obvious.	Information is presented with some interpretation or evaluation, but not enough to develop a coherent analysis or synthesis.	Specific position takes into account the complexities of an issue and acknowledges other viewpoints.	Questions are examined from a range of viewpoints, taking into account the complexities of an issue.	
	Conclusions and Related Outcomes (Implications and Consequences) (15%)	Conclusion is inconsistently tied to some of the information discussed; related outcomes (consequences and implications) are oversimplified.	Conclusion is logically tied to information (because information is chosen to fit the desired conclusion); some related outcomes (consequences and implications) are identified clearly.	Conclusion is logically tied to a range of information, including opposing viewpoints; related outcomes (consequences and implications) are identified clearly.	Conclusions and related outcomes (consequences and implications) are logical and reflect student's informed evaluation and ability to place evidence and perspective discussed in priority order.	
Total:						

1. Introduction

Climate change is a major issue that is being faced by many parts of the world today. It is expected to result in around 250,000 additional deaths yearly from heat stress, malnutrition, diarrhoea and malaria between 2030 and 2050 (World Health Organisation, 2021). One aspect that is affected by climate change is the temperature. The changes in temperature have clearly affected Malaysia in several ways, which include agriculture, water resources and human health. In 2000, the previous Ministry of Science, Technology and Environment roughly calculated that Malaysia will face temperature changes from 0.7 to 2.6 °C (Rogers, 2023). Projections also show that by 2050, Malaysia will become hotter with an average monthly temperature rise of 1.5 °C (Rogers, 2023).

Furthermore, calculations show that for every 1°C rise in temperature, grain yields decrease by up to 10% and wet paddy cannot be sustained in prolonged droughts. Since rising sea levels can be caused by the increase in temperatures, people could be forced to abandon low-lying areas that are planted with various crops, resulting in those crops to be wasted. Apart from crops, the increase in heat could also lead to decreased meat production as livestock, such as chicken and pigs, are affected by direct heat (Rogers, 2023).

Because of the rising temperatures, there is a rise in the potential evaporation rate that is the net loss of moisture per year. In the dry season, the evaporation rates will escalate. When there is not much surface runoff, it could lead to the deterioration of the water quality of Malaysia's rivers and this could threaten the economies of local fishing and the associated eating places (Rogers, 2023).

Even if we are not aware of how Malaysia's agriculture and water resources are negatively impacted by the increases in temperature, one cannot deny that the increased temperatures these days have caused us, Malaysians, to use fans and air conditioners more frequently, causing our electricity bills to be more expensive than usual. One must not forget about the recent fatal heatwave in Malaysia that has caused the deaths of some individuals, such as the 11 year old boy and 19 month old girl in Kelantan who died in a single week (Gabungan Darurat Iklim Malaysia & et al, 2023).

Last but not least, the changes in temperature and rainfall could cause a rise in vector-borne diseases. Examples are dengue fever, Japanese encephalitis and malaria. Furthermore, diarrhoeal diseases, for example salmonella and cholera, could multiply via amoebic dysentery (Rogers, 2023). All of these clearly show that climate change is a huge issue in Malaysia that needs to be handled seriously.

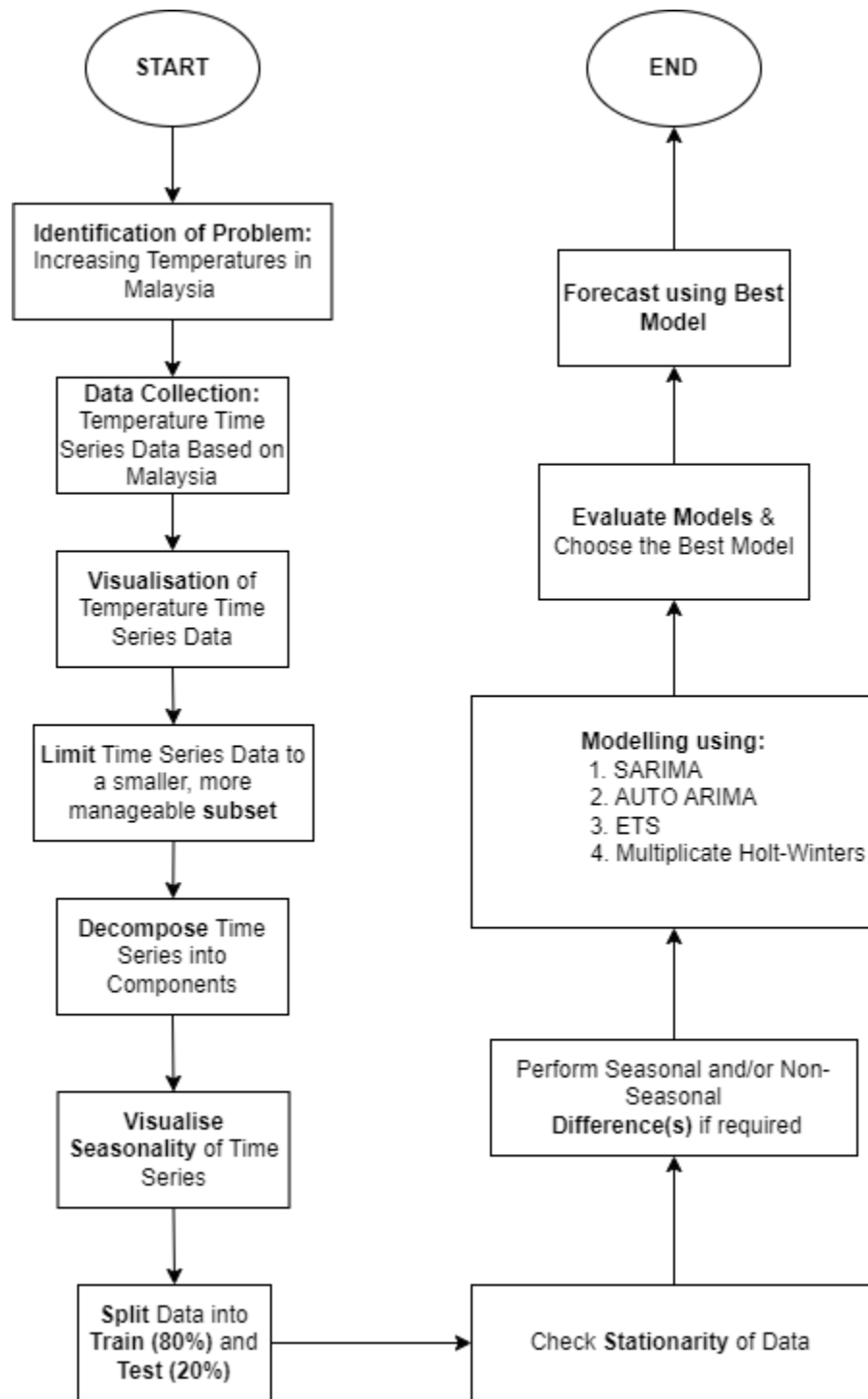
2. Objective

Due to the rising temperatures in Malaysia in recent years, it is important for the Malaysian government to have a forecasting method that accurately predicts future temperatures so that the Malaysian government can take the correct/necessary actions to ensure the safety of everyone in Malaysia. In order to achieve that goal, we have devised several objectives.

1. To visualise the time series dataset to comprehend past years temperature readings.
2. To study how the temperature in Malaysia progresses/changes as the years go by (pattern) and identify any predictable or regular changes that occur yearly.
3. To model the problem using several different Time Series Models, which are SARIMA, AUTO ARIMA, ETS and Multiplicative Holt-Winters, evaluate them based on certain evaluation metrics and choose the forecasting model that is the best at forecasting temperatures in Malaysia.
4. To warn/alert the Malaysian government ahead of time about the times where the temperatures in Malaysia would be extremely high so that the ministers are given more time to formulate solutions or plans to confront the issue, such as an incoming heatwave.
5. To raise public awareness by making temperature forecasts publicly available so that people can support policies that combat climate change and make informed decisions in their day-to-day lives. Examples are using public transportation, promoting afforestation and reforestation or using renewable energy. These actions can help to reduce carbon emissions.

3. Methodology

a) Methodology Process Flow



b) Visualisation

i) Time plot

Time plot is also known as time series plot. Time plots illustrate data points at a successive interval in time and are a graphical representation of data points (IBM, 2023). Time series that has been plotted must contain numerical values and are assumed to be in periods that are uniform. In the time plot the x axis always represents the time sequence that usually is in hours, months, years, minutes, seconds while the y axis is typically the variable that is measured, for example temperature, sales sold etc (IBM, 2023).

To use a time plot, we need to first collect data, which need to be all numerical values. The second step is to find a suitable graphical method to use for time plot because in R Programming there are many types of time plots that can be plotted, so depending on the dataset the user must select the best plot for representation. The next step is to plot using the data that has been collected.

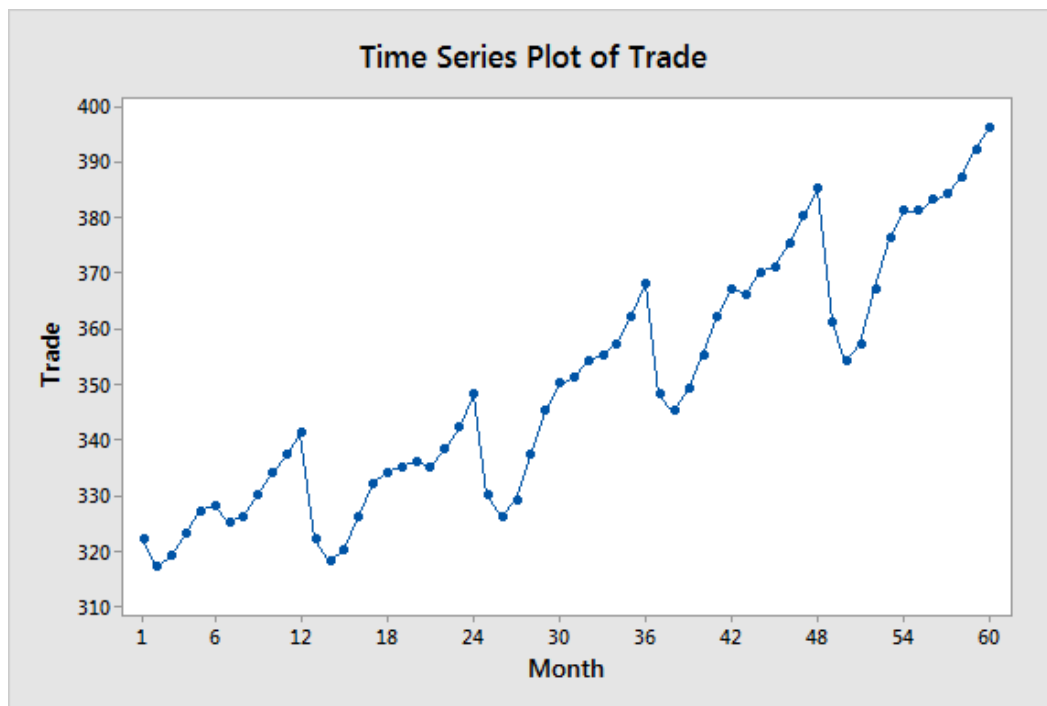


Figure 3.b.1: Example of a Time Plot

ii) ACF and PACF

When dealing with ARIMA models, ACF (autocorrelation) and PACF(Partial autocorrelation) are essential because they are a crucial tool for selecting the suitable ARIMA selection. ARIMA is characterised by three parameters, which are p (order of autoregression), d (degree of differencing) and q(order of moving average(MA) term). ACF and PACF help to determine the suitable value for p and q. To be specific, ACF is used to identify MA order(q) and PACF is used to identify AR order (p). In addition, ACF and PACF are essential to check whether the series is stationary. If non-stationary, differencing(d) might be needed. After fitting the model, PACF and ACF can be used to ensure the model captures essential information in the data where no data is left unexplained (*Autocorrelation Function and Stationarity - SPUR ECONOMICS*, 2023).

ACF (autocorrelation)

ACF is an important step for exploratory analysis, where ACF helps to identify patterns and check for randomness in the time series to determine whether the time series is stationary or non-stationary. ACF measures the linear relationship between an observation and its lag(s). If the ACF is slowly decaying, it means the futures values of the series are influenced significantly by past values in the series, which means it is non stationary as the mean value would change over time. On the other hand, if the series is stationary, ACF experiences a sharp drop indicating the time series does not have any trend and seasonality. If the model has a negative spike at lag 1 in the ACF graph, it suggests a potential over differencing. It is used to determine the order of the MA process. (Monigatti, 2022)

PACF (Partial autocorrelation function)

PACF measures the direct correlation/relationship between lag and observation. It is used to determine the order of AR(autoregression) models. In general, PACF is a conditional correlation where the correlation is split into two variables under the assumption that we know the other set of variables (Sachin, 2023). As an example

$$T_i = \beta_0 + \beta_1 \times T_{(i-1)} + \beta_2 \times T_{(i-2)}$$

The general formula for PACF(X, lag=k)

As mentioned above, the two variables we can use in this equation to understand more about the equation in general.

T_i = formula to represent PACF

$T_{(i-1)}$ = the variance of the time series

$T_{(i-2)}$ = Predict today's value

Variable 1: The amount of variance in T_i that is **not explained** by the variance in $T_{(i-1)}$

Variable 2: The amount of variance in $T_{(i-2)}$ that is **not explained** by the variance in $T_{(i-1)}$

The concept of 'variable 2' might seem counterintuitive (Sachin, 2023). To put it in simple terms, sales from one day don't necessarily predict the next day's sales, just as sales from the previous month or year might not directly forecast future sales. This is where the PACF becomes vital. It aids in predicting future sales by considering data from the past, even when multiple historical data points are involved. To better grasp the PACF's intricacies, one can plot a graph using the right model parameters, helping to clarify the relationships between different variables (Sachin, 2023).

iii) Seasonal Plots

What are seasonal plots ?

Seasonal plots offer a way to visualise data trends across specific timestamps, such as months or weeks, minute, seconds, spanning multiple years (otexts.com, n.d.). By presenting this data in a single graph, these plots display underlying seasonal patterns (otexts.com, n.d.). Especially when working with extensive time series datasets, seasonal plots are invaluable in identifying recurring trends. A primary reason many researchers turn to them is their efficacy in pinpointing/identifying anomalies. When a time series encompasses a vast range of values, it's easy to overlook inconsistencies or missing patterns. Overlooking these can jeopardise the accuracy of modelling or forecasting results (otexts.com, n.d.). Therefore, seasonal plots serve as a critical tool to ensure comprehensive data analysis.

How to plot a seasonal plot ?

Typically, when working with vast amounts of time series data, researchers select a specific timestamp as their reference point for plotting (otexts.com, n.d.). Following this, they must decide on the seasonal interval that can be in daily, monthly, quarterly, or yearly that is based on

their research objectives and achievements. Once these parameters are set and timestamp is selected, they can proceed to visualise the data, allowing for in-depth observation and analysis.

iv) Seasonal Box Plots

Seasonal box plots offer a compelling way to analyse the distribution and central tendency of data across various seasons (otexts.com, n.d.). While they adeptly highlight disparities within the data, they don't effectively represent patterns compared to seasonal plots (otexts.com, n.d.). Typically, researchers favour seasonal box plots for extensive datasets since they provide a clearer view than seasonal plots and seasonal subseries (otexts.com, n.d.). However, one limitation is that these plots presuppose knowledge of the seasonal periods before visualisation. Fundamental elements of a box plot include the median, the interquartile range (IQR), and any outliers. Interestingly, the process to create a box plot shares many similarities with that of generating a seasonal plot.

v) Seasonal subseries

Seasonal subseries are also known as seasonal plots to diagnose the seasonality of a dataset. It graphically shows the seasonality in different plots. Seasonal subseries are used to break down the time series into individual seasons (otexts.com, n.d.). Doing these steps can show the seasonality as a more precise and clear visualisation of the seasonal pattern as patterns that contain in the time series may not be visible to the researcher (otexts.com, n.d.). That's why using the seasonal subseries can make the seasonality pattern to show more in-depth images. This plot is useful if the seasonality of the plot is already known. If the seasonality is not known, auto-arima can find the seasonality easily (otexts.com, n.d.). Seasonal subseries plots are important if there is a significant seasonality effect in the time series.

c) Decomposition

Time series data are a combination of these components: **Level**, **trend**, **seasonality**, and **noise**, which represent the average value in the series, the underlying pattern in the time series over some time (either decreasing or increasing), a repeating pattern or cycle that occurs within a fixed period and random variation in the series, which is not explained by the trend and seasonality, respectively. All time series data have Level and noise, but seasonality and trend are optional; Level would be disregarded as it is implicitly included in the trend component in decomposition models. The decomposition methods are additive decomposition and multiplicative decomposition. (Brownlee, 2017)

i) Additive decomposition

Seasonal variation for additive decomposition is roughly constant over time, and it assumes that time series is the sum of residual, trend and seasonal. Variance of data does not change over different values of time series. The formula for additive decomposition is. (“Chapter 6 Time Series Decomposition | Forecasting: Principles and Practice (2nd Ed),” n.d.)

$$\text{Additive decomposition}$$
$$y_t = \hat{T}_t + \hat{S}_t + \hat{R}_t$$

ii) Multiplicative decomposition

Assume time series is the product of trend, seasonal and residual components. It is suitable for data with increasing/decreasing trends with repeating seasonal patterns with increasing/decreasing amplitude. (Chourasia, 2020)

Multiplicative decomposition

$$y_t = \hat{T}_t \times \hat{S}_t \times \hat{R}_t$$

Trend
component

Seasonal
component

Residual
component

Before explaining the mechanics of how this could be done by both additive and multiplicative time series, decomposition is more effective when there is a clear trend and seasonality. Decomposition might only be effective if a series has a clear seasonality. (*Why Time Series Decomposition Is Performed*, n.d.)

iii) How Decomposition Works (PennState, n.d.)

1. Estimate the trend
 - Smoothing procedure (moving averages)
 - Model the trend with regression equation
2. “De-trend” the series
 - Multiplicative model
 - Dividing series by trend value
 - Additive model
 - Subtracting the trend estimates from the series
3. Estimate seasonal factor using de-trended series. Such as average the de-trended values for a specific season to prove “season” component.
4. Determine irregular(random) component
 - Additive model
 - Random = Series - Trend - Seasonal
 - Multiplicative model
 - Random = Series / (Trend * Seasonal)

d) Differencing

i) What is differencing ?

Differencing is a technique where the current value in a time series dataset is subtracted from its previous or a lagged value (Lagop, 2023). This method is instrumental in transforming non-stationary time series data into a stationary form. Stationary data, marked by a constant mean, variance, and autocorrelation over time, offers a more predictable basis for models, enhancing the accuracy of forecasts and evaluation results (Lagop, 2023). Utilising non-stationary data can lead to unreliable or misleading outcomes. Achieving stationarity is vital in time series analysis. Not only does it simplify models by reducing the required parameters for accurate predictions and forecasts, but differencing is also recognized as one of the most effective methods for removing seasonal trends (Analysis, S. D., 2023). To ascertain the stationarity of a time series dataset, especially after plotting, tools like the ACF, PACF, or KPSS tests can be employed and evaluated (Lagop, 2023). For instance, by leveraging ACF and PACF to examine the residuals and seasonality, any anomalies, like a point diverging from the typical range, can signal lingering patterns, suggesting the data isn't fully stationary yet. Hence the user has to do a second differencing if necessary that depends on the graph patterns on the plots.

ii) Seasonal differencing

Seasonal differencing targets time series data with the specific intent of eliminating seasonal patterns (Saylor academy, n.d.). This process adjusts the data by accounting for variations between successive seasons. However, it's important to note that differencing isn't universally applicable to all time series data; its necessity varies case by case. Although multiple seasonal differencing can be performed, the extent to which it's applied often hinges on the results from tools like the ACF (Autocorrelation Function) and PACF (Partial Autocorrelation Function) (PennState, n.d.). Some time series datasets might require just a single seasonal differencing, while others may benefit from multiple rounds of the process.

Formula for seasonal differencing

$$\Delta_s y_t = y_t - y_{t-s}$$

$$\Delta_s y_t$$

indicates the seasonal differences series value at time given t

$$y_t$$

indicates the original series that time given as t

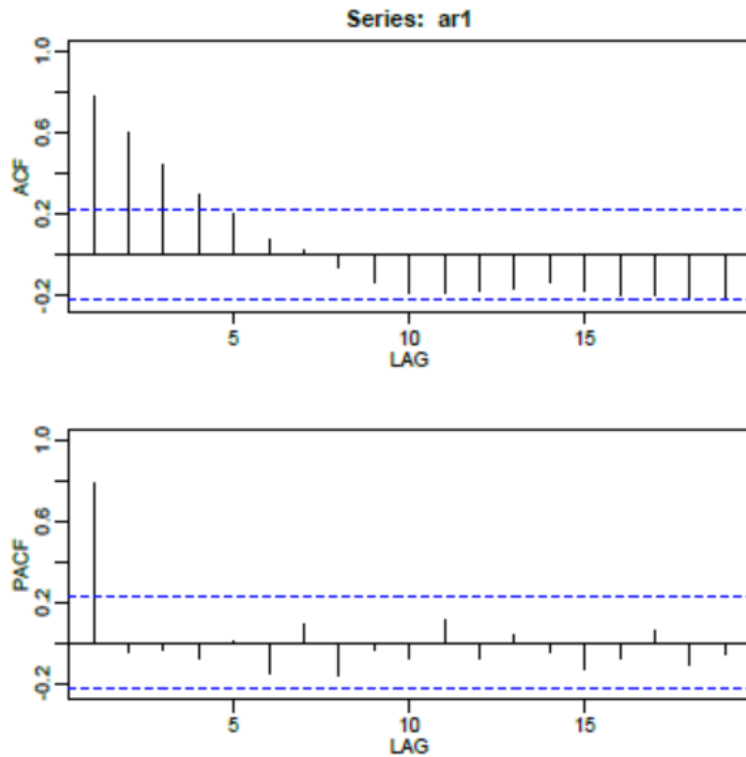
$$y_{t-s}$$

indicates the series value at time given t where indicates the previous season

Seasonal differencing, as a general rule in time series analysis, often employs a seasonal period of 12, symbolising 12 months in a year. Though it primarily targets seasonal patterns, it can be merged with non-seasonal differencing, depending on the specific situation and the resultant graph. The choice to combine the two also hinges on the analyst's understanding of the data. This compatibility stems from the fact that both methods revolve around the principle of differencing.

iii) Non-seasonal differencing

Non-seasonal differencing blends the concepts of differencing, autoregression, and moving average models. Unlike its seasonal counterpart, which emphasises seasonality, non-seasonal differencing focuses on removing trends without accounting for seasonal fluctuations. When working with non-seasonal differencing, yearly data is typically used in lieu of monthly data. To illustrate, while a year comprises 12 months, non-seasonal differencing would use the representation of 1, signifying a full year, because it doesn't factor in the individual seasonal trends within that year.



How does non-seasonal differencing work ?

Three steps are included in the non-seasonal differencing. Usually the first step is doing the first order of differencing which can be a seasonal differencing depending on the requirements of the result that has been plotted after the differencing has been done. We can use the formula above from seasonal differencing to understand better on seasonal differencing in depth. If the acf plot shows a lot of hidden patterns that have not been discovered we can do a second differencing. As shown above, if the ACF plot shows many lines that pass through the blue dotted lines, this can indicate that the model has still hidden patterns after the first seasonal differencing that has been made, which means the time series is still not stationary. Therefore in this scenario it's best to use non-seasonal differencing.

e) Models

i) SARIMA

ARIMA stands for “AutoRegressive Integrated Moving Average.” An ARIMA model is used to make forecasts in a Time Series based on its past data (MastersInDataScience, 2022). ARIMA consists of 3 components: AR, I, and MA. The “AR” (autoregression) model refers to a variable that regresses on its previous values called lags. In short, it makes predictions based on past values. The “I” (integration) model is concerned with achieving stationarity of the series, that is ensuring the statistical properties of the series, e.g. mean, variance and autocorrelation, remain constant over time (MastersInDataScience, 2022). This may require differencing the series 1 or more lags to make the series stationary. The “MA” (moving average) model forms a relationship, using regression, between a residual error at time t with past residual errors. Accordingly, ARIMA has 3 component functions: $AR(p)$ where p is the number of lags to regress, $I(d)$, where d is the number of non-seasonal difference(s) to be performed to make the series stationary, and $MA(q)$, where q is the number of past error terms to be used. When combined, the ARIMA model is referred to as $ARIMA(p, d, q)$. The AR and MA models can also be combined to form the ARMA model. The Seasonal ARIMA (SARIMA) model is used for a time series with a seasonal component. The SARIMA model is referred to as $ARIMA(p, d, q) (P, D, Q)$ where P, D, Q are the seasonal counterparts to the non-seasonal p, d, q (MastersInDataScience, 2022).

ii) ETS

What is the ETS model ?

ETS model stands for error, trend and seasonality, or exponential smoothing. The ETS model is considered as one of the time series families (statsmodel, 2023). ETS is commonly used for local statistical algorithms for time-series datasets (amazon, 2007). Inside the ETS model it requires specific parameters in order for the model to evaluate the information about the dataset. ETS must have either the trend component (T) or seasonal component (S). In a general idea, the ETS model computes a set of weighted averages as the overall observations in the input time of the series dataset in order for the model to predict. As the time increases, the weight average of the observations will also start to decrease rather than the simple moving average in different dataset.

How does the ETS model work ?

In the ETS model there are three main steps which are step 1 decomposition, step 2 model fitting and lastly step 3 forecasting (amazon, 2023). In step 1, decomposition usually happens after we have the pre-processed dataset, that contains no outliers, and all data has been found to have no severe problems such as inconsistent values, typo error etc. Now we break down the time series dataset into the three main components for ETS which are Error, Trend, Seasonality. After we have broken down these three main components, we can now use the train data from the time series dataset to feed into the model for training and evaluate the model for accuracy. After we have trained the model, we have to check the residual to make sure the model contains no additional patterns as the model that has been trained has to be stationary. Usually the model will determine the best fit parameter for us, however there is also another method we can use to find the best fit. The two methods are AIC/BIC (Akaike/Bayesian Information Criterion). After we have trained the model, we can proceed to forecast the result for predicting future periods.

Advantages of the ETS model

- **Flexibility**

The reason why ETS models can have flexibility is because ETS models can actually handle a range of time series datasets by just needing the parameters of Trend or Seasonality.

- **Robustness**

When a time series dataset exhibits unique trend and seasonality parameters, models can become more complex and require longer processing times. However, the presence of these clear and precise parameters often leads to more accurate results.

- **Easy to find out the parameters**

Typically, trend and seasonality can be readily identified and understood using built-in functions and libraries crafted by developers.

Disadvantages of the ETS model

- **Assumptions**

Not just the ETS model, but many models come with assumptions regarding the dataset's nature. We can take 'nsdiff' as an example: even though it's usually generated by a function, we must first confirm that the time series data is stationary before proceeding

with modelling. If the model's assumptions aren't met, the results could be compromised by inaccuracy and lead to bias by the majority of the datasets.

- **Complexity with non linear trends**

ETS models can struggle if the trend is non linear because the various trend points are different from each other thus, making a challenge for ETS models to fit the model.

- **Lack of external factors**

The ETS model solely depends on the past patterns of the time series dataset to make predictions for the model. If the past patterns are weak the predictions can be inaccurate and not solely correct, although the ETS model can predict. However, the result may not be exactly what the research wants and expects.

iii) Multiplicative Holt-Winters

Holt-winters, also known as the Triple Exponential Smoothing method, is used for time series forecasting. This model performs exceptionally well with series with trends and seasonalities. The method consists of three component, which are level (ℓ), the Average value in the series; Trend (b), which is used to indicate decreasing or increasing in the series; and seasonal(s), which indicate repeating short-term cycle in the series. This method involves three smoothing equations, which are

$$\begin{array}{l} \text{(Level)} \quad L_t = \alpha * (Y_t - S_{t-s}) + (1 - \alpha) * (L_{t-1} + b_{t-1}) \\ \text{(Trend)} \quad b_t = \beta * (L_t - L_{t-1}) + (1 - \beta) * b_{t-1} \\ \text{(Seasonal)} \quad S_t = \gamma * (Y_t - L_t) + (1 - \gamma) * S_{t-s} \end{array}$$

The smoothing parameters for the level, trend and seasonal component are represented by α, β, γ . Multiplicative is used when the seasonal variations change proportionally to the series level. As Malaysia is in an equatorial climate and the temperature variation is not extremely high, there are noticeable differences due to monsoon seasons. As the temperature in Malaysia might have

seasonal fluctuations, multiplication is a better fit in this case (7.3 *Holt-Winters' Seasonal Method* | *Forecasting: Principles and Practice*, 2016).

Advantages

- **Applicability**

suitable for various applications such as sales forecasting and resource allocation where seasonality is not constant.

- **Adaptability**

it can be used in many real-world scenarios as long as level, trend, and seasonal components are present

- **Forecasting**

Provide good forecasting performance on datasets with explicit multiplicative relationships between trend and seasonality.

- **Parameter optimisation**

Offer flexibility in optimising smoothing parameters to fit into different datasets.

Disadvantages

- **Overfitting risk**

It might result in poor generalisation to unseen data, especially with optimising smoothing parameters.

- **Non-stationary data**

It is designed for non-stationary data, but it might not be able to capture irregular and highly volatile time series data.

- **Sensitive to the parameter value**

Tuning (smoothing parameter) would significantly influence the model's performance. Different tuning parameters would result in significant differences in the results.

- **Zero or negative values**

The multiplicative model assumes all values are positive if the time series data contains zero or negative values.

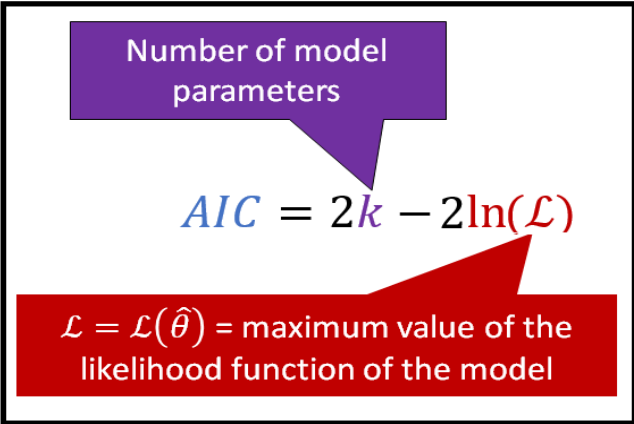
- **Computational intensive**

The model could be more computationally intensive compared to other simpler models.
(Snehal_bm, 2021)

f) Evaluation Metrics

i) AIC

For this project, we will be using the **Akaike information criterion (AIC)** when we are selecting the best ARIMA model. This “goodness-of-fit” metric assesses the quality of a model in comparison with other models (Date, 2021). When a model is fitted to the dataset, some information will be lost. As the AIC measures the amount of information lost, the lower the AIC value of the model, the higher the quality of the model (Wikiwand, n.d.).



The diagram shows the AIC formula $AIC = 2k - 2\ln(\mathcal{L})$ inside a black-bordered box. A purple callout bubble points to the variable k and contains the text "Number of model parameters". A red callout bubble points to the term $\ln(\mathcal{L})$ and contains the text " $\mathcal{L} = \mathcal{L}(\hat{\theta})$ = maximum value of the likelihood function of the model".

$$AIC = 2k - 2\ln(\mathcal{L})$$

$\mathcal{L} = \mathcal{L}(\hat{\theta})$ = maximum value of the likelihood function of the model

Figure 3.f.1: AIC Formula, where $k = p + q + P + Q$, the number of terms estimated in the model

Note: To measure the *accuracy* of our models, we will NOT be using the percentage-based error metrics MAPE and MPE because they are unsuitable for temperature data. MAPE and MPE assume the unit of measurement has a meaningful zero but the zero point in temperature

scales based on Celsius or Fahrenheit is arbitrary. Moreover, MAPE and MPE can result in large or infinite values when the data is zero or close to zero. Instead, we chose **MAE** (Mean Absolute Error), **RMSE** (Root Mean Square Error) and **MASE** (Mean Absolute Scaled Error) as our accuracy metrics. MAE is easy to understand and compute. RMSE is a little more involved but can be understood with a little effort. MASE measures the accuracy of forecasts by comparison with a naive forecast model. It does not have the problems of MAPE and MPE. The following are the more detailed explanations of **MAE**, **RMSE** and **MASE**.

ii) MAE

Also known as the Mean Absolute Error, it represents the average of (absolute) errors (Glen, 2020).

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |x_i - x|$$

Figure 3.f.2: Formula for MAE

n represents the number of errors, Σ represents summation and $|x_i - x|$ represents the absolute errors.

It is used for measuring the accuracy of variables that are continuous. In a set of forecasts, it measures the average magnitude of the errors without regarding their direction (*Mean absolute error (mae) and root mean squared error (RMSE)*, n.d.).

iii) RMSE

Also known as Root Mean Square Error, it represents the standard deviation of the residuals. In other words, it measures the spread of the residuals and this metric is used commonly in forecasting, climatology and regression analysis (Glen, 2023).

$$\text{RMSE} = \sqrt{(f - o)^2}$$

Figure 3.f.3: Formula for RMSE

f represents forecasts while **o** represents observed values.

iv) MASE

Also known as Mean Absolute Scaled Error, it gives each error as a ratio compared to an average error that is baseline. Since it never results in infinite or undefined values, it is a suitable option for series that are intermittent-demand (Glen, 2020).

$$q_t = \frac{e_t}{\frac{1}{n-1} \sum_{i=2}^n |y_i - y_{i-1}|}$$

Figure 3.f.4: Formula for MASE where t = 1...n represents the forecasting sample periods.

g) Tests

i) Ljung Box Test

What is Ljung Box test ?

The Ljung-Box test is a method used to check for the absence of autocorrelation up to a specified lag, k (Glen, 2018). It's an adaptation of the Box-Pierce Test. This test evaluates if a time series dataset might be characterised as white noise, or if the autocorrelation of errors or residuals is different from zero (Glen, 2018). If the autocorrelations show a low residual value, it indicates that the model hasn't adequately fit the time series data and still shows a hidden pattern in the time series. For further validation, the Durbin-Watson test can also be employed to assess the model's fit (Glen, 2018).

When to use the Ljung Box test ?

Usually Ljung Box is used in the context of an autoregressive integrated moving average (Glen, 2018). It can be another technique to verify whether the model has been fitted well as another indicator of different techniques. If the Ljung Box value is small, it indicates the model fitting can still improve dramatically.

Formula of Ljung Box Test

$$Q(m) = n(n + 2) \sum_{j=1}^m \frac{r_j^2}{n - j},$$

From the above formula for Ljung Box, we can also manually calculate the Ljung box value by hand. n is the sample size, whereas r^2 and m are explained below.

- r_j = the accumulated sample autocorrelations,
- m = the time lag.

Ljung Box Test Hypothesis

$$Q > \chi^2_{1-\alpha, h}$$

Where $\chi^2_{1-\alpha, h}$, indicates the chi-square distribution table for the significance level of alpha when h is the degree of freedom.

H₀: The model does not show lack of fitting of the model (the model is fitting fine)

H₁: The model shows a lack of fitting of the model

Therefore if Q is greater than the value calculated for the chi-square distribution table we can reject the null hypothesis and conclude that the model shows a lack of fitting of the model. If the Q value is less than the chi-square value, we fail to reject the null hypothesis and conclude that the model does not show a lack of fitting of the model.

Ljung-Box test

```
data: Residuals from HoltWinters  
Q* = 14.957, df = 17, p-value = 0.5986
```

```
Model df: 0.    Total lags used: 17
```

Based on the Ljung-Box test result above:

For this example, the Q^* value is 14.957. The total number of lags used is 17, representing the checking for autocorrelation from lag1 to lag 17. “Model df:” indicates several parameters estimated in the model. In this example, no parameter is used (Glen, 2018). Based on the chi-square with a significance level of 0.05 and the number of lags of 17, the value is 27.587. In this case, the Q value (14.957) is lower than 27.587, so we accept the null hypothesis and suggest no significant autocorrelation up to 17 lags. P-value is a measure of evidence against a null hypothesis. In this case, the p-value is 0.5986, which is greater than 0.05, so we fail to reject the null hypothesis and conclude there is no autocorrelation up to 17 lags.

ii) Coeftest

For this assignment, we will be using a function called `coefest` to test estimated coefficients. This can be done by performing z and t tests of the coefficients that were estimated (RDocumentation, n.d.).

$$H_0: \beta_i = 0$$

$$H_A: \beta_i \neq 0$$

Assuming that the threshold is $\alpha = 0.05$, if the p-value of the z-test or t-test is smaller than 0.05, we reject H_0 and conclude there is a significant relationship between the predictor and response variable (Zach, 2022).

4. Data Sources

The Malaysia Average Temperatures (1901-2021) dataset was downloaded from the world bank.org website at: <https://climateknowledgeportal.worldbank.org/download-data>

Below are screenshots of the first and last records in the dataset, showing monthly average temperatures (celsius) for each year. Originally, the month column labels were: Jan...Dec. They were replaced with: M01...M12 because after conversion to the “long” format (explained in the next section), we want to sort months in chronological order. We cannot do that with Jan...Dec.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	year	M01	M02	M03	M04	M05	M06	M07	M08	M09	M10	M11	M12
2	1901	25.07	25.37	25.86	26.29	26.44	26.12	25.83	25.82	25.75	25.71	25.49	25.21
3	1902	25.03	25.25	25.85	26.28	26.44	26.16	25.83	25.84	25.75	25.73	25.49	25.29
4	1903	25.11	25.3	25.86	26.28	26.47	26.1	25.75	25.8	25.66	25.62	25.39	25.01
5	1904	24.87	25.17	25.76	26.09	26.29	26.05	25.78	25.73	25.78	25.61	25.34	25.06
6	1905	25.06	25.32	25.86	26.3	26.42	26.09	25.74	25.78	25.68	25.66	25.41	25.35
7	1906	25.14	25.46	25.87	26.37	26.49	26.09	25.8	25.81	25.69	25.61	25.38	25.2
8	1907	24.98	25.3	25.72	26.19	26.34	26.07	25.73	25.71	25.68	25.66	25.46	25.11
9	1908	25.04	25.34	25.78	26.24	26.33	26.04	25.75	25.79	25.71	25.66	25.36	25.18
10	1909	25.06	25.36	25.8	26.24	26.36	26.12	25.73	25.82	25.67	25.67	25.37	25.08
11	1910	25.08	25.33	25.75	26.17	26.33	26.08	25.8	25.79	25.66	25.62	25.4	25.12
12	1911	25.12	25.65	26	26.45	26.47	26.33	26.23	25.93	25.95	25.85	25.98	25.65
13	1912	25.66	25.83	26.45	26.6	26.64	26.27	25.97	25.95	25.96	25.85	25.69	25.55
14	1913	25.34	25.74	26.34	26.25	26.46	26.08	25.97	26.01	25.96	25.97	25.79	25.64
15	1914	25.27	25.83	26.15	26.53	26.99	26.13	26.09	26.24	26.13	26.06	26.08	25.89
16	1915	25.43	25.78	26.17	26.62	26.72	26.54	25.96	26.38	25.98	25.96	25.96	25.49
17	1916	25.17	25.77	26.04	26.47	26.35	26.13	25.65	25.89	25.97	25.97	25.78	25.46

Figure 4.1: First Few Records of Dataset (1901 - 1916)

107	2006	25.71	26.05	26.49	26.46	26.35	26.21	26.75	26.55	26.08	26.17	26.12	26.19
108	2007	25.63	25.91	26.44	26.71	26.93	26.53	26.26	26.34	26.17	26.15	25.56	25.64
109	2008	25.73	25.36	25.68	26.32	26.52	26.13	25.93	26.25	26.26	26.17	25.98	25.63
110	2009	25.16	25.9	26.05	26.79	26.81	26.95	26.44	26.67	26.72	26.32	26	25.8
111	2010	25.85	26.6	26.92	27.24	27.46	26.75	26.02	26.35	26.25	26.28	25.92	25.46
112	2011	25.35	25.66	25.8	26.46	26.68	26.71	26.59	26.56	26.43	26.16	26.24	25.83
113	2012	25.91	26.1	26.28	26.56	26.95	26.92	26.29	26.54	26.48	26.31	26.24	26.17
114	2013	26.13	26.07	27.04	27.07	27.06	27.21	26.41	26.55	26.4	26.09	26.09	25.81
115	2014	24.94	25.85	26.53	26.98	27.08	27.39	26.74	26.2	26.54	26.29	26.3	25.97
116	2015	25.44	25.83	26.61	27.1	27.33	27.14	27.07	26.65	26.84	26.73	26.39	26.48
117	2016	26.66	26.71	27.3	27.78	27.63	26.92	26.81	27.26	26.62	26.69	26.41	26.27
118	2017	26.19	26.01	26.32	26.76	27.12	26.7	27.01	26.47	26.44	26.5	26.2	25.98
119	2018	25.54	25.87	26.58	27.04	26.9	26.71	26.99	27.1	26.27	26.35	26.53	26.32
120	2019	26.43	26.51	26.99	27.54	27.58	27.07	26.82	26.95	26.81	26.3	26.49	26.04
121	2020	26.57	26.55	27.15	27.39	27.59	26.59	26.39	26.91	26.31	26.34	26.22	26.02
122	2021	25.46	25.92	26.56	26.82	27.07	26.81	26.93	26.35	26.32	26.81	26.25	26.11

Figure 4.2: Last Few Records of Dataset (2006 - 2021)

5. Data Analysis

a) Data Re-Formatting

Minor changes were applied to the original data file (in CSV format) downloaded before it is used for analysis. The original data contained 2 lines/rows of header info (country name and nature of the data i.e. average temperatures) which were deleted from the file.

In addition, the Month column labels were changed from “Jan”, ... , “Dec” to “M01”, ... , “M12”. The reason for these changes is because the data downloaded is in a “wide” format (like a pivot table) which has to be converted to a “long” format before it can be converted into a Time Series object later. The month labels will be transposed into the Month column during the conversion process. If the month labels were left in their original form (“Jan”, ... , “Dec”), it will not be possible to sort the rows in chronological order. Hence the relabelling. A year column label was added as well to the first column.

Once the data is in the “long” format (3 columns: Year, Month, Temperature), it is used to create a Time Series object with frequency of 12 and with temperature data starting from the first month of 1901 and ending in the 12th month of 2021.

We decided to work with a smaller, more recent subset of the data as it is more relevant and manageable. The window function was used to subset the data, starting from the first month of 2012 to the 12th month of 2021, resulting in a 10-year dataset.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	year	M01	M02	M03	M04	M05	M06	M07	M08	M09	M10	M11	M12
2	1901	25.07	25.37	25.86	26.29	26.44	26.12	25.83	25.82	25.75	25.71	25.49	25.21
3	1902	25.03	25.25	25.85	26.28	26.44	26.16	25.83	25.84	25.75	25.73	25.49	25.29
4	1903	25.11	25.3	25.86	26.28	26.47	26.1	25.75	25.8	25.66	25.62	25.39	25.01
5	1904	24.87	25.17	25.76	26.09	26.29	26.05	25.78	25.73	25.78	25.61	25.34	25.06
6	1905	25.06	25.32	25.86	26.3	26.42	26.09	25.74	25.78	25.68	25.66	25.41	25.35
7	1906	25.14	25.46	25.87	26.37	26.49	26.09	25.8	25.81	25.69	25.61	25.38	25.2
8	1907	24.98	25.3	25.72	26.19	26.34	26.07	25.73	25.71	25.68	25.66	25.46	25.11
9	1908	25.04	25.34	25.78	26.24	26.33	26.04	25.75	25.79	25.71	25.66	25.36	25.18
10	1909	25.06	25.36	25.8	26.24	26.36	26.12	25.73	25.82	25.67	25.67	25.37	25.08
11	1910	25.08	25.33	25.75	26.17	26.33	26.08	25.8	25.79	25.66	25.62	25.4	25.12
12	1911	25.12	25.65	26	26.45	26.47	26.33	26.23	25.93	25.95	25.85	25.98	25.65
13	1912	25.66	25.83	26.45	26.6	26.64	26.27	25.97	25.95	25.96	25.85	25.69	25.55
14	1913	25.34	25.74	26.34	26.25	26.46	26.08	25.97	26.01	25.96	25.97	25.79	25.64
15	1914	25.27	25.83	26.15	26.53	26.99	26.13	26.09	26.24	26.13	26.06	26.08	25.89
16	1915	25.43	25.78	26.17	26.62	26.72	26.54	25.96	26.38	25.98	25.96	25.96	25.49
17	1916	25.17	25.72	26.04	26.47	26.35	26.13	25.65	25.89	25.97	25.97	25.78	25.46

Figure 5.a: Data in “Wide” Format

b) Detection of Missing Values

After that, we used the `ggplot_na_distribution()` plot to visualise whether there are any discontinuities in the data. With the `ggplot_na_distribution()` plot, we can get an overview of the locations of the missing values in the time series and their distribution. It also allows us to know how many missing values are in various intervals of the time series. If there are missing values, they will be indicated as highlighted regions.

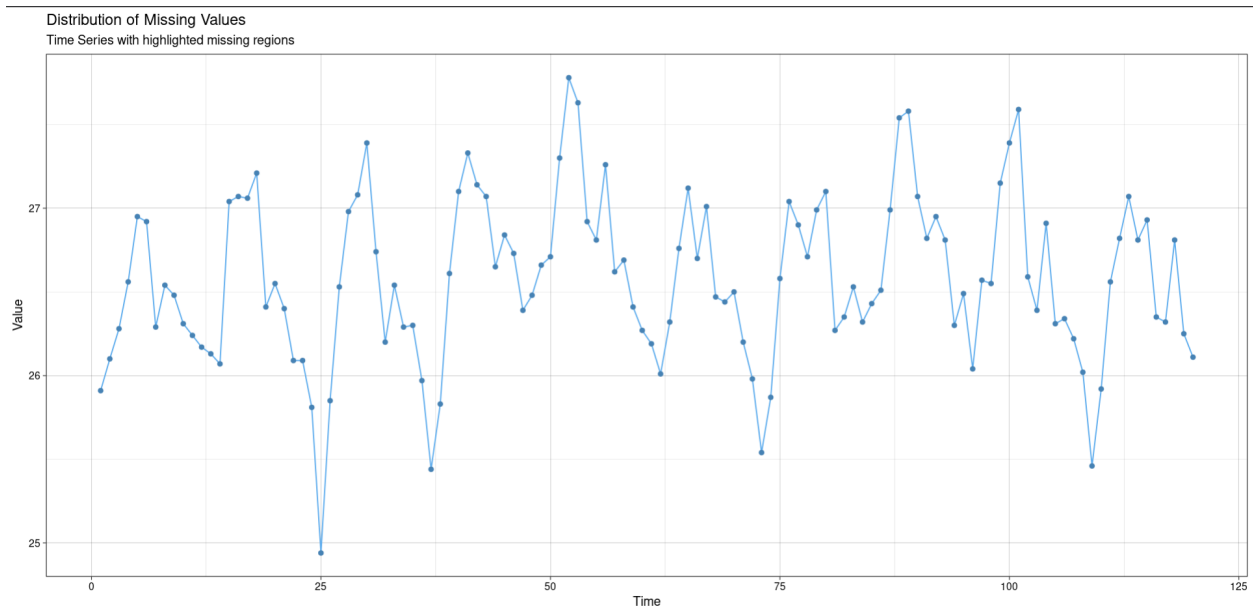


Figure 5.b.1: Visualisation of Missing Values in Time series

Since no highlighted regions can be found in **Figure 5.b.1**, we can conclude that there are no missing values.

What will be the impact if missing values are contained in the time series ?

It can lead to biased estimators and incorrect inference because the data is not stationary (Adrian_PAdrian_P, KontorusKontorus, & bappers2bappers2, 1962). The dataset may also not be enough for the model to be trained. If the data has missing values, the accuracy of the model can also be drastically affected. Forecasting can also be severely affected because forecasting depends on the previous data in time series in order to predict.

c) Exploratory Data Analysis

Next, we plotted the time series for the entire new dataset (2012 to 2021). From Figure 5.c.1, we can see that there is a seasonality pattern with peak temperatures occurring just before the middle of the year and troughs occurring in the cross-over period between one year to the next.

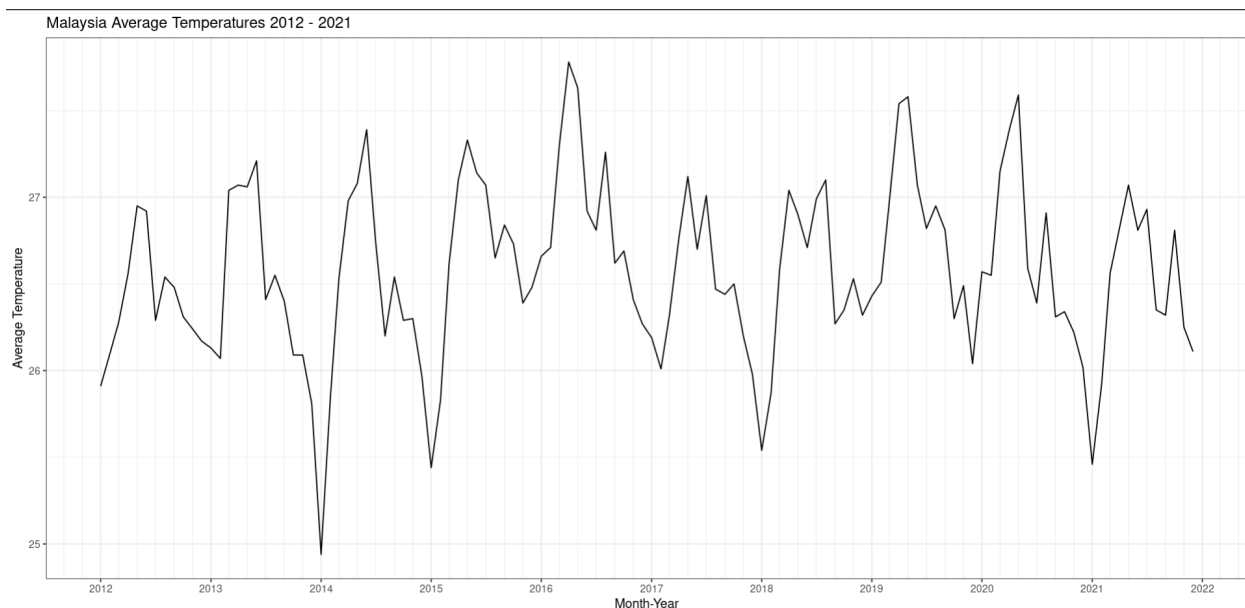


Figure 5.c.1: Plot of Time Series Data from 2012 to 2021

To further analyse the existence of trends and patterns, we use the `stl()` function to decompose the time series into its components.

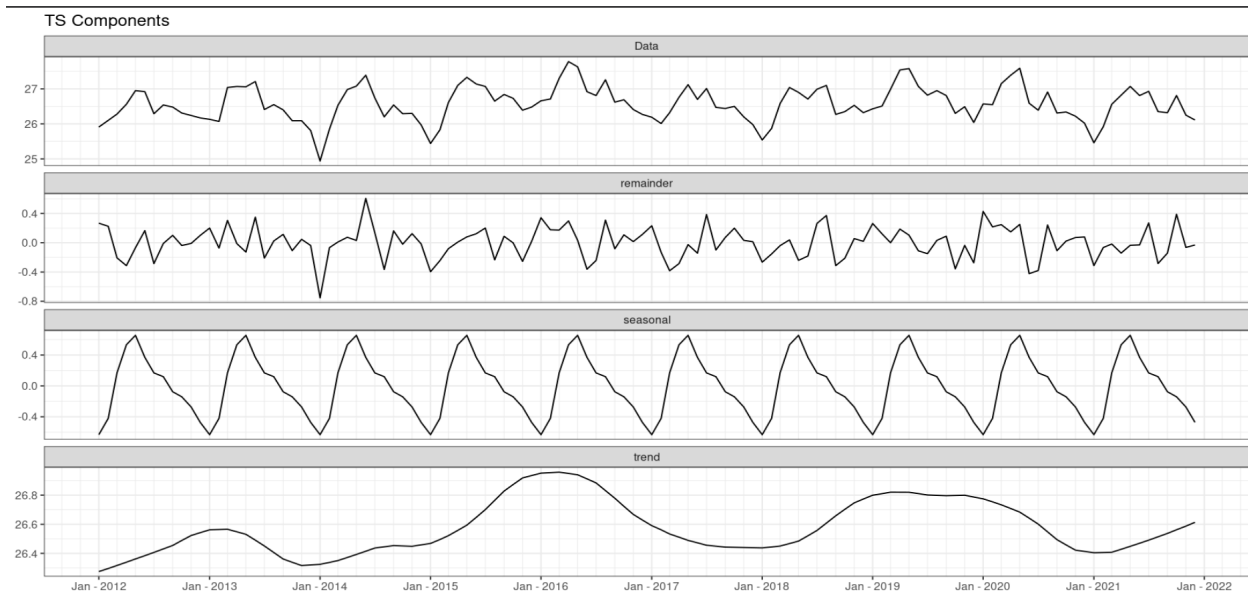


Figure 5.c.2: Decomposition of Time Series

From Figure 5.c.2, we can see that there is definitely a seasonal pattern but it is uncertain whether or not there is a trend. To see if our observations are correct, we calculated the Trend Strength and Seasonal Strength.

F_t: Trend Strength

$$F_T = \max \left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(T_t + R_t)} \right)$$

F_s: Seasonal Strength

$$F_S = \max \left(0, 1 - \frac{\text{Var}(R_t)}{\text{Var}(S_t + R_t)} \right)$$

```
> data.frame('Trend Strength' = TrSt , 'Seasonal Strength' = SnSt)
  Trend.Strength Seasonal.Strength
1             0.5              0.8
```

From the calculations, the Trend Strength is 0.5 whereas the Seasonal Strength is **0.8**. Since 1 is the highest value, we can see that there is quite a strong seasonality in the time series.

To visualise this seasonality in further details, we generated the Seasonal Plot (**Figure 5.c.3**) where each line represents each year. This plot further reinforces our earlier observation that a trough occurs in the crossover period between one year to the next and peaks occur around the beginning of May.

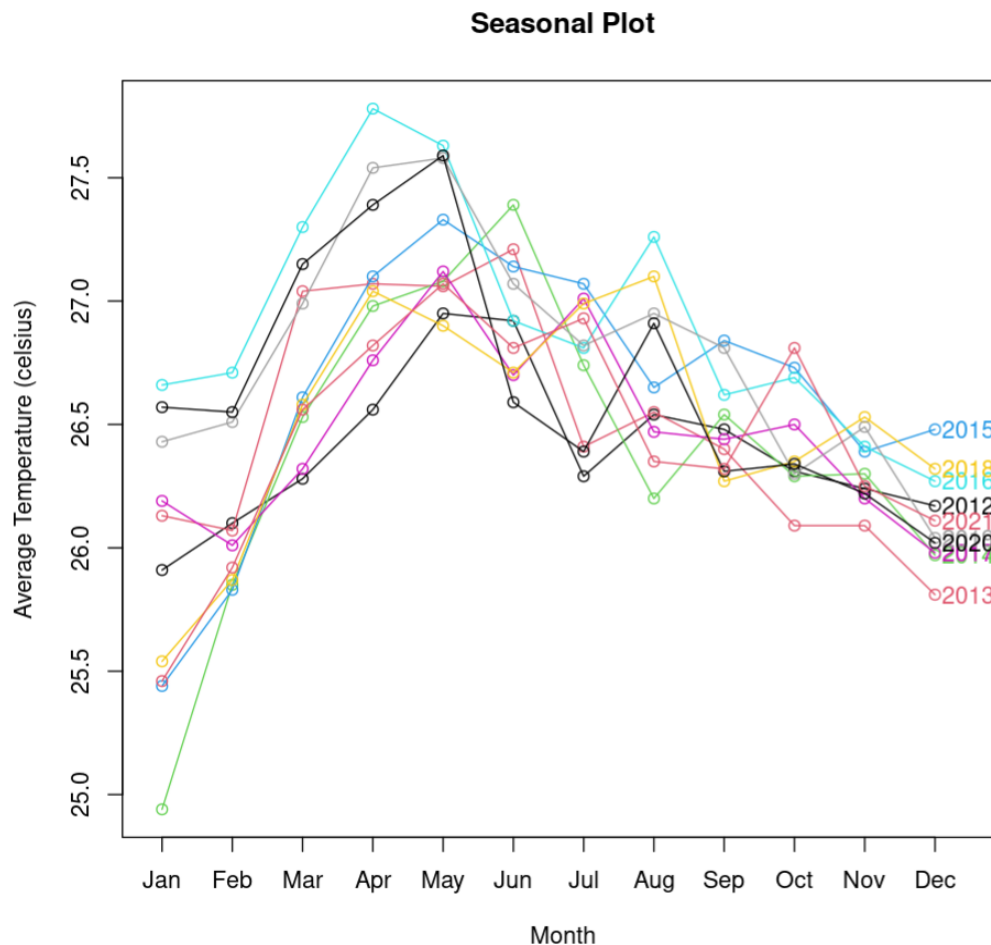


Figure 5.c.3: Seasonal Plot of Time Series Data

Two further plots, Seasonal Subseries Plot (**Figure 5.c.4**) and Seasonal Box Plot (**Figure 5.c.5**), were generated. The horizontal lines in both plots represent the temperature means for a duration of 1 month.

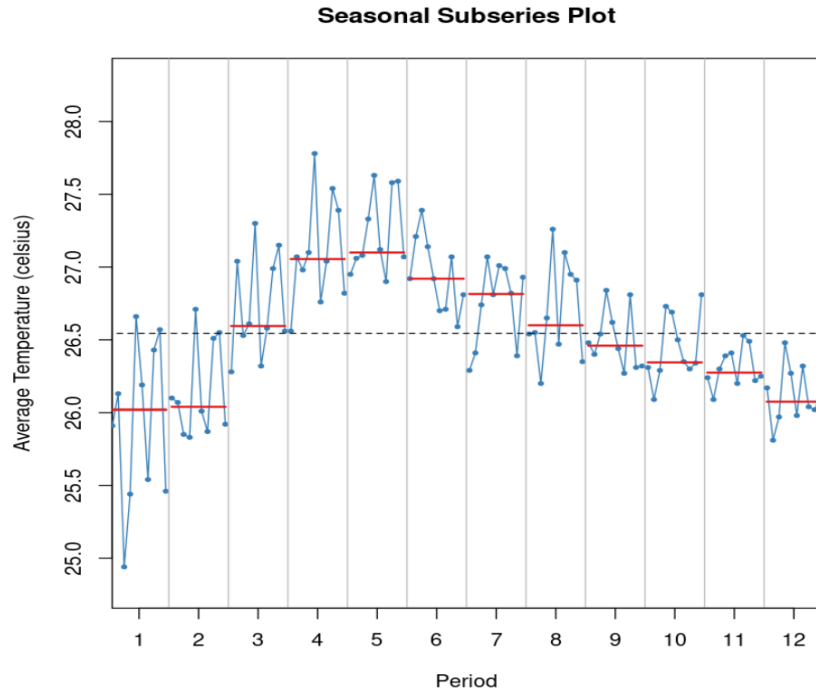


Figure 5.c.4: Seasonal Subseries Plot of Time Series Data

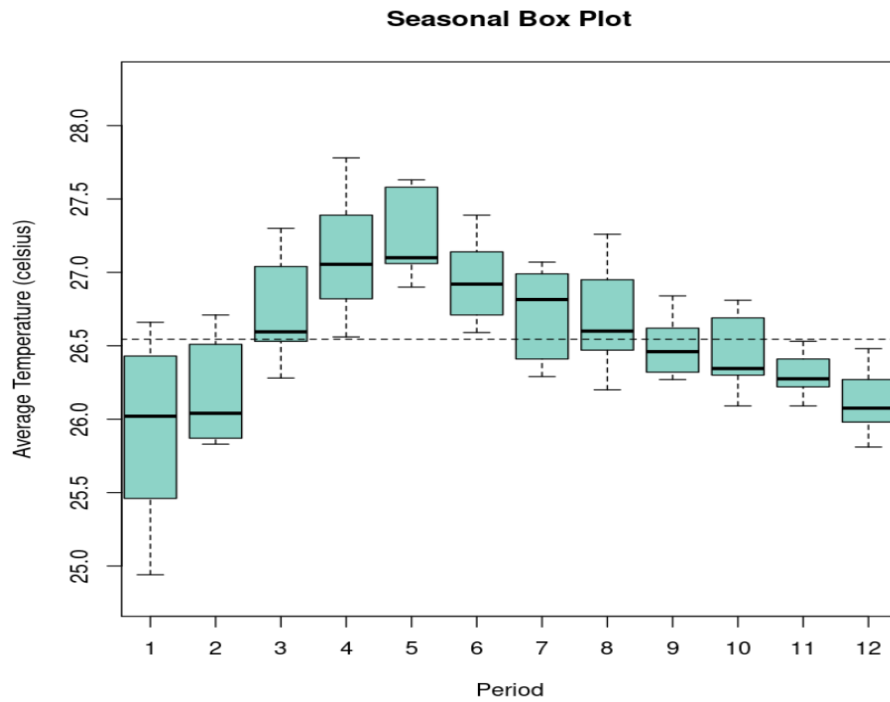


Figure 5.c.5: Seasonal Box Plot of Time Series Data

These 2 plots further confirm the existence of a seasonality pattern (**note that the text output below for each plot indicates “Evidence of seasonality: TRUE”**).

```
> #Seasonal Sub-Series Plot
> seasplot(tempdata_ts, outplot = 3, trend = FALSE,
+           main = "Seasonal Subseries Plot", ylab= "Average
Temperature (celsius)")
Results of statistical testing
Presence of trend not tested.
Evidence of seasonality: TRUE (pval: 0)
>
> #Seasonal Boxplot
> seasplot(tempdata_ts, outplot = 2, trend = FALSE,
+           main = "Seasonal Box Plot", ylab= "Average Temperature
(celsius)")
Results of statistical testing
Presence of trend not tested.
Evidence of seasonality: TRUE (pval: 0)
```

d) Checking the Stationarity of the Data

At this point, we split our new subsetting data (2012 to 2021) into 80:20 proportions. The earlier 80% portion (2012 to 2019) will be used for training the model whereas the later 20% will be used as a test set (2020 to 2021) for the models.

Before we can proceed to modelling, we want to check whether or not our time series is *stationary*. We ran the `nsdiffs()` function on the training set to determine how much seasonal differencing can be done in order to make the series stationary. The value returned is 1, meaning 1 seasonal difference.

```
> nsdiffs(tempdata_train)
[1] 1
```

However, before performing any differencing, we want to examine the ACF (Autocorrelation Function) and PACF (Partial Autocorrelation Function) graphs. Referring to **Figure 5.d.1**, the ACF graph shows that the autocorrelations are significant (cuts the dotted lines) for quite a number of lags, decreasing quite gradually. This is a characteristic of a Non-Stationary series, compared to a Stationary series whose autocorrelations drop to zero quite rapidly. Moreover, there are large autocorrelations at the seasonal lags (12, 24, 36, 48), indicating that there is

seasonality in the series. Neither the Augmented Dickey-Fuller and KPSS tests were performed as they are only meant to check the stationarity of *non-seasonal* data.

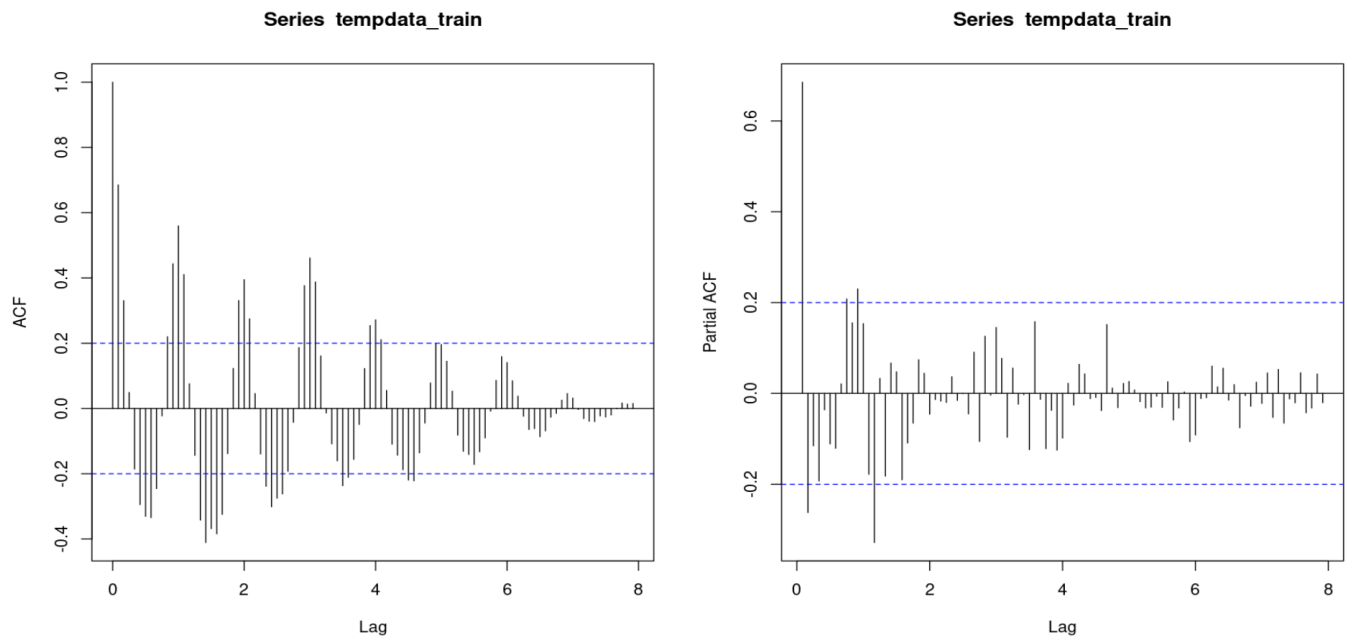


Figure 5.d.1: ACF and PACF Before Any Differencing

Based on the observations in the paragraph above, we start with 1 seasonal differencing and see what the effect it has.

```
> tempdata_train_d12 <- diff(tempdata_train, 12)
```

We now proceed to plot the *differenced* time series (**Figure. 5.d.2**).

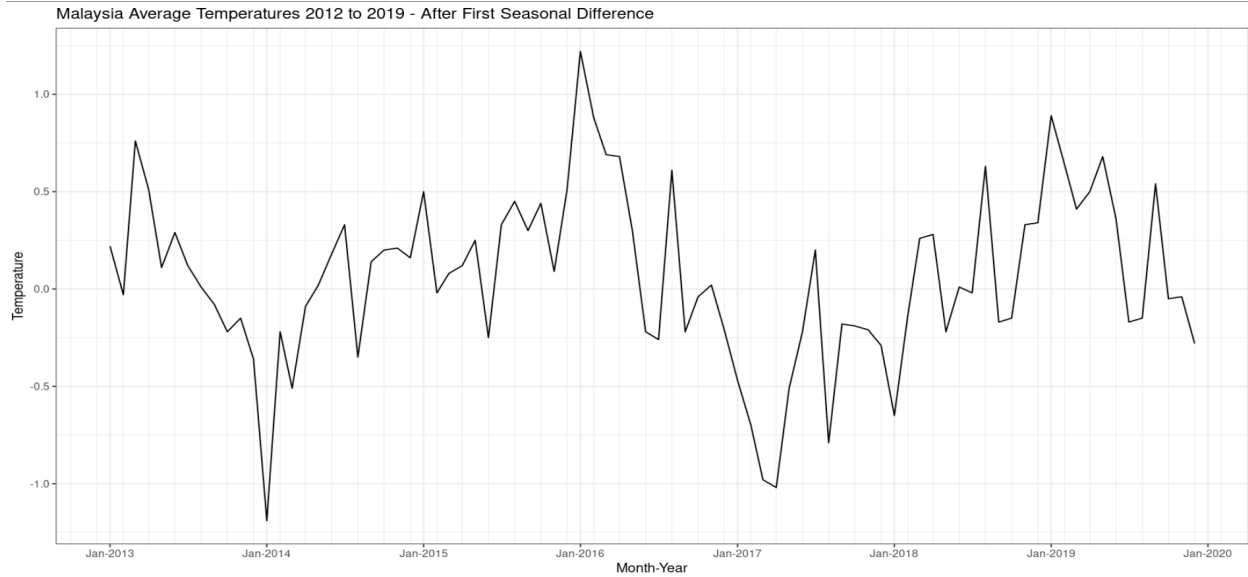


Figure 5.d.2: Time Series Data After One Seasonal Differencing

From **Figure 5.d.2**, we observe that the variances are not uniform. This can be further supported by **Figure 5.d.3**, which still shows some signs of the autocorrelations of the data decreasing gradually as the number of time lags increases.

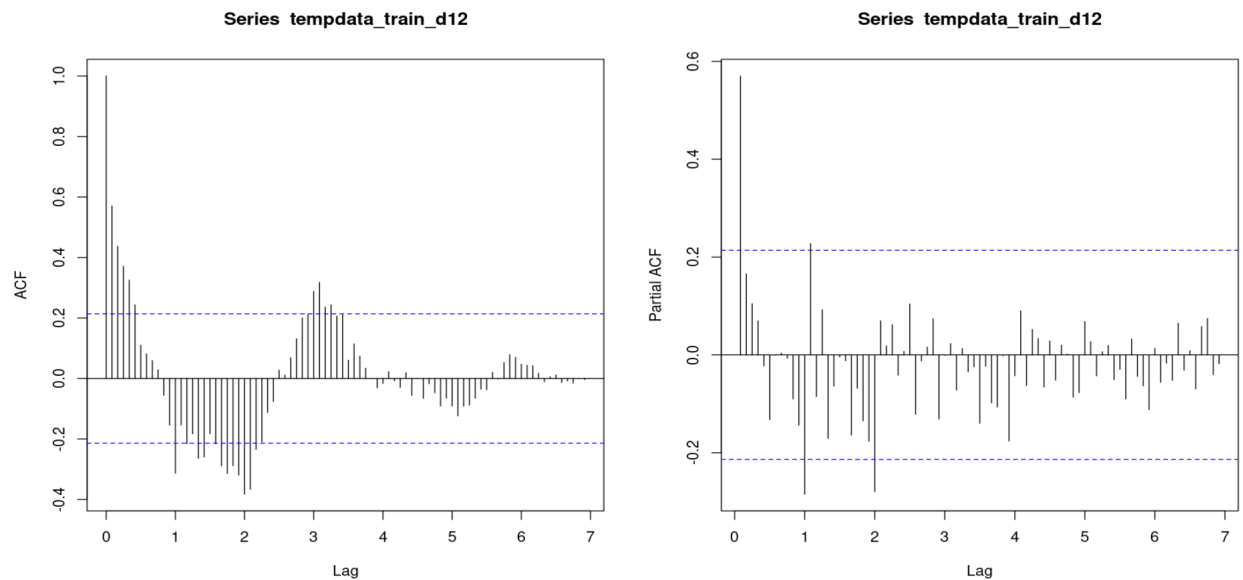


Figure 5.d.3: ACF and PACF After One Seasonal Differencing

To stabilise these variances, we perform 1 Non-Seasonal Difference and generate the plot again. In **Figure 5.d.4**, the variances now appear to be more centred around zero (the mean).

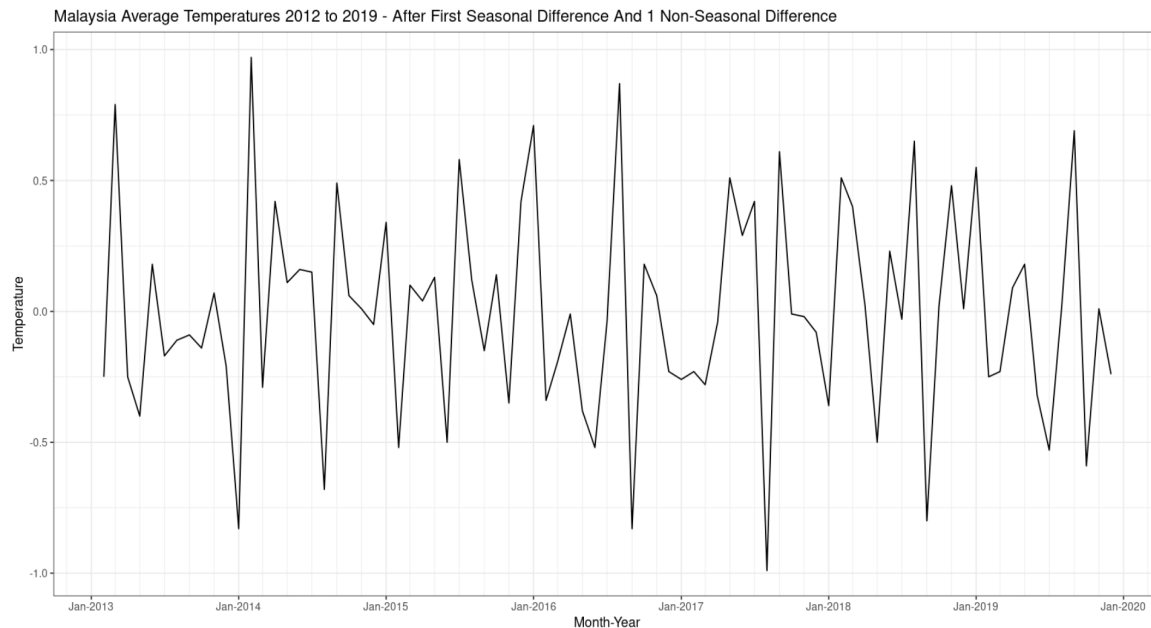


Figure 5.d.4: Time Series Data After An Additional Non-Seasonal Differencing

Now, let's review the ACF and PACF to see what changes have taken place (**Figure 5.d.5**).

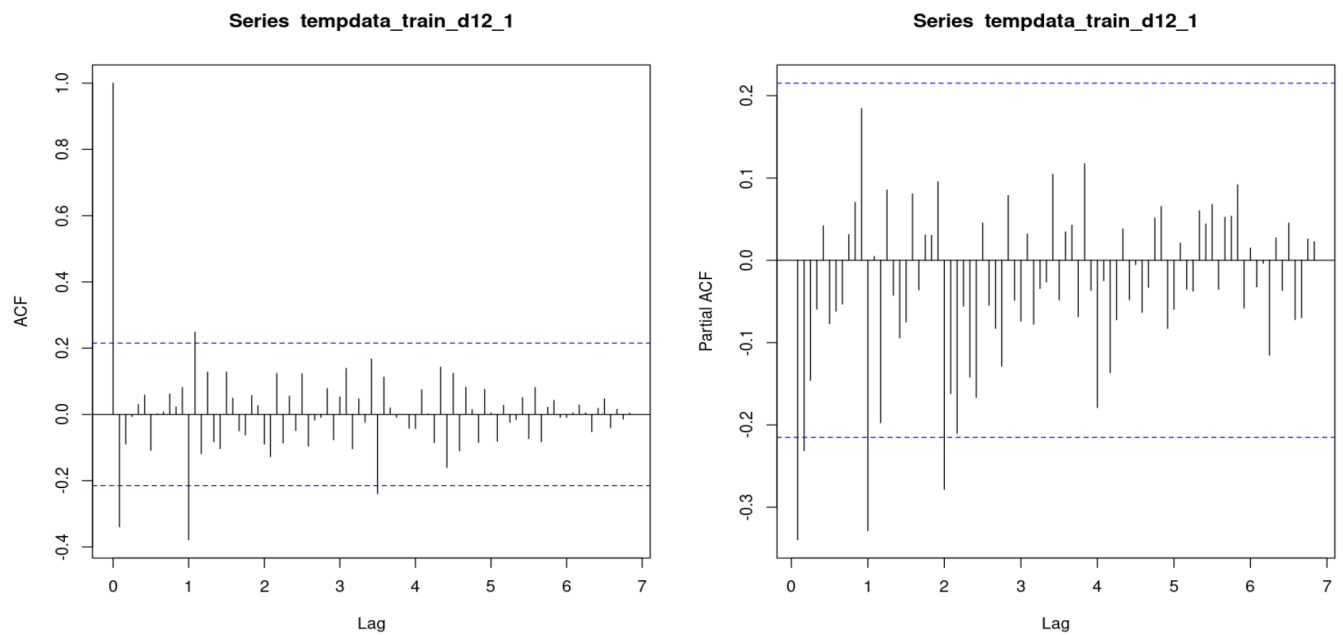


Figure 5.d.5: ACF and PACF After An Additional Non-Seasonal Differencing

Based on **Figure 5.d.5**, we can see that the data is made stationary because the autocorrelations of the data drop to zero relatively quickly. The autocorrelations no longer decrease slowly as the

number of time lags increases. Therefore, we can conclude that the data is now stationary with a constant mean and variance.

6. Results

a) SARIMA & AUTO ARIMA

i) SARIMA

Referring to **Figure 5.d.5** in the previous chapter, the PACF tapers off with a decay pattern. Hence, a MA process is applicable here. On the ACF graph, there is a spike at lag 1 and then at lag 12. Therefore, we can assume that the ARIMA model's parameters $(p,d,q)(P,D,Q)$ are $(0,1,1)(0,1,1)$ where $D=1$ and $d=1$ correspond to the single Seasonal Difference and single Non-Seasonal Difference respectively. The Non-Seasonal $q=1$ refers to the spike at Lag 1 whereas the Seasonal $Q=1$ refers to the spike at Lag 12.

We now proceed to train the ARIMA model we have identified i.e. ARIMA $(0,1,1)(0,1,1)$. To be certain that this model is the best model, we create additional models by incrementing or decrementing the p, q, P, Q values by 1. This can be seen in **Figure 6.1.a** below.

```
fit1 <- arima(tempdata_train, order = c(0,1,1), seasonal = c(0,1,1))
fit2 <- arima(tempdata_train, order = c(1,1,1), seasonal = c(0,1,1))
fit3 <- arima(tempdata_train, order = c(0,1,1), seasonal = c(1,1,1))
fit4 <- arima(tempdata_train, order = c(1,1,1), seasonal = c(1,1,1))
```

Figure 6.1.a: Additional SARIMA Models

```
Call:
arima(x = tempdata_train, order = c(0, 1, 1), seasonal = c(0, 1, 1))

Coefficients:
      ma1      sma1
 -0.5735 -0.8171
s.e.    0.1049  0.1883

sigma^2 estimated as 0.08314: log likelihood = -21.11, aic = 48.23
```

Result of fit1

```
Call:
arima(x = tempdata_train, order = c(1, 1, 1), seasonal = c(0, 1, 1))

Coefficients:
      ar1      ma1      sma1
  0.1735 -0.6927 -0.8106
s.e.    0.2017  0.1564  0.1822

sigma^2 estimated as 0.08264: log likelihood = -20.72, aic = 49.44
```

Result of fit2

```
Call:
arima(x = tempdata_train, order = c(0, 1, 1), seasonal = c(1, 1, 1))

Coefficients:
      ma1      sar1      sma1
 -0.5680  0.1275 -1.000
s.e.    0.1024  0.1271  0.355

sigma^2 estimated as 0.0735: log likelihood = -20.72, aic = 49.44
```

Result of fit3

```
Call:
arima(x = tempdata_train, order = c(1, 1, 1), seasonal = c(1, 1, 1))

Coefficients:
      ar1      ma1      sar1      sma1
  0.1697 -0.6854  0.1278 -0.9996
s.e.    0.2010  0.1572  0.1267  0.3747

sigma^2 estimated as 0.07284: log likelihood = -20.35, aic = 50.69
```

Result of fit4

Figure 6.1.b: Computer Outputs of the SARIMA Models in Figure 6.1.a

Model	AIC (training set)
arima (0,1,1)(0,1,1)	48.23
arima (1,1,1)(0,1,1)	49.44
arima (0,1,1)(1,1,1)	49.44
arima (1,1,1)(1,1,1)	50.69

Figure 6.1.c: Summary of the Results from Figure 6.1.b

By comparing the AIC values among the 4 models using **Figure 6.1.c** , we find that **ARIMA (0,1,1)(0,1,1)[12]** indeed has the lowest AIC and we will select it as our best SARIMA model.

Next, we check the residuals of the model. This is done by running the `checkresiduals()` function, which generated the graphs in **Figure 6.1.d**. The ACF plot shows that there are no autocorrelations. The residuals plot at the bottom right shows that the residuals are normally distributed.

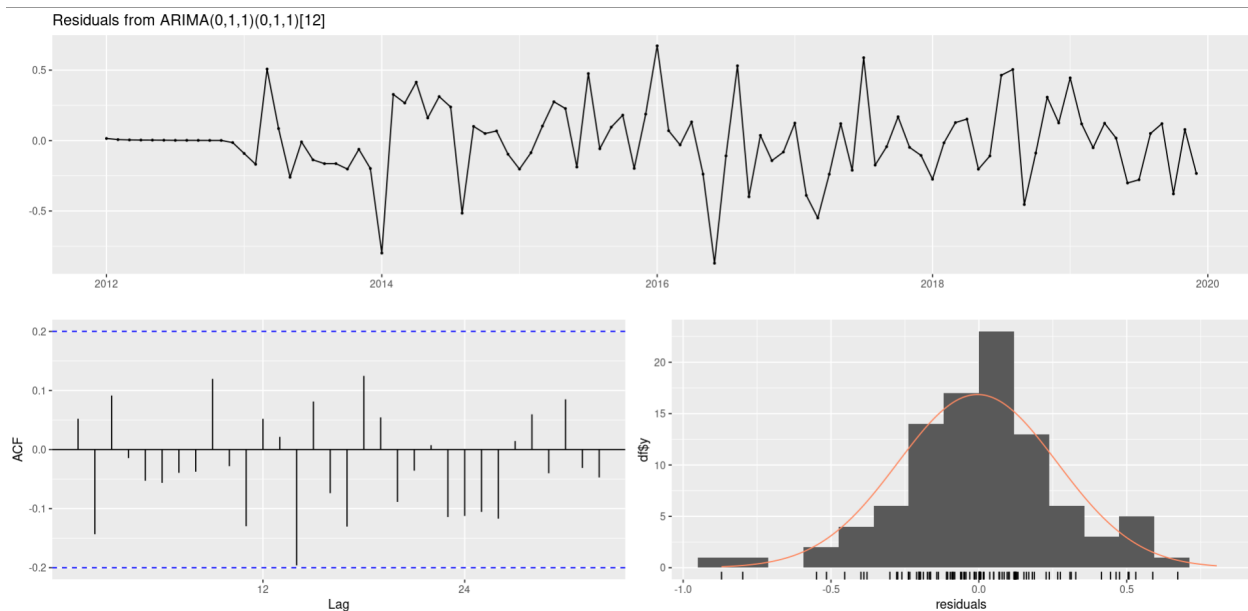


Figure 6.1.d: Result of `checkresiduals()` function on SARIMA model

In addition, the Ljung-Box test in **Figure 6.1.e** (which is also generated by `checkresiduals`) confirms that there are no autocorrelations in the residuals. The p-value of $0.3857 > 0.05$ means that the null hypothesis that the residuals are white noise cannot be rejected.

Ljung-Box test

```
data: Residuals from ARIMA(0,1,1)(0,1,1)[12]
Q* = 18.05, df = 17, p-value = 0.3857
```

```
Model df: 2. Total lags used: 19
```

Figure 6.1.e: Ljung-Box Test Result on SARIMA Model

Next, we examine the evaluation metrics of the SARIMA model.

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	-0.004458134	0.2682867	0.1961117	-0.02406512	0.7379281	0.6417709	0.05211899

Figure 6.1.f: Evaluation Metrics of SARIMA Model on Training Set

ii) AUTO ARIMA

Now that we have completed our manually-tuned ARIMA model, we want to compare it against other alternative models. The most natural one to start with is the `auto.arima` function. We can run `auto.arima` on the training set with default parameters but it will not return a well-tuned model. Therefore, we have tuned several parameters to ensure that `auto.arima` returns a well-tuned model.

```
> auto.arima(tempdata_train, ic="aic", max.order=6, D=1, d=1,
stepwise=FALSE, approximation=FALSE)
Series: tempdata_train
ARIMA(0,1,1)(0,1,1)[12]
Coefficients:
            ma1      sma1
        -0.5735  -0.8171
s.e.      0.1049   0.1883
sigma^2 = 0.08531: log likelihood = -21.11
AIC=48.23   AICc=48.53   BIC=55.48
```

Figure 6.1.g: Result of Auto Arima

The argument **max.order** was set to 6 so that the **maximum value of $p + q + P + Q$ is less than or equal to 6** if the selection of the model is not stepwise (Hyndman et al., n.d.). The argument **approximation** is set to FALSE or else the minimum AICc model will not be discovered because of these approximations or the usage of a procedure that is stepwise (otexts.com, n.d.). The argument **stepwise** is set to FALSE so that a much bigger set of models is explored (otexts.com, n.d.). Based on Figure 6.g, the `auto.arima` function returns a model that is the same as our manually tuned SARIMA model that is **ARIMA (0,1,1)(0,1,1)[12]**.

To evaluate whether the SARIMA model is overfitting or underfitting, we look at the prediction errors on the training and testing datasets.

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	-0.004458134	0.2682867	0.1961117	-0.02406512	0.7379281	0.5715954	0.05211899	NA
Test set	-0.075795569	0.3153403	0.2805038	-0.29697397	1.0572196	0.8175684	0.34317807	0.6967918

Figure 6.1.h: Results of Training and Testing Evaluation Metrics

Evaluation Metric	Training Result	Testing Result	Difference (Test - Train)
RMSE	0.2682867	0.3153403	0.0470536
MAE	0.1961117	0.2805038	0.0843921
MASE	0.5715954	0.8175684	0.245973

Figure 6.1.i: Differences Between Testing and Training Results

We will only compare the results for RMSE, MAE and MASE (our chosen metrics). Based on Figure 6.1.i, we can see that the differences between the testing and training results are almost negligible except for the MASE metric, where there is a larger difference.

b) ETS

After fitting the ETS model with the training dataset, we check the residuals of the model by running the `checkresiduals()` function (results shown in Figure 6.2.a). Although the ACF plot shows that there is only 1 significant spike, most of the spikes are within the significance limits. The residuals plot at the bottom right shows that the residuals are normally distributed.

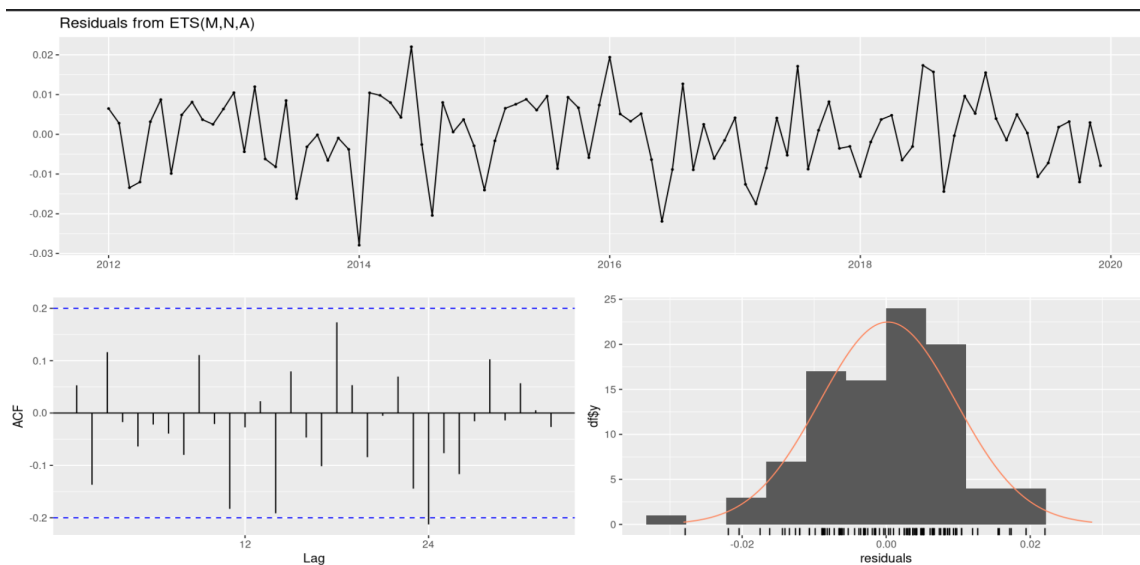


Figure 6.2.a: Result of checkresiduals() function on ETS model

In addition, the Ljung-Box test in **Figure 6.2.b** (which is also generated by `checkresiduals()`) confirms that there are no autocorrelations in the residuals. The p-value of $0.366 > 0.05$ means that the null hypothesis that the residuals are white noise cannot be rejected.

Ljung-Box test

```
data: Residuals from ETS(M,N,A)
Q* = 20.483, df = 19, p-value = 0.366
```

```
Model df: 0. Total lags used: 19
```

Figure 6.2.b: Ljung-Box Test Result on ETS Model

Next, we examine the evaluation metrics of the ETS model.

```
ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
Training set 0.005352646 0.2498626 0.2039062 0.01312114 0.7679375 0.5943136 0.05428529
```

Figure 6.2.c: Evaluation Metrics of ETS Model on Training Set

```
ME      RMSE      MAE      MPE      MAPE      MASE      ACF1 Theil's U
Training set 0.005352646 0.2498626 0.2039062 0.01312114 0.7679375 0.5943136 0.05428529 NA
Test set    -0.058053828 0.3111778 0.2766003 -0.23002321 1.0415712 0.8061910 0.33855151 0.677829
```

Figure 6.2.d: Results of Training and Testing Evaluation Metrics

Evaluation Metric	Training Result	Testing Result	Difference (Test - Train)
RMSE	0.2498626	0.3111778	0.0613152
MAE	0.2039062	0.2766003	0.0726941
MASE	0.5943136	0.8061910	0.2118774

Figure 6.2.e: Differences Between Testing and Training Results

We will only compare the results for RMSE, MAE and MASE (our chosen metrics). Based on Figure 6.2.e, we can see that the differences between the testing and training results are almost negligible except for the MASE metric, where there is a larger difference. The results for the differences in Figure 6.2.e is similar to the results for the differences in Figure 6.1.i.

c) Multiplicative Holt-Winters

After fitting the Multiplicative Holt-Winters model with the training dataset, we check the residuals of the model by running the `checkresiduals()` function (results shown in Figure 6.3.a). The ACF plot shows that there are no autocorrelations. The residuals plot at the bottom right shows that the residuals are normally distributed.

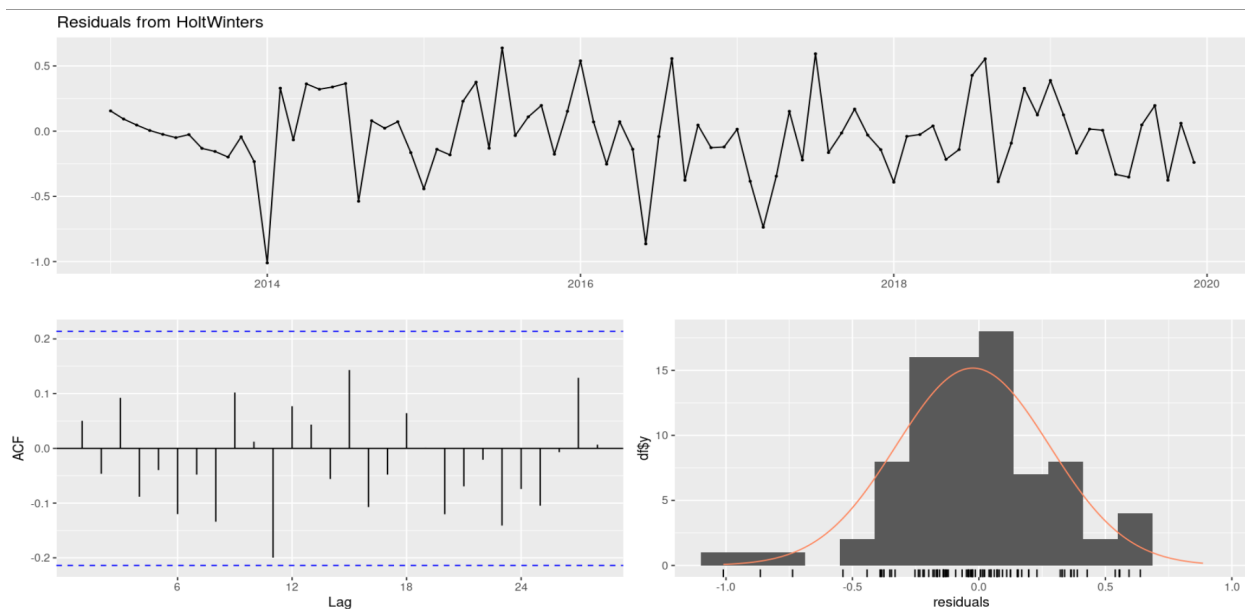


Figure 6.3.a: Result of `checkresiduals()` function on Multiplicative Holt-Winters model

In addition, the Ljung-Box test in Figure 6.3.b (which is also generated by `checkresiduals()`) confirms that there are no autocorrelations in the residuals. The p-value of $0.5986 > 0.05$ means that the null hypothesis that the residuals are white noise cannot be rejected.

Ljung-Box test

```
data: Residuals from HoltWinters
Q* = 14.957, df = 17, p-value = 0.5986

Model df: 0.    Total lags used: 17
```

Figure 6.3.b: Ljung-Box Test Result on Multiplicative Holt-Winters Model

Next, we examine the evaluation metrics of the Multiplicative Holt-Winters model.

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	-0.02414627	0.302593	0.2249439	-0.1001601	0.8480236	0.6556311	0.05034468

Figure 6.3.c: Evaluation Metrics of Multiplicative Holt-Winters Model on Training Set

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	-0.02414627	0.3025930	0.2249439	-0.1001601	0.8480236	0.6556311	0.05034468	NA
Test set	-0.24454148	0.4219497	0.3791228	-0.9327747	1.4335844	1.1050074	0.42722318	0.9772984

Figure 6.3.d: Results of Training and Testing Evaluation Metrics

Evaluation Metric	Training Result	Testing Result	Difference (Test - Train)
RMSE	0.3025930	0.4219497	0.1193567
MAE	0.2249439	0.3791228	0.1541789
MASE	0.6556311	1.1050074	0.4493763

Figure 6.3.e: Differences Between Testing and Training Results

We will only compare the results for RMSE, MAE and MASE (our chosen metrics). Based on Figure 6.3.e, we can see that the errors in the testing results are significantly larger than the errors in the training results. The differences in the errors here are larger than the ones for SARIMA and ETS models. These results may indicate the model is *overfitting* on the training data.

d) Comparison of Models & the Selection of the Best Model

Models Evaluation Metrics	SARIMA/Auto Arima	ETS	Holt-Winters Multiplicative
1) ME			
i) Training	-0.004458134	0.005352646	-0.02414627
ii) Testing	-0.075795569	-0.058053828	-0.24454148
2) RMSE			
i) Training	0.2682867	0.2498626	0.3025930
ii) Testing	0.3153403	0.3111778	0.4219497
3) MAE			
i) Training	0.1961117	0.2039062	0.2249439
ii) Testing	0.2805038	0.2766003	0.3791228
4) MPE			
i) Training	-0.02406512	0.01312114	-0.1001601
ii) Testing	-0.29697397	-0.23002321	-0.9327747
5) MAPE			
i) Training	0.7379281	0.7679375	0.8480236
ii) Testing	1.0572196	1.0415712	1.4335844
6) MASE			
i) Training	0.5715954	0.5943136	0.6556311
ii) Testing	0.8175684	0.8061910	1.1050074
7) ACF1			
i) Training	0.05211899	0.05428529	0.05034468
ii) Testing	0.34317807	0.33855151	0.42722318

Figure 6.5.a: Evaluation Metrics for All Models

i) Based on Results for Training set:

- ETS has the lowest RMSE (0.2498626), followed by SARIMA (0.2682867). Multiplicative Holt-Winters has the highest RMSE (0.3025930).
- SARIMA has the lowest MAE (0.1961117), followed by ETS (0.2039062). Multiplicative Holt-Winters has the highest MAE (0.2249439).
- SARIMA has the lowest MASE (0.5715954), followed by ETS (0.5943136). Multiplicative Holt-Winters has the highest MASE (0.6556311).

ii) Based on Results for Testing set:

- ETS has the lowest RMSE (0.3111778), followed by SARIMA (0.3153403). Multiplicative Holt-Winters has the highest RMSE (0.4219497).
- ETS has the lowest MAE (0.2766003), followed by SARIMA (0.2805038). Multiplicative Holt-Winters has the highest MAE (0.3791228).
- ETS has the lowest MASE (0.8061910), followed by SARIMA (0.8175684). Multiplicative Holt-Winters has the highest MASE (1.1050074).

iii) Choosing the best model:

Among the 3 models we looked into, **SARIMA / ARIMA(0,1,1)(0,1,1)[12]** and **ETS(M,N,A)** are the highest performing models. Based on the accuracy metrics we have chosen (**MAE, RMSE, MASE**), the ETS model performed slightly better than the SARIMA model especially in forecasting the test set although the SARIMA model for the most part, showed higher accuracy in fitting the training set. Despite the slight edge in forecast accuracy that ETS demonstrated for our dataset, **we decided to go with the SARIMA model as our best model**. This is because, if we look at the residuals plot for ETS(M,N,A), on the ACF graph we see that the long negative line at lag 24 breaches the threshold. Moreover, there are 2 other negative lines just before and after lag 12 which almost breach the threshold. Compared to the residual plot of SARIMA where there is no breach of the threshold on the ACF graph and except for 1, most lines are at a considerable distance from the threshold. This means that the residuals are pure white noise and that no autocorrelation exists between the residuals and their lags. The residuals are normally

distributed for both models. From this analysis, we can conclude that the ETS(M,N,A) model failed to completely capture all the patterns that exist in the time series and therefore is still considered as a “work-in-progress”.

e) Equation for the Best Model

The best ARIMA model (and overall model) is: ARIMA (0,1,1)(0,1,1)₁₂. Here, the coefficient of each parameter is checked whether it is significant with the 0.05 significance value (Biradar, 2021). If all coefficients are significant, the model cannot be discarded.

```
Call:
arima(x = tempdata_ts, order = c(0, 1, 1), seasonal = list(order = c(0, 1, 1)))

Coefficients:
          ma1      sma1
        -0.6192  -0.9241
s.e.      0.0863   0.2700
```

Figure 6.e.1: Estimates of the Parameters of the Model

From Figure 6.e.1, we can see the estimated coefficients for the model.

```
z test of coefficients:

      Estimate Std. Error z value Pr(>|z|)
ma1  -0.619231   0.086311 -7.1744 7.263e-13 ***
sma1 -0.924085   0.269988 -3.4227 0.00062 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 6.e.2: Result of z-test for each estimated coefficient in the model

Based on Figure 6.e.2, all of the coefficients of the model are significant as their respective p-values are less than 0.05. This model will not be discarded.

On the left-hand side of the equation, there are no AR terms. The Non-Seasonal Difference is $(1 - B)$ and Seasonal Difference is $(1 - B^{12})$. On the right-hand side, there is a Non-Seasonal MA(1) and Seasonal MA(1).

$$(1 - B)(1 - B^{12})Y_t = (1 - \theta_1 B)(1 - \Theta_1 B^{12})\varepsilon_t$$

We obtain the model coefficients from the one which was run on the *entire* dataset (not just training set). Substituting the MA (θ_1) and SMA (Θ_1) coefficients, which are 0.619231 and 0.924085 respectively, we have:

$$(1 - B)(1 - B^{12})Y_t = (1 - 0.619231B)(1 - 0.924085B^{12})\varepsilon_t$$

The estimated model is:

$$\begin{aligned}(1 - B - B^{12} + B^{13})Y_t &= (1 - 0.619231B - 0.924085B^{12} + 0.572222B^{13})\varepsilon_t \\ Y_t - BY_t - B^{12}Y_t + B^{13}Y_t &= \varepsilon_t - 0.619231B\varepsilon_t - 0.924085B^{12}\varepsilon_t + 0.572222B^{13}\varepsilon_t \\ Y_t &= BY_t + B^{12}Y_t - B^{13}Y_t + \varepsilon_t - 0.619231B\varepsilon_t - 0.924085B^{12}\varepsilon_t + 0.572222B^{13}\varepsilon_t \\ Y_t &= Y_{t-1} + Y_{t-12} - Y_{t-13} + \varepsilon_t - 0.619231\varepsilon_{t-1} - 0.924085\varepsilon_{t-12} + 0.572222\varepsilon_{t-13}\end{aligned}$$

f) Forecasting with the Best Model

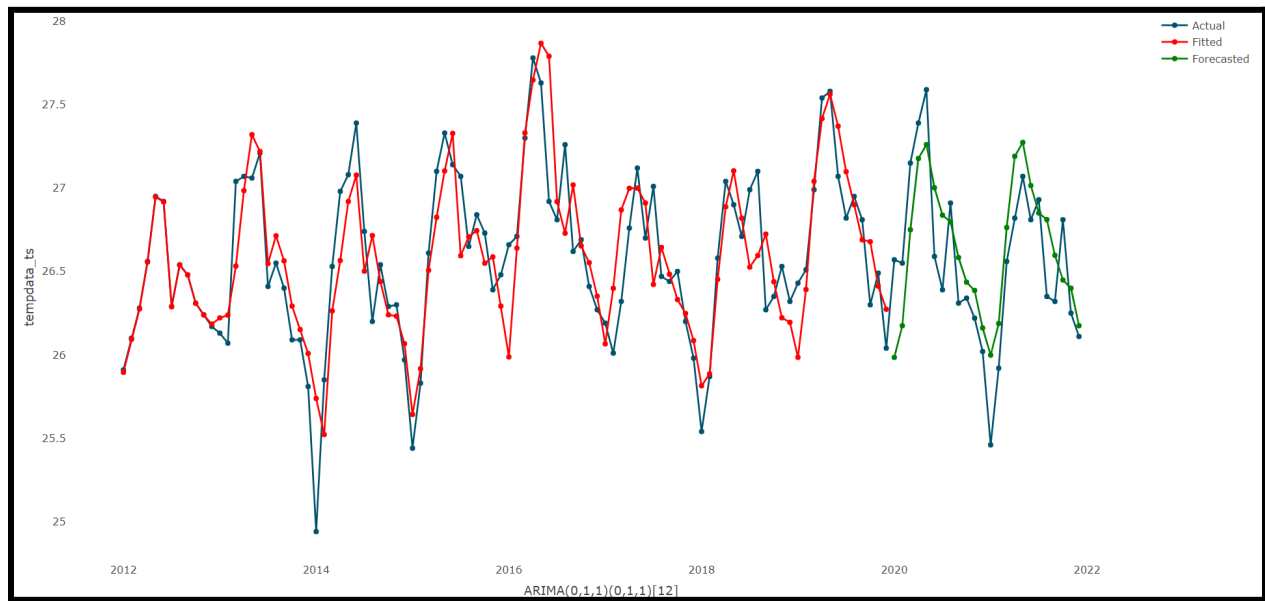


Figure 6.f.1: Result of test_forecast function

Figure 6.f.1 shows a visualisation of the fitted values of the training dataset and the forecast values of the testing dataset against the actual values of the time series. From the figure, we can observe that the SARIMA model is able to fit the training dataset well. When the model forecasts for the years 2020 and 2021, the forecast line is quite close to the “actual values” line. Therefore, we can conclude that the SARIMA model has performed quite well for data that it has not seen before.

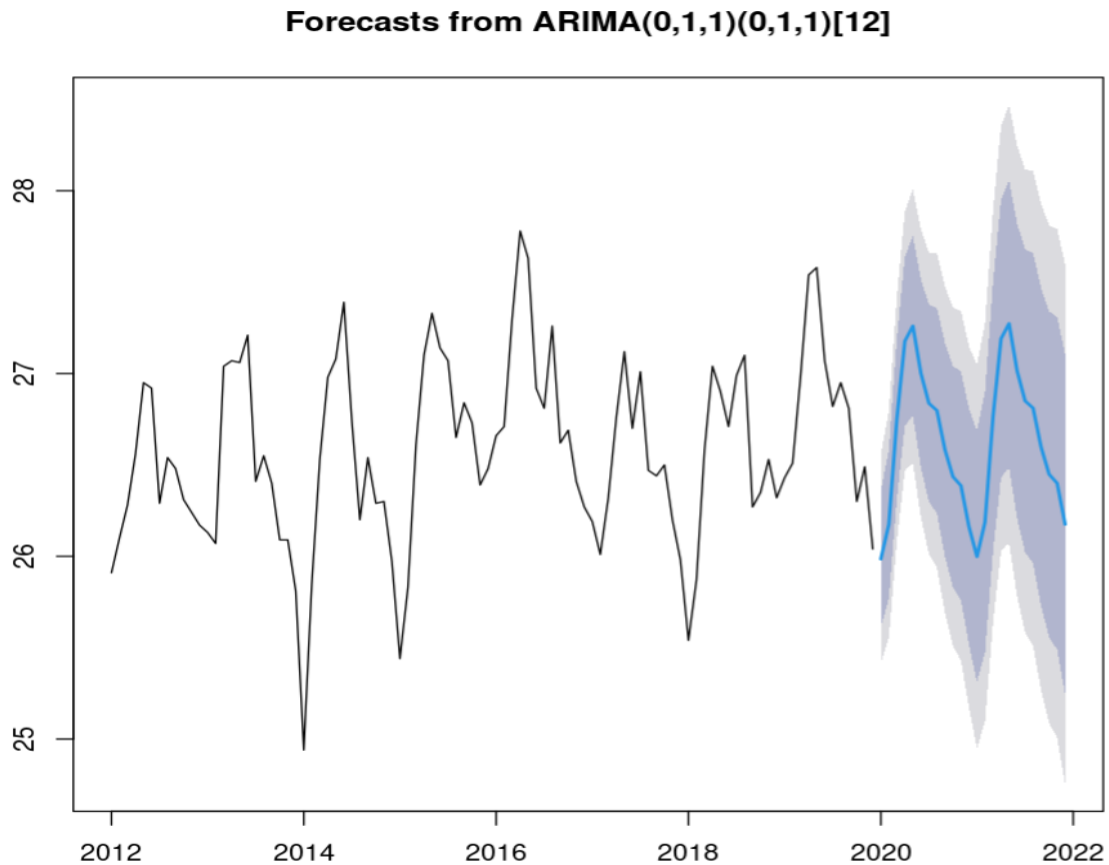


Figure 6.f.2: Result of plot function

Figure 6.f.2 shows the result of plotting historical data with prediction intervals and forecasts for the next 2 years. Based on our observation and the intervals, we cannot infer that the temperature will increase or decrease as we cannot observe either an upward or downward trend. We can still see that there is a seasonality pattern with peak temperatures occurring just before the middle of the year and troughs occurring in the cross-over period between one year to the next.

7. Discussions and Interpretations (may include limitations and recommendations)

The SARIMA / ARIMA(0,1,1)(0,1,1)[12] was selected as the best model for forecasting temperature in Malaysia on the basis of RMSE, MAE and MASE results and the extent to which the residuals are white noise. Based on the model's result on the training set, it was found that it had the lowest MAE and MASE among all the models and its forecast accuracy errors were quite close to those of ETS.

Limitations:

Findings aside, we have also identified several limitations with our problem-solution approach. Firstly, we have only experimented using SARIMA, AUTO ARIMA, ETS and Multiplicative Holt-Winters models. It is possible that there are other models, such as TBATS, which could perform better than our best model. Secondly, although the original dataset actually spans from 1901 to 2021, we have only used 10 years of data for this assignment that is from 2012 to 2021 and we are only forecasting for 2 years, which are 2020 and 2021. This dataset does not have the most recent monthly average temperatures that is for the year 2022. Another concern is that, since we are analysing the *average* temperatures in Malaysia, this means that East Malaysia's temperatures are also included. So, how representative are the *average* temperatures in the dataset?

Recommendations:

Even though we managed to find the best time series model, there is still room for improvement. Firstly, we could experiment using other kinds of models, such as TBATS and Neural Network models, as it is possible for them to have better accuracy than our selected model. Next, we could try to conduct our study using more than 10 years of data. Although 10 years may seem acceptable, using more than 10 years of data may be able to provide more insights in the analysis process. Moreover, forecasting more than 2 years may be able to help us in further assessing the performance of the different models as some models may be reasonably good at short-term forecasting but not so good for longer term forecasts. Lastly, we can use our models on different

datasets (e.g. from other countries) to validate not just our models but also our methodology/approach.

8. Conclusion

The project was intended to investigate rising temperatures in Malaysia, which may be due to climate change. To facilitate this, we downloaded a dataset from world bank.org, which contains Malaysia's Average Temperatures from 1901 to 2021. For our study, we limited the data to 2012 to 2021 as these years are more recent and more relevant and also more manageable.

Based on the time plot of the time series data, we could observe a seasonality pattern with peak temperatures occurring just before the middle of the year and troughs occurring in the cross-over period between one year to the next. The presence of seasonality was also confirmed by the “nsdiffs” function and also the calculation of seasonal strength (0.8). From the ACF and PACF, we confirmed that the time series is non-stationary. As we intended to build a SARIMA model for forecasting, we performed 1 seasonal difference on the data. After that, we observed that the variances were unstable. Since our earlier calculations revealed a trend strength of 0.5, we decided to further difference that data by performing 1 non-seasonal difference. After that, the variances were stabilised and more centred around 0 (mean). From the differenced data, we re-generated the ACF and PACF which indicated that the suitable SARIMA model may be $ARIMA(0,1,1)(0,1,1)$. This was further confirmed by fitting several other SARIMA models and checking their AIC values. Besides our manually-tuned SARIMA model, we also use the Auto ARIMA function (with tuned hyperparameters) to find the best SARIMA model. Auto ARIMA recommended our manually-tuned model i.e. $ARIMA(0,1,1)(0,1,1)$. In addition, we use 2 other commonly used time series forecasting models, ETS and Multiplicative Holt-Winters, to analyse our time series. After comparing the SARIMA, ETS and Multiplicative Holt-Winters models (using MAE, MASE and RMSE as our accuracy metrics), we found out that the Multiplicative Holt-Winters model was the worst performing model (it could be overfitting on the training set) while SARIMA and ETS were the highest performing models. Overall, the SARIMA model fits the training set better than ETS but ETS forecasts more accurately on the test data than SARIMA. In the end, we selected the SARIMA model (**$ARIMA(0,1,1)(0,1,1)[12]$**) as the best model because residual analysis showed that it is better at capturing the patterns that exist in the time series data than ETS.

When we forecasted for the next 2 years (ie 2020 and 2021) using the best model, the model's forecast line was quite close to the "actual values" line and the forecast line was still able to show a seasonality pattern with peak temperatures occurring just before the middle of the year and troughs occurring in the cross-over period between one year to the next. Although we were not able to explore several kinds of time series models for this project due to a smaller team size, our project has allowed us to achieve the objectives that were stated in Chapter 2.

9. Reference (APA Referencing Style)

Adrian_PAdrian_P, KontorusKontorus, & bappers2bappers2. (1962, October 1). Why does slowly decaying ACF indicate that a time series is non-stationary?. Economics Stack Exchange. <https://economics.stackexchange.com/questions/12497/why-does-slowly-decaying-acf-indicate-that-a-time-series-is-non-stationary#:~:text=If%20the%20ACF%20is%20slowly,positive%20auto%2Dcorrelation%20here>

Analysis, S. D. (2023, July 27). What are the advantages and disadvantages of differencing for time series analysis?. Differencing for Time Series Analysis: Pros and Cons. <https://www.linkedin.com/advice/1/what-advantages-disadvantages-differencing>

Arima and seasonal Arima models: Non-seasonal Arima Models. Saylor Academy. (n.d.). <https://learn.saylor.org/mod/book/view.php?id=55330&chapterid=40908>

Autocorrelation function and Stationarity - SPUR ECONOMICS. (2023, January 25). <https://spureconomics.com/autocorrelation-function-and-stationarity/#:~:text=For%20a%20stationary%20time%20series>

Biradar, A. L. (2021). Time Series Analysis of Airline passengers. RPubs. https://rpubs.com/Anilbiradar/time_series_sarima

Brownlee, J. (2017, January 29). *How to Decompose Time Series Data into Trend and Seasonality*. Machine Learning Mastery. <https://machinelearningmastery.com/decompose-time-series-data-trend-seasonality/>

Chapter 6 Time series decomposition | Forecasting: Principles and Practice (2nd ed). (n.d.). In *otexts.com*. <https://otexts.com/fpp2/decomposition.html>

Chourasia, A. (2020, January 22). Decomposition in Time Series Data. Medium. <https://medium.com/analytics-vidhya/decomposition-in-time-series-data-b20764946d63>

Date, S. (2021, October 1). The Akaike Information Criterion. Time Series Analysis, Regression, and Forecasting. <https://timeseriesreasoning.com/contents/akaike-information-criterion/>

doriendorien, Richard HardyRichard Hardy , StatguyUserStatguyUser, & EstatisticsEstatistics . (1962, June 1). Interpreting accuracy results for an Arima model fit. Cross Validated.<https://stats.stackexchange.com/questions/194453/interpreting-accuracy-results-for-an-arima-model-fit#:~:text=ACF1%3A%20Autocorrelation%20of%20errors%20at,values%20in%20a%20time%20series>

ETS models. ETS models - statsmodels 0.15.0 (+59). (n.d.).
<https://www.statsmodels.org/dev/examples/notebooks/generated/ets.html>

Forecasting: Principles and practice (2nd ed). 8.9 Seasonal ARIMA models. (n.d.).
<https://otexts.com/fpp2/seasonal-arima.html>

Gabungan Darurat Iklim Malaysia & et al. (2023, May 12). Letter: Heatstroke-related deaths: Govt must act now. Malaysiakini. <https://www.malaysiakini.com/letters/664896>

Glen, S. (2018, September 7). *Ljung Box Test: Definition*. Statistics How To.
<https://www.statisticshowto.com/ljung-box-test/>

Glen, S. (2020, December 28). Absolute error & mean absolute error (MAE). Statistics How To.
<https://www.statisticshowto.com/absolute-error/>

Glen, S. (2020a, June 8). Mean absolute scaled error: Definition, example. Statistics How To.
<https://www.statisticshowto.com/mean-absolute-scaled-error/>

Glen, S. (2023, March 3). RMSE: Root mean square error. Statistics How To.
<https://www.statisticshowto.com/probability-and-statistics/regression-analysis/rmse-root-mean-square-error/>

Hyndman, R., Athanasopoulos, G., & Bergmeir, C. (n.d.). *Fit best arima model to Univariate time series - auto.arima*. robjhyndman.
<https://pkg.robjhyndman.com/forecast/reference/auto.arima.html>

Is it correct to impute missing values in a time series when the ... (n.d.).
https://www.researchgate.net/post/Is_it_correct_to_impute_missing_values_in_a_time_series_when_the_percentage_of_missing_observations_is_36_of_the_total_observations

Lagop. MATLAB & Simulink. (2023).

<https://www.mathworks.com/help/econ/nonseasonal-and-seasonal-differencing.html>

Lendave, V. (2021, October 29). A guide to different evaluation metrics for time series forecasting models. Analytics India Magazine.

<https://analyticsindiamag.com/a-guide-to-different-evaluation-metrics-for-time-series-forecasting-models/>

Liu, B. (2021, June 9). Seasonal data forecasting using R. Medium.

<https://bozliu.medium.com/seasonal-data-forecasting-using-r-b4cb791493c0>

MastersInDataScience. (2022, August 4). Arima modeling. CORP-MIDS1 (MDS).

<https://www.mastersindatascience.org/learning/statistics-data-science/what-is-arima-modeling/#:~:text=The%20ARIMA%20model%20predicts%20a,over%20a%20period%20of%20time>

Mean absolute error (mae) and root mean squared error (RMSE). (n.d.).

https://resources.eumetrain.org/data/4/451/english/msg/ver_cont_var/uos3/uos3_ko1.htm#:~:text=The%20MAE%20measures%20the%20average,measures%20accuracy%20for%20continuous%20variables.

Monigatti, L. (2022, August 2). *Interpreting ACF and PACF Plots for Time Series Forecasting*. Medium.

<https://towardsdatascience.com/interpreting-acf-and-pacf-plots-for-time-series-forecasting-af0d6db4061c#:~:text=in%20your%20browser.->

Moritz, S. (2022, September 9). Gallery: Times Series Missing Data Visualizations. Gallery: Times series missing data visualizations.

https://cran.r-project.org/web/packages/imputeTS/vignettes/gallery_visualizations.html

otexts.com. (n.d.). *Forecasting: Principles and practice (2nd ed)*. 2.4 Seasonal plots.

<https://otexts.com/fpp2/seasonal-plots.html>

otexts.com. (n.d.). *Forecasting: Principles and practice (2nd ed)*. 8.7 ARIMA modelling in R.

<https://otexts.com/fpp2/arima-r.html>

Pallante, M. (2020, February 20). Time Series Forecasting models. Medium.
<https://medium.com/analytics-vidhya/time-series-forecasting-models-726f7968a2c1>

PennState. (n.d.). 2.2 *partial autocorrelation function (PACF): Stat 510*. PennState: Statistics Online Courses. <https://online.stat.psu.edu/stat510/lesson/2/2.2>

PennState. (n.d.). 5.1 *decomposition models: Stat 510*. PennState: Statistics Online Courses. <https://online.stat.psu.edu/stat510/lesson/5/5.1#:~:text=For%20a%20multiplicative%20decomposition%2C%20this>

RDocumentation. (n.d.). *Coefest: Testing estimated coefficients*. RDocumentation. <https://www.rdocumentation.org/packages/lmtest/versions/0.9-30/topics/coefest>

Rogers, A. (2023, May 27). How climate change is affecting Malaysia. Borneo Post Online. <https://www.theborneopost.com/2023/05/28/how-climate-change-is-affecting-malaysia/>

royerroyer 62511 gold badge66 silver badges2020 bronze badges, & PaulPaul 8. (1967, February 1). Winter's method result accuracy in R. Stack Overflow. <https://stackoverflow.com/questions/64181044/winters-method-result-accuracy-in-r>

Sachin Date (2023, July 25). *Understanding partial auto-correlation and the PACF*. Time Series Analysis, Regression, and Forecasting. <https://timeseriesreasoning.com/contents/partial-auto-correlation/>

Seasonality Box Plot. 6.4.4.3. seasonality. (n.d.). <https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc443.htm#:~:text=The%20box%20plot%20shows%20the,the%20seasonal%20periods%20are%20known.>

Seasonality Sub Series. 6.4.4.3.1. seasonal subseries plot. (n.d.). <https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc4431.htm>

Snehal_bm. (2021, August 3). *Holt Winter's Method for Time Series Analysis*. Analytics Vidhya. <https://www.analyticsvidhya.com/blog/2021/08/holt-winters-method-for-time-series-analysis/#:~:text=The%20Holt%2DWinters%20method%20is>

Snehal_bm. (2023, April 26). Holt Winter's method for time series analysis. Analytics Vidhya.
<https://www.analyticsvidhya.com/blog/2021/08/holt-winters-method-for-time-series-analysis/#:~:text=The%20Holt%2DWinters%20algorithm%20is,make%20forecasts%20for%20future%20pe,riods.>

Tara, J. (2007). Forecast. Amazon.
<https://docs.aws.amazon.com/forecast/latest/dg/aws-forecast-recipe-ets.html>

Time Plot. Time Plots. (n.d.).
<https://www.ibm.com/docs/en/spss-modeler/18.3.0?topic=types-time-plots>

Why Time series decomposition is performed. (n.d.). Cross Validated. Retrieved September 30, 2023, from
<https://stats.stackexchange.com/questions/525092/why-time-series-decomposition-is-performed>

Wikiwand. (n.d.). *Akaike information criterion*. Wikiwand.
https://www.wikiwand.com/en/Akaike_information_criterion

World Health Organization. (2021). Climate change and health. World Health Organization.
<https://www.who.int/news-room/fact-sheets/detail/climate-change-and-health#:~:text=Key%20facts,malaria%2C%20diarrhoea%20and%20heat%20stress.>

Zach. (2020, October 15). Ljung-box test: Definition + example. Statology.
<https://www.statology.org/ljung-box-test/>

Zach. (2022, November 19). *How to use the coefstest() function in R*. Statology.
<https://www.statology.org/coefstest-r/>

7.3 Holt-Winters' seasonal method | Forecasting: Principles and Practice. (2016). Otexts.com.
<https://otexts.com/fpp2/holt-winters.html>

10. Appendix

Source Code:

```
library(lubridate)
library(ggplot2)
library(tidyverse)
library(dplyr)
library(reshape2)
library(forecast)
library(zoo)
library(ggfortify)
library(tseries)
library(TSstudio)
library(tsutils)
library(lmtest)
library("imputeTS")

# Read data from CSV file into data frame
tempdata <- read.csv("my_tmp_1901-2021.csv")

# Convert from Wide format to Long format. Month column labels in file were
renamed to
# "M01"... "M12". These will become values in the "month" column in the long
format.
# Renaming to "M01"... "M12" will allow sorting by chronological order.
tempdata_long <- melt(tempdata, id.vars=c("year"), measure.vars=c("M01", "M02",
"M03", "M04", "M05", "M06", "M07", "M08", "M09", "M10", "M11", "M12"),
variable.name="month", value.name="avgtemp")
# Sort by Year and Month
tempdata_long <- tempdata_long[ order(tempdata_long$year, tempdata_long$month), ]

# Create TS object from reformatted data
tempdata_ts <- ts(data = tempdata_long[,3], frequency = 12, start = c(1901,1), end
= c(2021, 12))

# Display TS data
tempdata_ts

# Subset the TS data to years 2012 - 2021
tempdata_ts <- window(tempdata_ts, start=c(2012,1))

ggplot_na_distribution(tempdata_ts)

# Plot TS for 2012 - 2021
autoplot(tempdata_ts) + ylab("Average Temperature") + xlab("Month-Year") +
scale_x_date(date_labels = '%Y', breaks = '1 year', minor_breaks = '2 month') +
theme_bw() + ggtitle("Malaysia Average Temperatures 2012 - 2021")

# Decompose TS into components
```

```

tscomponents <- stl(tempdata_ts, s.window = 'periodic')

# Display TS components
autoplot(tscomponents) + theme_bw() + scale_x_date(date_labels = '%b - %Y', breaks
= '1 year', minor_breaks = '2 month') + ggtitle("TS Components")

# Calculate Seasonal and Trend strengths
Tc <- trendcycle(tscomponents)
Sc <- seasonal(tscomponents)
Rc <- remainder(tscomponents)
TrSt <- round(max(0,1 - (var(Rc)/var(Tc + Rc))),1)
SnSt <- round(max(0,1 - (var(Rc)/var(Sc + Rc))),1)
data.frame('Trend Strength' = TrSt , 'Seasonal Strength' = SnSt)

# Display Seasonal Plot
seasonplot(tempdata_ts, year.labels = TRUE, col = 1:13, main = "Seasonal Plot",
ylab= "Average Temperature (celsius)")

#Seasonal Sub-Series Plot
seasplot(tempdata_ts, outplot = 3, trend = FALSE,
          main = "Seasonal Subseries Plot", ylab= "Average Temperature (celsius)")

#Seasonal Boxplot
seasplot(tempdata_ts, outplot = 2, trend = FALSE,
          main = "Seasonal Box Plot", ylab= "Average Temperature (celsius)")

# Split TS data into Training and Test sets (80%-20% split)
# Training set 2012 to 2019
tempdata_train <- window(tempdata_ts, start = c(2012,1), end = c(2019,12))

# Test set 2020 to 2021 (2 years)
tempdata_test <- window(tempdata_ts, start = c(2020,1), end = c(2021,12))

# Returns the number of Seasonal Differences required to make TS stationary
nsdiffs(tempdata_train)

# Plot ACF and PACF for training set (display 8 yrs)
par(mfrow=c(1,2))
acf(tempdata_train, lag.max = 96)
pacf(tempdata_train, lag.max = 96)

# Perform 1 Seasonal Difference (frequency of 12) on training set
tempdata_train_d12 <- diff(tempdata_train, 12)

# tempdata_train_d24 <- diff(tempdata_train_d12, 12)

# Display TS plot after taking 1 Seasonal Difference
autoplot(tempdata_train_d12) + ylab("Temperature") + xlab("Month-Year") +
  scale_x_date(date_labels = '%b-%Y', breaks = '1 year', minor_breaks = '2 month')
+ theme_bw() + ggtitle("Malaysia Average Temperatures 2012 to 2019 - After First
Seasonal Difference")

```

```

par(mfrow=c(1,2))
acf(tempdata_train_d12, lag.max = 96)
pacf(tempdata_train_d12, lag.max = 96)

# Perform 1 Non-Seasonal Difference (1 lag) on TS to stabilize variance
tempdata_train_d12_1 <- diff(tempdata_train_d12, 1)

# Display TS plot after taking 1 Seasonal Difference and 1 Non-Seasonal Difference
autoplot(tempdata_train_d12_1) + ylab("Temperature") + xlab("Month-Year") +
  scale_x_date(date_labels = '%b-%Y', breaks = '1 year', minor_breaks = '2 month')
+ theme_bw() + ggtitle("Malaysia Average Temperatures 2012 to 2019 - After First
Seasonal Difference And 1 Non-Seasonal Difference")

# Plot ACF and PACF to see the difference
par(mfrow=c(1,2))
acf(tempdata_train_d12_1, lag.max = 96)
pacf(tempdata_train_d12_1, lag.max = 96)

##### 1. SARIMA MODEL #####

# Set Arima (p,d,q) and (P,D,Q) parameters. Based on analysis of ACF and PACF, a
suitable
# model may be (0,1,1) (0,1,1). Construct additional Arima models with parameters
with
# small differences in parameter values. Note that D and d should be 1 since we
performed
# differencing for both seasonal and non-seasonal.
fit1 <- arima(tempdata_train, order = c(0,1,1), seasonal = c(0,1,1))
fit2 <- arima(tempdata_train, order = c(1,1,1), seasonal = c(0,1,1))
fit3 <- arima(tempdata_train, order = c(0,1,1), seasonal = c(1,1,1))
fit4 <- arima(tempdata_train, order = c(1,1,1), seasonal = c(1,1,1))

# The model which returns the lowest AIC score is (0,1,1) (0,1,1)
# Run this best model again on training set
best_md <- arima(tempdata_train, order = c(0,1,1), seasonal = list(order =
c(0,1,1)))

# Use auto arima function to find a good performing model
auto_md <- auto.arima(tempdata_train, ic="aic", max.order=6, D=1, d=1,
stepwise=FALSE, approximation=FALSE)
auto_md

# Check Residuals to see if it contains White Noise
#checkresiduals(selected_md)
checkresiduals(best_md)
accuracy(best_md)
#accuracy(selected_md)

```

```

# Test the significance of the coefficients
# coeftest(selected_md)

# Use best model to forecast Test set for 24 periods (2 yrs)
test_fc <- forecast(best_md, h = 24)

# Display forecasted values
#test_fc

# Display Fitted, Actual and Forecasted values on same plot
test_forecast(actual = tempdata_ts, forecast.obj = test_fc, test = tempdata_test)

# Display metrics values (RMSE, MAE)
accuracy(test_fc, tempdata_test)

##### END OF SARIMA MODEL
#####

#####2. ETS MODEL
#####
#stands for error, trend, seasonality
#step 4
etsmodel <- ets(tempdata_train)
#plot graph for ets after training
autoplot(etsmodel)

#step 6
#test for white noise in the ets model
checkresiduals(etsmodel)

#step 7
#testing the accuracy
accuracy(etsmodel)
#giving out the summary of the data for ets model (same result)
summary(etsmodel)

test_fc_ets <- forecast(etsmodel, h = 24)
test_forecast(actual = tempdata_ts, forecast.obj = test_fc_ets, test =
tempdata_test)
accuracy(test_fc_ets, tempdata_test)

##### END OF ETS MODEL
#####

#####3. HOLT WINTERS - MULTIPLICATIVE MODEL

```

```
#####

#fitting the Holt Winter Model
hw_model <- HoltWinters(tempdata_train, seasonal = "multi")

print(hw_model)

checkresiduals(hw_model)

# Compute in-sample accuracy by comparing fitted values with the training data
#in_sample_accuracy <- accuracy(fitted(hw_model), tempdata_train)
hw_modelf <- forecast(hw_model)
in_sample_accuracy <- accuracy(hw_modelf)

# Print the in-sample accuracy measures
print(in_sample_accuracy)

test_fc_hw_m <- forecast(hw_modelf, h = 24)
test_forecast(actual = tempdata_ts, forecast.obj = test_fc_hw_m, test =
tempdata_test)
accuracy(test_fc_hw_m, tempdata_test)

##### END OF HOLT WINTERS - MULTIPLICATIVE MODEL
#####

##### Coefficients of the Best Model
#####
sel_best_md <- arima(tempdata_ts, order = c(0,1,1), seasonal = list(order =
c(0,1,1), period=12))

coeftest(sel_best_md)

sel_best_test_fc <- forecast(best_md, h = 24)

# Display Fitted, Actual and Forecasted values on same plot
test_forecast(actual = tempdata_ts, forecast.obj = sel_best_test_fc, test =
tempdata_test)

plot(forecast(best_md,h=24))
```