



Uio • University of Oslo

Forecasting Norwegian oil production using time series forecasting models

Nelly Askjer

Thesis submitted for the degree of Master of Economics
30 credits

Department of Economics

Faculty of Social Sciences

UNIVERSITY OF OSLO

May 2021

Summary

Oil production has been of great importance for Norway and its economy since it was discovered back in the 70s. It is estimated that oil production and the oil related industry are accountable for 25% of Norway's Gross Domestic Product. Understanding the different variables that impact oil production in the country while predicting the levels of production can therefore result in better decision making process for the economy and society as whole. This is the goal of the Norwegian Petroleum Directorate (NPD), which collects data of oil production in the country and forecasts the oil production for the subsequent 12 months ahead, starting every December of each year.

The objective of this thesis is to evaluate if statistical forecasting methods used to forecast oil production in Norway can result in a higher accuracy compared to the forecasts done by the NPD. The data collection process has been done through the NPD website, while all the evaluations for different forecast methods have been done by using RStudio and its core programming language R.

This thesis focus on time series forecasting based on historical data rather than a deep understanding of the components affecting oil production. Different forecasting methods were constructed to find the method that performed the best for the data of interest. Forecasts for the monthly Norwegian oil production was fitted by the use of several simple forecast methods along with more advanced ARIMA models to investigate if the simple methods could give higher accuracy than the forecast given by the ARIMA models.

Table of content

<i>Summary</i>	<i>II</i>
1. Introduction	1
1.1. The time series of interest	2
1.2. Thesis structure.....	2
2. Review of time series theory	3
2.1. Time series.....	3
2.2. Types of variation	3
2.2.1. Trend.....	4
2.2.2. Seasonal variation	4
2.2.3. Cyclic variation	4
2.2.4. Irregular fluctuations	4
2.3. Stationarity time series.....	4
2.3.1. White noise.....	5
2.4. Non-stationarity and differencing	7
2.4.1. Random walk.....	8
2.5. ARIMA modelling.....	10
2.5.1. Autoregressive processes AR(p)	10
2.5.2. Moving average processes MA(q).....	11
2.5.3. Autoregressive moving average models ARMA(p,q).....	11
2.5.4. Integrated ARMA models ARIMA(p,d,q).....	12
2.5.5. Seasonal ARIMA models.....	13
2.6. Measuring forecast accuracy	13
2.6.1. Mean absolute error.....	14
2.6.2. Mean squared error.....	14
3. Model selection in ARIMA modelling	15
3.1. Decomposition	16
3.2. Selecting the value of d	17
3.2.1. Dickey-Fuller test	17
3.2.2. Augmented Dickey-Fuller test	18
3.2.3. KPSS Unit root test.....	19
3.2.4. Selecting the value of D.....	21
3.3. Inspecting the autocorrelation function.....	22
3.3.1. Determining the order of a MA process.....	22
3.3.2. Determining the order of an AR process	22
3.3.3. Maximum likelihood estimation	24
3.3.4. Seasonal differenced series.....	25
3.4. The automated ARIMA model	26
3.4.1. The non-seasonal automated ARIMA.....	26
3.4.2. The seasonal automated ARIMA.....	27
3.4.3. Selecting the appropriate ARIMA model manually	28
3.5. Simple forecast methods.....	30
3.5.1. Mean method.....	30
3.5.2. Naïve method.....	30
3.5.3. Seasonal naïve method.....	30
3.5.4. Drift method	31
3.6. Comparing the simple forecasts methods.....	31

<i>4. Comparing forecast accuracy</i>	32
<i>5. Discussion</i>	34
<i>6. References</i>	36
<i>7. Appendix</i>	39

1. Introduction

Crude oil is one of the most important export commodities for the Norwegian economy, and almost all the oil produced in Norway is exported. During 2019, a total of 248 Million NOK have been contributed to the Norwegian Gross Product Income (GPD), also being responsible for 27% of the total external trade in the same period (Norwegian Petroleum, 2020).

The income from producing and exporting oil is central for the wealth of the country of Norway. If future oil production is predicted as accurately as possible, informed decisions about future transportation, investments, income and personnel can be planned. Forecasting is therefore an important aid to effective and strategic planning, and should be considered under decision-making for a business, as it can play an important part role in an economy's management.

Several components can affect the demand and production of Norwegian oil. This thesis will focus on time series forecasting based on the use of historical data of oil production rather than a deep understanding of the components affecting future oil production.

Time series forecasting is the use of historical data to predict future values. By selecting the forecasting method that forecasts known data points of the series with the highest accuracy available, the probability of accurate forecasted values for the future increases. It is therefore important to compare different forecasting models to find the method that performs the best for the data of interest.

The Norwegian Petroleum Directorate posted a forecast for Norwegian oil production for every month in 2018, obtained in December the year before (Norwegian Petroleum Directorate, 2019). This thesis aims to investigate if a more precise prediction for the same data points can be fitted.

1.1. The time series of interest

The time series of interest in this thesis is the Norwegian monthly oil production from June 1971 to December 2019, measured in million barrels.

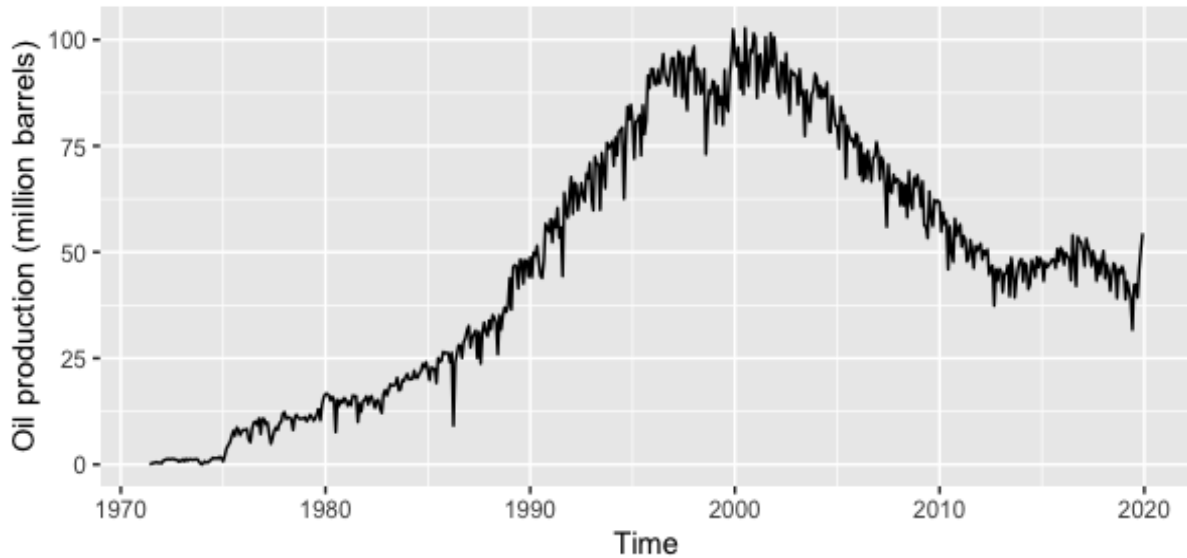


Figure 1.1: Time plot of Norwegian monthly oil production from 1971 to 2019 measured in million barrels. The plot illustrates that the production reached a peak in year 2000. The plot is constructed in RStudio (Appendix 1.A).

As illustrated in Figure 1.1, Norwegian oil production reached a peak in year 2000 with a production of 1140 barrels, and then declined until 2013. After 2014 the production has been increasing (Norwegian Petroleum, 2021).

The time series was originally obtained in standard cubic metres of oil equivalents (Sm^3 o.e.) and been transformed into million barrels in this study. The years 1971 - 2018 in the data will be used to estimate how well the forecasts performs on the time series of interest. The year 2019 will be forecasted with the most accurate forecast method from this study.

1.2. Thesis structure

This thesis will start with an introduction of time series theory. Different variations and types of time series will be presented before introducing the forecast models and measures of accuracy that will be applied to compare the time series forecasts in this study.

Different time series forecasting models will be constructed in the software RStudio using programming language R to forecast Norwegian monthly oil production. Various cases of ARIMA models will be fitted to investigate which ARIMA model fits the data most appropriately according to the Akaike Information Criterion (AIC). To investigate if the time series needs differencing to obtain stationarity, the series will be investigated by decomposition and unit root tests.

Forecasts from simple forecast methods will be compared with the advanced appropriate ARIMA model and the advanced forecast from the NPD to investigate if simple forecast methods can give lower forecast errors for this time series. If any advanced forecast methods give a higher error than the simple methods, they will not be considered. The most accurate forecast model will be selected comparing the forecasts by the accuracy measures mean absolute error (MAE) and mean squared error (MSE). The most accurate forecast model in this study will be selected to predict the last year of the time series of interest.

2. Review of time series theory

2.1. Time series

A time series is a collection of observations arranged by the date of each observation (Hamilton, 1994, page 25). It is generally assumed that time is a discrete variable. A time series is discrete when observations are taken at specific times, generally with equal space between every observation (Kirchgässner & Wolters, 2007, page 1). The time series for variable y data beginning at time $t = 1$, and ending at another $t = T$ can be presented as:

$$(y_1, y_2, y_3 \dots y_T) \quad (2.1)$$

2.2. Types of variation

A traditional method for analysing time series contains separating variations in the series into the components trend, seasonal variation and other cyclic variation. Any remaining variation can be referred to as irregular fluctuations.

2.2.1. Trend

A trend is defined as a long-term change in the mean level of the series (Chatfield, 2004, page 26). The difficulty of defining a trend is deciding what is meant by “long-term”. The number of observations available in the data will be taken into account when making a subjective assessment of what long term change is in the time series of interest (Springer, 2008).

2.2.2. Seasonal variation

Seasonal variation can be defined as the changes that is repeated regularly through periods. These variations usually have the same or similar pattern year after year. When data is reported daily, weekly, monthly or quarterly, it is possible to look for seasonal variations. The data of interest is monthly, and it is therefore possible to investigate for seasonal variation in this data (Springer, 2008).

2.2.3. Cyclic variation

Cyclical variations comes from increases and decreases recurring after periods at irregular times. The four phases of a business cycle, a recession, depression, recovery or boom can cause these variations (Springer, 2008).

2.2.4. Irregular fluctuations

Irregular fluctuations are results of unpredictable forces. These forces operate randomly and do not have any definite pattern. These fluctuations can come from earthquakes, strikes or pandemics (Springer, 2008).

2.3. Stationarity time series

A time series is said to be weakly stationary or second order stationary if the mean μ_t and the autocovariance γ_{jt} does not depend on the date t:

$$E(Y_t) = \mu \quad \text{For all } t$$

$$E(Y_t - \mu)(Y_{t-j} - \mu) = \gamma_{jt} \quad \text{for all } t \text{ and } j$$

(Hamilton, 1994, page 46).

A time series is said to be strictly stationary if the joint distribution of $(Y_t, Y_{t+j_1}, Y_{t+j_2}, \dots, Y_{t+j_n})$ only depends on the intervals separating the dates (j_1, j_2, \dots, j_n) and not the date t itself for any values on j (Hamilton, 1994, page 46). Many of the properties of stationary processes only depend on the structure of the process that are specified by its first and second order moments. In this text the term “stationarity” will imply weak stationarity.

A time series is stationary if there is no systematic change in the mean from trend, if there is no systematic change in variance, and if strictly periodic variations have been removed (Chatfield, 2004, page 13). Trend and seasonal variation affect the mean and variance of the time series differently through time, and time series with trends or seasonality are therefore not stationary. A stationary time series will have no predictable patterns in a long-term perspective, and time series plots will therefore be roughly horizontal with constant variance.

Since the property of stationarity is defined for a model, there are no time-series that is stationary when talking in strict terms (Chatfield, 2004, page 13). The phrase “stationary time-series” will in this thesis be referred to as time-series data that contain characteristics that implies that a stationary model can be fitted properly.

A times series can also have variation from cyclic behaviour that will show variation in the time plots of the series. If the variation in the series comes from cyclic behaviour, the mean and variance will not be affected since it is no trend or seasonality present. A time series with variation from cyclic behaviour can therefore be stationary (Hyndman & Athanasopoulos, 2018, chapter 8.1).

2.3.1. White noise

White noise is a purely random process defined as a sequence of independent and identically distributed (i.i.d) random variables (Chatfield, 2004, page 221). A white noise times series is stationary, and therefore the first and second order moments, mean and variance, does not depend on the time the series are observed (Hamilton, 1994, page 50).

A white noise series will have no predictable patterns, and time series plots of white noise will therefore be roughly horizontal with constant variance.

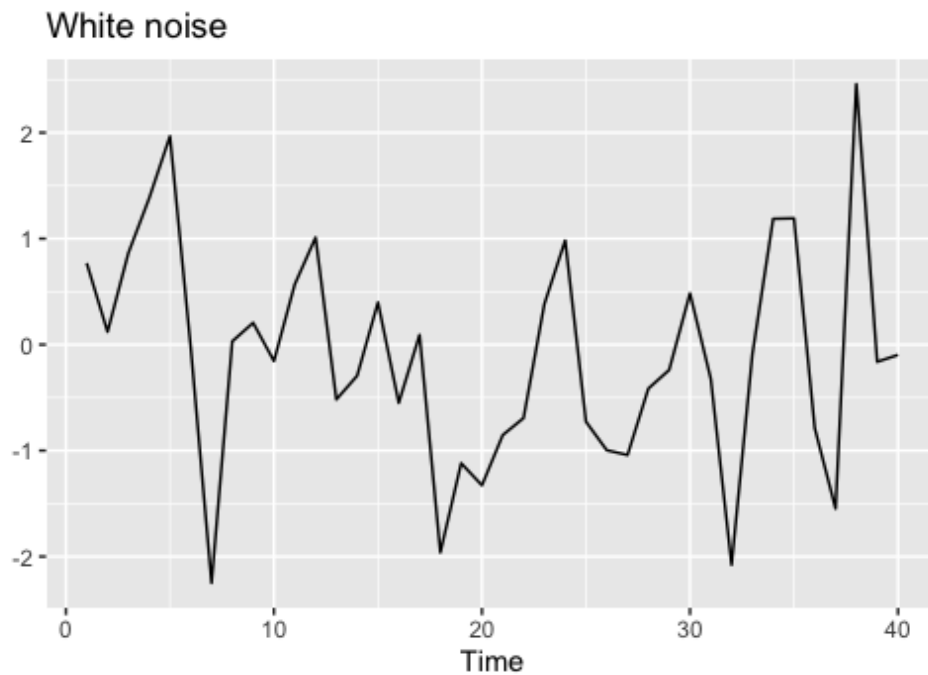


Figure 2.1: Time plot for white noise. The plot illustrates no predictable patterns in the time series. The plot is constructed in RStudio (Appendix 2.A).

It is expected that each autocorrelation is close to zero for white noise series. Due to random variation, the autocorrelation will not be exactly equal to zero. Autocorrelation can be represented by the autocorrelation function (ACF) plot. The ACF plot in Figure 2.2 is an autocorrelation function illustrating how well the present value of the series is related with its past values. The autocorrelation function will therefore investigate for autocorrelation between present production and the production at different times. The dotted blue lines in Figure 2.2 are at value $\pm 2/\sqrt{T}$, where T is the number of observations in the time series. Values outside the dotted lines are significantly different from zero (Chatfield, 2004, page 27).

With no autocorrelation, 95% of the spikes in the autocorrelation function plot is expected to lie within the blue dotted bounds in Figure 2.2. If one or more large spikes are outside these limits, or if more than 5% of the spikes are outside these limits, the series is probably not white noise (Hyndman & Athanasopoulos, 2018, chapter 2.9).

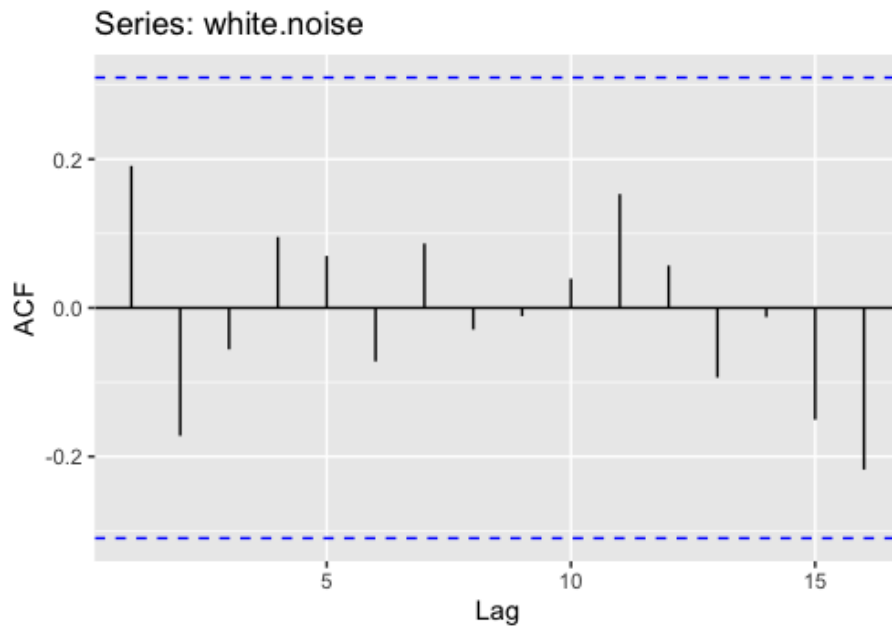


Figure 2.2: Autocorrelation function plot for white noise series. All spikes lie within the significance bounds, implying no autocorrelation between the observations. The plot is constructed in RStudio (Appendix 2.B).

All the autocorrelation coefficients lie within the blue dotted lines, which confirms that the data in this example are white noise and stationary.

2.4. Non-stationarity and differencing

The use of non-stationary data in time series models can produce unreliable and spurious results. Spurious results implies that series seems to be correlated, but are not in reality. High residual autocorrelation can be signs of spurious regression (Ventosa-Santaulària, 2009, page 2). Unreliable and spurious results can lead to poor forecasting and understanding of the data. Forecasts with spurious regression will not in general work well in long-term forecasts, but can still give accurate short-term forecasts.

An important consideration when estimating an ARMA model, is that all the variables in the model must be stationary before modelling. To choose to estimate the model when the data is non-stationary can give inconsistent parameter estimates, and can again lead to meaningless forecasts. Non-stationary time series therefore needs to be transformed into stationary data by differencing to obtain a reliable result in long-term forecasting.

The differenced series equals the change between the observations in the original time series. By computing the differences between the observations, the mean of a time series can be stabilized by removing the changes in the level of the time series. This stabilization can reduce or remove seasonality and trend, and therefore help to obtain stationary time series (Brockwell & Davis, 1991, page 19).

2.4.1. Random walk

The random walk is a simple non-stationary time series. A random walk has the ability to fit a wide range of data despite its simplicity.

A random walk with no constant is a time series y_t where

$$y_t = y_{t-1} + \varepsilon_t \quad (2.2)$$

y_t is the production at present time

y_{t-1} is the production at a previous time

And ε_t is a white noise series where all values are independent and identically distributed (IID) with a zero mean. It is assumed that ε_t is a Gaussian white noise, which implies $\varepsilon_t \sim N(0, q)$.

A random walk model with a drift is characterized by an upward or downward trend and unforeseen changes in direction. A random walk with drift can be represented as:

$$y_t = a + y_{t-1} + \varepsilon_t \quad (2.3)$$

Where $\alpha = E(y_t - y_{t-1})$ Represents the time trend of the production, y_t is the drift of the model (Iordanova, 2020).

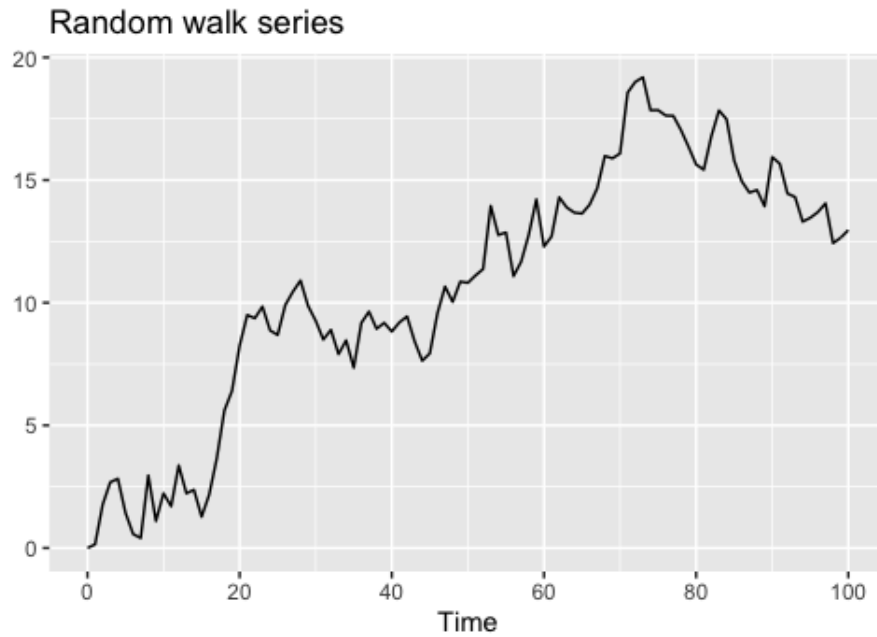


Figure 2.3: Time plot of random walk series. The Plot illustrates a predictable pattern of trend in the series. The plot is constructed in RStudio (Appendix 2.C).

For random walk series it is expected that one or more large spikes are outside the significance limits in the autocorrelation function plot (Iordanova, 2020).

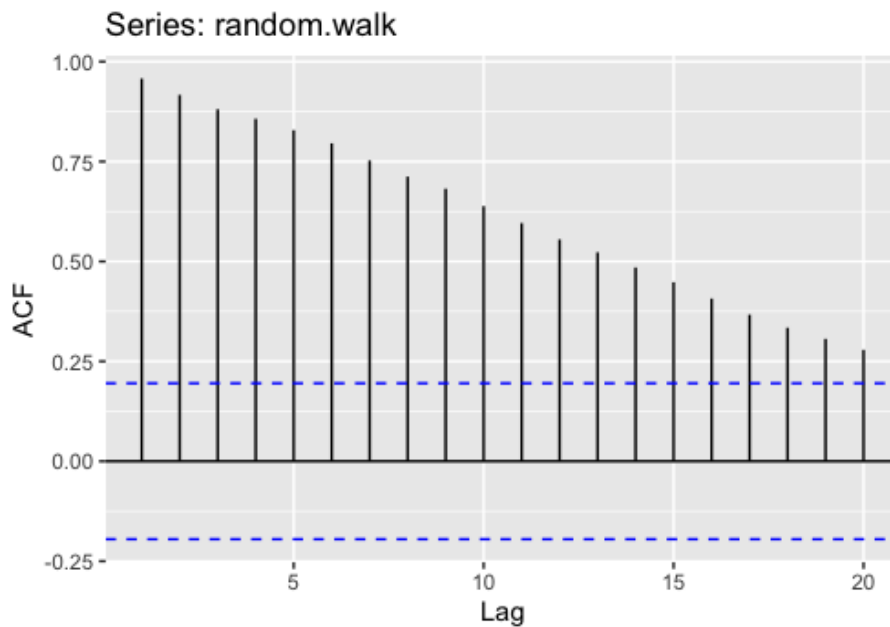


Figure 2.4: Autocorrelation function plot for white noise series. All spikes lie outside the significance bounds, implying autocorrelation between the observations. The plot is constructed in RStudio (Appendix 2.D).

All the autocorrelation coefficients lie outside the blue dotted lines, which confirms that the data in this example are not stationary. A random walk can therefore be considered as a special type of an ARIMA model where both p and q equals zero, and d equals one. This forms a ARIMA(0,1,0) model.

2.5. ARIMA modelling

ARIMA modelling is a well-known and much used tool for forecasting time series. This section will investigate different variations from the class of ARIMA models. ARIMA is an acronym for Autoregressive Integrated Moving Average. The model includes a number of autoregressive (AR) terms p , a degree of differencing d that removes non-stationarity (d), and a number of moving-average (MA) terms q . The model can therefore be specified as an ARIMA(p,d,q) model (Hyndman & Athanasopoulos, 2018, chapter 8.1).

2.5.1. Autoregressive processes AR(p)

A time series is autoregressive when the present values in the series can be obtained by using previous values (Salvi, 2019). An autoregressive model (AR) can be considered as a special case of an ARIMA model, with the parameters q and d equals zero; ARIMA($p,0,0$). The equation (2.4) illustrated an AR(p) process, where p is the number of autoregressive terms:

$$y_t = \varphi_1 y_{t-1} + \dots + \varphi_p y_{t-p} + \varepsilon_t \quad (2.4)$$

For simplicity, p is given the value one:

$$y_t = \varphi_1 y_{t-1} + \varepsilon_t \quad (2.5)$$

Equation 2.5 represents the first order AR process, also called AR(1), and implies that the variable y only depends on its past values. This implies that the variable is explained by its own previous lag.

The φ is the coefficient of interest when determining stationarity. If φ is lower than one, the shocks in the variable y will be transitory and therefore disappear over time. A coefficient with a value lower than one therefore implies stationarity for the time series (Özcan & Öztürk,

2019, chapter 8.1). ε_t is a purely random process with mean zero and variance σ_ε^2 (Chatfield, 2004, page 65).

When the coefficient φ equals one, shocks in y are permanent and will not disappear over time. This implies presence of a unit root in the series, also called non-stationarity. When the coefficient φ takes a value higher than one, the series will explode (Özcan & Öztürk, 2019, chapter 8.1).

2.5.2. Moving average processes MA(q)

A moving average process defines the present value of a time series as a linear combination of past errors. The MA(q) model can be considered as an ARIMA model where p and d equals zero; ARIMA(0,0,q):

$$y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} \quad (2.6)$$

Where ε_t is a purely random process with mean zero and variance σ_ε^2 , also called white noise. θ_i are a constant and μ is the mean. q is the number of moving-average terms.

when $q = 1$, the first-order moving average process MA(1):

$$y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} \quad (2.7)$$

Where θ_i and μ could be any real number.

The phrase “moving average” comes from the fact that y_t is constructed from the weighted sum of the two latest values of ε (Hamilton, 1994, page 48).

2.5.3. Autoregressive moving average models ARMA(p,q)

By combining MA and AR processes, a useful class of time series models can be formed. A mixed autoregressive/moving-average process with p AR terms and q MA terms is called an ARMA process of order (p,q), and is given by:

$$y_t = \varphi_1 y_{t-1} + \dots + \varphi_p y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} \quad (2.8)$$

The importance of ARMA processes lies in the fact that stationary time series can be generally only approximated in a proper way in an ARMA process with fewer parameters involved than a pure AR or MA process. This is an example of the Principle of Parsimony, which implies that it is wished to find a model that gives an adequate representation of the data with as few variables as possible (Brockwell & Davis, 1991, page 320).

2.5.4. Integrated ARMA models ARIMA(p,d,q)

Most time series are non-stationary in practice. In order to fit a stationary model, it is necessary to remove non-stationary sources of variation. Differencing is used to make non-stationary time series stationary.

The differenced series equals the change between the observations in the original series. When the time-series are non-stationary in the mean, we can difference the series to generate stationarity by eliminating trend and seasonal variation (Brockwell & Davis, 1991, page 47). The use of differencing can therefore extend the ARMA(p,q) model into the ARIMA(p,d,q) to handle non-stationary time series.

The first difference operator ∇^1 is defined by:

$$\nabla^1 y_t = y_t - y_{t-1} = (1 - B)y_t \quad (2.9)$$

Where B is the backward shift operator, also called lag-operator.

$$By_t = y_{t-1} \quad (2.10)$$

The full ARIMA(p,d,q) model is given by:

$$\nabla^d y_t = \varphi_1 \nabla^d y_{t-1} + \dots + \varphi_p \nabla^d y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} \quad (2.11)$$

Where d is the number of first-differencing necessary to obtain stationarity in the model (Chatfield, 2004, page 48).

2.5.5. Seasonal ARIMA models

The representation of ARIMA models have been restricted to non-seasonal models until now. ARIMA models also have the possibility to handle seasonal series. Seasonal series are characterized by a fluctuation pattern in the autocorrelation function. Instead of assuming that oscillations are due to equal cycles in the time series, the seasonal ARIMA allows for randomness in the seasonal pattern form one cycle to the next (Brockwell & Davis, 1991, page 320).

A seasonal ARIMA model is formed by including a seasonal term, $(P,D,Q)_m$, in addition to the $ARIMA(p,d,q)$ model already represented. The full seasonal model can be represented as:

$ARIMA(p,d,q)(P,D,Q)_m$.

Where m is the number of observations per year, which is months for the data of interest in this thesis, and m is therefore given the value 12. The uppercase P , D and Q refer to the autoregressive, differencing and moving average terms for the seasonal part of the ARIMA model.

The seasonal components of this model are similar to the non-seasonal part of the model, but also includes backshifts of the seasonal period (Hyndman & Athanasopoulos, 2018, chapter 8.9). The $ARIMA(3,1,2)(3,1,2)_{12}$ model can be written:

$$(1 - \varphi_1 B) (1 - \varphi_1 B^{12})(1 - B)(1 - B^{12})y_t = (1 + \theta_1 B) (1 + \theta_1 B^{12})\varepsilon_t \quad (2.12)$$

2.6. Measuring forecast accuracy

It is important to evaluate forecast accuracy to obtain an estimation of how believable a forecast for the future could be. Even if an accurate forecast on known data points do not guarantee the same accuracy on forecasts for the future, calculating the level of error in earlier forecasts can help us say something about the error in future forecasts.

When measuring forecast accuracy, it is common practice to separate the time series of interest into a training and test set. The training part of the series is used to estimate parameters and residuals of forecasting methods (AIC), and the test part of the data is used

to evaluate the forecast accuracy. The first part of the time series, the training part, will be used to train the forecasting methods to forecast the second part of the time series, the test part.

The accuracy of a forecast can only be determined by considering how well a model performs on new data that were not used when fitting the model. The measure AIC will therefore not be sufficient to determine accuracy. AIC can only be used to compare models from the same class. In this thesis it will be compared models from different classes, and other measures will therefore be calculated (Hyndman & Athanasopoulos, 2018, chapter 9.4).

The measures mean absolute error (MAE) and mean squared error (MSE) will be used to compare forecast accuracy in this thesis. A lower error implies a more accurate forecast in all the methods presented in this section.

2.6.1. Mean absolute error

The mean absolute error is a measure of errors between predicted values and actual values. This algorithm measures the average of the absolute deviation between these values. Every forecast error will get the same weight in this measure (Kirchgässner & Wolters, 2007, page 86). This method is therefore preferred when overall deviation is wanted to be lowest possible over time, and few big errors are not a major concern for the result.

The mean absolute error can be calculated as:

$$MAE = \frac{1}{T} \sum |\varepsilon_i| \quad (2.13)$$

Where ε_i = true value - predicted value

\sum is the sum of all data points, and T is the number of observations in the series.

2.6.2. Mean squared error

MSE is short for mean squared error and measures the average of the squared differences between the predicted values and true values (Kirchgässner & Wolters, 2007, page 86).

The mean squared error can be calculated as:

$$MSE = \frac{1}{T} \sum (\varepsilon_i)^2 \quad (2.14)$$

Due to the fact that the errors in MSE are squared, this measure will give large errors a stronger weight than in MAE. This measure is therefore preferred when big errors are a major concern for affecting the result of the forecast (Kirchgässner & Wolters, 2007, page 86).

3. Model selection in ARIMA modelling

This section will focus on selecting the appropriate order of the ARIMA model. AIC is an acronym for the Akaike Information Criterion, and is a mathematical method to evaluate how well a model fits the data it was generated from. AIC is used to compare different models and determine which model that is the best fit for the data, by determining the relative information value of the model using the maximum likelihood estimate and the number of parameters in the model. The forecast model that performs the best usually has the lowest AIC (Bevans, 2020). AIC can be calculated as:

$$AIC = 2K - 2\ln(L) \quad (3.1)$$

Where K is the number of parameters, and L is the log-likelihood estimate. The log-likelihood is the logarithm of the probability that the observed data are coming from the estimated model (Bevans, 2020).

Unlike the accuracy measures MAE and MSE in section 2.6, AIC does only measure how well the model residuals fits the training data, and do not compare forecasted values with the test part of the time series to find accuracy. When comparing different ARIMA models in this section, AIC will be used as a measurement because it is less time consuming than calculating the accuracy measures MAE and MSE.

Various cases of ARIMA models will be fitted in this section to investigate which ARIMA model fits the data most appropriately according to AIC values. To investigate if the time

series needs differencing to obtain stationarity, the series will be investigated by decomposition and unit root tests in the start of this section.

3.1. Decomposition

To investigate if the time series of interest has any clear trend, seasonality, or cyclic behaviour, the time series is decomposed with the function `decompose()`. Decomposing is to separate elements of the time series into simpler compounds. Extracting these components from a time series is done to help improve understanding of the time series and improve forecast accuracy (Brockwell & Davis, 1991, page 14). After splitting the time series in parts illustrating different pattern categories, it will be easier to look for trend, seasonality and cycles.

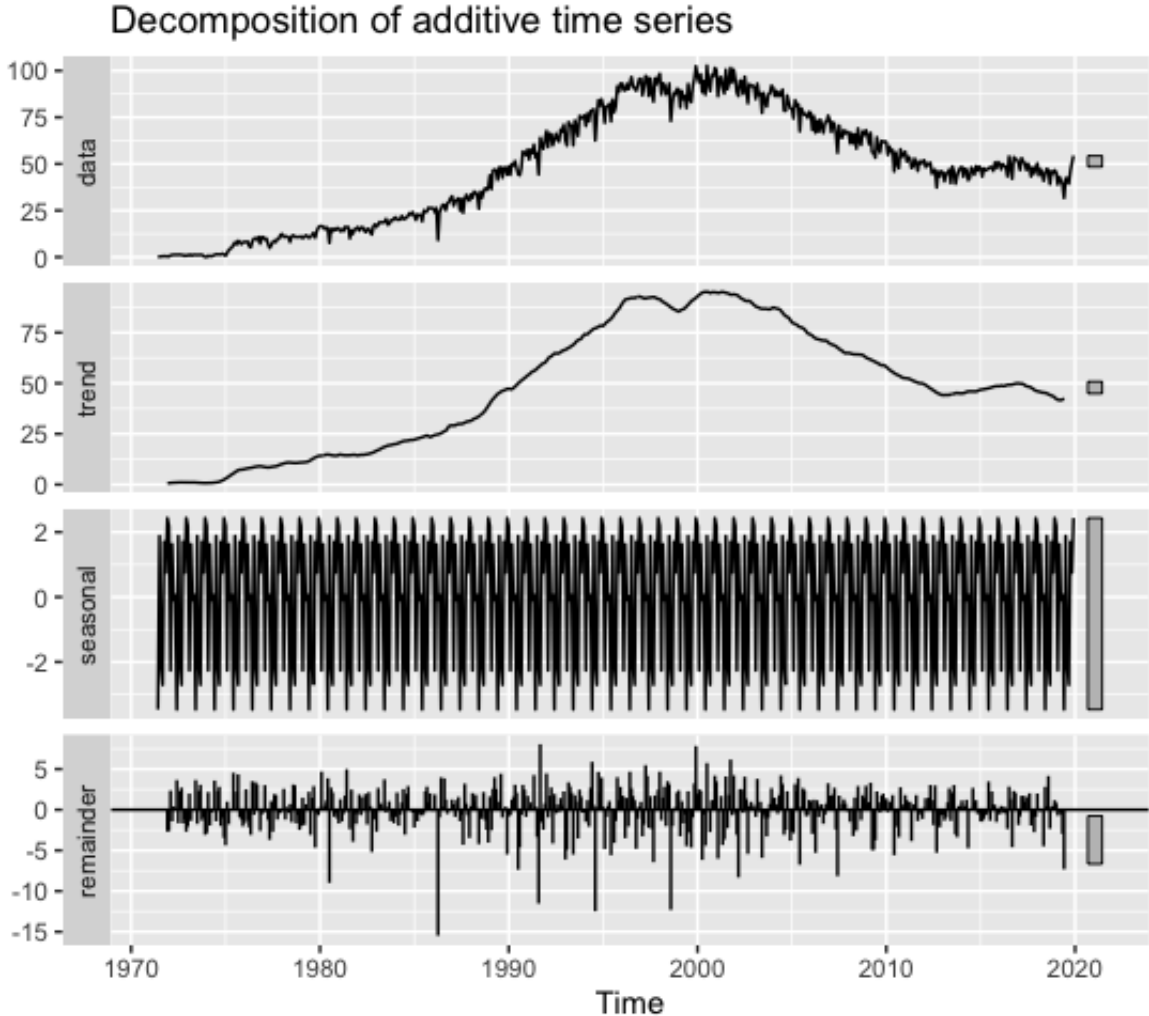


Figure 3.1: The decomposed time series. The plot illustrates trend and seasonal variation. The series was decomposed with the function `decompose()` and plotted in RStudio (Appendix 3.A)

Figure 3.1 illustrates that there is a trend in the time series. The figure also contains variation in the seasonal row of the plot. This variation can be by cycles, and not seasonality that makes data non-stationary. The figure above does not provide enough information to conclude with the number of differencing that is required to make the data stationary. A test for stationarity will therefore be proceeded with before concluding with appropriate number of differencing.

3.2. Selecting the value of d

The value of d in the $ARIMA(p,d,q)$ model should be equal to the number of differencing required to make the time-series stationary. Unit roots can cause unpredictable results in a time series analysis. Unit root tests will therefore be computed to determine if differencing is required to obtain stationarity in the time series. If a time-series contains one unit root, $I(1)$, the series needs to be differenced once to obtain stationarity. If the series contains two unit roots, $I(2)$, the series would need to be differenced twice to obtain stationarity (Özcan & Öztürk, 2019, chapter 8.1).

Different tests can measure the properties of a time-series differently, and it can therefore be useful to use several tests to confirm the results. The augmented Dickey-Fuller (ADF) test and the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test will be computed to determine the appropriate number of differencing required to obtain stationarity in the time series.

Typically for most unit root tests, both the ADF and the KPSS test has a high type I error, which means to reject the null hypothesis when it is actually true. A method to correct with potential high Type I error is to use both the KPSS test and the ADF test to test for stationarity. If the result from both tests suggests that the time series is stationary, it probably is (Wang, 2006, page 33).

3.2.1. Dickey-Fuller test

The Dickey-Fuller test was developed and popularized by Dickey and Fuller in 1979 (Dickey & Fuller, 1979). The null hypothesis of a Dickey Fuller test is that there is a unit root in the autoregressive model, which implies that the data series is non-stationary. The general alternative hypothesis is that the data is stationarity or trend stationarity, depending on what type of test being used (Özcan & Öztürk, 2019, chapter 8.1).

Consider the time series equation below to understand a Dickey Fuller test:

$$y_t = \varphi_1 y_{t-1} + \varepsilon_t \quad (3.2)$$

This equation represents the first order autoregressive process, also called AR(1), and implies that the variable y only depends on its past values. This means that the variable is explained by its own previous lag.

The φ is the coefficient of interest when determining stationarity. If φ is lower than one, the shocks in the variable y will be transitory and therefore disappear over time. This implies stationarity for the time series. When the coefficient φ equals one, shocks in y are permanent and will not disappear over time. This implies presence of a unit root in the series, and therefore not stationarity (Özcan & Öztürk, 2019, chapter 8.1).

The general Dickey-Fuller (DF) test is a simple approach to test for a unit root, but cannot capture the complicated structure times series often have in practice. This is due to the estimate of φ can act inappropriate under the null hypothesis that $\varphi = 1$ because of non-stationarity. An augmented Dickey-Fuller (ADF) test is therefore preferred for the time series in this thesis (StatsDirect, 2020).

3.2.2. Augmented Dickey-Fuller test

The ADF test is a more powerful unit root test that can handle more complex models than the Dickey-Fuller test. Similar to the original Dickey-Fuller test, an augmented Dickey-Fuller test has a null hypothesis that the series contains a unit root, which implies non-stationarity

The p-value of the hypothesis test is the limit for which significance level the test will have to reject the null hypothesis (H_0). A p-value lower than 0.05 will result in a rejection of the null hypothesis that assumes that the series contains a unit root. For a series to be stationary, the p-value of the test should be lower than 0.05 (Özcan & Öztürk, 2019, chapter 8.1).

The hypotheses of interest in an ADF-test are:

$$H_0 : \varphi = 1 \text{ (unit root)} \Rightarrow y_t \sim I(1)$$

$$H_1 : |\varphi| < 1 \Rightarrow y_t \sim I(0)$$

An augmented Dickey-Fuller test is constructed to test for the presence of a unit root in the time series. The output of the test can be found in Appendix 3.B. It can be seen in Appendix 3.B that the p-value of the test equals 0.99, which is greater than 0.05. The null hypothesis fails to reject at a 95% confidence interval and cannot conclude that $|\varphi| < 1$ (stationarity).

Due to the fact that it cannot be concluded with stationarity from the ADF-test, the time series are first-differenced once with the function `diff()`, shown in Appendix 3.C.

An augmented Dickey-Fuller test on the differenced time series is constructed to investigate if it can be concluded to reject the presence of a unit root after first-differencing the data once. Results of the test can be found in Appendix 3.D. As shown in Appendix 3.D, the p-value in the hypothesis test for the differenced data is 0.01. This value is below 0.05, and the null hypothesis that the data contains a unit root can be rejected with a 95% confidence interval. The augmented Dickey-Fuller rejects the presence of a unit root in the first-differenced time series, which implies stationarity.

3.2.3. KPSS Unit root test

The Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test investigates with linear regression if a time series is stationary around a linear trend or mean, or is non-stationary due to a unit root (Kwiatkowski et al., 1992). Unlike the ADF-test, the KPSS hypothesis test has the null hypothesis that the data is stationary, which implies no need for differencing. The alternative hypothesis is equal to that the data is not stationary (Wang, 2006, page 33).

To be able to confirm that the time series needs one first differencing to obtain stationarity, the null hypothesis of stationarity has to be rejected on the original series, and fail to reject the null hypothesis when unit root testing the differenced series. The null hypothesis will be rejected if the value of the t-statistic is higher than the critical values for the chosen significant level.

The significant level of the statistical test is the probability that the event could have occurred. A low significant level implies a low probability of the event occurring is quite small, the event can be called significant (Riffenburgh, 2012, page 325).

A KPSS unit root test on the original series can be found in Appendix 3.E. As shown in Appendix 3.E, the value of the test-statistic is 4.6988, which is higher than the 1% critical value of the test. This indicates that the null hypothesis is rejected at all significance levels. This proves that the data is not stationary, and that applying differencing on the time series is necessary to remove one or more-unit roots. One first-differencing is therefore applied to the time series.

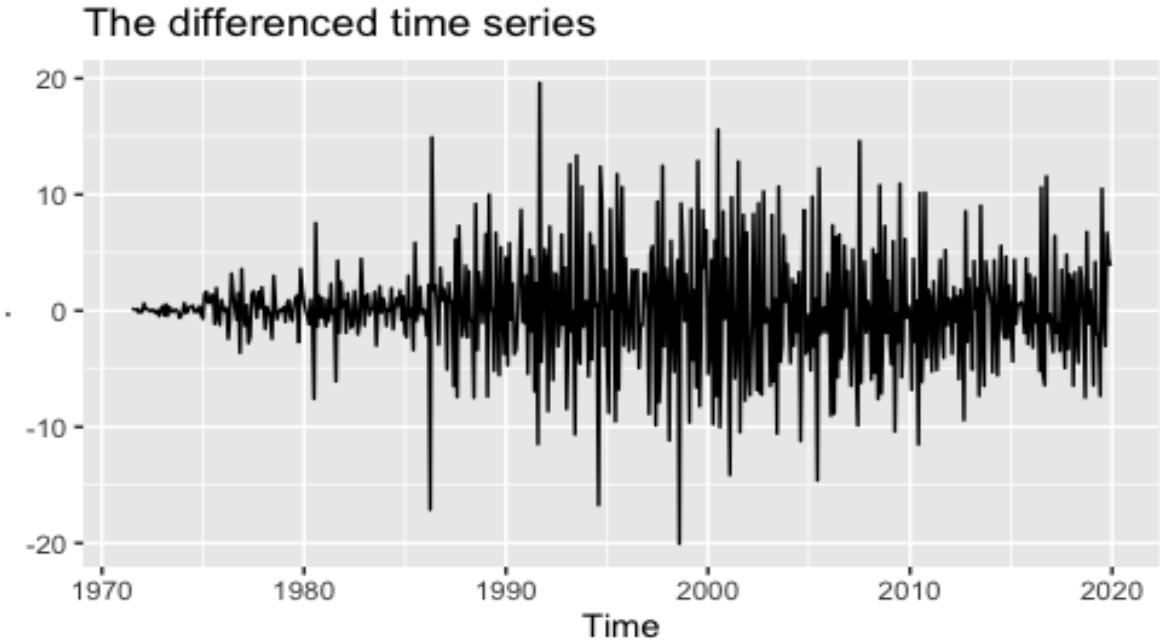


Figure 3.2: The differenced time series. The plot illustrates the difference between the observations in the original time series, and are therefore negative due to a decrease between observations. The series was differenced with the function `diff()` and plotted in RStudio (Appendix 3.C)

After the time series is differenced, several of the values of oil production in Figure 3.2 are negative. This is because the values of the series are now the difference between the observations in the original series, and can therefore be negative from a decrease between the observations.

A KPSS unit root test is constructed on the differenced time series in Appendix 3.F, that illustrates the value of the test-statistic for the differenced data is 0.3347. This is lower than all critical values listed, and therefore indicates failing to reject the null hypothesis that the data is stationary. It is therefore assumed that the first differenced data is stationary when using a KPSS unit root test.

As an alternative to trying different differences and tests, the functions `ndiffs()` and `nsdiffs()` can be used to determine the number of differences needed to obtain stationarity in the data. This automates the searching of the number of differences needed. The function `ndiffs()` determines the appropriate number of first differences that is necessary to make the time series stationary, and the function `nsdiffs()` determines the appropriate number of seasonal differencing. First-differencing removes linear trend in a time-series, and seasonal differencing removes seasonal variation (Holmes et al., 2021, chapter 5.5).

The automated testing of appropriate number of first differences is represented in Appendix 3.G, and the given output “1” implies that one first-differencing is needed to obtain stationarity in the time series. A test for appropriate number of seasonal differencing required to obtain stationarity can also be found in Appendix 3.G. The given output “0” from the test implies that no seasonal differencing is needed to obtain stationarity in the time series.

To test if any additional differencing is necessary, the functions `ndiffs()` and `nsdiffs()` was used on the differenced time series in Appendix 3.H. The received output “0” of both procedures confirms that no first differencing or seasonal differencing is needed to make the differenced time series stationary, and that one first-differencing is sufficient to obtain stationarity for this series.

One first differencing is applied on the time series with the function `diff()`, and the value of d in the ARIMA model is therefore one, forming a $ARIMA(p,1,q)$ model.

3.2.4. Selecting the value of D

When fitting a seasonal $ARIMA(p,d,q)(P,D,Q)_{12}$ model to the data, the value of the seasonal differencing D also needs to be interpreted. The value of D should be equal to number of seasonal differences needed to obtain stationarity in the time series.

A test for appropriate number of seasonal differences required to obtain stationarity can be found in Appendix 3.G. The given output “0” from the test implies that no seasonal differencing is needed to obtain stationarity in the time series. D is given the value zero, forming a $ARIMA(p,1,q)(P,0,Q)_{12}$ model.

3.3. Inspecting the autocorrelation function

A correlogram can be helpful to identify suitable classes of models for the time series, especially for selecting the most appropriate type of an autoregressive integrated moving average model. The use of the ACF and PACF plots can help finding the order of the autoregressive (AR) and moving average (MA) in several time series.

The ACF plot is an autocorrelation function illustrating how well the present value of the series is related with its past values. The autocorrelation function will therefore investigate for autocorrelation between present production and the production at different times. PACF is short for the partial autocorrelation function, and illustrates the correlation between the residuals (Salvi, 2019).

3.3.1. Determining the order of a MA process

If a moving average process is considered to be appropriate for the time series of interest, the autocorrelation function could be able to illustrate the value of q . The theoretical autocorrelation function cuts off at lag q for a $MA(q)$ process.

To look for the lag beyond the first spike that are close to zero is a common strategy to determine the order of an moving average process. The partial autocorrelation function (PACF) is usually less helpful for identifying MA models because of the partial functions attenuated from, which means that the lags would slowly die out instead of cut off quickly. This makes it harder to interpret where the function tends to zero (Chatfield, 2004, page 62).

3.3.2. Determining the order of an AR process

It can be tricky to determine the order of an AR process from the sample autocorrelation function. For a first-order autoregressive process, $AR(1)$, the theoretical autocorrelation function decreases exponentially with a sample function of similar shape. Processes of higher orders can have an autocorrelation function that contains sinusoidal or damped exponential functions that will make the order hard to identify (Chatfield, 2004, page 62). The spikes in an autocorrelation function tends slower to zero for an AR or ARMA than a MA process, due to the fact that the ACF for these processes will rather attenuate than cut off.

The partial autocorrelation function can therefore be useful to determine the order of the AR process. A partial autocorrelation function of a moving average process often cuts off at lag p , so the value of p is devoted to the spike beyond the first lag that are not significantly different from zero (Chatfield, 2004, page 62). Figure 3.3 illustrates the autocorrelation function for the series. Spikes outside the dotted blue lines are significantly different from zero.

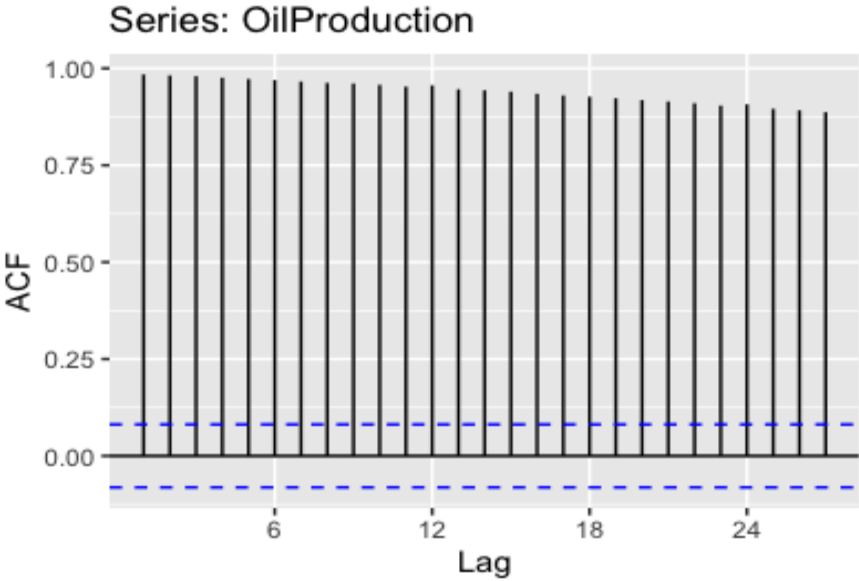


Figure 3.3: Autocorrelation function for the original time series. The plot illustrates significant correlation between the lags. The plot is constructed in RStudio (Appendix 3.1).

The autocorrelation function in Figure 3.3 does not decay quickly along the lags, and show significant correlation between the lags. This is typical for times series that contains trend. An autocorrelation function where the lags does not tend to zero quickly indicates non-stationarity and implies a need for differencing. The ACF-plot for the non-differenced series in Figure 3.3 therefore confirms non-stationarity in the series.

The sample autocorrelation function is only meaningful for data from a stationary time series model. Any trend should therefore be removed before observing the ACF plot (Chatfield, 2004, page 27). The autocorrelation for the first differenced series will therefore be investigated to say something about the appropriate parameters for the autoregressive and moving average processes. Figure 3.4 illustrates the autocorrelation function, the partial autocorrelation function and the time plot for the differenced series.

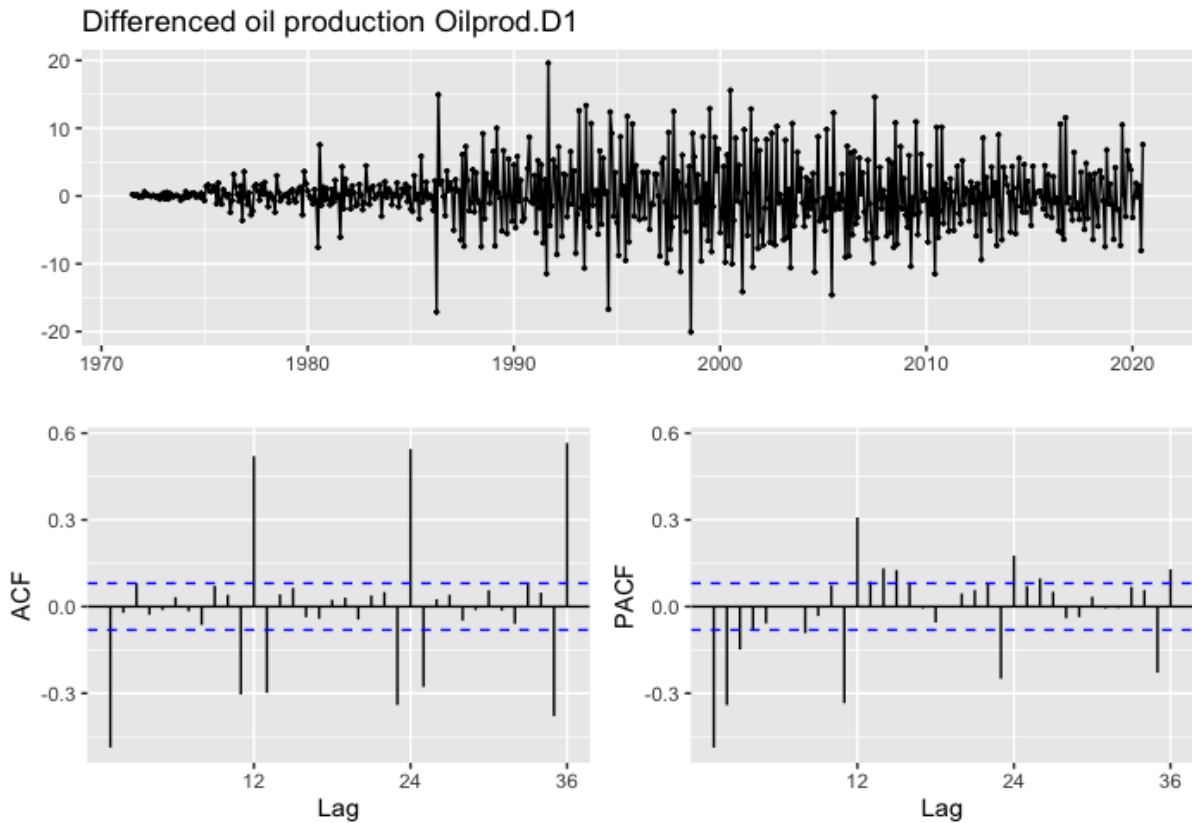


Figure 3.4: The time plot, autocorrelation function (ACF) and the partial autocorrelation function (PACF) for the differenced time series. There are less significant lags for the differenced series than the original series. The plot is constructed in RStudio (Appendix 3.I).

As illustrated in Figure 3.4, the series still include significant spikes in the ACF-plot after differencing. This can make the process of selecting the parameters p and q difficult. A known procedure to determine the parameters for the autoregressive and Moving average processes is by investigating the ACF- and PACF-plots for the processes. Considerable experience is required to interpret autocorrelation coefficients. This thesis will therefore focus on comparing residuals from the different forecast models to determine the appropriate model for time series of interest. ARIMA models with different p and q values will be fitted and compared, and the model with the lowest AIC will be chosen as the optimal ARIMA model.

3.3.3. Maximum likelihood estimation

After the values of p, d and q are determined, the parameters φ and θ needs to be estimated. When the program R estimates the ARIMA model, maximum likelihood estimation (MLE) is applied. The technique finds the parameter values that maximizes the probability of obtaining

the data observed. Maximum likelihood estimation is similar to the least squares estimates for ARIMA models, which minimizes the squared sum of errors $\sum \varepsilon_i^2$ (Hyndman, & Athanasopoulos, 2018, chapter 8.6).

For simple time series regression models, MLE and least square estimation often gives the same parameter estimates. ARIMA models are more complicated to estimate than regressions, and different techniques can therefore give different parameter estimates for these models. When maximum likelihood estimation is used in R, the value of the log likelihood of the data is reported, which is the logarithm of the probability that the observed data is obtained from the estimated model. R attempts to maximize the log likelihood when determining parameter estimates for the given values of p,d,q.

3.3.4. Seasonal differenced series

If a time series have seasonal variation, the autocorrelation function will also contain oscillations that can disturb the result. For series with monthly variation, it can be expected that lag number six is large and negative, and lag number 12 is large and positive in the ACF plot. If the lags in the ACF-plot follows a sinusoidal pattern, it is generally assumed that there is seasonal variation in the data (Chatfield, 2004, page 27).

If seasonal variation is removed from seasonal data, useful information can be provided by the autocorrelation function. It occurs to be oscillations that can remind of a sinusoidal pattern in Figure 3.4. Seasonal variation is therefore removed by seasonal differencing, and the autocorrelation plot is illustrated in Figure 3.5.

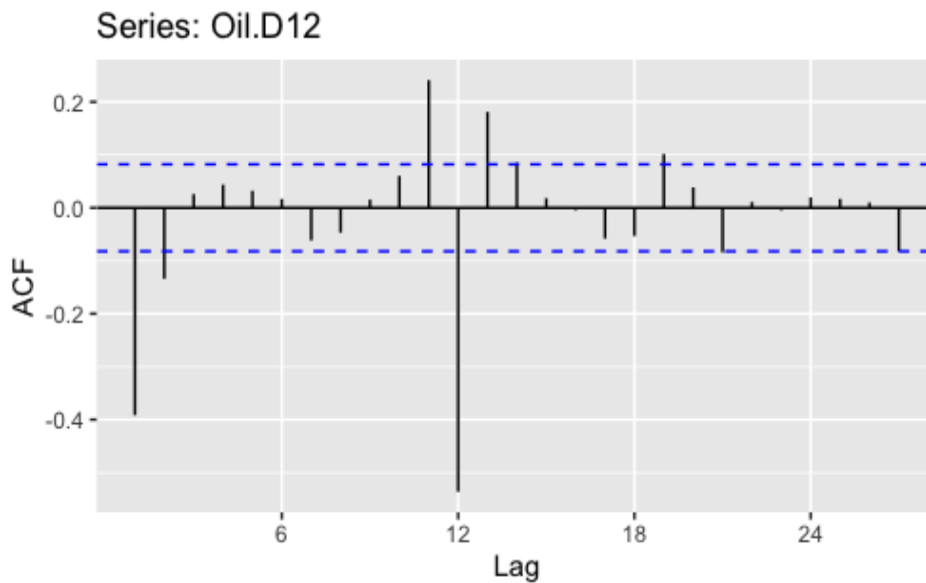


Figure 3.5: The autocorrelation function for the seasonal differenced series illustrates less significant spikes than the earlier autocorrelation plots. The plot is constructed in RStudio (Appendix 3.I).

After seasonal differencing, there are still a presence of significant spikes in the series. This autocorrelation tends to zero after lag 12, which should lead to suspicion of seasonal variation in the series. An ARIMA model that includes a seasonal term can therefore seem to be more appropriate for these time series than a non-seasonal model.

3.4. The automated ARIMA model

Determining the values for p , d and q is not straight-forward, and the appropriate order of processes will vary from time series to time series. The function `auto.arima()` can simplify the process by selecting these values for us. The function fits the best ARIMA model for the time series model by conducting a search over possible models within the constraints provided (Hyndman & Khandakar, 2008, page 35).

3.4.1. The non-seasonal automated ARIMA

An adjustment of the `auto.arima()` function that restricts the search of models to only allow non-seasonal alternatives is applied in the first automated forecast. The forecast for 2018 with the suggested non-seasonal model is illustrated in Figure 3.6. Observations before 2014 are cut from the plot to focus on the forecasted values. A plot containing all the observations in the time series with the forecasted values can be found in Appendix 3.J.

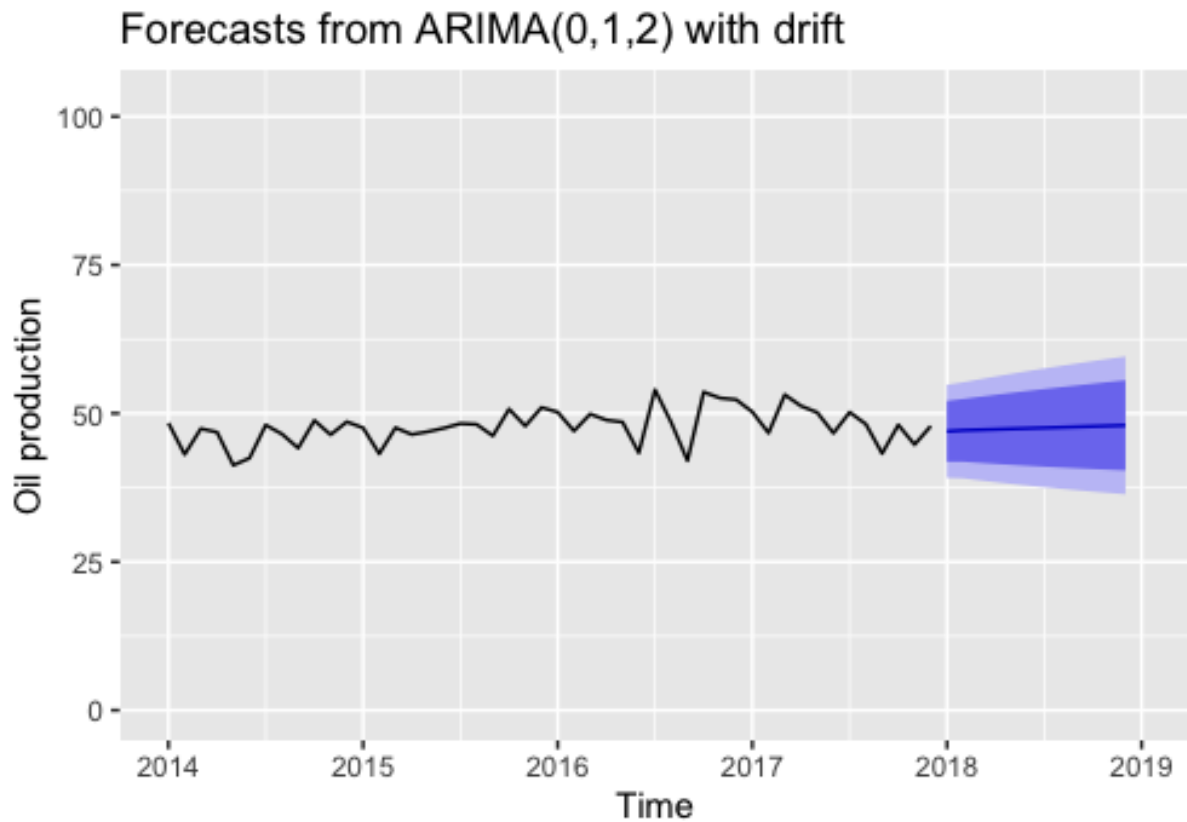


Figure 3.6: Forecast of the automated non-seasonal ARIMA model. Due to the seasonal restriction in the function, no seasonal variation can be seen in the forecast. The plot is constructed in RStudio (Appendix 3.J).

When a forecast is obtained, the middle of the range of possible values the future production could take with a given probability are estimated, called a prediction interval. If a 95% prediction interval is chosen, a random observation from the future will be within the prediction interval 95 of 100 times (Hyndman, & Athanasopoulos, 2018, chapter 3.1).

The light blue shaded area in Figure 3.6 represents the 95% interval for monthly Norwegian oil production, and the darker blue shaded area represents the 80% interval. The blue line is the average of the possible future values, which is called the point forecasts.

3.4.2. The seasonal automated ARIMA

Allowing a seasonal term to be included when fitting the model makes the possibility to adjust for seasonal variations. Including a seasonal term can therefore give more accurate

predictions if the model includes a seasonal term. Figure 3.7 illustrates the automated seasonal ARIMA(p,d,q)(P,D,Q)₁₂ model.

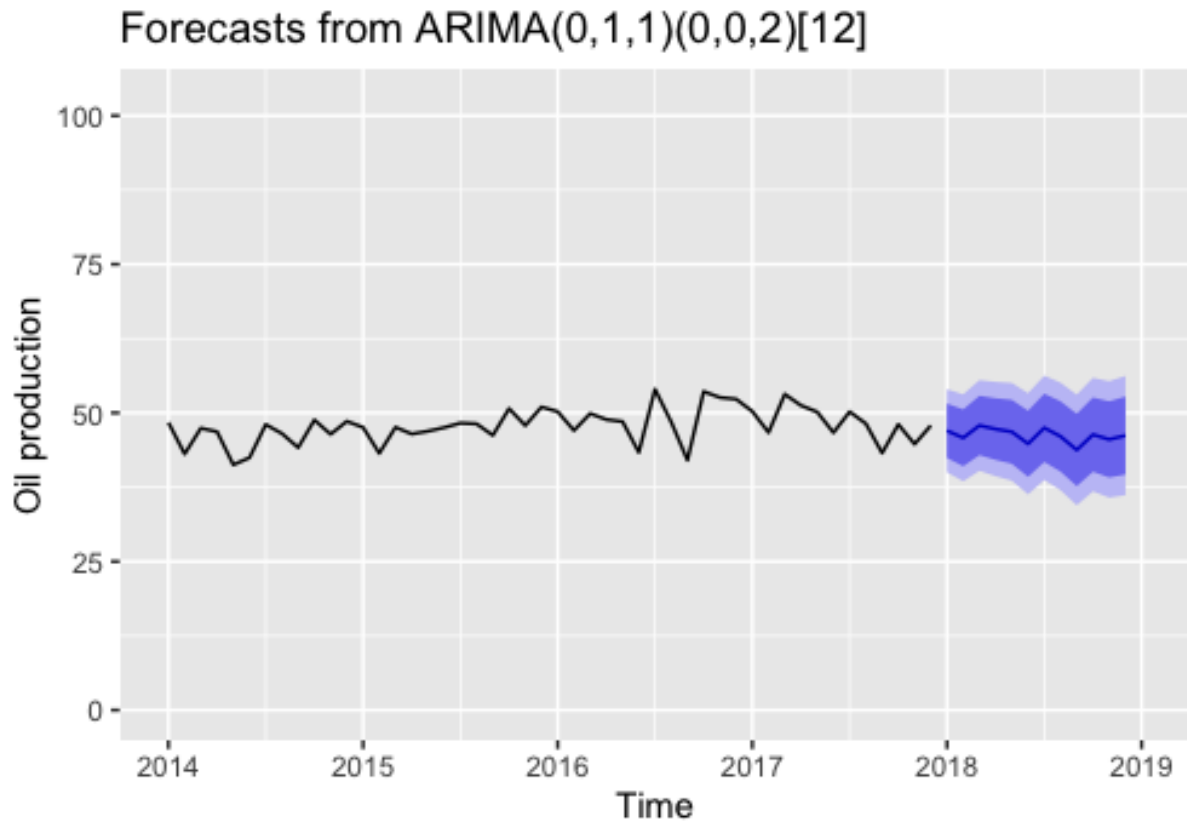


Figure 3.7: Forecast of the automated seasonal ARIMA model. Seasonal variation is allowed this function, which explains the seasonal variation in the forecast. The plot is constructed in RStudio (Appendix 3.K).

3.4.3. Selecting the appropriate ARIMA model manually

The function `auto.arima()` can be useful, but an automated model can also be unreliable in few cases. Several ARIMA models with different p,d and q values are therefore fitted to investigate if any manual fitted ARIMA models can perform better on the data than the automated models for this time series.

The automated models constructed with the `auto.arima()` function is illustrated in the top rows of Table 3.1. To investigate if any models could give a lower AIC value than the Auto ARIMA forecasts, several ARIMA models are fitted and compared with the automated models in Table 3.1.

Table 3.1: Comparing AIC between the ARIMA models to find the most appropriate ARIMA for this series. The values are found with RStudio. See example of AIC estimation in Appendix 3.L.

Model	AIC
(Auto) ARIMA(0,1,2) with drift	3130.19
(Auto) ARIMA(0,1,1)(0,0,2)[12]	3002.52
ARIMA(0,0,0)	5409.7
ARIMA(0,1,0)	3369.61
ARIMA(1,0,0)	3379.72
ARIMA(2,0,0)	3228.2
ARIMA(0,0,1)	4811.29
ARIMA(0,0,2)	4398.14
ARIMA(1,1,0)	3215.82
ARIMA(2,1,1)	3130.05
ARIMA(3,1,2)	3133.72
ARIMA(3,1,3)	3081.98
ARIMA(3,1,12)	3059.5
ARIMA(1,1,2)(0,0,2)[12]	3219.27
ARIMA(0,1,1)(0,0,1)[12]	3045.21
ARIMA(0,1,0)(0,0,2)[12]	3219.27
ARIMA(1,1,0)(0,0,2)[12] with drift	3097.05

As illustrated in Table 3.1, the seasonal automated $ARIMA(0,1,1)(0,0,2)_{12}$ has a lower AIC than the non-seasonal automated $ARIMA(0,1,2)$. This implies that the series can contain seasonal variation, and that including a seasonal term in the model can help obtain more accurate predictions for the future.

The seasonal automated $ARIMA(0,1,1)(0,0,2)_{12}$ has the lowest AIC of all ARIMA models fitted in Table 3.1, and it is therefore concluded to be the best fitted ARIMA model for this time series. This model is selected as the optimal ARIMA model for this series, and will be compared in accuracy with the best benchmark forecasting method and the prediction of the Norwegian petroleum directorate in section 4.

3.5. Simple forecast methods

Certain forecasting methods are surprisingly effective for their simplicity. The following simple forecasting methods will be used as a benchmark through this thesis. More advanced forecast methods will be compared with these simple methods to ensure that the new methods are achieving a higher accuracy than the accuracy achieved with the simpler models. If any advanced forecast methods have a higher error than the simple methods, they will not be considered.

3.5.1. Mean method

In the mean method, the sum of forecasted future values equals the average of the historical data (Hyndman, & Athanasopoulos, 2018, chapter 3.1). When the historical data is denoted by $y_1, y_2, y_3 \dots y_T$, the forecast can be presented as:

$$\hat{y}_{T+h|T} = \bar{y} = (y_1, y_2, y_3 \dots y_T)/T \quad (3.3)$$

Where $\hat{y}_{T+h|T}$ the estimate of is y_{T+h} based on the historical data $y_1, y_2, y_3 \dots y_T$

3.5.2. Naïve method

The naïve forecast method set all forecast values to be the equal to the last observation of the historical data:

$$\hat{y}_{T+h|T} = y_T \quad (3.4)$$

This method often works well on economic and financial time series. A naïve forecast is optimal when the data follow a random walk, and can therefore also be called random walk forecasts (Hyndman, & Athanasopoulos, 2018, chapter 3.1).

3.5.3. Seasonal naïve method

For seasonal data, it can be helpful to set each future forecast value equal to the same month of the last year:

$$\hat{y}_{T+h|T} = y_{T+h-m(k+1)} \quad (3.5)$$

Where m is the month and k is the integer part of $(h-1)/m$, which is the number of complete years in the forecast horizon before the date $T+h$. This implies that each forecasted future August is equal to the latest August in the historical data (Hyndman & Athanasopoulos, 2018, chapter 3.1).

3.5.4. Drift method

Another variation of the random walk is to allow the forecast to increase or decrease over time, where the change over time is set to be the average change in the historical data. This change is called the drift.

This method is identical to drawing a line between the first and last observations in the historical data and continuing the line into the future (Hyndman & Athanasopoulos, 2018, chapter 3.1). For a naïve method with drift, the best forecast of tomorrow's production is today's production plus a drift term:

$$\hat{y}_{T+h|T} = y_T + \frac{h}{T-1} \sum_{t=2}^T (y_t - y_{t-1}) = y_T + h \left(\frac{y_T - y_1}{T-1} \right) \quad (3.6)$$

3.6. Comparing the simple forecasts methods

Figure 3.8 illustrates the different simple forecast methods with a forecast horizon of 12 months. Earlier observations than the year 2016 are not illustrated in Figure 3.8 to focus on the representation of the forecasted data points.

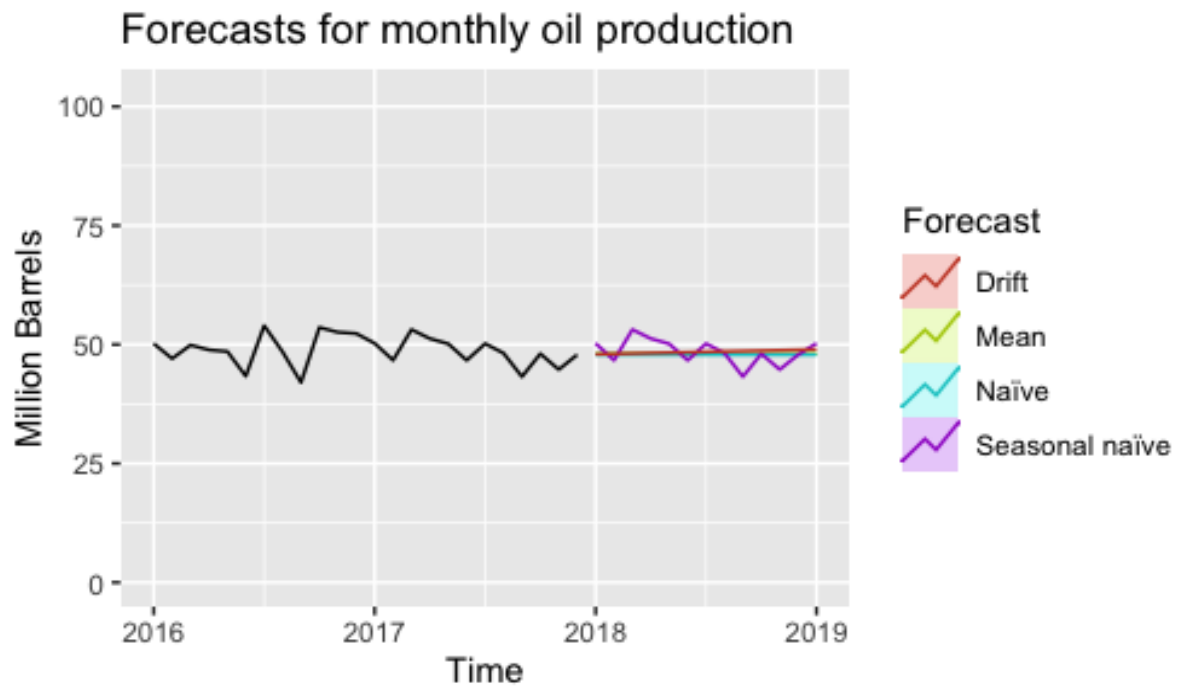


Figure 3.8: Forecasts for every month in the year 2018 by the simple forecast methods. The plot is constructed in RStudio (Appendix 3.M).

The accuracy of these benchmark forecast methods are compared with the most appropriate ARIMA forecast and the forecast from the Norwegian petroleum directorate in section 4.

4. Comparing forecast accuracy

Table 4.1 gives an overview of the accuracy of different forecasts of every month of the year 2018 constructed in this study, along with the forecast obtained by the Norwegian Petroleum Directorate.

Table 4.1: Comparing the forecast accuracy between the simple forecast methods, the most appropriate ARIMA and the forecast by the NPD for monthly oil production in 2018. See example of MAE and MSE calculation in Appendix 4.A

Method	MAE	MSE
Mean	3.187363	15.90779
Naïve	3.036304	14.91907
Seasonal naïve	3.16667	16.84814
Naïve with drift	3.578016	18.60098
ARIMA(0,1,1)(0,0,2) ₁₂	1.740171	6.341439
Forecast from the NPD	2.035762	6.04833

The forecast from the NPD obtained the lowest mean squared error (MSE) of all the forecasts in the Table 4.1. MSE gives large errors high weight due to the squared errors in the technique. This technique is preferred when it is preferred to deviate from high errors rather than obtaining the forecast that predicts closest possible to the actual production.

The forecast obtained by the ARIMA(0,1,1)(0,0,2)₁₂ model obtained the lowest mean absolute error (MAE) of all the models in Table 4.1. MAE is a preferred technique when the goal of the forecast is to predict values closest possible to the actual production. In this study it is preferred to prioritize a model with overall closest forecasted values to the actual production. The forecast by the ARIMA(0,1,1)(0,0,2)₁₂ model is therefore selected as the most accurate forecast in this study.

The seasonal ARIMA model is selected to predict the year 2019, that is the last year of the time series of interest. Figure 3.9 illustrates the monthly forecast for 2019 with a blue line compared to actual production illustrated with a red line. Earlier observations than year 2016 are not illustrated in Figure 3.9 to focus on the representation of the forecasted data points.

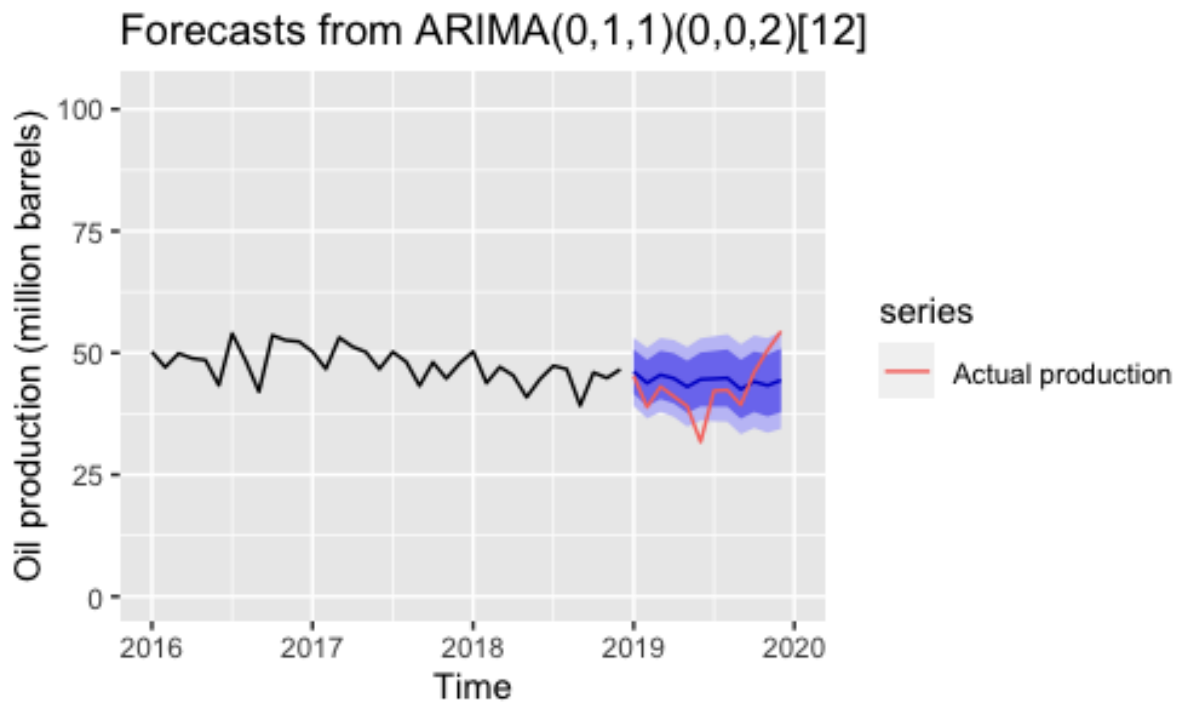


Figure 3.9: Monthly forecast for the year 2019 by the most accurate forecast method in this study, the seasonal ARIMA, compared with actual production for 2019. The figure illustrates that most of the actual production lies within the prediction interval. The plot is constructed in RStudio (Appendix 4.B).

The blue line in Figure 3.9 illustrates the forecast for 2019 by the most accurate model in this study, the seasonal automated ARIMA model. The red line represents actual production for 2019. It can be seen in Figure 3.9 that most of the actual production lies within the prediction interval, represented by the blue shaded areas.

5. Discussion

This thesis focused on time series forecasting based on historical data rather than a deep understanding of the components affecting oil production. Different time series forecasting models were constructed in the software RStudio using programming language R to forecast Norwegian monthly oil production to find the forecasting method that performed the best for the data of interest.

Various cases of ARIMA models were fitted to investigate which ARIMA model fitted the data most appropriately according to the Akaike Information Criterion (AIC).

The most appropriate ARIMA model and the forecast from the Norwegian Petroleum Directorate were compared with simple forecast methods to ensure these advanced methods were achieving a higher accuracy than the accuracy achieved with the simpler models. The measures mean absolute error (MAE) and mean squared error (MSE) were applied to compare accuracy between the different forecasts in this thesis.

The simple forecast methods gave higher MAE and MSE than the most appropriate ARIMA model for the series and the forecast from the NPD, implying that the simple forecast methods were less accurate. The advanced forecast models were therefore compared in accuracy.

The goal for this thesis was to construct a more accurate forecast than the prediction obtained by the Norwegian petroleum directorate. Due to the fact that the forecasts from the NPD and the ARIMA model gave the lowest error with different accuracy measures, the selection of the most accurate forecast was depending on the accuracy measure being used. Overall closest prediction to actual values was preferred rather than deviation from big errors in this study, which implied that the forecast from the ARIMA model was most accurate forecast. The goal to construct a more accurate forecast than the Norwegian Petroleum Directorate was therefore achieved according to the accuracy measure mean absolute error.

It is also worth mentioning that the constructed ARIMA forecast uses less data to predict oil production than the forecast method from the NPD. This implies a higher transparency, which makes the forecast method easier to interpret.

6. References

Bevans, R. (2020, March 26.). An introduction to the Akaike Information Criterion. Retrieved February 3, 2021, from: <https://www.scribbr.com/statistics/akaike-information-criterion/>

Brockwell, P. & Davis, R. (1991) *Time series: Theory and Methods* (Second edition) Springer-Verlag.

Chatfield, C. (2004). *The analysis of Time Series: An introduction* (sixth edition). Chapman & Hall/CRC.

Dickey, D., & Fuller, W. (1979). Distribution of the Estimators for Autoregressive Time Series With a Unit Root. *Journal of the American Statistical Association*, 74(366), 427-431.
doi:10.2307/2286348

Hamilton, J. D. (1994). *Time series analysis*. United Kingdom: Princeton University Press.

Holmes E. E., Scheuerell M. D., and Ward E. J., (2021, January 1.) *Applied Time Series Analysis for Fisheries and Environmental Sciences*. Retrieved march 12, 2021, from: <https://nwfsc-timeseries.github.io/atsa-labs/sec-boxjenkins-non-stationarity.html>

Hyndman, R., & Khandakar, Y. (2008). Automatic Time Series Forecasting: The forecast Package for R. *Journal of Statistical Software*, 27(3), 1 – 22 :<https://dx.doi.org/10.18637/jss.v027.i03>

Hyndman, R.J., & Athanasopoulos, G. (2018) *Forecasting: principles and practice* (Second edition). Otexts: Melbourne, Australia: <https://otexts.com/fpp2/>

Iordanova, T., (2020, 26. April). An Introduction to Stationary and Non-Stationary Processes. Retrieved February 4, 2021, from: <https://www.investopedia.com/articles/trading/07/stationary.asp>

Kirchgässner G. & Wolters J. (2007) *Introduction to Modern Time Series Analysis*. St Gallen/Berlin: Springer

Kwiatkowski, D.; Phillips, P. C. B.; Schmidt, P.; Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics*. 54 (1–3): 159–178. [doi:10.1016/0304-4076\(92\)90104-Y](https://doi.org/10.1016/0304-4076(92)90104-Y)

Norwegian Petroleum (2020, April 24.). Exports of oil and gas. Retrieved February 2, 2021, from: <https://www.norskpetroleum.no/en/production-and-exports/exports-of-oil-and-gas/>

Norwegian Petroleum (2021, February 17.) Historical production: Retrieved March 24, 2021, from: <https://www.norskpetroleum.no/en/facts/historical-production/>

Norwegian Petroleum Directorate (2019, September 15.) Production figures September 2019. Retrieved February 13, 2021, from: <https://www.npd.no/en/facts/news/Production-figures/2019/production-figures-september-2019/>

Özcan, B. & Öztürk, I. (2019) *Environmental Kuznets Curve (EKC): A Manual*. Turkey: Academic press. Retrieved April 3, 2021, from: <https://www.sciencedirect.com/topics/economics-econometrics-and-finance/dickey-fuller-test>

Prediction Intervals: Definition & Examples. (2018, August 10) Retrieved April 27, 2021, from: <https://study.com/academy/lesson/prediction-intervals-definition-examples.html>

R.H. Riffenburgh (2012). *Statistics in Medicine* (Third Edition). Academic Press: <https://doi.org/10.1016/B978-0-12-384864-2.00015-9>.

Salvi J., (2019, March 27.) Significance of ACF and PACF Plots In Time Series Analysis. Retrieved January 30, 2021, from: <https://towardsdatascience.com/significance-of-acf-and-pacf-plots-in-time-series-analysis-2fa11a5d10a8>

Springer (2008) *Time Series*. In: The Concise Encyclopedia of Statistics. Springer, New York, NY. https://doi.org/10.1007/978-0-387-32833-1_401

StatsDirect (2020) P values. Retrieved February 30, 2021, from: https://www.statsdirect.com/help/basics/p_values.htm

Ventosa-Santaulària, D. (2009). Spurious Regression: *Journal of Probability and Statistics*, Retrieved February 30, 2021, from: <https://doi.org/10.1155/2009/802975>

Wang, W. (2006). *Stochasticity, Nonlinearity and Forecasting of Streamflow Processes*. Deft University Press, Amsterdam

7. Appendix

1.A. The time series plot:

```
OilProduction %>%  
  autoplot(ylab="Oil production (million barrels)")
```

2.A. white noise series plot:

```
set.seed(123)  
white.noise<-ts(rnorm(40))  
autoplot(white.noise, ylab = "") + ggtitle("White noise")
```

2.B. Auto correlation plot for white noise:

```
ggAcf(white.noise)
```

2.C. Random walk:

```
random.walk <- arima.sim(model= list(order=c(0,1,0)), n=100)  
autoplot(random.walk, ylab="", main="Random walk series")
```

2.D. Auto correlation plot for Random walk:

```
ggAcf(random.walk)
```

3.A. Decomposed series:

```
decomposed <- decompose(OilProduction)  
autoplot(decomposed)
```

3.B. ADF-test of original series:

```
adf.test(OilProduction)
```

Output from RStudio:

```
Augmented Dickey-Fuller Test

data: OilProduction
Dickey-Fuller = -0.11998, Lag order = 8, p-value = 0.99
alternative hypothesis: stationary

Warning message:
In adf.test(OilProduction) : p-value greater than printed p-value
```

3.C. First-differenced series:

```
Oilprod.D1 <- diff(OilProduction, differences=1)
```

Plot:

```
Oilprod.D1 %>%
  autoplot(main="The differenced time series")
```

3.D. ADF-test of differenced series:

```
adf.test(Oilprod.D1, alternative = "stationary")
```

Output from RStudio:

```
Augmented Dickey-Fuller Test

data: .
Dickey-Fuller = -10.543, Lag order = 8, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test(.) : p-value smaller than printed p-value
```

3.E. KPSS Unit root test for original series :

```
Production %>% ur.kpss() %>% summary()
```

Output from RStudio:

```
#####  
# KPSS Unit Root Test #  
#####  
  
Test is of type: mu with 6 lags.  
  
Value of test-statistic is: 4.6988  
  
Critical value for a significance level of:  
10pct 5pct 2.5pct 1pct  
critical values 0.347 0.463 0.574 0.739
```

3.F. KPSS Unit root test for the differenced series:

```
Oilprod.D1 %>% ur.kpss() %>% summary()
```

Output from RStudio:

```
#####  
# KPSS Unit Root Test #  
#####  
  
Test is of type: mu with 6 lags.  
  
Value of test-statistic is: 0.3347  
  
Critical value for a significance level of:  
10pct 5pct 2.5pct 1pct  
critical values 0.347 0.463 0.574 0.739
```

3.G. Test for appropriate number of differences on original series:

```
ndiffs(OilProduction)
```

Output from RStudio:

```
[1] 1
```

Seasonal differences:

```
nsdiffs(OilProduction)
```

Output from RStudio:

```
[1] 0
```

3.H Test for appropriate number of differences on differenced series:

```
ndiffs(Oilprod.D1)
```

Output from RStudio:

```
[1] 0
```

Seasonal differences:

```
nsdiffs(Oilprod.D1)
```

Output from RStudio:

```
[1] 0
```

3.I. Auto correlation Plots:

```
ggAcf(OilProduction)
```

```
ggtsdisplay(Oilprod.D1)
```

```
ggAcf(Oil.D12)
```

3.J. First splitting train and test part of the series:

```
Train.oil <- window(OilProduction, start=c(1971,7), end=c(2017,12), frequency=12)
```

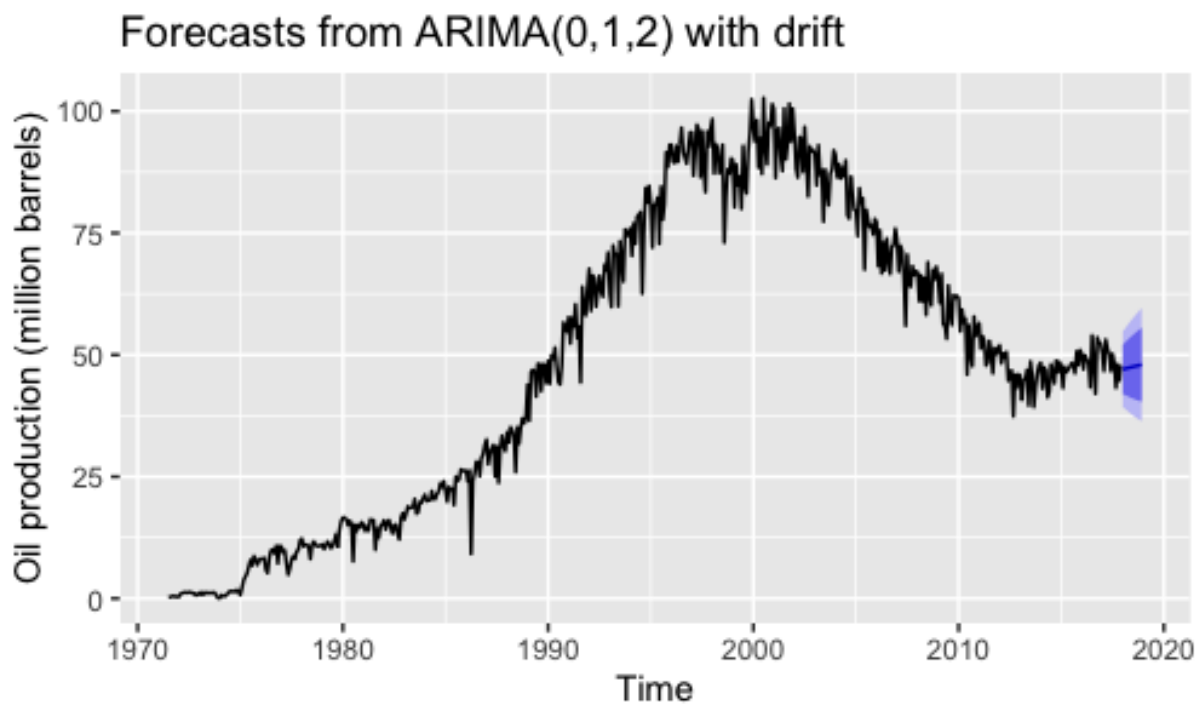
```
Test.oil <- window(OilProduction, start=c(2018,1), end=c(2018,12), frequency=12)
```

Constructing an automated ARIMA forecast:

```
Auto.arima<-auto.arima(Train.oil, seasonal=FALSE)
autoforecast <- forecast(Auto.arima, h=12)
```

Complete time series with forecast:

```
autoforecast %>%
  autoplot(ylab="Oil production (million barrels)")
```



Cutted plot by the restriction “xlim”:

```
autoplot(autoforecast, ylab = "Oil production", xlim=c(2014,2019))
```

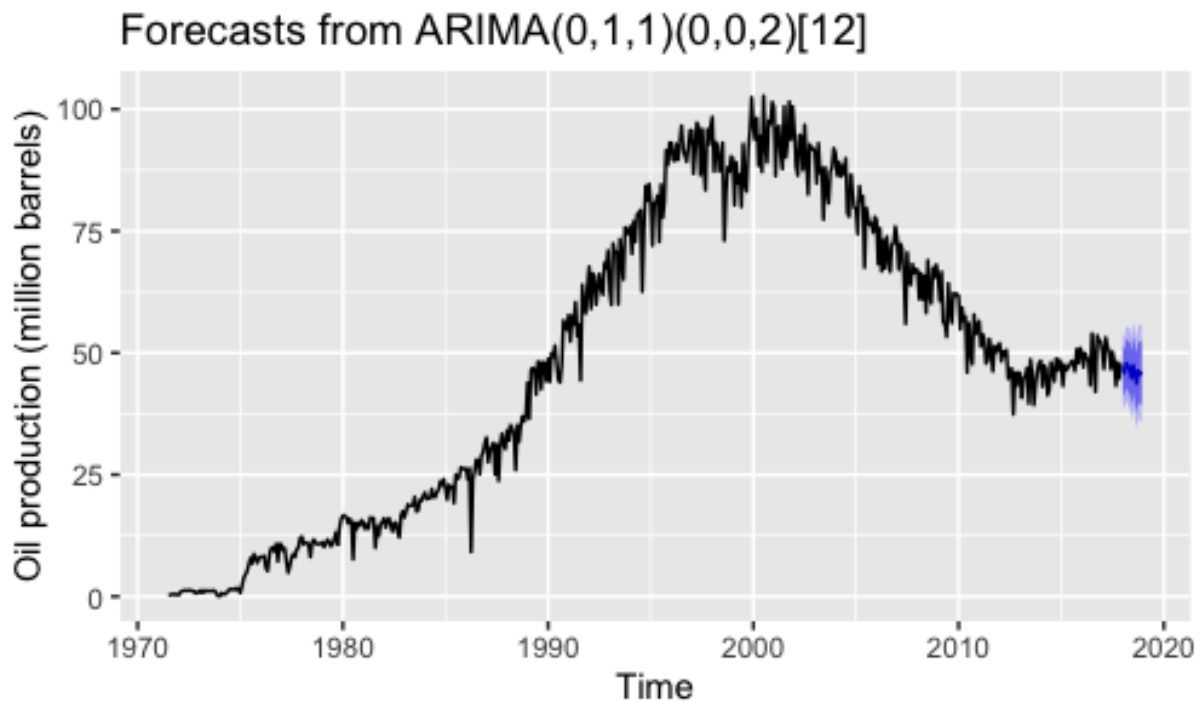
3.K. Automated ARIMA forecast model:

```
Autoseasonal<-auto.arima(Train.oil, seasonal=TRUE)
autoseasonalforecast <- forecast(Autoseasonal, h=12)
```

Plot:

```
autoseasonalforecast %>%
  autoplot(ylab="Oil production (million barrels)")
```

Complete time series with forecast:



Cutted plot in thesis:

```
autoplot(autoseasonalforecast, ylab = "Oil production", xlim=c(2014,2019))
```

3.L. Find AIC by calling the name of the model in RStudio:

For ARIMA(0,1,0):

```
Arima010 <- Arima(Train.oil, order=c(0,1,0))
```

```
Arima010
```

Output from RStudio:

```
Series: Train.oil
ARIMA(0,1,0)
sigma^2 estimated as 24.73: log likelihood=-1683.81
AIC=3369.61 AICc=3369.62 BIC=3373.93
```


For the automated non-seasonal ARIMA:

Auto.arima

Output from RStudio:

```
Series: Train.oil
ARIMA(0,1,2) with drift

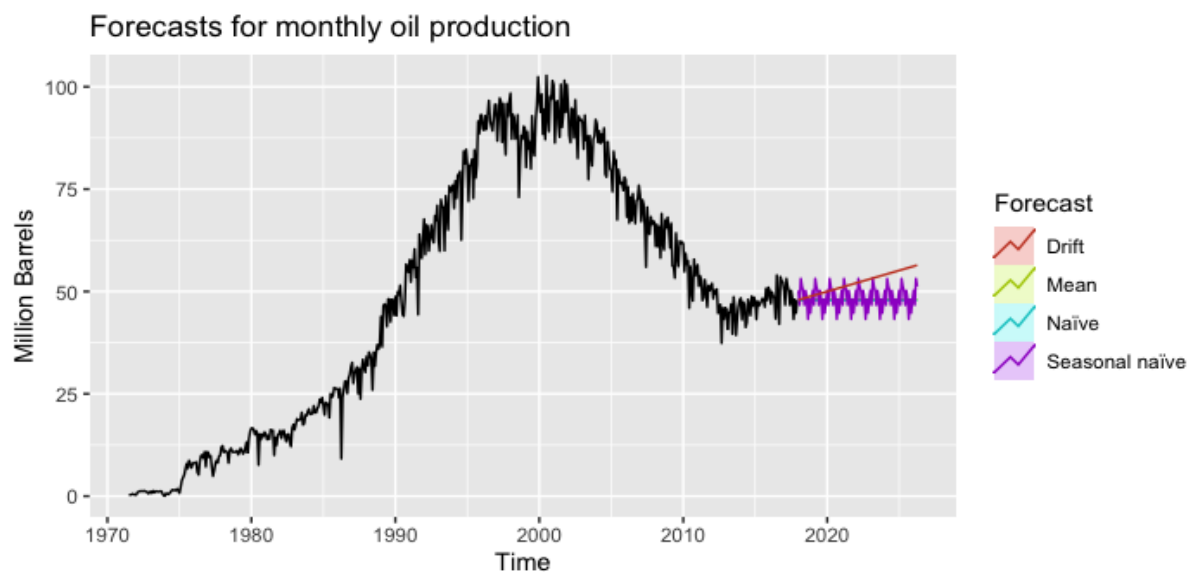
Coefficients:
      ma1  ma2  drift
-0.7479 0.0835 0.0839
s.e. 0.0437 0.0432 0.0569

sigma^2 estimated as 15.99: log likelihood=-1561.09
AIC=3130.19 AICc=3130.26 BIC=3147.48
```

3.M. Comparing the simple forecast methods:

```
autoplot(Train.oil) +
  autolayer(meanf(Train.oil, h=12),
    series="Mean", PI=FALSE) +
  autolayer(naive(Train.oil, h=12),
    series="Naïve", PI=FALSE) +
  autolayer(snaive(Train.oil, h=12),
    series="Seasonal naïve", PI=FALSE) +
  autolayer(rwf(Train.oil, drift=TRUE, h=12),
    series="Drift", PI=FALSE) +
  ggtitle("Forecasts for monthly oil production") +
  xlab("Time") + ylab("Million Barrels") + xlim(c(2016,2019)) +
  guides(colour=guide_legend(title="Forecast"))
```

Full time series with forecast horizon to 2025:



4.A. Construction of forecast and transform into time series:

```
meanforecast <- meanf(Train.oil,h=12)
mean.ts <-ts(as.vector(unlist(meanforecast[5])),frequency=12,start=c(2018,1),
end=c(2018,12))
```

Accuracy measures for that forecast:

```
mse(Test.oil, mean.ts)
mae(Test.oil, mean.ts)
```

4.B. Construct new test and train part for the series:

```
until.2019 <- window(OilProduction, start=c(1971,7), end=c(2018,12), frequency=12)
just.2019 <- window(OilProduction, start=c(2019,1), end=c(2019,12), frequency=12)
```

The seasonal automated ARIMA model:

```
SARIMA<-auto.arima(until.2019, seasonal=TRUE)
```

```
SARIMAforecast <- forecast(SARIMA, h=12)
```

Plotting the forecast along with actual production:

SARIMAForecast %>%

```
autoplot(ylab="Oil production (million barrels)",xlim=c(2016,2020), series= "ARIMA  
forecast") + autolayer(just.2019) , Series="Actual production")
```