Assignment - 2

MATH1318 Time Series Analysis

Apr 23, 2023 • RMIT University

Tran Ngoc Anh Thu (s3879312)

Table of Contents

1. Executive Summary

This report analyses the yearly Arctic Sea ice minimum extent data from 1979 to 2022. The goal was to identify a suitable set of ARIMA(p, d, q) models for the data. First, we use R to perform a descriptive analysis to understand the data's underlying characteristics and identify potential trends and patterns. We proposed a set of possible ARIMA(p, d, q) models using the model specification tools introduced in the course modules. We demonstrated using ACF-PACF, EACF, and BIC table methods to identify the appropriate (p, d, q) orders for our models. Next, we fitted the possible ARIMA(p, d, q) models and analysed the parameter estimates. Finally, we perform several diagnostic checks on the chosen model. In the process, we eliminated the trend by first differencing the time series. Furthermore, we considered the normality assumption throughout the analysis and selected the CSS-ML method for fitting our ARIMA models. This method was chosen because it does not rely on any probability density function and offers a better balance between the ML and CSS methods. The report concludes that the ARIMA(1,1,2) model is the smallest model has the best performance across all metrics. The result of this report is used to make forecasts of future Arctic Sea ice minimum extent values in 10 years and to inform policymakers about the potential consequences of climate change.

2. Introduction

The Arctic Sea ice minimum extent is an important indicator of climate change, and its variations have significant implications for global climate patterns. In recent decades, there has been a growing concern about the rapid decline of Arctic Sea ice minimum extent due to climate change. Therefore, studying the behaviour of Arctic Sea ice minimum extent over time is essential to understand climate change dynamics better. The data from NASA consists of the Arctic Sea ice minimum extent (million square km) observed in September of each year. The report proposes various model specification tools to identify the best fit ARIMA(p, d, q) model for the Arctic Sea ice minimum extent time series and to analyse the potential impact of climate change on Arctic Sea ice.

3. Data Exploration and Visualization

We loaded the dataset into RStudio as a data frame object and conducted a descriptive analysis of the Arctic Sea ice minimum extent. Then we convert it to a time series object. The data consists of 44 observations of the Arctic Sea ice minimum extent (million square km) from 1979 to 2022. We provide a numerical summary of the data's main features and help inform the modelling process. We will also create data visualizations to understand its patterns and trends better.

3.1. Descriptive Analysis

We begin by loading the data and conducting a descriptive analysis, including measures of central tendency

Table 1. Summary Statistics of the Arctic Sea ice minimum extent over years (1979-2022)

Statistic value	Year	Arctic.Sea.Ice.Extent..million.square.km.
Min.	1979	3.390
1st Qu.	1990	4.707
Median	2000	5.995
Mean	2000	5.800

3rd Qu.	2011	6.900
Max.	2022	7.540

The summary statistics indicate that the minimum Arctic Sea ice minimum extent observed was 3.39 million square km in 2012, while the maximum was 7.54 million square km in 1980. The mean Arctic Sea ice minimum extent over the entire period is 5.8 million square km, with a median of 5.99 million square km. The interquartile range (IQR) is 2.19 million square km, with the first quartile (Q1) at 4.71 million square km and the third quartile (Q3) at 6.90 million square km.

3.2. Find the Frequency and Convert data to Time Series Object

The frequency of a time series refers to the number of observations in one complete cycle. In this case, since the observations were recorded once per year, the frequency of the arctic_sea_ice time series is 1.

3.3. Scatter plot

The scatter plot demonstrates the association between the current year's Arctic Sea Ice extent and the previous year's extent. The x-axis displays the extent from the previous year, and the y-axis shows the extent for the current year. The connection between these two variables can be ascertained by computing the Pearson correlation coefficient, symbolized by 'r.' This coefficient quantifies the two variables' intensity and direction of the linear link.

r = cov(X,Y) / (sd(X) * sd(Y)) = 0.8014121

Where cov(X,Y) is the covariance between X and Y, sd(X) is the standard deviation of X, and sd(Y) is the standard deviation of Y.

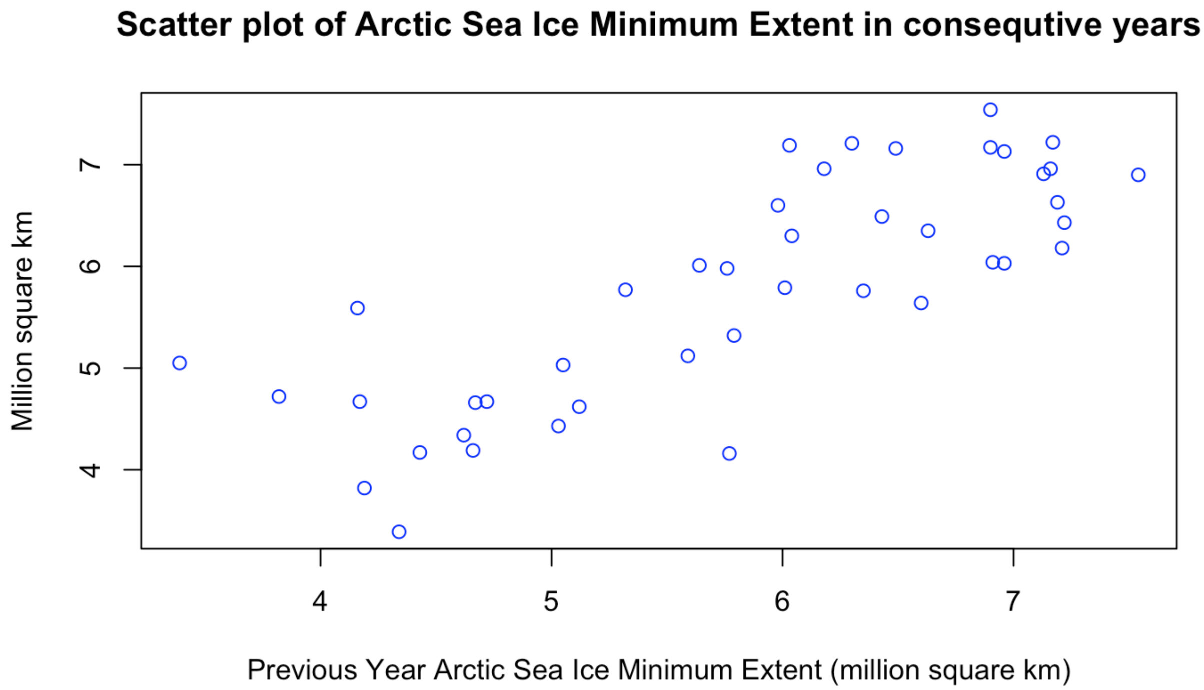

Figure 1. Scatter plot of yearly Artic Sea Ice Minimum Extent in Consecutive Years

We calculate the correlation between the Arctic Sea Ice Minimum Extent and its first lag. The correlation value obtained is 0.8014121. The interpretation of this correlation value is that there is a strong positive linear relationship between pairs of consecutive Arctic Sea Ice Minimum Extent values. In other words, if the extent in a particular year is elevated, the extent in the following year will likely be high, and vice versa. This strong correlation suggests that the time series has some persistence, and previous values can be useful in predicting future values.

3.4. Time series plot

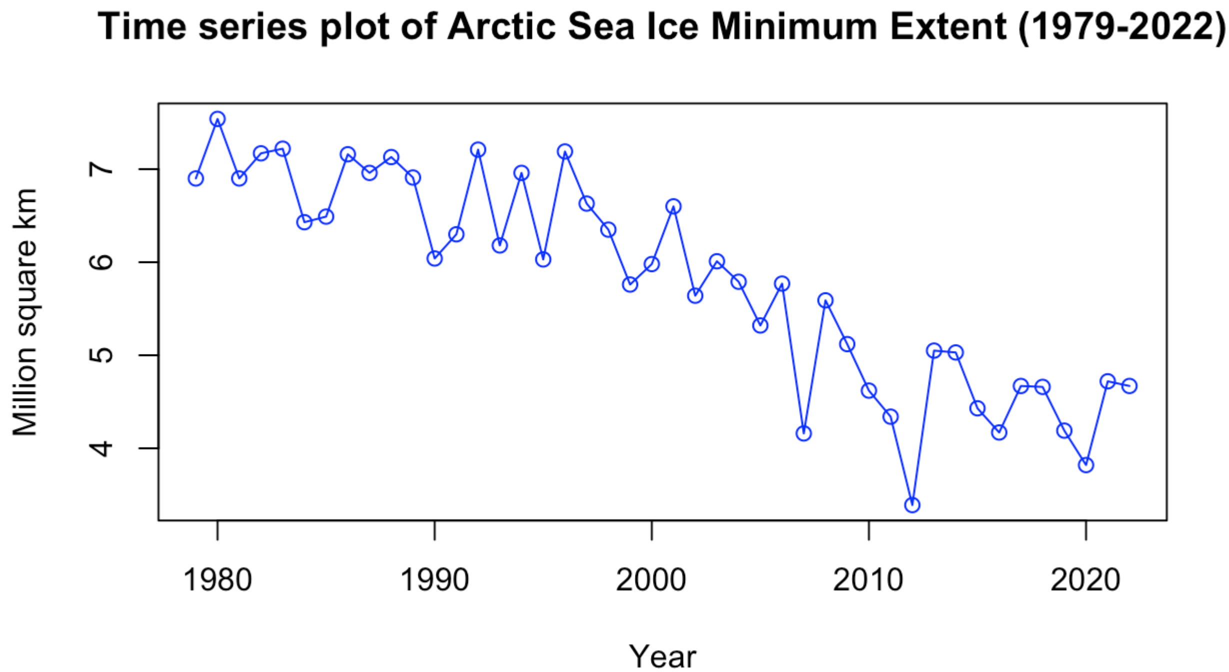

Figure 2. Time series plot of Sea Ice Minimum Extent (1979-2022)

+ **Trend** – The time series plot suggests a general downward trend over the years, indicating a decrease in the ice extent from 1979 to 2022.
+ **Seasonality** - Since the Arctic Sea ice minimum extent data is measured annually, it exhibits no clear seasonal patterns. However, if the data were recorded at a monthly or daily frequency, we might observe seasonality due to the annual cycle of ice formation and melting.
+ **Changing variance** - In the Arctic Sea ice minimum extent data, the variance seems relatively constant until the mid-2000s, after which the fluctuations appear to become more significant, indicating a possible increase in the variability of the ice extent.
+ **Behaviour** - A moving average of the Arctic Sea ice minimum extent data would show a downward trend over the years, indicating a decrease in ice extent. There might be some autoregressive behaviour in the Arctic Sea ice minimum extent data, as the ice extent each year is likely to be influenced by the ice extent in previous years. Analysing the ACF and PACF plots can help determine the degree of autoregressive behaviour in the data.
+ **Changepoint** - The Arctic Sea ice minimum extent data shows no distinct change points. However, the increased variability in the data since the mid-2000s may indicate a potential shift in the underlying processes driving the ice extent, possibly due to the acceleration of climate change.

The fluctuation showing MA behaviour and the succeeding time point showing AR behaviour so we would have positive p and q values in ARIMA or ARMA models for this data.

3.5. Normality Check

We employ both numerical and visual methods to interpret the normality of the original time series data.

3.5.1 Q-Q Plot

A Q-Q plot (quantile-quantile plot) is a graphical method used to assess the similarity between the distribution of a given dataset and a reference probability distribution, typically the normal distribution. The plot is created by comparing the dataset's quantiles against the reference distribution's quantiles. If the data points in the Q-Q plot closely follow a straight line, it suggests that the dataset's distribution is like the reference distribution.

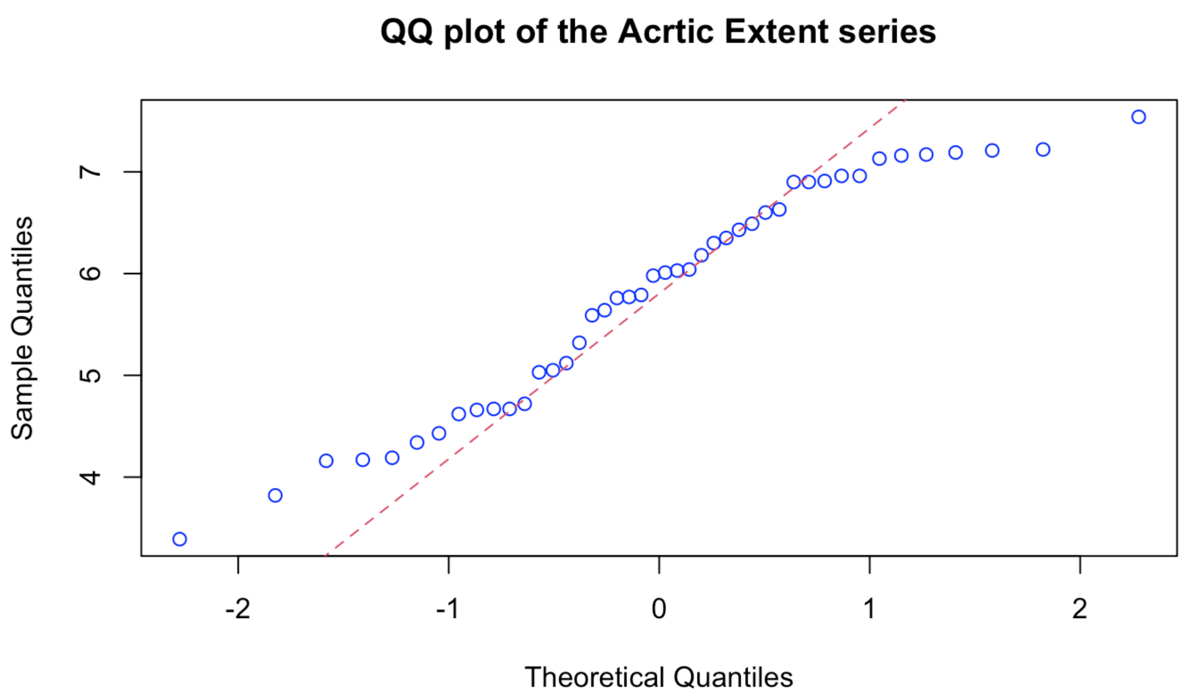

Figure 3. Q-Q plot shows the normality of the data.

The Q-Q plot for the Arctic Sea time series data (blue dots) shows a departure from the straight line (red straight line) at both tails. This observation indicates that our dataset may not conform to the normal distribution in the extreme values (the data points in the distribution's tails may exhibit heavier or lighter tails than a normal distribution). This finding is essential as it helps us understand the distributional characteristics of our dataset, which in turn informs our model selection and fitting process.

3.5.2 Shapiro-Wilk test

The Shapiro-Wilk test is a statistical test used to check the normality of a given sample. In this case, we have applied the test to the original time series data of arctic sea ice with a significance level $\alpha = 0.05$.

- The null hypothesis (Ho): data is normal distributed.
- The alternative hypothesis (Ha): data is not normal distributed.

The test provides two results: the W statistic and the p-value. In our case, the W statistic is 0.94543, and the p-value is 0.0373. The p-value of 0.0373 is less than 0.05, so we would reject the null hypothesis. In addition to the Q-Q plot, we can conclude and confirm that the Arctic Sea Ice time series data does not come from a normal distribution.

3.6. Stationary check

3.6.1 ACF

ACF(k) = Corr(Yt,Yt+k)

The autocorrelation function (ACF) quantifies the relationship between occurrences of the same time series {Yt, t=1,2,...,n}. Specifically, ACF(k) represents the correlation between data points separated by k time intervals.

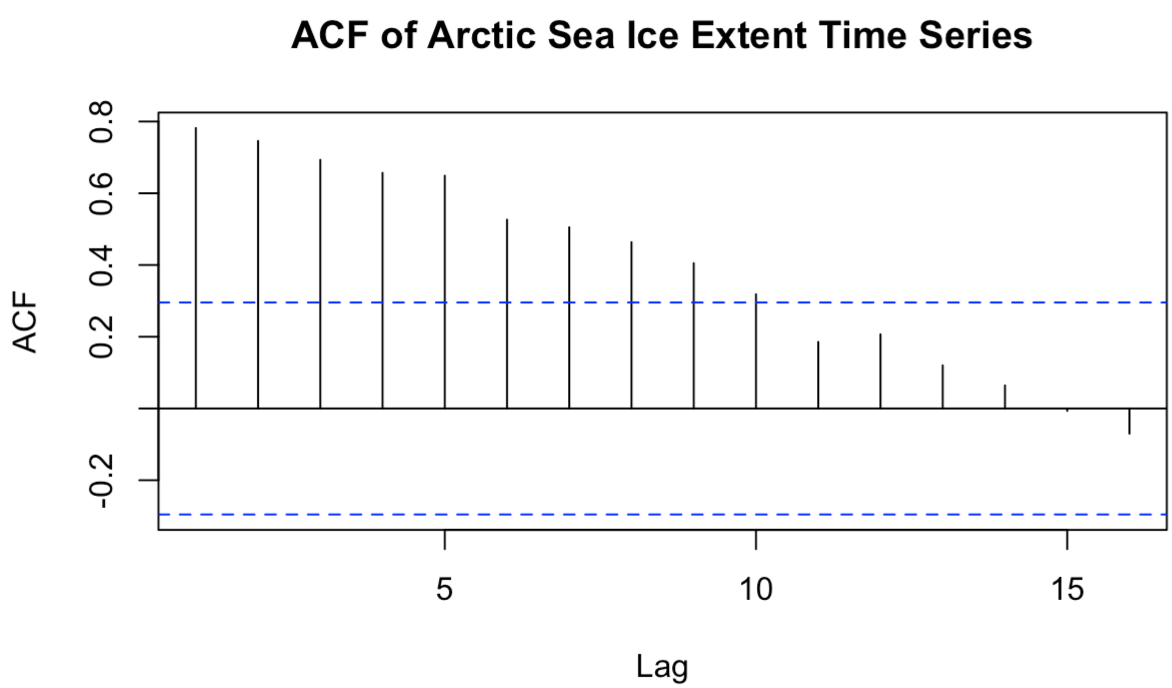

Figure 4. The autocorrelation function plot of the Arctic Sea time series.

The ACF plot indicates that the Arctic Sea Ice Extent Time Series exhibits strong positive autocorrelation at lower lags, suggesting that the extent values are persistent and that past values can be useful in predicting future values. The autocorrelation gradually tails off (decreases) as the lag increases, with a weak negative correlation observed at lag 15. A slowly decaying ACF pattern indicates that the time series is likely non-stationary and exhibits a trend. In such cases, we consider differencing the time series to remove the trend and make it stationary. Once the time series is stationary, we can proceed with time series modelling, such as ARIMA, to make forecasts.

3.6.2 PACF

The partial autocorrelation function (PACF) is derived from the autocorrelation function by eliminating the indirect effects of intermediate data points.

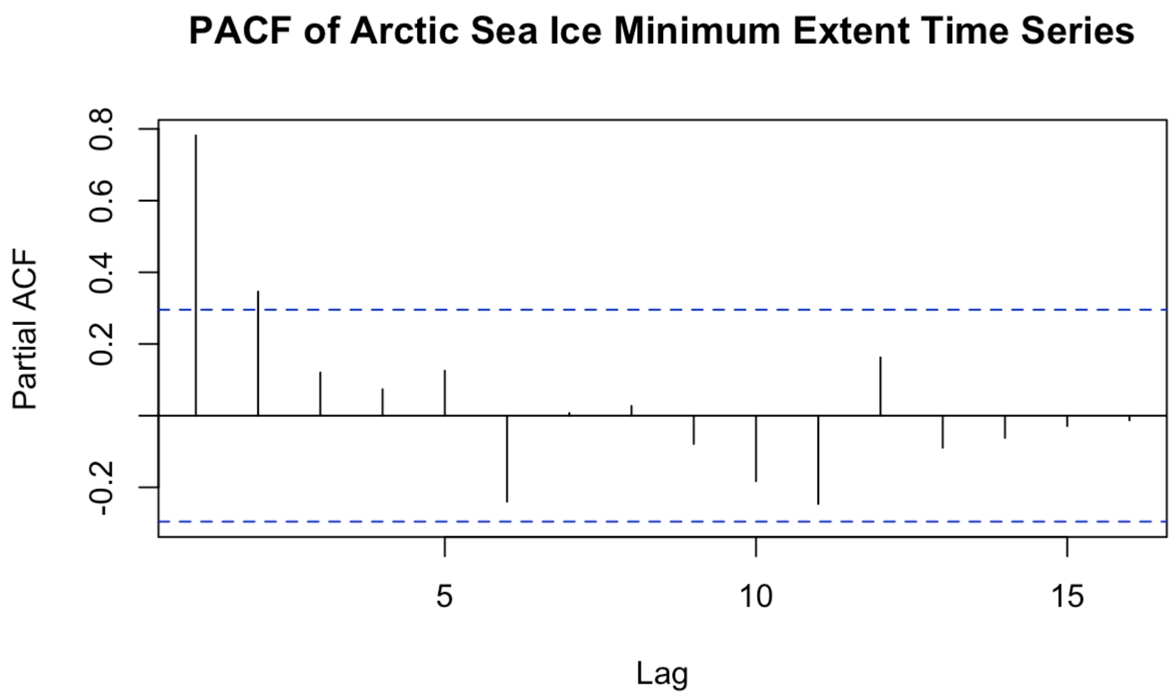

Figure 5. The partial autocorrelation function plot of the Artic Sea time series

The PACF plot shows an exceptionally large partial autocorrelation at lag 1, followed by weaker partial autocorrelations at higher lags. This information can be useful when selecting the order of an AR (autoregressive) component in time series models, such as ARIMA.

3.6.3 The Dickey-Fuller Unit-Root Test (ADF test)

The Augmented Dickey-Fuller (ADF) test is a statistical test used to determine the presence of unit roots in a time series, helping us assess whether the data is stationary. A stationary time series has a constant mean, constant variance, and no time-dependent structure. If the test fails to reject the null hypothesis, a unit root is present, and the time series is non-stationary.

- The null hypothesis (Ho): data is stationary.
- The alternative hypothesis (Ha): data is non-stationary.

The ADF test result for the original Arctic Sea ice time series data shows a Dickey-Fuller value of -2.3296 with a lag order of 3. The p-value for the test is 0.4433.
Since the p-value is significantly greater than the typical significance level α of 0.05, we fail to reject the null hypothesis, suggesting that the time series data is non-stationary. The original Arctic Sea ice time series data exhibits a time-dependent structure. Further pre-processing, such as differencing, may be required to achieve stationarity before fitting ARIMA or other time series models.

4. Methodology

4.1. Deterministic Versus Stochastic Trends

We will explore models that account for stochastic trends, as the time series under investigation exhibits a highly variable trend. Employing a deterministic stationary model is not suitable for this dataset. Instead, we will focus on non-stationary models incorporating stochastic trends, providing a more reasonable approach to understanding and analysing the data. Our initial focus will be addressing and eliminating the trends within the dataset. Once this has been accomplished, we will model the autocorrelation structure inherent in the time series, ensuring a comprehensive understanding of the underlying patterns and relationships.

4.2. Variance Stabilising Transformation (Box-Cox Transformation)

In our analysis, we utilize the R programming language to explore a variety of lambda values, computing the log-likelihood value for each lambda using a normal likelihood function.

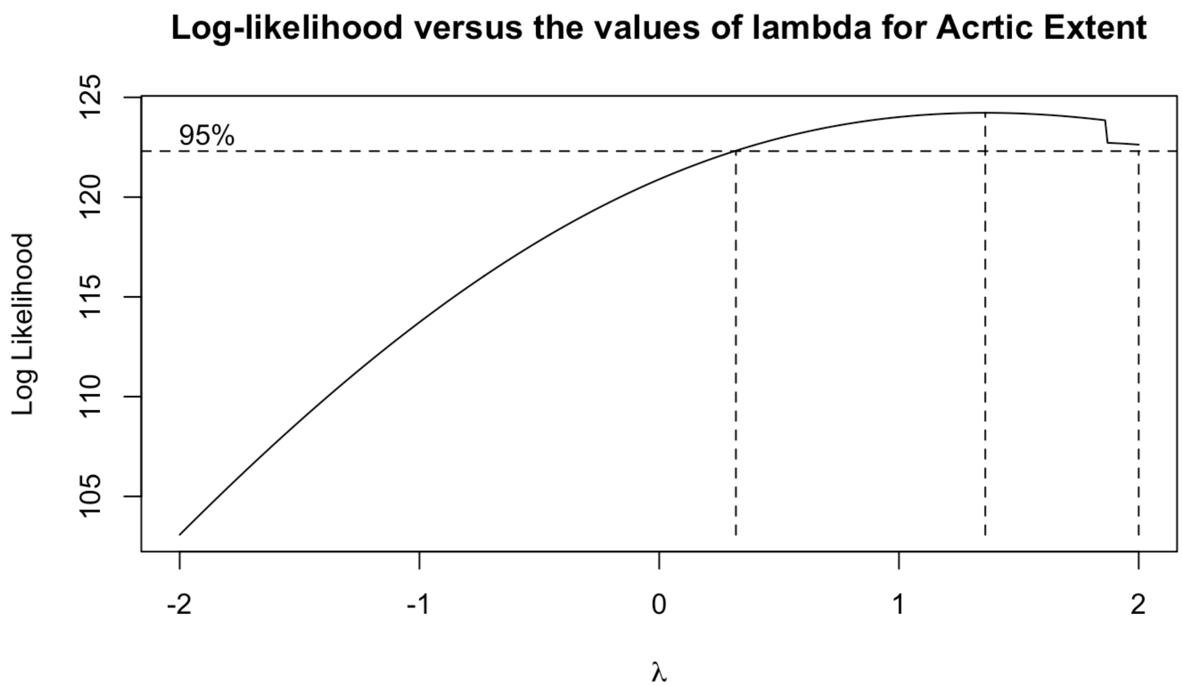

Figure 6. Log-likelihood versus the values of 1.36 lambda. The first and third vertical line are 0.32 and 2 respectively.

It is generally recommended to apply transformations before differencing, as this can help reveal the underlying structure of the data, making it more accessible and easier to model. In our dataset, we observed fluctuations in variance that could potentially impact the performance of our time series model. To address this issue, we can apply a transformation technique, such as the logarithmic transformation, which helps stabilize the variance and make the time series more suitable for modelling. To determine the optimal transformation for our dataset, we will employ the Box-Cox transformation instead of natural logarithm transformation (assumes lambda $\lambda = 0$), which provides an optimal lambda λ value as the best-suited transformation.
The Box-Cox transformation was applied to the data to improve its normality. While the transformation resulted in a slight improvement, the difference in the Shapiro-Wilk normality test p-value was not substantial. After applying the Box-Cox transformation, the p-value increased from 0.0373 to 0.04554 (borderline p-value). Despite the improvement, the p-value remains below the significance level ($\alpha = 0.05$), indicating that the null hypothesis of normality cannot be accepted. Consequently, we have decided not to use the Box-Cox transformation on our data as it does not significantly improve normality.

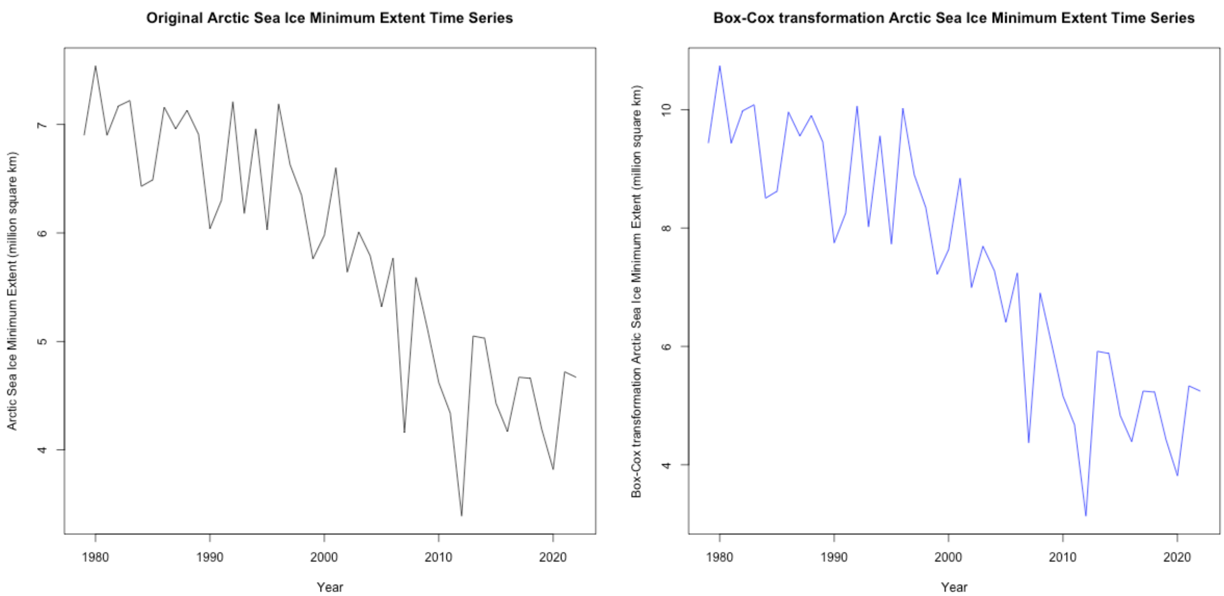

Figure 7. Comparison of Original and Box-Cox Transformed Arctic Sea Ice Minimum Extent Time Series

We see little to no difference between the two plots in the "Comparison of Original and Box-Cox Transformed Arctic Sea Ice Minimum Extent Time Series" plot. It could be because the optimal lambda value chosen by the Box-Cox transformation is close to 1. When the lambda value is 1, the Box-Cox transformation does not change the data, as the transformation formula becomes $(y^1 - 1) / 1$, equal to $y - 1$. In this case, the transformed time series would be like the original one, with only a constant difference of 1.

$$(y^1 - 1) / 1 = y - 1$$

Our lambda value is 1.36, close to 1, which would explain the similarity between the two plots. In this case, the Box-Cox transformation might not help stabilize the time series data variance; we could consider other transformations or stick to the original data. We will still transform the data here to increase the possibility to improve the normality.

4.3. Addressing Non-Stationarity in Time Series Data

4.3.1 Detrend the data through Differencing

We use differencing to transform a non-stationary time series into a stationary one by computing the differences between consecutive observations. The purpose of differencing is to remove trends from the data, common causes of non-stationarity. Detrending the data is important because many time series model, such as ARIMA, require stationary input data for accurate modelling and forecasting.

It is crucial to avoid over-difference in the data, as excessive differencing can obscure the underlying autocorrelation structure, making it challenging to identify suitable models. We typically start with first-order differencing to achieve the right balance, which calculates the differences between consecutive observations ($Y_t - Y_{t-1}$). Further differencing may be necessary if the first-order differencing does not result in a stationary time series. However, it is essential to assess the stationarity after each differencing step to avoid over-differencing and preserve the autocorrelation structure of the data.

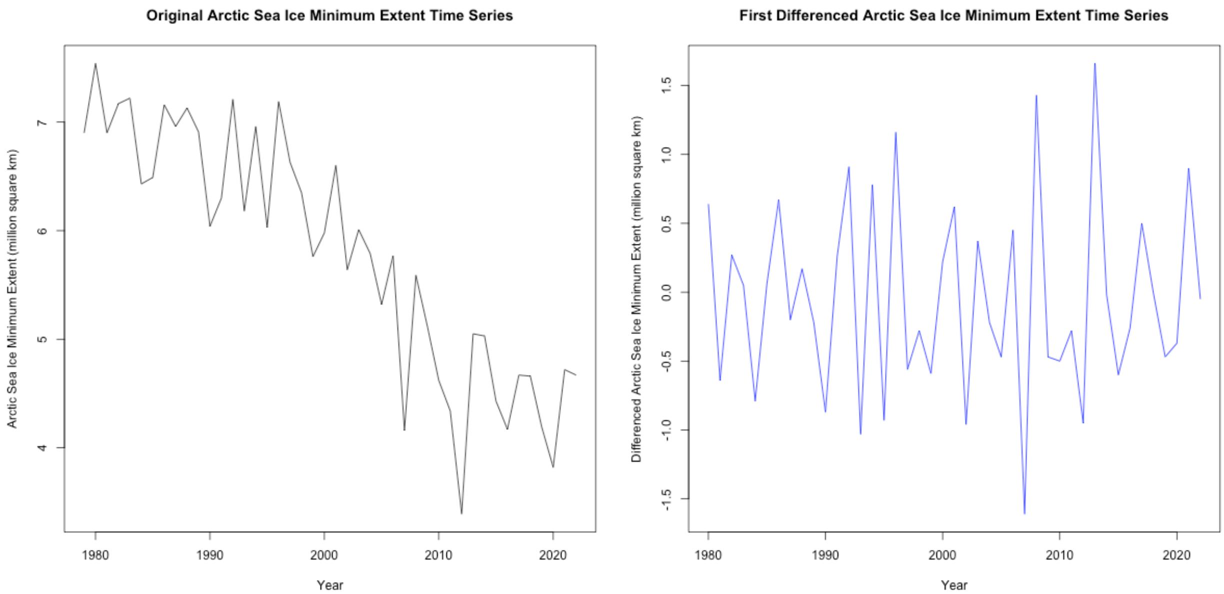

Figure 8. Comparison of Original and First Differenced Arctic Sea Ice Minimum Extent Time Series

After applying the first differencing to the time series data, the mean level appears relatively flat, suggesting that the data may be stationary. However, to confirm the stationarity of the transformed data, we will perform a series of stationarity tests for Level Stationarity. These tests will provide a more rigorous assessment of the stationary properties of the time series data and help ensure the validity of our subsequent analyses.

4.3.2 Stationarity tests

All three tests, including Augmented Dickey-Fuller, Phillips-Perron Unit Root, and KPSS, were performed on the first-differenced Arctic Sea ice time series to test for stationarity with $\alpha = 0.05$.

Table 2. Comparison of different stationarity tests after the First Differenced time series data

Test	Data	Test Statistic	Lag Parameter	p-value	Alternative Hypothesis
Augmented Dickey-Fuller	first_diff_arctic_sea_ice	Dickey-Fuller = -6.659	3	0.01	Stationary
Phillips-Perron Unit Root	first_diff_arctic_sea_ice	Z(alpha) = -53.498	3	0.01	Stationary
KPSS Test for Level Stationarity	first_diff_arctic_sea_ice	KPSS Level = 0.07461	3	0.1	Stationary (null hypothesis)

The Augmented Dickey-Fuller and Phillips-Perron tests have p-values of 0.01, smaller than the commonly used significance level of 0.05. According to these tests, this indicates that the first-differenced Arctic Sea ice time series is stationary. On the other hand, the KPSS test has a p-value of 0.1, which is greater than 0.05. Since the null hypothesis of the KPSS test is that the time series is stationary, we fail to reject the null hypothesis, suggesting that the time series is also stationary. All three tests conclude that the first-differenced Arctic Sea ice time series is stationary after the first differencing at a 5% level of significance. Therefore, we decide to choose the order of difference for our ARIMA model is d = 1.

4.4. Model selection

ARMA(p,q) and ARIMA(p,d,q) models are used for time series analysis but differ in how they handle the data.

- **ARMA (Autoregressive Moving Average)** model combines two components: autoregressive (AR) and moving average (MA). The AR component represents the relationship between the current observation and its previous observations, while the MA component captures the relationship between the current observation and the past errors. An ARMA model is suitable for stationary time series data, meaning it has constant mean and variance over time.
- **ARIMA (Autoregressive Integrated Moving Average)** model extends the ARMA model by adding an "integrated" (I) component. The integrated component represents the differencing applied to the data to make it stationary. In other words, an ARIMA model first differentiates the data to remove any trends or seasonality, then fits an ARMA model to the transformed data. The ARIMA model is suitable for time series data that is non-stationary, as it can handle trends and seasonal patterns.

Since our data is non-stationary, we should consider using an ARIMA model of order p, q, and d time(s) of differencing.

4.5. Model Specification

The ARIMA model consists of two main components: Autoregressive (AR) and Moving Average (MA). The order of the AR component, denoted as 'p', represents the number of previous time points that influence the current value in the time series. In an AR(p) model, 'p' determines the extent of the autoregressive behaviour. On the other hand, the MA component, denoted as 'q', represents the number of previous error terms that influence the current value. In an MA(q) model, 'q' determines the extent of the moving average behaviour.
To select the most appropriate values for 'p' and 'q' in our ARIMA model, we will utilize various diagnostic tools, including the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots, the Extended Autocorrelation Function (EACF) table, and the Bayesian Information Criterion (BIC) table. By evaluating these tools, we can identify the optimal combination of 'p' and 'q' that best captures the underlying patterns in our time series data.

4.5.1 ACF Plot

Upon eliminating the trend through the first differencing, the ACF of the differenced series exhibits a smooth decay pattern, which is indicative of an MA(q) characteristic. We will closely examine the ACF plot to determine the possible q values for our ARIMA model.

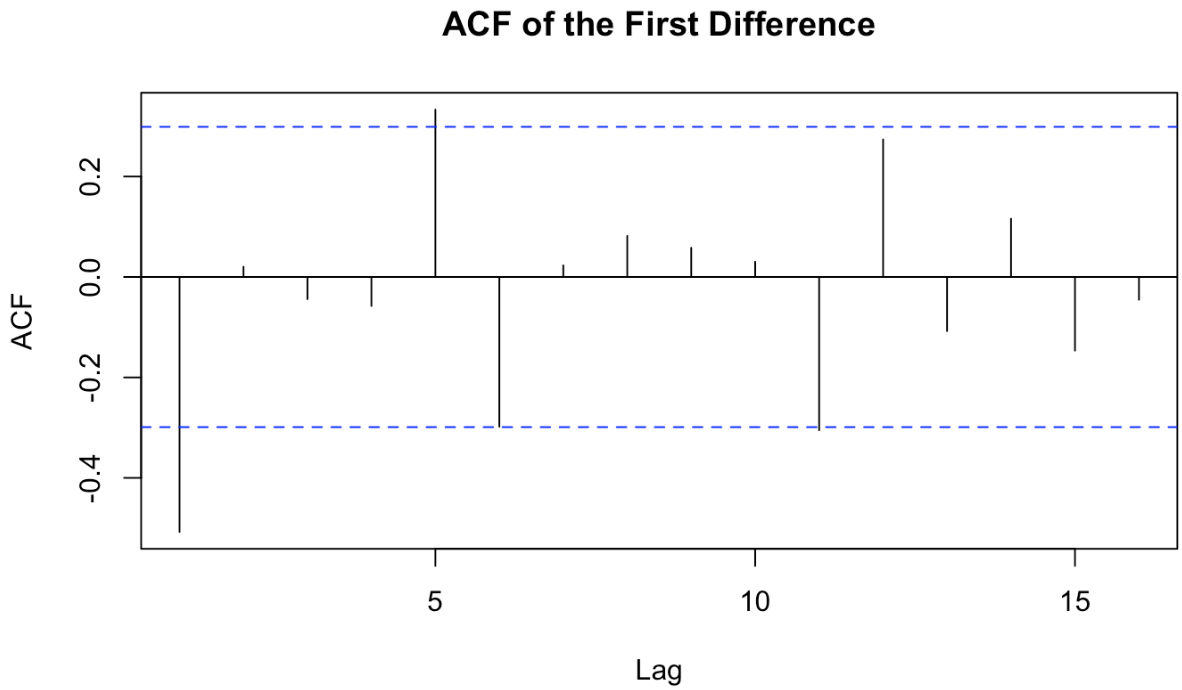

Figure 9. ACF plot to determine the q value for the ARIMA model.

4.5.2 PACF

Upon eliminating the trend through the first differencing, the PACF of the differenced series exhibits a large first lag, which is indicative of an AR(p) characteristic. We will closely examine the PACF plot to determine the possible p values for our ARIMA model.

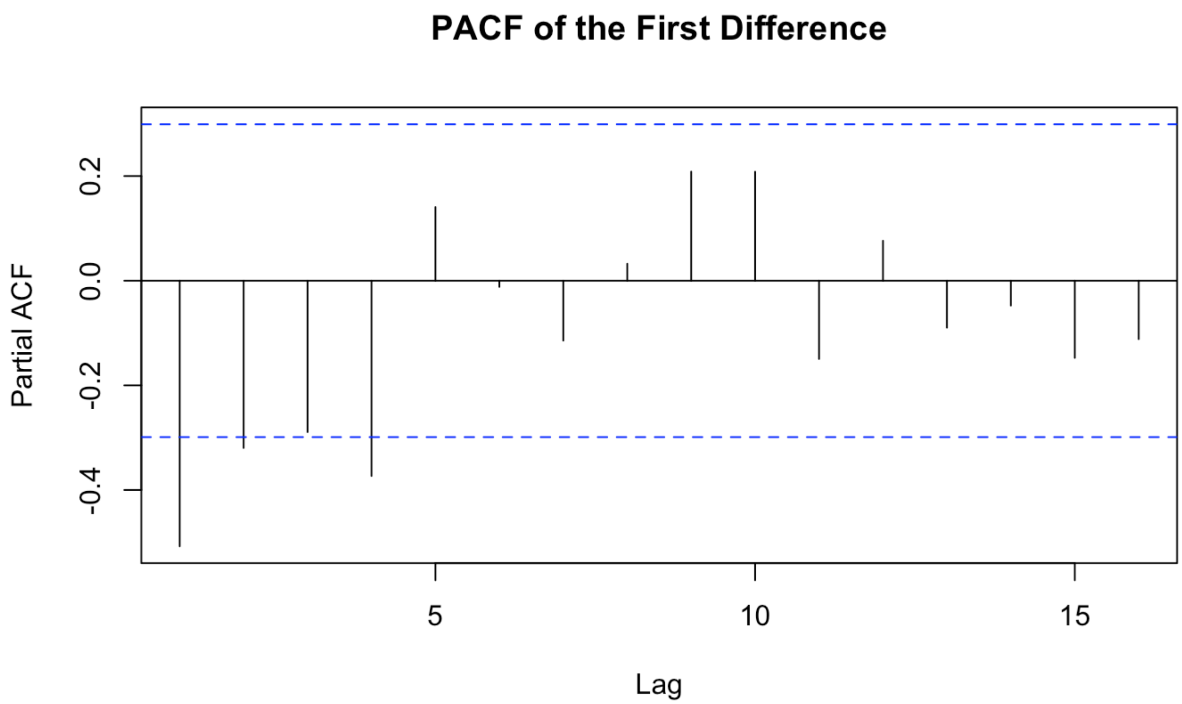

Figure 10. PACF plot to determine the p value for the ARIMA model.

After examining the ACF and PACF plots of the differenced time series, we observed at least two significant autocorrelations in the ACF plot and three in the PACF plot. This pattern of significant lags in the ACF and PACF plots provides useful information to determine the potential ARIMA models for our data.

Considering the observed significant lags in the ACF and PACF plots, we have chosen the following candidate models for further evaluation: ARIMA(1,1,1), ARIMA(2,1,1), ARIMA(2,1,2), ARIMA(3,1,1), and ARIMA(3,1,2). These models represent different combinations of autoregressive (AR) and moving average (MA) components, capturing the potential structure in our time series data. We will now fit these candidate models with more model selection tools, such as the EACF and BIC tables, to determine our dataset's most suitable ARIMA model.

4.5.3 EACF Table

The Extended Autocorrelation Function (EACF) table is a useful diagnostic tool for identifying the orders of AR and MA components (p and q) in an ARIMA model. The EACF table presents the partial autocorrelations for various combinations of 'p' and 'q', allowing us to identify patterns that suggest the optimal model order. Typically, we look for a combination of 'p' and 'q' that results in a "sequence" of non-significant partial autocorrelations, indicating a suitable ARIMA(p, d, q) model.

AR/MA	0	1	2	3	4	5	6	7	8	9	10
0	x	o	o	o	x	o	o	o	o	o	o
1	x	o	o	o	x	o	o	o	o	o	o
2	x	o	o	o	o	o	o	o	o	o	o
3	x	o	o	o	o	o	o	o	o	o	o
4	o	o	o	o	o	o	o	o	o	o	o
5	o	o	o	o	o	o	o	o	o	o	o
6	o	x	o	o	o	o	o	o	o	o	o
7	o	x	o	o	o	o	o	o	o	o	o
8	o	x	o	o	o	o	o	o	o	o	o
9	x	o	o	o	o	o	o	o	o	o	o
10	x	x	o	o	o	o	o	o	o	o	o

Figure 11. EACF Table to determine set values of p and q for Arctic Sea series

To select an appropriate model from the EACF table, we first identify the vertex located at the top left corner of the table to choose the smallest possible model. The vertex model in the EACF table is characterized by the "o" symbol, which occurs at the intersection of AR=0 and MA=1. The row in which the "o" symbol is located represents the autoregressive (AR) order 'p', while the column it is situated in corresponds to the moving average (MA) order 'q'. We then examine the neighbouring models surrounding the vertex. Considering its adjacent models, the subsequent downward vertex such as AR=1 and MA=1.

Taking into account the results from the EACF table, we have selected the following models for further consideration: ARIMA(0,1,1), ARIMA(0,1,2), ARIMA(1,1,1), and ARIMA(1,1,2). These models represent different combinations of AR and MA components, capturing the potential structure in our time series data. We will now fit these candidate models and evaluate their performance using other model selection tools, such as the BIC table, to determine the most suitable ARIMA model for our dataset.

4.5.4 BIC Table

The Bayesian Information Criterion (BIC) table is another valuable tool for model selection. It measures the model's goodness of fit while penalizing models with more parameters to avoid overfitting. The BIC table helps us balance the trade-off between model complexity and model performance. A lower BIC value indicates a better model fit when comparing multiple models, considering the model's complexity and ability to explain the data. By evaluating the BIC values for different combinations of 'p' and 'q', we can select the most appropriate ARIMA model for our time series data.

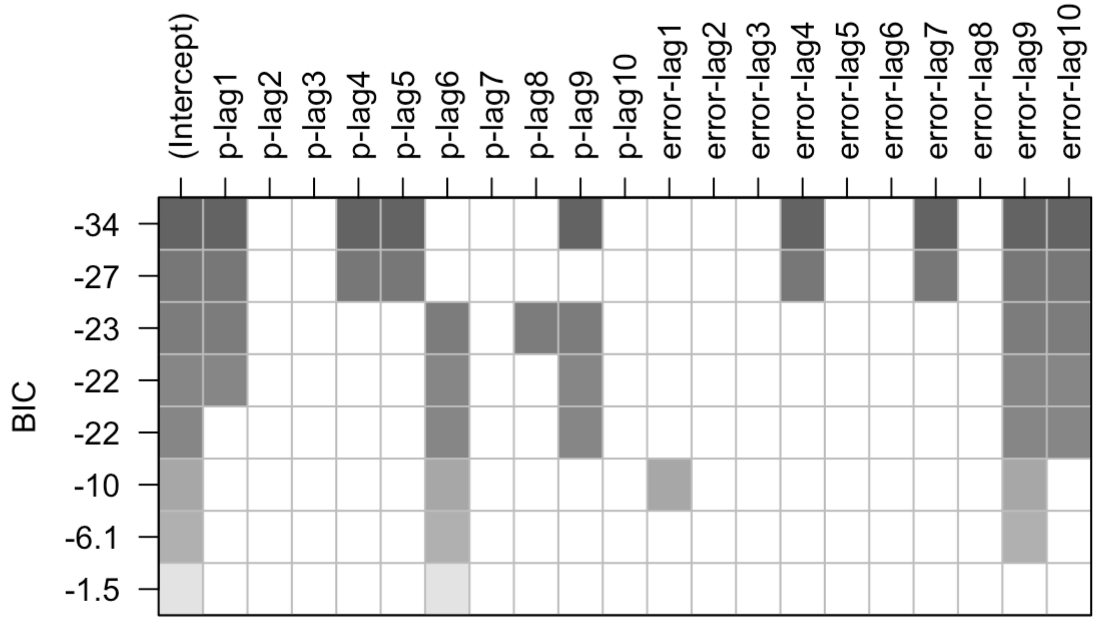

Figure 12. BIC Table to determine set values of p and q for Arctic Sea series

We selected two additional ARIMA models from the BIC table: ARIMA(1,1,9) and ARIMA(6,1,9). These models were chosen due to their relatively lower BIC values than other models in the table, suggesting that they provide a better fit for our time series data while maintaining an appropriate level of complexity.

With the full set of candidate models identified through ACF and PACF analysis, EACF table evaluation, and BIC table examination, we will now proceed to fit each of these models and compare their performance to select the most suitable ARIMA model for our Arctic Sea time series data.

4.6. Model Fitting

Although we have already applied difference to the original time series data to transform it into stationary data, this step was primarily taken to determine the order of differencing needed to achieve stationarity. Later in the analysis, we will detrend the model by fitting the original, non-stationary time series data into an ARIMA model, which will consider the appropriate order of differencing as determined earlier. This way, the ARIMA model can effectively capture the underlying patterns and relationships in the data while accounting for any trends and effects present in the original time series.

4.6.1. Parameter Estimation

We consider three estimations of parameters of ARIMA models:

- **'CSS': Conditional Sum of Squares**. This method estimates the parameters by minimizing the conditional sum of squares, considering only the residuals from the specified AR and MA lags onwards. It does not use the full likelihood function, which means it is faster to compute but may produce less accurate estimates than maximum likelihood estimation. The Conditional Sum of Squares (CSS) method does not explicitly assume normality. It focuses on minimizing the sum of squared residuals, which can be applied regardless of the data distribution.

- **'ML': Maximum Likelihood**. This method estimates the parameters by maximizing the full likelihood function, which considers the model's structure and the data-generating process. Maximum likelihood estimation is more accurate but computationally more expensive than the CSS method. The Maximum Likelihood (ML) method assumes the normality of the data. When estimating the parameters of an ARIMA model using the ML method, the likelihood function assumes that the innovations (residuals) of the model follow a Gaussian distribution.
- **'CSS-ML': A hybrid method**. First, it uses the CSS method to provide initial estimates of the parameters. Then, it refines these estimates using the maximum likelihood method. This approach can be more computationally efficient than the 'ML' method alone, especially for larger datasets or complex models, as it starts the optimization process with reasonable initial values.

Our analysis could not confirm the normality of the input data for the fitted model. As a result, we have opted to use the **CSS-ML** method as a parameter for our model fitting. The choice of CSS-ML over the ML and CSS methods is due to several reasons.

While the maximum likelihood estimation (MLE) assumes that the input data follows a normal distribution, the CSS method does not depend on any probability distribution function. This makes CSS more robust in cases where the normality assumption does not hold. However, more than CSS is needed for calculating AIC and BIC, as they require likelihood in their formulation. On the other hand, CSS-ML combines the advantages of both CSS and ML methods. In the CSS-ML approach, the iterative search for MLE commences with CSS estimates, which aids in the faster convergence of the MLE search. This allows us to use likelihood information for AIC and BIC calculations while still benefiting from the robustness of the CSS method in cases where normality may not be guaranteed. Therefore, by choosing the CSS-ML method, we can balance robustness and the ability to compute model selection criteria such as AIC and BIC, making it the most suitable option for fitting our ARIMA model.

4.6.2 Model Fitting and Significance Tests of Coefficients

This section will interpret the coefficient significance tests for each ARIMA model considered. The z-test of coefficients is used to determine the significance of the estimated parameters in each model. We expect to see all coefficients in each model are significant. Smaller p-values (below 0.05) indicate a significant relationship between the predictor and the response variable. Significant coefficients are marked with asterisks, where:

- ***: p-value < 0.001
- **: 0.001 <= p-value < 0.01
- *: 0.01 <= p-value < 0.05
- .: 0.05 <= p-value < 0.1
- (blank): p-value >= 0.1

Model: ARIMA(1,1,1) The ma1 coefficient is highly significant ($p < 0.001$). However, the ar1 coefficient is not significant ($p = 0.4072$).

Model: ARIMA(2,1,1) The ma1 coefficient is significant ($p = 0.01358$). The ar1 and ar2 coefficients are insignificant ($p = 0.27804$ and $p = 0.47646$, respectively).

Model: ARIMA(2,1,2) All coefficients, except ar2, are highly significant (ar1 and ma1: $p < 0.001$, ma2: $p < 0.01$). The ar2 coefficient is not significant ($p = 0.4038$).

Model: ARIMA(3,1,1) None of the coefficients are significant (ar1: $p = 0.1198$, ar2: $p = 0.2046$, ar3: $p = 0.2769$, ma1: $p = 0.2575$).
Model: ARIMA(3,1,2) All coefficients, except ar1, are highly significant (ar2 and ma1: $p < 0.001$, ma2: $p < 0.01$). The ar1 coefficient is significant ($p = 0.0119809$).
Model: ARIMA(0,1,1) The ma1 coefficient is highly significant ($p < 0.001$).
Model: ARIMA(0,1,2) The ma1 coefficient is highly significant ($p < 0.001$), while the ma2 coefficient is not significant ($p = 0.334$).
Model: ARIMA(1,1,2) All coefficients are highly significant (ar1 and ma1: $p < 0.001$, ma2: $p < 0.01$).
Model: ARIMA(1,1,9) The ar1, ma2, and ma5 coefficients are significant (ar1: $p < 0.001$, ma2: $p = 0.020628$, ma5: $p = 0.008485$). The ma7 and ma8 coefficients are also significant at $p < 0.05$ (ma7: $p = 0.023291$, ma8: $p = 0.050916$).
Model: ARIMA(6,1,9) The ar4, ar5, ma1, ma4, ma5, ma8, and ma9 coefficients are significant (ar4 and ma9: $p < 0.001$, ma1 and ma4: $p < 0.01$, ar5, ma5, and ma8: $p < 0.05$).
Based on the significance of the coefficients, the models ARIMA(2,1,2), ARIMA(3,1,2), Therefore, **ARIMA(0,1,1), and ARIMA(1,1,2)** show the most promising results, with the majority of their coefficients being significant. These models should be considered for further evaluation and comparison based on their performance in terms of AIC and BIC.

4.7. Diagnostics Checking

In the diagnostic check of our fitted models, we aim to assess the residuals' behaviour to ensure they meet the assumptions of a well-fitted model. Ideally, the residuals should be white noise, exhibiting no discernible patterns, correlations, or seasonality. This means the residuals should have a constant mean (usually close to zero), constant variance, and no autocorrelation. When examining the time series plot of the residuals, we expect to see random fluctuations around zero without any evident trends or seasonal patterns. However, let us examine some of the evaluation metrics first.

4.7.1 AIC and BIC

The Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) are used to compare different models and select the best one based on their relative goodness of fit. Lower AIC and BIC values indicate a better model, considering the trade-off between model complexity and goodness of fit.

Table 3. AIC and BIC scores for each ARIMA model.

Model	AIC	BIC
ARIMA(1,1,1)	79.06531	84.34891
ARIMA(2,1,1)	80.59482	87.63962
ARIMA(2,1,2)	75.34409	84.15009
ARIMA(3,1,1)	81.62889	90.43489
ARIMA(3,1,2)	79.63786	90.20506
ARIMA(0,1,1)	77.7177	81.2401
ARIMA(0,1,2)	78.67759	83.9612
ARIMA(1,1,2)	73.9841	81.0289
ARIMA(1,1,9)	77.46541	96.83861
ARIMA(6,1,9)	81.3445	109.5237

Based on the AIC scores, the ARIMA(1,1,2) model performs the best, followed closely by the ARIMA(2,1,2) model. When considering the BIC scores, which penalize model complexity more heavily, the ARIMA(0,1,1) model has the lowest BIC score. However, the ARIMA(1,1,2) model still performs relatively well, with a slightly higher BIC score than the ARIMA(0,1,1) model.

Based on the AIC and BIC scores and the significance of the coefficients, the **ARIMA(1,1,2)** model appears to be the most appropriate model for the Arctic Sea Ice time series data. This model balances model complexity and fit while still having significant coefficients.

4.7.2 ME, RMSE, MAE, MPE, MAPE, MASE, ACF1 Scores

Each table row corresponds to an ARIMA model with specified p, d, and q values (in the format ARIMA(p,d,q)). Lower values of these accuracy measures generally indicate a better fit of the model to the data. The table includes the following accuracy measures for each model:

- ME: Mean Error
- RMSE: Root Mean Squared Error
- MAE: Mean Absolute Error
- MPE: Mean Percentage Error
- MAPE: Mean Absolute Percentage Error
- MASE: Mean Absolute Scaled Error
- ACF1: Autocorrelation of residuals at lag 1

Table 4. Comparison of ME, RMSE, MAE, MPE, MAPE, MASE, ACF1 Scores for each model

Model	ME	RMSE	MAE	MPE	MAPE
ARIMA(1,1,1)	-0.15086239	0.5557352	0.4482326	-3.632910	8.629260
ARIMA(2,1,1)	-0.15399186	0.5523046	0.4409619	-3.687298	8.467021
ARIMA(2,1,2)	-0.08845814	0.4807002	0.3848922	-2.248990	7.329126
ARIMA(3,1,1)	-0.16158089	0.5455148	0.4265380	-3.812230	8.187461
ARIMA(3,1,2)	-0.09485837	0.4940196	0.3930167	-2.388306	7.250946
ARIMA(0,1,1)	-0.15257478	0.5604544	0.4548486	-3.678329	8.722628
ARIMA(0,1,2)	-0.15036950	0.5526560	0.4412597	-3.610873	8.495991
ARIMA(1,1,2)	-0.07996605	0.4964359	0.3989249	-2.091620	7.591672
ARIMA(1,1,9)	-0.06850758	0.3945930	0.3316751	-1.670089	6.184845
ARIMA(6,1,9)	-0.04577465	0.3604483	0.2971254	-1.134684	5.422577

This table shows that the ARIMA(6,1,9) model has the lowest RMSE, MAE, MPE, MAPE, MASE, and ACF1, suggesting that it may be the best model among those considered for this dataset. However, it is important to consider model simplicity and other diagnostic tools like residual plots and ACF plots of residuals when deciding the best model.

That is why we are looking for a simpler model with relatively good performance for all metrics. The ARIMA(1,1,2) model is still a good choice. It has fewer parameters than the ARIMA(6,1,9) model and performs reasonably well across all metrics. The ARIMA(1,1,2) model has an RMSE of 0.4964359, which is higher than the best model ARIMA(6,1,9) but still relatively low. The other metrics (MAE, MPE, MAPE, MASE, and ACF1) are also reasonably good, indicating that the model performs well.

4.7.3 Plots for the Residuals Analysis

residual = actual - predicted

For an effective model that captures data's underlying structure, several conditions are expected:

- The time series plot of the residuals should display a rectangular scatter around a horizontal zero level without any observable trends.
- The histogram of standardized residuals should be symmetric, with values ranging between -3 and +3, indicative of a normal distribution.
- The Quantile-Quantile (QQ) plot should display dots closely aligned with the reference line, suggesting that the residuals are normally distributed.
- The Autocorrelation Function (ACF) plot should not exhibit any significant autocorrelations, as this would indicate remaining patterns in the residuals that the model has not accounted for.

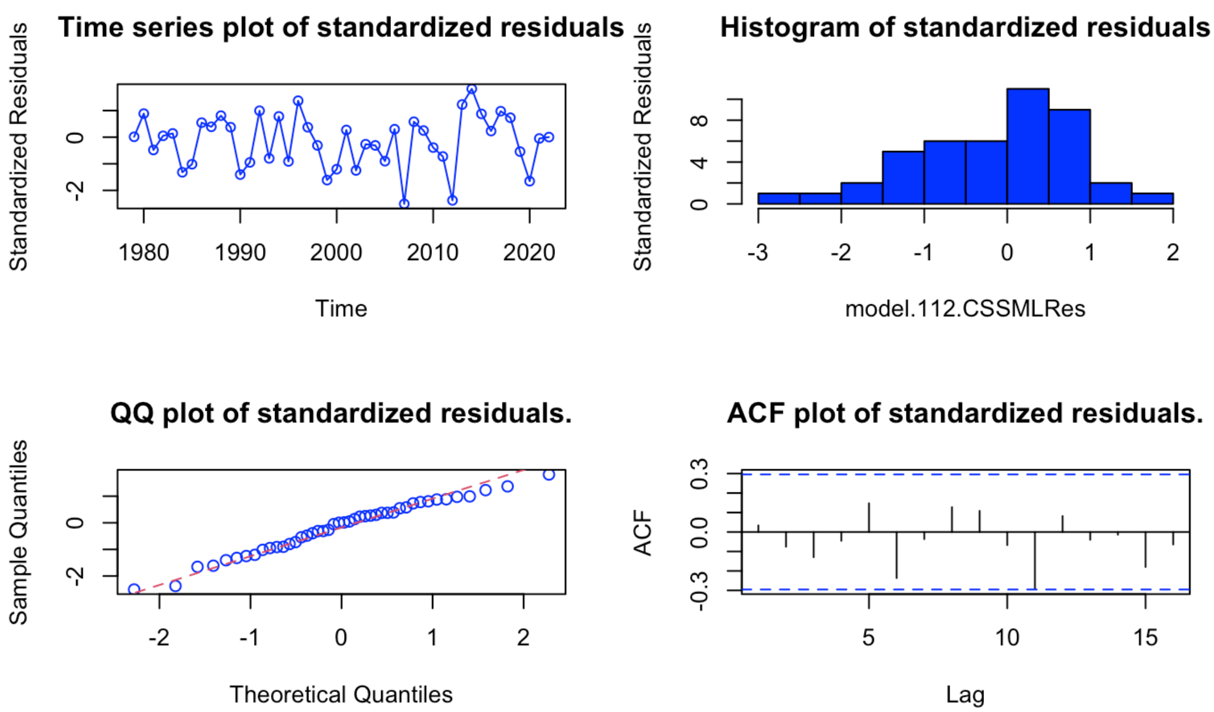

Figure 13. Standardized Residuals Plots

- The time series plot of the residuals does not exhibit any discernible trend, which is a positive indication. However, some variances in the plot suggest that the model may have yet to capture all underlying patterns in the data fully.
- The histogram of standardized residuals predominantly lies within the range of -3 and +3, as expected for a normal distribution. However, the plot appears somewhat asymmetric, raising concerns about the normality assumption for the residuals.
- The Quantile-Quantile (QQ) plot shows that some data points deviate from the reference line at both tails, indicating potential deviations from normality.
- The ACF plot of the residuals, it is evident that no significant correlations or patterns remain in the series.

These observations suggest that while our model has captured some of the essential features of the data, there may still be room for improvement. Further analysis or exploration of alternative models could lead to a better fit and more accurate forecasts.

4.7.4 Box-Ljung test

The Box-Ljung test (also known as the Ljung-Box test) is used to check for autocorrelation in the residuals of a time series model. A high p-value (greater than the chosen significance level of 0.05) indicates no significant evidence of autocorrelation in the residuals. A low p-value (less than the significance level) suggests that autocorrelation is present.

In this case, the Box-Ljung test result is as follows:

- X-squared: 0.054517
- Degrees of freedom (df): 1
- p-value: 0.8154

Since the p-value (0.8154) is greater than the common significance level of 0.05, we fail to reject the null hypothesis, meaning there is no significant evidence of autocorrelation in the residuals. This result suggests that the ARIMA(1,1,2) model has captured the structure of the time series, and the residuals can be considered white noise.

4.7.5 Shapiro-Wilk normality test

- W: 0.97998
- p-value: 0.6328

Since the p-value (0.6328) is greater than the significance level of 0.05, we fail to reject the null hypothesis, meaning that the residuals are normally distributed at a 5% significance level in this model. This result suggests that the ARIMA(1,1,2) model's residuals are consistent with the assumption of normality, which is important for many statistical tests and model validity.

4.8. Forecasting

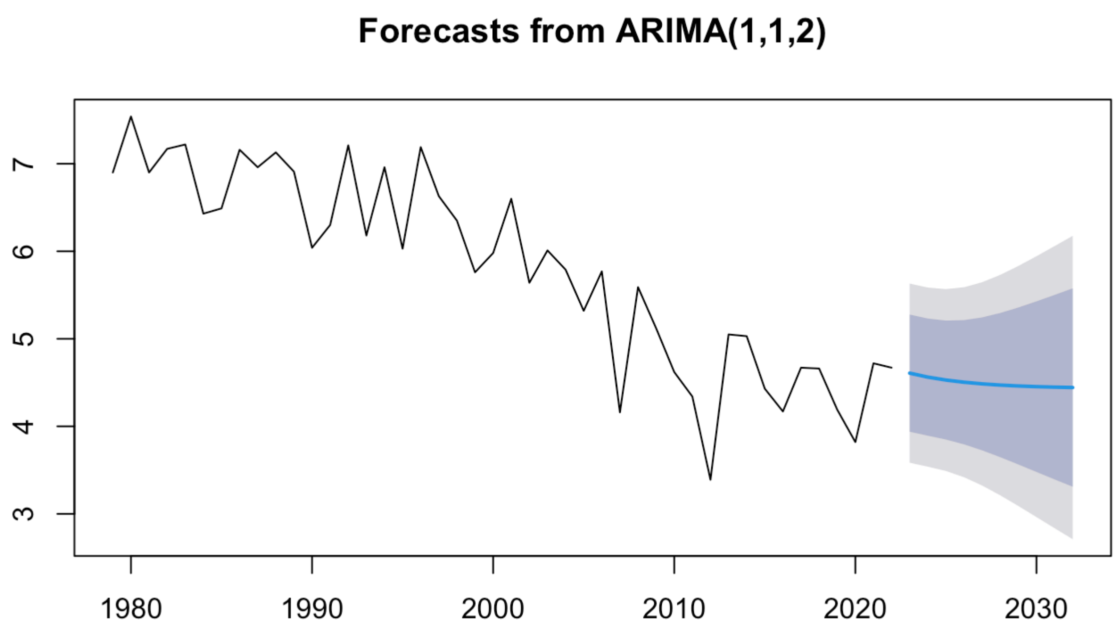

Figure 14. Forecasts from ARIMA(1,1,2) model.

Table 5. The point forecasts for the years 2023 to 2032, along with the 80% and 95% prediction intervals.

Year	Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
2023	4.607644	3.938988	5.276300	3.585023	5.630265
2024	4.562466	3.893945	5.230986	3.540051	5.584880
2025	4.528840	3.850089	5.207592	3.490780	5.566901
2026	4.503814	3.794784	5.212844	3.419446	5.588181
2027	4.485187	3.726476	5.243899	3.324838	5.645536
2028	4.471324	3.648502	5.294146	3.212926	5.729722
2029	4.461006	3.565096	5.356916	3.090829	5.831182
2030	4.453326	3.479681	5.426971	2.964265	5.942387
2031	4.447610	3.394560	5.500660	2.837109	6.058111
2032	4.443356	3.311114	5.575598	2.711742	6.174971

In 2023, the point forecast is 4.607644. This means that the expected value for that year is 4.607644. There is an 80% chance that the actual value will be between 3.938988 and 5.276300 and a 95% chance that it will be between 3.585023 and 5.630265. The forecast intervals become wider as we move further into the future, reflecting increased uncertainty.

5. Results

The analysis involved fitting multiple ARIMA models to the time series data and comparing their performance based on various evaluation metrics, such as AIC, BIC, ME, RMSE, MAE, MPE, MAPE, MASE, and ACF1. Models with different combinations of autoregressive (AR), integrated (I), and moving average (MA) components were tested. Additionally, the Box-Cox transformation was applied to assess its impact on data normality but not used in the model.

6. Discussion

The ARIMA(1,1,2) model emerged as the simplest model with relatively good scores for all evaluation metrics. This model had a lower AIC and BIC than the other models, suggesting that it is a better fit for the data. Although The Box-Cox transformation slightly improved data normality, with the Shapiro-Wilk normality test p-value increasing from 0.0373 to 0.04554, the p-value remained below the significance level, indicating that the data deviated from normality even after the transformation.

7. Conclusion

Based on the evaluation metrics and the principle of parsimony, the ARIMA(1,1,2) model is recommended as the best model for forecasting in this case. Although the Box-Cox transformation improved data normality to some extent, it did not provide a significant improvement. Therefore, the transformation will not be used for the final analysis.

8. Recommendations

- Utilize the ARIMA(1,1,2) model for future forecasting, as it provides the best balance between simplicity and performance.
- Monitor the residuals of the chosen model for any potential deviations from normality or changes in variance over time, which may indicate a need for model reassessment.
- We consider exploring other forecasting methods, such as state-space models or machine-learning techniques, to improve forecasting performance.

- If data normality is a major concern, it may be beneficial to investigate other data transformations or pre-processing techniques to address this issue better.

9. Appendices: R code

````
---
title: "assignment2Solution_s3879312"
author: "Thu Tran"
date: "`r Sys.Date()`"
output: pdf_document
---

```{r import libraries}
rm(list=ls())
library(TSA)
library(tseries)
library(ggplot2)
library(lmtest)
library(forecast)
```



```{r data overview}
# read data into a data frame
arctic_sea_ice <- read.csv("assignment2Data2023.csv", header = TRUE)
# display first 3 rows of data frame
head(arctic_sea_ice, n=3)
# class of the data
class(arctic_sea_ice)
# display summary statistics
summary(arctic_sea_ice)
```



```{r Create the ACF plot for finding the frequency - no need since this is annual data}
# Create the ACF plot with custom title, y-axis label, x-axis label, and color
acf(arctic_sea_ice, main = "ACF of Arctic Sea Ice Extent")
# confirm the frequency
frequency(arctic_sea_ice)
```



```{r convert data frame to time series (ts) object}
# Create a vector of the Arctic sea ice minimum extent values
````
````

```
arctic_sea_ice_ts <- arctic_sea_ice$Arctic.Sea.Ice.Extent..million.square.km.
# Convert the vector to a time series object
arctic_sea_ice_ts <- ts(arctic_sea_ice_ts, start = 1979, end = 2022, frequency = 1)
class(arctic_sea_ice_ts)
arctic_sea_ice_ts
```

```{r time series plot}
plot(arctic_sea_ice, type='o', main = "Time series plot of Arctic Sea Ice Minimum Extent
(1979-2022)", ylab = "Million square km", xlab = "Year", col="blue")
```

```{r normality test of the original time series data}
qqnorm(y = arctic_sea_ice_ts, main = "QQ plot of the Acrtic Extent series", col = "blue")
qqline(y = arctic_sea_ice_ts, col = 2, lwd = 1, lty = 2)
shapiro.test(arctic_sea_ice_ts)
# p-value = 0.0373
```

```{r scatter plot to show the relationship between pairs of consecutive Arctic Sea Ice
Minimum Extent}
plot(y=arctic_sea_ice_ts,x=zlag(arctic_sea_ice_ts),ylab='Million square km', xlab='Previous
Year Arctic Sea Ice Minimum Extent (million square km)', main = "Scatter plot of Arctic Sea
Ice Minimum Extent in consequtive years", col="blue")

# the correlation show linear relationship between pairs of consecutive Arctic Sea Ice
Minimum Extent
y = arctic_sea_ice_ts            # Read the annual data into y
x = zlag(arctic_sea_ice_ts)      # Generate first lag of the annual series
index = 2:length(x)    # Create an index to get rid of the first NA value in x
cor(y[index],x[index]) # Calculate correlation between numerical values in x and y
```

```{r Stationary Check: ACF plot and ACF numerical representation}
# Create the ACF plot: slowly decay trend
acf(arctic_sea_ice_ts, main = "ACF of Arctic Sea Ice Extent Time Series")

# convert the ACF plot into a numerical representation
# Calculate ACF values for the Arctic Sea Ice Extent Time Series
acf_values <- acf(arctic_sea_ice_ts, plot = FALSE)
# Print the ACF values
```

````
acf_values$acf
```


```{r Stationary Check: PACF plot and PACF numerical representation}
# large first lag in the pacf
pacf(arctic_sea_ice_ts, main = "PACF of Arctic Sea Ice Minimum Extent Time Series")

# convert the ACF plot into a numerical representation
# Calculate PACF values for the Arctic Sea Ice Minimum Extent Time Series
pacf_values <- pacf(arctic_sea_ice_ts, plot = FALSE)
# Print the PACF values
pacf_values$acf
```

```{r Stationary Check: of the original time series data}
adf.test(arctic_sea_ice_ts)
# p-value = 0.4433: non-stationary
```




```{r Box-Cox transformation}
BC <- suppressWarnings(BoxCox.ar(y = arctic_sea_ice_ts, lambda = seq(-2, 2, 0.01)))
title(main = "Log-likelihood versus the values of lambda for Acrtic Extent")

BC$ci # Values of the first and third vertical lines

# To find the lambda value of the middle vertical line
lambda <- BC$lambda[which(max(BC$loglike) == BC$loglike)]
lambda
```


```{r}
# Apply Box-Cox transformation using the optimal lambda value
BC_arctic_sea_ice_ts <- ((arctic_sea_ice_ts^lambda) - 1) / lambda

# Set the output file and dimensions (width, height) in pixels
png("BC_arctic_sea_ice_ts.png", width = 1200, height = 600)

# Set up the layout for side-by-side plots
par(mfrow = c(1, 2))

# Create a time series plot of the original data
````

```
plot(arctic_sea_ice_ts,
    type = "l",
    col = "black",
    main = "Original Arctic Sea Ice Minimum Extent Time Series",
    xlab = "Year",
    ylab = "Arctic Sea Ice Minimum Extent (million square km)")

# Create a time series plot of the Box-Cox transformed data
plot(BC_arctic_sea_ice_ts,
    type = "l",
    col = "blue",
    main = "Box-Cox transformation Arctic Sea Ice Minimum Extent Time Series",
    xlab = "Year",
    ylab = "Box-Cox transformation Arctic Sea Ice Minimum Extent (million square km)")

# Reset the layout
par(mfrow = c(1, 1))
```

```{r Transformation improve normality?}
qqnorm(y = BC_arctic_sea_ice_ts, main = "QQ plot of the Acrtic Extent series", col = "blue")
qqline(y = BC_arctic_sea_ice_ts, col = 2, lwd = 1, lty = 2)
shapiro.test(BC_arctic_sea_ice_ts)
```



```{r First Difference}
# First Difference
first_diff_arctic_sea_ice <- diff(arctic_sea_ice_ts, differences = 1)

# Set the output file and dimensions (width, height) in pixels
png("first_differenced_arctic_sea_ice_plot.png", width = 1200, height = 600)
# Set up the layout for side-by-side plots
par(mfrow = c(1, 2))
# Create a time series plot of the original data
plot(arctic_sea_ice_ts,
    type = "l",
    col = "black",
    main = "Original Arctic Sea Ice Minimum Extent Time Series",
    xlab = "Year",
    ylab = "Arctic Sea Ice Minimum Extent (million square km)")
# Create a time series plot of the first differenced data
plot(first_diff_arctic_sea_ice,
    type = "l",
    col = "blue",
    main = "First Differenced Arctic Sea Ice Minimum Extent Time Series",
```

```
    xlab = "Year",
    ylab = "Differenced Arctic Sea Ice Minimum Extent (million square km)")
# Reset the layout
par(mfrow = c(1, 1))
```

```{r Stationary Tests}
suppressWarnings({
  adf <- adf.test(first_diff_arctic_sea_ice, alternative = "stationary")
  pp <- pp.test(first_diff_arctic_sea_ice)
  kpss <- kpss.test(first_diff_arctic_sea_ice, null = "Level")
})

# Print the test results without the warnings
adf
pp
kpss
```

```{r}
# Plot the ACF of the first differenced time series
acf(first_diff_arctic_sea_ice, main = "ACF of the First Difference")
# Plot the PACF of the first differenced time series
pacf(first_diff_arctic_sea_ice, main = "PACF of the First Difference")
# {ARIMA(1,1, 1), ARMA(2,1,1), ARMA(2,1,2), ARMA(3,1,1), ARMA(3,1,2)}
```

```{r Model Selection: EACF table}
# EACF
eacf(first_diff_arctic_sea_ice, ar.max = 10, ma.max = 10)
# {ARIMA(0,1,1), ARIMA(0,1,2), ARIMA(1,1,1), ARIMA(1,1,2)}
```

```{r BIC Table}
res = armasubsets(y=first_diff_arctic_sea_ice,nar=10,nma=10,y.name='p',ar.method='ols')
plot(res)
# {ARIMA(1,1,9), ARIMA(6,1,9)}
```

```{r Model Fitting Parameter Estimations, BIC and AIC scores}
```

```
fit_arima_models <- function(time_series, arima_orders) {
  models <- list()
  for (order in arima_orders) {
    model <- arima(time_series, order = order, method = 'CSS-ML')
    coef_test <- coeftest(model)
    aic_score <- AIC(model)
    bic_score <- BIC(model)
    models[[paste("ARIMA(", paste(order, collapse = ","), ")", sep = "")]] <- list(model =
model, coef_test = coef_test, AIC = aic_score, BIC = bic_score)
  }
  return(models)
}

# Define the list of ARIMA models
arima_orders <- list(
  c(1, 1, 1), c(2, 1, 1), c(2, 1, 2), c(3, 1, 1), c(3, 1, 2),
  c(0, 1, 1), c(0, 1, 2), c(1, 1, 1), c(1, 1, 2),
  c(1, 1, 9), c(6, 1, 9)
)
arctic_sea_ice_models <- fit_arima_models(arctic_sea_ice_ts, arima_orders)

# Accessing the models and their coefficient tests, AIC and BIC scores:
for (model_name in names(arctic_sea_ice_models)) {
  cat("Model:", model_name, "\n")
  cat("Coefficient test:\n")
  print(arctic_sea_ice_models[[model_name]]$coef_test)
  cat("AIC:", arctic_sea_ice_models[[model_name]]$AIC, "\n")
  cat("BIC:", arctic_sea_ice_models[[model_name]]$BIC, "\n\n")
}
```

```
```{r "ME", "RMSE", "MAE", "MPE", "MAPE", "MASE", "ACF1" scores}
# Function to fit ARIMA models and compute accuracy
fit_arima_models <- function(data, arima_orders) {
  models <- list()
  accuracy_measures <- list()

  for (order in arima_orders) {
    model <- Arima(data, order = order, method = 'CSS-ML')
    models[[paste0("ARIMA(", paste(order, collapse = ","), ")")]] <- model
    accuracy_measures[[paste0("ARIMA(", paste(order, collapse = ","), ")")]] <-
accuracy(model)[1:7]
  }
```
```

```
  df_accuracy <- data.frame(do.call(rbind, accuracy_measures))
  colnames(df_accuracy) <- c("ME", "RMSE", "MAE", "MPE", "MAPE", "MASE", "ACF1")

  return(df_accuracy)
}

# Define the list of ARIMA models
arima_orders <- list(
  c(1, 1, 1), c(2, 1, 1), c(2, 1, 2), c(3, 1, 1), c(3, 1, 2),
  c(0, 1, 1), c(0, 1, 2), c(1, 1, 1), c(1, 1, 2),
  c(1, 1, 9), c(6, 1, 9)
)

# Call the function with your data and the list of ARIMA models
accuracy_results <- fit_arima_models(arctic_sea_ice_ts, arima_orders)
print(accuracy_results)

```


We found ARIMA(1,1,2) model is the most promissing one for Diagnostic Checking

```{r 4 residual plots}
# Fit the ARIMA model
model.112.CSSML <- arima(arctic_sea_ice_ts, order = c(1, 1, 2), method = 'CSS-ML')
model.112.CSSMLRes <- rstandard(model.112.CSSML)

# Set up the multi-panel plot layout
par(mfrow = c(2, 2))

# Time series plot of standardized residuals
plot(model.112.CSSMLRes, xlab = 'Time', ylab = 'Standardized Residuals', type = 'o',
     main = "Time series plot of standardized residuals", col = "blue")

# Histogram of standardized residuals
hist(model.112.CSSMLRes, ylab = 'Standardized Residuals',
     main = "Histogram of standardized residuals", col = "blue")

# QQ plot of standardized residuals
qqnorm(model.112.CSSMLRes, main = "QQ plot of standardized residuals.", col = "blue")
qqline(model.112.CSSMLRes, col = 2, lwd = 1, lty = 2)

# ACF plot of standardized residuals
acf(model.112.CSSMLRes, main = "ACF plot of standardized residuals.")
```

````
```



```{r autocorrelation test}
Box.test(model.112.CSSMLRes, type = "Ljung-Box")
```

```{r normality test}
shapiro.test(model.112.CSSMLRes)
```



```{r Forecasting consecutive 10 years}
model.112.CSSMLA = Arima(arctic_sea_ice_ts,order=c(1,1,2),method='CSS-ML')
model.112.CSSMLAfrc = forecast::forecast(model.112.CSSMLA, h = 10)
plot(model.112.CSSMLAfrc)
model.112.CSSMLAfrc
```
````
````

References

[1] Demirhan, H., "Lecture slides for Time Series Analysis, Semester 1, 2023," RMIT University, Melbourne, Australia.

[2] A. M. De Livera, R. J. Hyndman, and R. D. Snyder, “Forecasting Time Series With Complex Seasonal Patterns Using Exponential Smoothing,” Journal of the American Statistical Association, vol. 106, no. 496, pp. 1513–1527, Dec. 2011, doi: https://doi.org/10.1198/jasa.2011.tm09771.

[3] S. Fan and R. J. Hyndman, “Short-Term Load Forecasting Based on a Semi-Parametric Additive Model,” *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 134–141, Feb. 2012, doi: https://doi.org/10.1109/tpwrs.2011.2162082.

[4] E. Parzen, “ARARMA models for time series analysis and forecasting,” *Journal of Forecasting*, vol. 1, no. 1, pp. 67–82, Jan. 1982, doi: https://doi.org/10.1002/for.3980010108.

[5] “11.1 Complex seasonality | Forecasting: Principles and Practice,” *Otexts.com*, 2011. https://otexts.com/fpp2/complexseasonality.html

[6] J. D. Cryer and K.-S. Chan, *Time series analysis: with applications in R*. New York: Springer, 2008.