Dataset 1: Daily Mean Measurements of Nitrous Oxides Level

Table of contents:

1. Introduction
2. Dataset Description
3. Data Analysis and Modeling
4. Model Diagnostics and Fit
5. Conclusion
6. Appendix: R Code

INTRODUCTION:

Welcome to this report, where we explore the analysis of daily mean measurements of nitrous oxide (NOx) levels recorded on the A23 Purley Way road in Croydon. Our objective is to gain a comprehensive understanding of the variations and trends in NOx levels, with the aim of assessing and improving air quality in the area. By employing a suitable time series model, we seek to uncover valuable insights into the factors influencing NOx levels and provide actionable information for environmental management.

DATA DESCRIPTION:

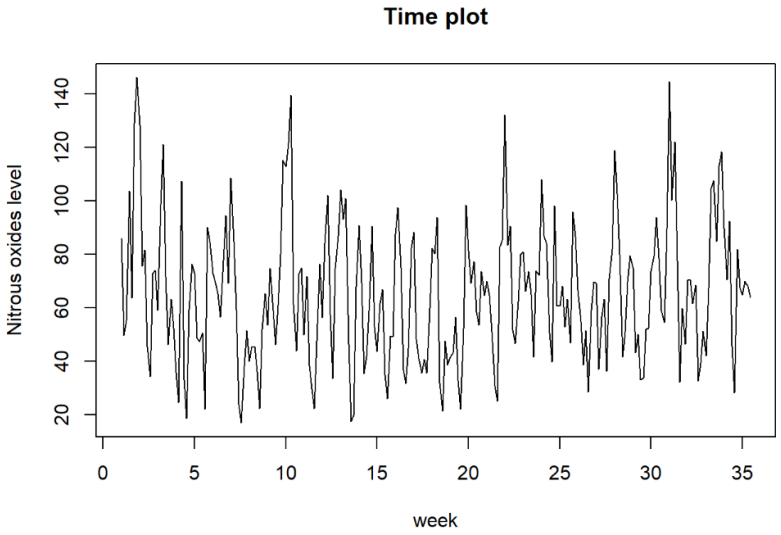
The dataset used in this analysis consists of daily mean measurements of nitrous oxide (NOx) levels, expressed in micrograms per cubic meter. The data spans a period from 1st February 2017 to 30th September 2017 and represents measurements taken at a specific location along the A23 Purley Way road in the London Borough of Croydon.

By examining this dataset, our goal is to detect patterns, establish potential correlations, and construct a robust time series model that accurately characterizes the behavior of NOx levels over time. These findings will play a crucial role in making informed decisions related to air quality management, contributing to the well-being of the local community and promoting sustainable environmental practices.

DATA ANALYSIS AND MODELLING:

We begin by importing the dataset and converting it into a time series format as the given data are collected sequentially over time.

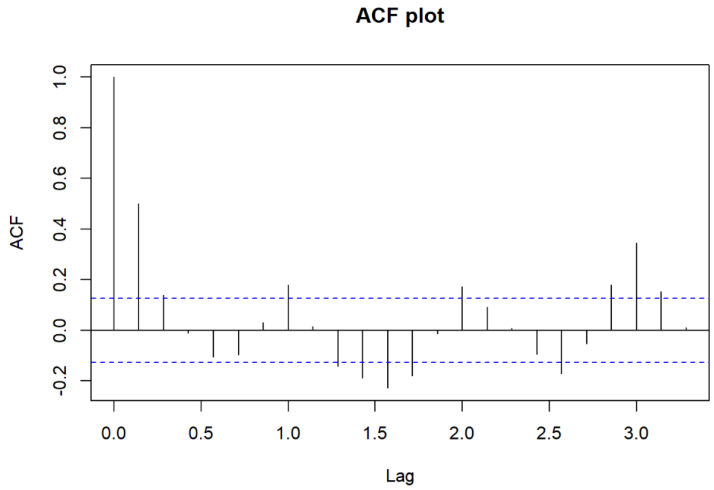
We will plot time plot, as the time plot is the initial stage in any examination of time series data. It could show the presence of a trend as well as seasonal variation. Also, any unusual observations or outliers may be visible.

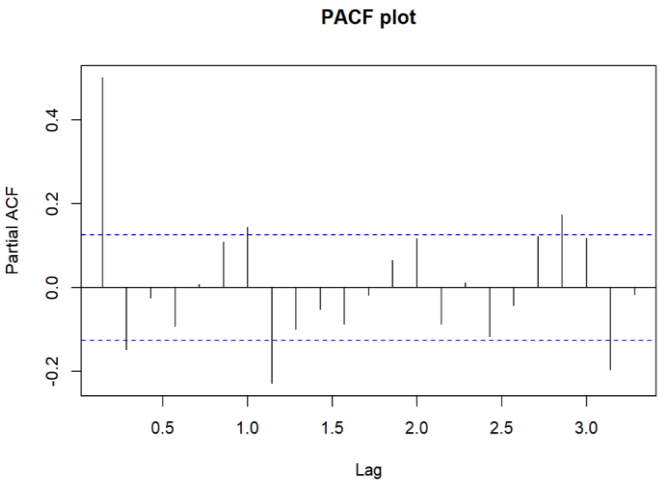


The time series plot suggests that the data is relatively stable and consistent based on a few key observations:

- Firstly, there doesn't seem to be a significant change in the average value of the data points over time.
- Secondly, the data points show relatively minor variation or unexpected fluctuations.
- However, it's worth noting that there appears to be a noticeable pattern of increasing variability over specific time intervals, indicating some seasonal effects in the data.

As a Next step, we will produce ACF and PACF plot to verify them.





After carefully analyzing both the ACF and PACF plots, we can confidently determine that the time series exhibits stationarity. The ACF plot also reveals that the spikes diminish quickly, indicating that there is no need to apply seasonal differencing, despite the presence of seasonality in the data.

We will verify the stationarity of the data using the Augmented Dickey-Fuller (ADF) test,

- In the ADF test, the null hypothesis assumes that the time series is non-stationary, while the alternative hypothesis suggests stationarity.
- The p-value produced by ADF test for our data is 0.01, which is less than the significance level of 0.05, hence we can reject the null hypothesis.
- Therefore, we can firmly conclude that the data is having a stationary behavior, indicating a consistent mean, variance, and autocovariance over time.

Now, we are ready to proceed with the model fitting process. To determine the appropriate model, we will examine the ACF (Autocorrelation Function) and PACF (Partial Autocorrelation Function) plots:

- Upon inspecting the ACF plot, we observe a gradual decrease in the spikes, indicating the absence of a Moving Average (MA) component. However, there appears to be a presence of an Autoregressive (AR) component.
- In the PACF plot, we notice a significant spike at lag 1, followed by a sharp cutoff. This pattern further confirms the suitability of an Autoregressive model.

Based on these observations, we will explore fitting both an AR(1) and AR(2) model to our data. We will evaluate the goodness of fit for each model and determine which one best captures the underlying dynamics of the data.

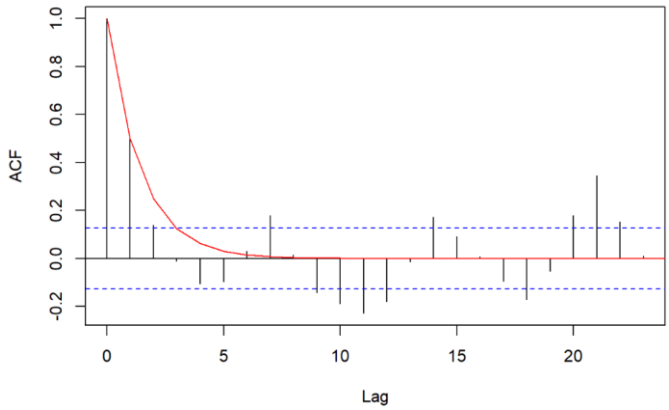
Model Diagnostics and Fit:

We will start by fitting AR(1) model, and examine some diagnostics:

Below is the R output for the AR(1) model with parameters using Maximum likelihood,

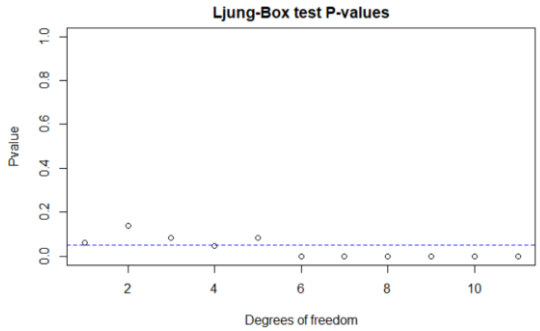
```
##
## Call:
## arima(x = ts_data, order = c(1, 0, 0), method = "ML")
##
## Coefficients:
##          ar1  intercept
##       0.4994    65.3338
## s.e.  0.0555     2.8826
##
## sigma^2 estimated as 508.1:  log likelihood = -1097.44,  aic = 2200.87
```

1) We will examine the fit of the model with its corresponding theoretical ACF:



By analysing the plot seems AR(1) model is not a good fit.

2) we will use Ljung test further to test this,



Almost all the p values are nearer to zero indicating to try AR(2) model as AR(1) is not a good choice.

It is clearly, visible that the AR(1) model is not the best fit.

We will try fitting AR(2) model:

Below is the R output for the AR(2) model with parameters using Maximum likelihood,

```
##
## Call:
## arima(x = ts_data, order = c(2, 0, 0), method = "ML")
##
## Coefficients:
##          ar1      ar2  intercept
##       0.5753  -0.1501    65.3332
## s.e.  0.0636   0.0635     2.4875
##
## sigma^2 estimated as 496.6:  log likelihood = -1094.68,  aic = 2197.36
```

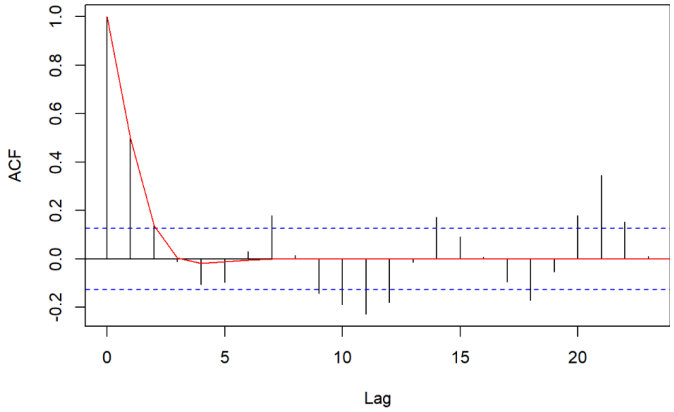
Using hypothesis testing we will evaluate the model:

Here we define the null hypothesis (H0) and alternative hypothesis (H1):

H0: $\varphi 1 = 0$ (AR coefficient is zero)
H1: $\varphi 1 \neq 0$ (AR coefficient is not zero)

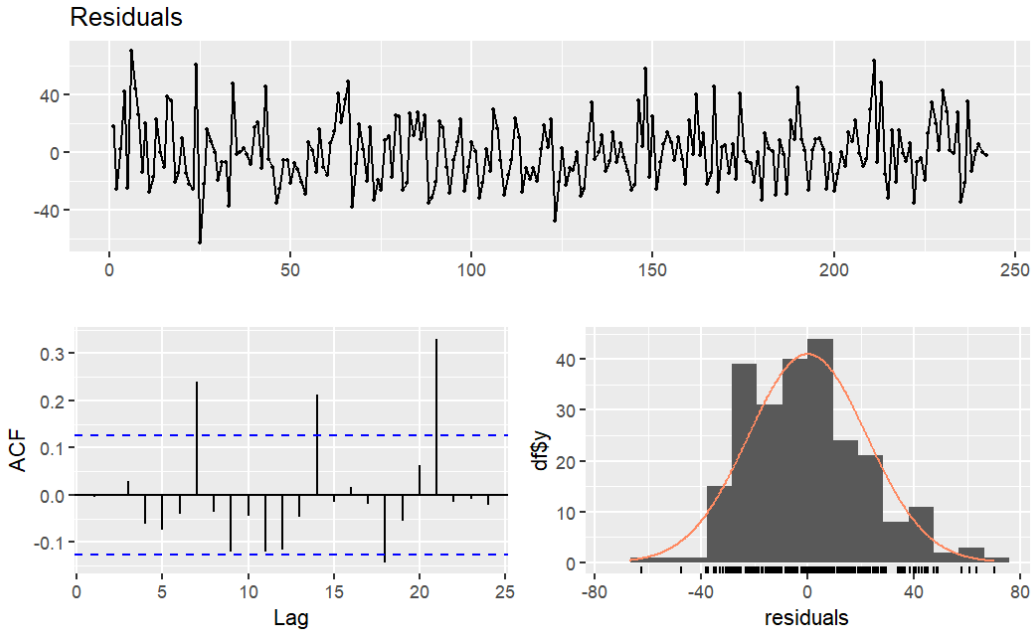
Then we will find test statistic by diving AR co-efficient by its Standard error.
We find out that absolute test statistic value for our data is 2.364 > 1.96, hence we reject Null hypothesis at 5 % level, indicating AR(2) model is good fit.

Now, we will try examining the fit of the model using its corresponding theoretical ACF:



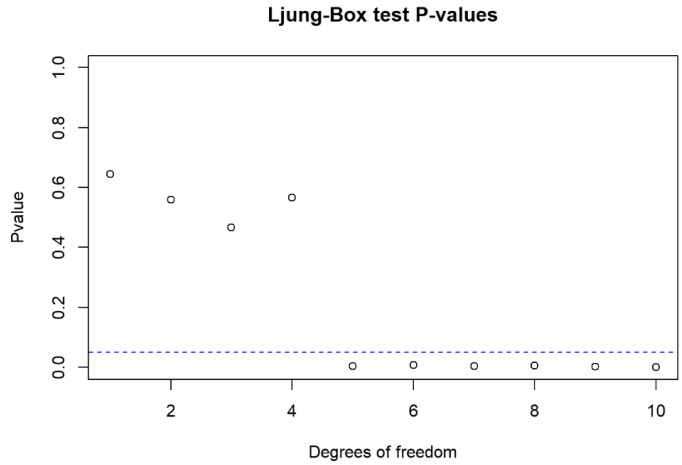
This model seems to fit accurately, also we will check with the other diagnostics methods.

We will try to produce residuals for the fitted model and analyze them:



Time plot of the residuals seems to be white noice but in ACF plot we can see the spikes at the cyclic lag (i.e.,7,14,21..) indicating a recurring pattern that occurs over regular intervals, resembling weekly cycles. However, it is worth noting that in this specific dataset, the presence of seasonality does not significantly influence the overall trend or the forecasting objectives we aim to achieve hence we proceed further with this model.

Finally, we will evalute the model with Ljung Box test:



From jung Box test we can notice that there is just 4 significant points and remaining close to zero this could be because of the seasonality present in the data.

Also, after fitting the AR(3) and ARMA(1,0,1) model and evaluating them it can be observed that the AR(2) model outperforms the other models in terms of several criteria:

- The Akaike Information Criterion (AIC) value for the AR(2) model is lower compared to the other models, indicating a better fit.
- Moreover, the results of hypothesis testing also suggest that the AR(2) model is a suitable choice for the data.
- The Ljung-Box test, which assesses the presence of autocorrelation, reveals that the AR(2) model exhibits a higher number of P-values greater than 0.05, further supporting its appropriateness.

Based on these findings, we conclude that the AR(2) model provides the best fit for the given dataset.

The equation for the AR(2) model based on the given data is:

X(t) = 0.5753 * X(t-1) - 0.1501 * X(t-2) + 65.3332

Where:
X(t) represents the value of the nitrous oxide (NOx) level at time t.
X(t-1) represents the value of the NOx level at the previous time step (t-1).
X(t-2) represents the value of the NOx level at the time step before the previous time step (t-2).

CONCLUSION:

Finally, based on our study of the daily mean observations of nitrous oxide (NOx) levels, we chose the AR(2) model as the best fit for capturing the underlying patterns and dynamics of the data. Based on numerous evaluation criteria, the AR(2) model outperformed other models.

Several factors influenced the choice of the AR(2) model. To begin, the AR(2) model has a lower Akaike Information Criterion (AIC) value, indicating a better fit to the data. Furthermore, hypothesis testing confirmed the AR(2) model's significance, bolstering its suitability for representing NOx levels. The Ljung-Box test results also indicated that the AR(2) model has a higher number of p-values greater than 0.05, indicating a better fit in terms of autocorrelation.

We can confidently estimate and analyses future NOx levels by using the AR(2) model, which aids in effective air quality management and decision-making. These findings are vital for implementing suitable steps to improve environmental conditions and ensure the local community's well-being.

APPENDIX: R Code

```
data = read.csv('a23_nox.csv', header=TRUE)       #loading the data
ts_data = ts(data$daily_mean_nox, frequency = 7)    #Setting the data as Time series data
plot(ts_data, main="Time plot", xlab="week", ylab=" Nitrous oxides level ")   #time plot
acf(ts_data,main="ACF plot")   #ACF plot
pacf(ts_data,main="PACF plot")  #PACF plot
library(tseries)                                #importing library to use ADF test
adf_test = adf.test(ts_data)                      #checking stationary using ADF test
adf_test
if ((adf_test$p.value)<0.05)
  print(paste("as p-value ",adf_test$p.value," is less than 0.05, hence rejecting the null hypothesis. Therefore,
it is stationary time series " ))else
    print("Non stationary time series")
AR_model1<-arima(ts_data,order=c(1,0,0),method="ML")   #fitting AR(1) model
AR_model1
test_Statistic=abs(0.4994/0.0555)       #Hypothesis testing for AR(1) model
if(test_Statistic<1.96){
   print(paste("As test statistic : ",test_Statistic," < 1.96, we retain Null hypothesis at 5 % level and should fit
AR(1) model.  "))}else{
  print(paste("As test statistic : ",test_Statistic," > 1.96, we reject Null hypothesis at 5 % level and should fit
AR(2) model.  "))
}
AR1_acf= ARMAacf(ma=c(0), ar=c(0.4994), lag=100)  # Compute the ACF for an AR(1) model
y_lag = c(0:100)                                # Create lag values from 0 to 100
acf(data$daily_mean_nox)                          # Compute ACF of the given data
lines(y_lag,AR1_acf,col="red")                     #overlay the ACF values computed for the AR(1) model on the
ACF plot
#Ljung-Box test AR(1)
LB_test<-function(resid,max.k,p,q){
 lb_result<-list()
 df<-list()
 p_value<-list()
  for(i in (p+q+1):max.k){
   lb_result[[i]]<-Box.test(resid,lag=i,type=c("Ljung-Box"),fitdf=(p+q))
   df[[i]]<-lb_result[[i]]$parameter
   p_value[[i]]<-lb_result[[i]]$p.value
  }
 df<-as.vector(unlist(df))
 p_value<-as.vector(unlist(p_value))
 test_output<-data.frame(df,p_value)
 names(test_output)<-c("deg_freedom","LB_p_value")
 return(test_output)
}
# Perform the Ljung-Box test on AR(1) residuals
AR1.LB<-LB_test(resid_AR1,max.k=12,p=1,q=0)
# Plot the Ljung-Box test P-values
plot(AR1.LB$deg_freedom,AR1.LB$LB_p_value,xlab="Degrees of freedom",ylab="Pvalue",main="Ljung-
Box test P-values",ylim=c(0,1))
# Add a horizontal line at significance level 0.05
abline(h=0.05,col="blue",lty=2)
AR_model2<-arima(ts_data,order=c(2,0,0),method="ML") #fitting AR(2) model
AR_model2
```

```
test_Statistic=abs(-0.1501/0.0635)  #hypothesis testing for AR(2) model
#plotting sample ACF and Theoretical ACF
AR2_acf= ARMAacf(ma=c(0), ar=c(0.5753,-0.1501), lag=100)
y_lag = c(0:100)
acf(data$daily_mean_nox)
lines(y_lag,AR2_acf,col="red")
library(forecast)              #importing forecast library to use checkresiduals function
checkresiduals(AR_model2)      #checking residuals
#Ljung-Box test AR(2)
LB_test<-function(resid,max.k,p,q){
 lb_result<-list()
 df<-list()
 p_value<-list()
  for(i in (p+q+1):max.k){
   lb_result[[i]]<-Box.test(resid,lag=i,type=c("Ljung-Box"),fitdf=(p+q))
   df[[i]]<-lb_result[[i]]$parameter
   p_value[[i]]<-lb_result[[i]]$p.value
  }
 df<-as.vector(unlist(df))
 p_value<-as.vector(unlist(p_value))
 test_output<-data.frame(df,p_value)
 names(test_output)<-c("deg_freedom","LB_p_value")
 return(test_output)
}
resid_AR2=residuals(AR_model2)
# Perform the Ljung-Box test on AR(2) residuals
AR1.LB<-LB_test(resid_AR2,max.k=12,p=2,q=0)
# Plot the Ljung-Box test P-values
plot(AR1.LB$deg_freedom,AR1.LB$LB_p_value,xlab="Degrees of freedom",ylab="Pvalue",main="Ljung-
Box test P-values",ylim=c(0,1))
# Add a horizontal line at significance level 0.05
abline(h=0.05,col="blue",lty=2)
AR_model3<-arima(ts_data,order=c(3,0,0),method="ML") #fitting AR(3) model

AR_model3

test_Statistic=abs(-0.0236/0.0641)  #hypothesis testing for AR(3) model

#plotting sample ACF and Theoretical ACF

AR3_acf= ARMAacf(ma=c(0), ar=c(0.5717,-0.1364,-0.0236), lag=100)

y_lag = c(0:100)

acf(data$daily_mean_nox)

lines(y_lag,AR3_acf,col="red")

library(forecast)              #importing forecast library to use checkresiduals function

checkresiduals(AR_model3)      #checking residuals

#Ljung-Box test AR(3)

LB_test<-function(resid,max.k,p,q){

 lb_result<-list()

 df<-list()
```

```
 p_value<-list()
  for(i in (p+q+1):max.k){
   lb_result[[i]]<-Box.test(resid,lag=i,type=c("Ljung-Box"),fitdf=(p+q))
   df[[i]]<-lb_result[[i]]$parameter
   p_value[[i]]<-lb_result[[i]]$p.value
  }
 df<-as.vector(unlist(df))
 p_value<-as.vector(unlist(p_value))
 test_output<-data.frame(df,p_value)
 names(test_output)<-c("deg_freedom","LB_p_value")
 return(test_output)
}

# Perform the Ljung-Box test on AR(3) residuals
resid_AR3=residuals(AR_model3)
AR1.LB<-LB_test(resid_AR3,max.k=12,p=3,q=0)
# Plot the Ljung-Box test P-values
plot(AR1.LB$deg_freedom,AR1.LB$LB_p_value,xlab="Degrees of freedom",ylab="Pvalue",main="Ljung-Box test P-values",ylim=c(0,1))
# Add a horizontal line at significance level 0.05
abline(h=0.05,col="blue",lty=2)
ARMA_model<-arima(ts_data,order=c(1,0,1),method="ML") #fitting ARMA(1,1) model
ARMA_model
#plotting sample ACF and Theoretical ACF
ARMA_acf= ARMAacf(ma=c(0.2723), ar=c(0.3013), lag=100)
y_lag = c(0:100)
acf(data$daily_mean_nox)
lines(y_lag,ARMA_acf,col="red")
library(forecast)              #importing forecast library to use checkresiduals function
checkresiduals(ARMA_model)       #checking residuals
#Ljung-Box test ARMA(1,1)
LB_test<-function(resid,max.k,p,q){
 lb_result<-list()
 df<-list()
 p_value<-list()
```

```
 for(i in (p+q+1):max.k){
  lb_result[[i]]<-Box.test(resid,lag=i,type=c("Ljung-Box"),fitdf=(p+q))
  df[[i]]<-lb_result[[i]]$parameter
  p_value[[i]]<-lb_result[[i]]$p.value
 }
df<-as.vector(unlist(df))
p_value<-as.vector(unlist(p_value))
test_output<-data.frame(df,p_value)
names(test_output)<-c("deg_freedom","LB_p_value")
return(test_output)
}
# Perform the Ljung-Box test on ARMA(1,1) residuals
resid_ARMA=residuals(ARMA_model)
AR1.LB<-LB_test(resid_ARMA,max.k=12,p=2,q=0)
# Plot the Ljung-Box test P-values
plot(AR1.LB$deg_freedom,AR1.LB$LB_p_value,xlab="Degrees of freedom",ylab="Pvalue",main="Ljung-Box test P-values",ylim=c(0,1))
# Add a horizontal line at significance level 0.05
abline(h=0.05,col="blue",lty=2)
```

Dataset 2: Quarterly New Car Registrations in England

Table of contents:

1. Executive Summary
2. Introduction
3. Dataset Description
4. Data Analysis and Modeling
5. Model Diagnostics and Fit
6. Forecasting
7. Conclusion
8. Appendix: R Code

EXECUTIVE SUMMARY

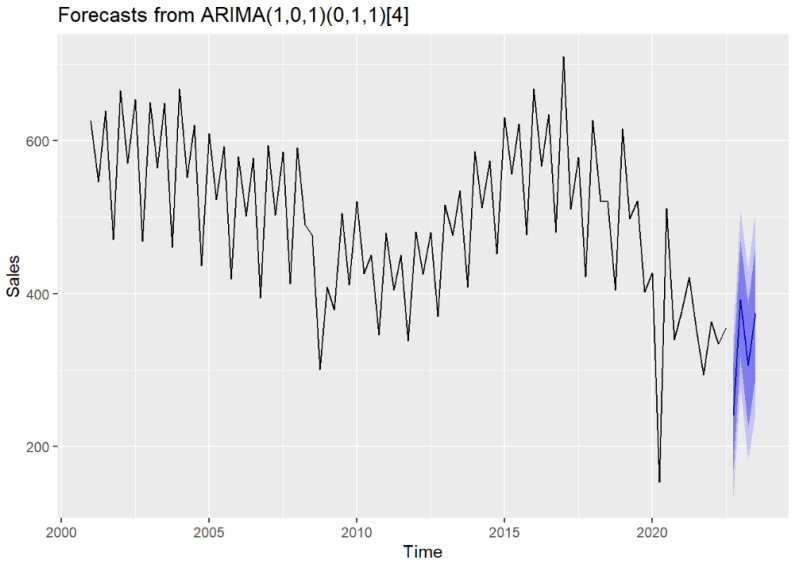
As a statistical advisor to the car industry, my task was to analyze a dataset containing quarterly counts of new car registrations in England and forecast future registration numbers. By examining the dataset from Q1 2001 to Q3 2022, we aimed to identify trends and seasonal patterns that would help us make accurate predictions.

Upon analyzing the data, we observed that new car registrations in England fluctuated over time, indicating non-stationarity. However, by applying seasonal differencing, we were able to remove the seasonal patterns and make the data stationary. This allowed us to proceed with modeling.

Based on the analysis of the autocorrelation and partial autocorrelation functions, we determined that an ARIMA(1,0,1)(0,1,1)[4] model was the most suitable for the data. This model included autoregressive, moving average, and seasonal moving average components.

We thoroughly evaluated the chosen model using various diagnostics methods, such as examining the residuals and performing hypothesis and Ljung-Box tests. The results indicated that the ARIMA(1,0,1)(0,1,1)[4] model accurately captured the underlying patterns and minimized residual autocorrelation.

Using this model, we forecasted the number of new car registrations for Q1 2022, Q2 2023, Q3 2023, and Q4 2023. The forecasted numbers are as follows: 240.2288 thousand for Q1 2022, 392.2740 thousand for Q2 2023, 305.7431 thousand for Q3 2023, and 374.2759 thousand for Q4 2023. The forecast plot is below:



These forecasts provide valuable insights for the car industry, allowing stakeholders to anticipate market trends and make informed decisions regarding production and inventory levels. By aligning their strategies with the expected demand, car manufacturers and dealerships can remain competitive in the market.

In conclusion, our analysis and modeling techniques have enabled us to accurately forecast the future numbers of new car registrations in England. These forecasts serve as valuable tools for industry professionals, empowering them to plan effectively and adapt to changing market dynamics.

INTRODUCTION:

This analysis report explores the dataset of new car registrations in England. Our goal as a statistical advisor to the industry of cars is to analyses past data and forecast future quantities of new car registrations. This report intends to provide significant insights and aid in strategic decision-making for the England automobile sector.

DATA DESCRIPTION:

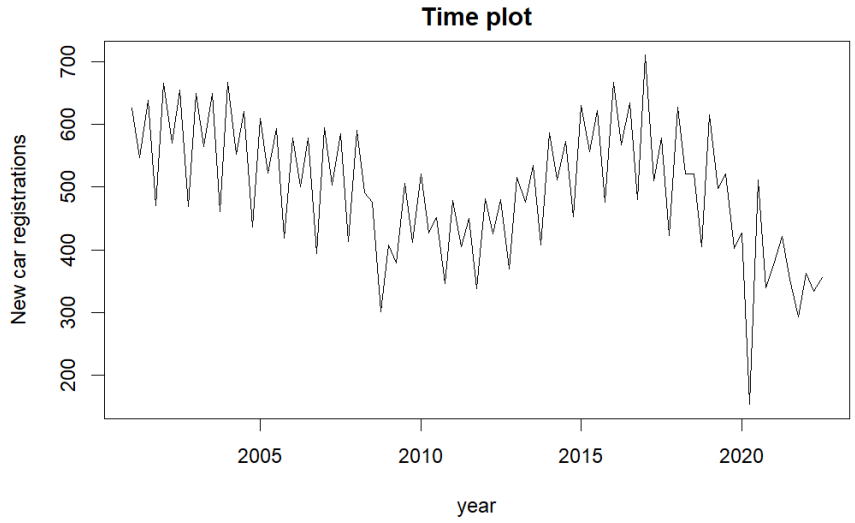
The dataset under consideration includes quarterly counts of new car registrations in England from Q1 2001 to Q3 2022. Each quarter represents a three-month period, with Q1 corresponding to January to March, Q2 corresponding to April to June, Q3 corresponding to July to September, and Q4 corresponding to October to December.

The data is collected in thousands, providing an idea of the number of new car registrations throughout each quarter. We intend to discover trends, seasonal patterns, and other factors impacting new car registrations in England by analysing this dataset. This analysis will allow for more precise forecasting of future figures, allowing the automobile sector to make informed choices and plan for future market dynamics.

DATA ANALYSIS AND MODELLING:

To commence our analysis, we shall import the dataset and transform it into a time series format, considering the sequential nature of the collected data over time.

We will proceed by creating a time plot, which serves as a fundamental step in exploring time series data. This visual representation allows us to identify potential trends and seasonal variations. Moreover, any noteworthy observations such as outliers can be easily detected through this plot.



Upon analyzing a forementioned time plot, a notable observation emerges:

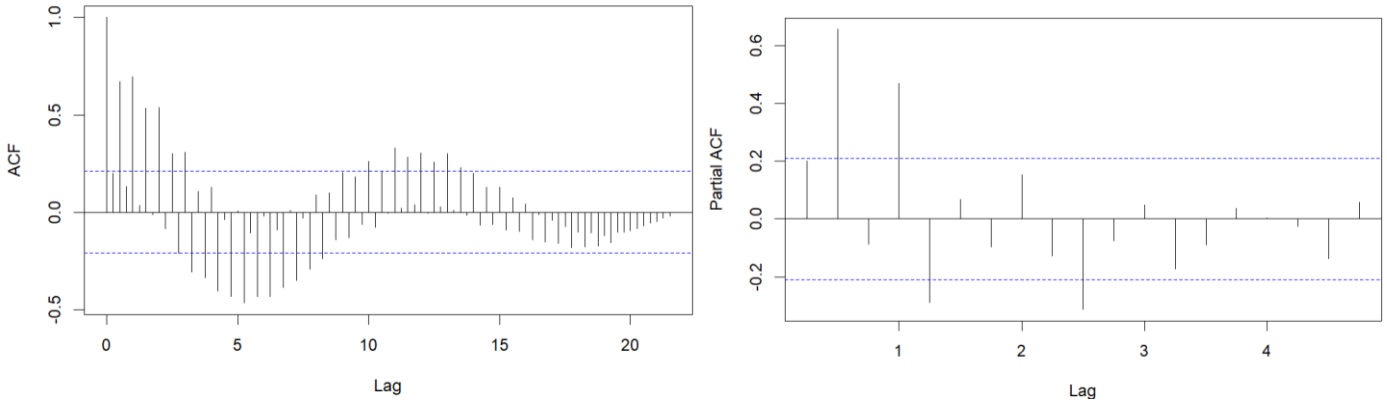
- As the mean does not appear to be constant and there are variations over time, the time series appears to be non-stationary. This means that the statistical qualities of the data appear to

vary with time, making certain analytical approaches difficult to apply.

- It is also essential to recognize the presence of outliers, notably around 2009 and 2020. When analysing the impact of outliers on our analysis objectives, it does not appear to have a significant impact on forecasting.

- Furthermore, when the data is examined, a distinct cyclic pattern emerges. There is a constant trend of greater registration numbers in the first and third quarters of each year, with lower numbers in the second and fourth quarters.

Let's plot ACF and PACF and analyses them,

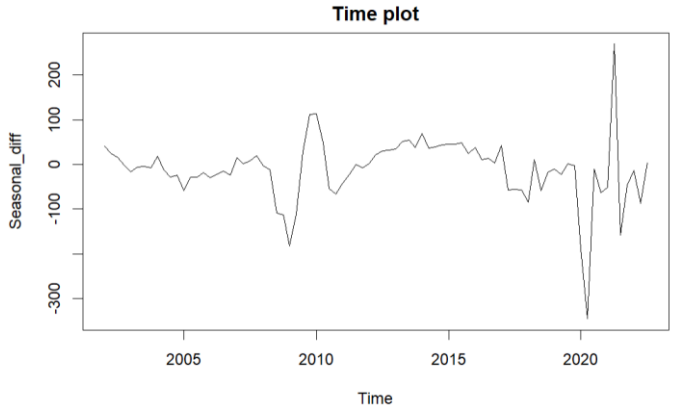


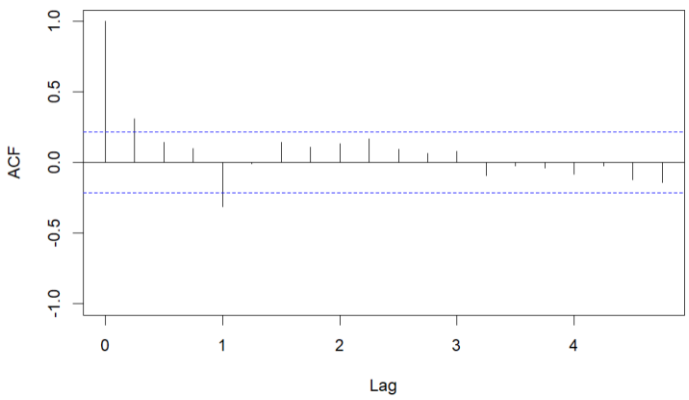
The autocorrelation function (ACF) indicates a slow decay of sample values as the lag increases, which suggests non-stationarity. This could potentially be attributed to the presence of seasonality in the data.

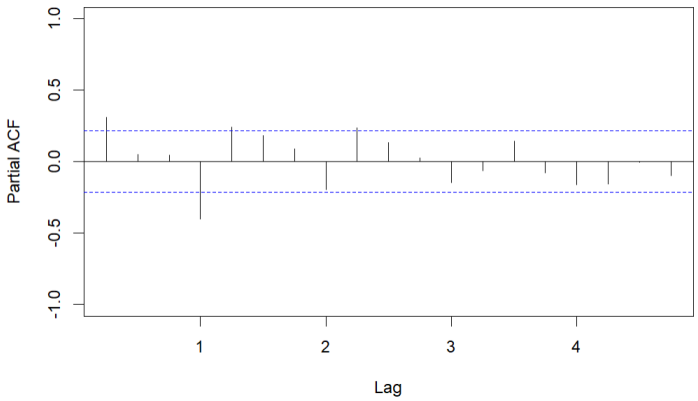
To address this issue, we could perform a seasonal difference on the data, considering it is collected quarterly with a period of 4. By applying this seasonal differencing, we aim to remove the seasonal patterns and make the data more stationary, thereby facilitating accurate statistical analysis.

i.e., $Y(t) = X(t) - X(t-4)$

the seasonally differenced plots are below:







Based on a examination of the plots, it becomes apparent that the seasonally differenced data exhibit characteristics of stationarity, as visual analysis reveals that the majority of the spikes in both the autocorrelation function (ACF) and partial autocorrelation function (PACF) are insignificant, with only one or two notable spikes that deviate from this pattern.

We will verify it by performing Augmented Dickey-Fuller (ADF) Test.

Upon conducting the Augmented Dickey-Fuller (ADF) Test, the resulting p-value was found to be 0.0468. This value, being less than the commonly chosen significance level of 0.05, provides substantial evidence to reject the null hypothesis. Consequently, we can confidently state that the time series in question exhibits stationarity.

Now we will proceed to choose our model,

Based on the analysis of the seasonally differenced ACF and PACF plots, we can make certain observations:

- The presence of a significant spike at lag 1 in the ACF plot suggests the inclusion of a non-seasonal Moving Average (MA) component in our model.
- Similarly, the significant spike at lag 4 in the ACF plot indicates the need for a seasonal MA component.

Considering these findings, we can construct our model as ARIMA(0,0,1)(0,1,1)[4].
This notation signifies the inclusion of a non-seasonal MA(1) component, a seasonal MA(1) component, and a seasonal difference with a lag of 4.

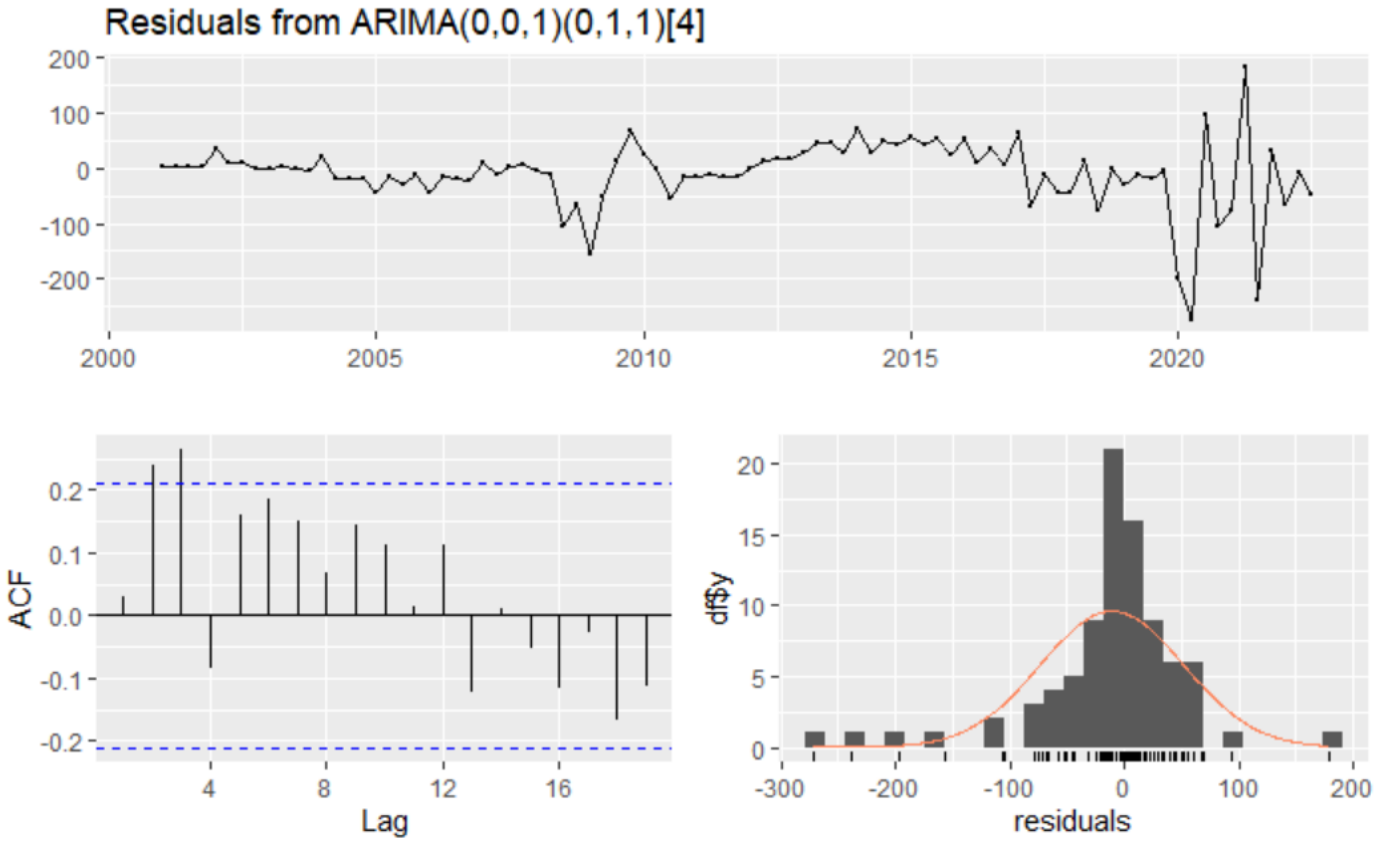
Model Diagnostics and Fit:

We will start fitting ARIMA(0,0,1)(0,1,1)[4] model and examine it,

Below is it R output of the fitted model:

```
##
## Call:
## arima(x = ts_data, order = c(0, 0, 1), seasonal = list(order = c(0, 1, 1), period = 4),
##     method = "ML")
##
## Coefficients:
##          ma1     sma1
##       0.4139  -0.3426
## s.e.  0.1061   0.1004
##
## sigma^2 estimated as 4178:  log likelihood = -464.13,  aic = 934.27
```

We will perform Model Diagnostics : Residuals to examine the fit,



- The residuals from the fitted model do not appear to exhibit the characteristics of a white noise process.
- The existence of significant spikes at lags 2 and 3 in both the autocorrelation function (ACF) and partial autocorrelation function (PACF) suggests that the model requires an additional component.

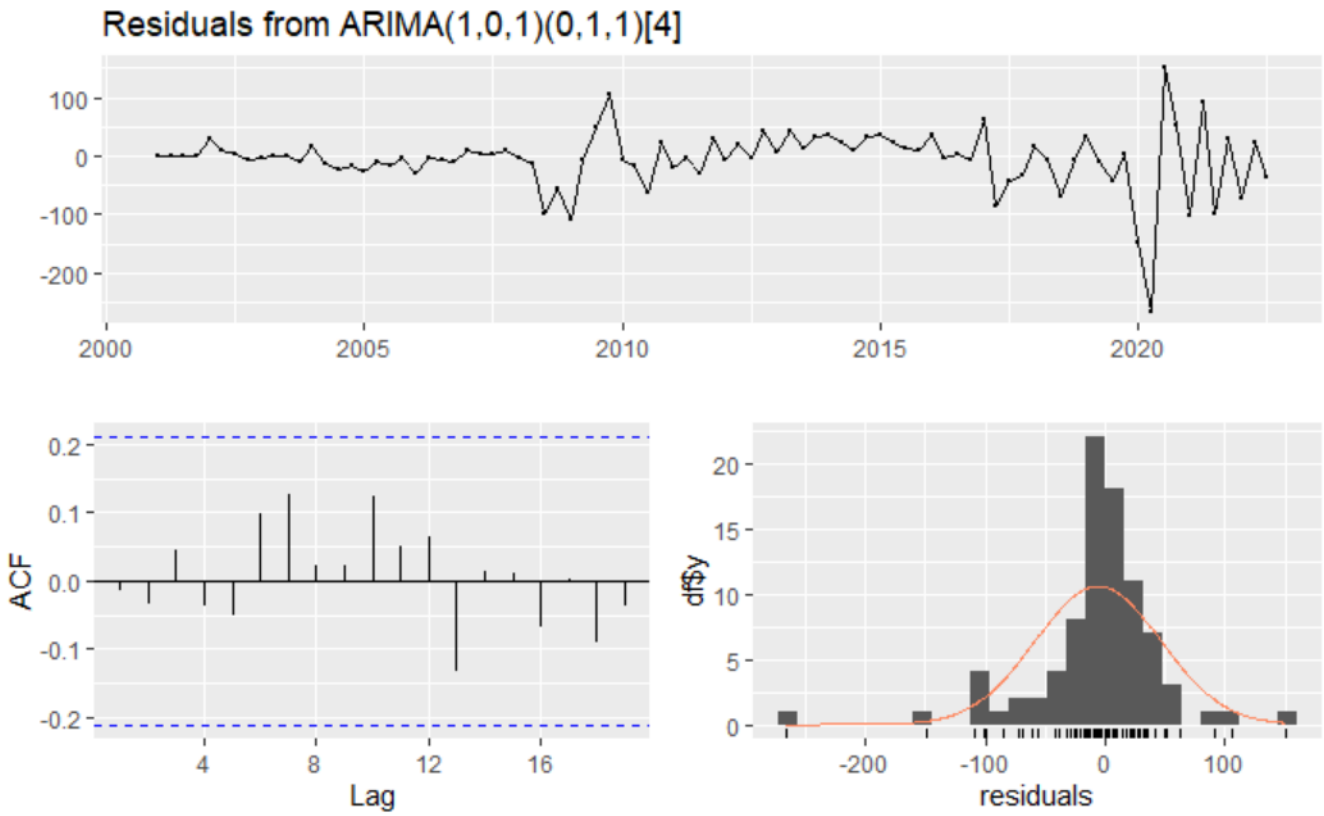
In this case, it appears reasonable to include an autoregressive term of order 1 (AR(1)).

We will fit ARIMA(1,0,1)(0,1,1)[4] model and evaluate it,

Below is it R output of the fitted model:

```
##
## Call:
## arima(x = ts_data, order = c(1, 0, 1), seasonal = list(order = c(0, 1, 1), period = 4),
##     method = "ML")
##
## Coefficients:
##          ar1      ma1     sma1
##       0.9845  -0.5401  -0.9360
## s.e.  0.0548   0.0969   0.1561
##
## sigma^2 estimated as 2878:  log likelihood = -451.52,  aic = 911.04
```

The AIC of the ARIMA(1,0,1)(0,1,1)[4] model (911.04) appears to be lower than previously fitted model (934.27) suggesting the good fit, further we will analyses the residuals to check the whether the current model identifies the underlying patterns,



After incorporating the AR(1) non-seasonal component into the model, we have successfully addressed the significant spikes observed at lag 2 and 3 in both the ACF and PACF.
This addition has improved the model's ability to capture the underlying patterns and correlations in the data.

To verify the effectiveness of the augmented model, we will evaluate it using hypothesis test and Ljung-Box test:

Hypothesis testing,
Here we define the null hypothesis (H0) and alternative hypothesis (H1):

H0: $\varphi 1 = 0$ (AR and MA coefficient is zero)
H1: $\varphi 1 \neq 0$ (AR and MA coefficient is not zero)

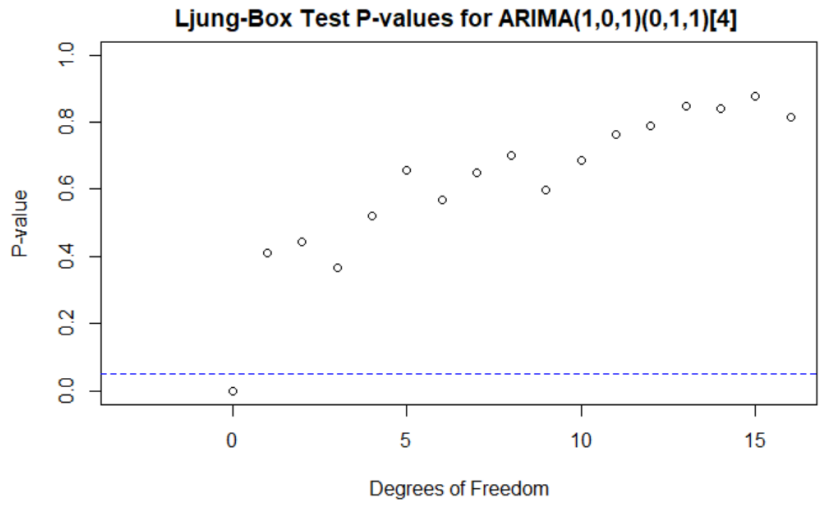
Then we will find test statistics by diving AR co-efficient by its Standard error and MA co-efficient by its standard error.

- We find out that absolute test statistic value for the AR co-efficient is 17.96533 > 1.96, and for the MA co-efficient is 5.573787 > 1.96, hence we reject Null hypothesis at 5 % level.

This supports the conclusion that the AR(1) and MA(1) components added to the model are contributing significantly to the fit and capturing important dynamics in the data.

Now, we will confirm this with Ljung-Box test:

We will plot the residuals p value against the degree of freedom to assess the independence of residuals by examining their autocorrelation at different lags,
Below is the plot generated by R:



The p-values are continuously greater than the significance level (0.05), it implies the residuals are independent and have no significant autocorrelation. Hence confirms the fitted model is accurate.

Now, we will add one more AR component to our model and check whether it increases the accuracy of the model or not.

We will fit ARIMA(2,0,1)(0,1,1)[4] model and evaluate it,

Below is it R output of the fitted model:

```
##
## Call:
## arima(x = ts_data, order = c(2, 0, 1), seasonal = list(order = c(0, 1, 1), period = 4),
##     method = "ML")
##
## Coefficients:
##          ar1     ar2     ma1     sma1
##       0.9672  0.0151  -0.529  -0.9328
## s.e.  0.1961  0.1799   0.159   0.1410
##
## sigma^2 estimated as 2882:  log likelihood = -451.52,  aic = 913.04
```

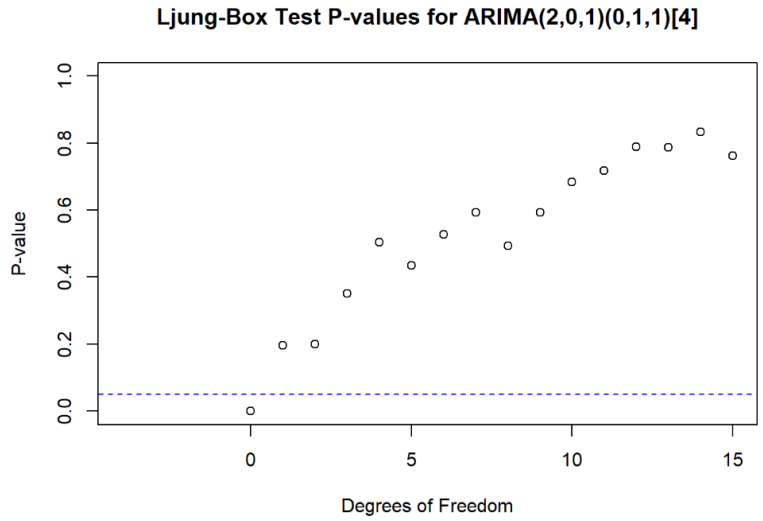
The AIC of ARIMA(2,0,1)(0,1,1)[4] model (913.04) is slightly greater than ARIMA(1,0,1)(0,1,1)[4] model (911.04).

We will perform hypothesis test on ARIMA(2,0,1)(0,1,1)[4] model:

- We find out that absolute test statistic value for the AR co-efficient is 0.0839 < 1.96, and for the MA co-efficient is 3.327 > 1.96, hence we reject Null hypothesis for MA co-efficient at 5 % level and retain it for AR co-efficient.

This indicates that p=1 (i.e., an autoregressive term of order 1) is a suitable choice for our model.

Now, we will proceed with Ljung- Box test:



The p-values obtained from this test are slightly less significant compared to the previous model, indicating that the ARIMA(1,0,1)(0,1,1)[4] model performs better in capturing the underlying patterns and reducing residual autocorrelation.

Taking into account the results from the various diagnostics methods performed above, we can confidently assert that the ARIMA(1,0,1)(0,1,1)[4] model is the most accurate representation for our data. This model incorporates an autoregressive term of order 1, a moving average term of order 1, and a seasonal moving average term of order 1 with a seasonal period of 4.

By following these rigorous statistical methods, we have successfully identified a robust model that captures the intricate dynamics within our time series data, allowing for more reliable inferences and predictions moving forward.

The estimated coefficients for the ARIMA(1,0,1)(0,1,1)[4] model, obtained through the maximum likelihood method, are as follows:

- AR(1) coefficient (non-seasonal): (phi_1 = 0.9845) (standard error = 0.0548)
- MA(1) coefficient (non-seasonal): (theta1 = -0.5401) (standard error = 0.0969)
- Seasonal MA(1) coefficient: (Theta_1 = -0.9360) (standard error = 0.1561)

The estimated variance of the error term (sigma^2)) is 2878. The log likelihood of the model is -451.52, and the AIC (Akaike Information Criterion) is 911.04.

Using these estimates, the equation for the ARIMA(1,0,1)(0,1,1)[4] model can be written as:

(1-0.9845B)(1-B^4)(1+0.5401B)Delta Xt = c + (1+0.9360B^4)Zt

where:

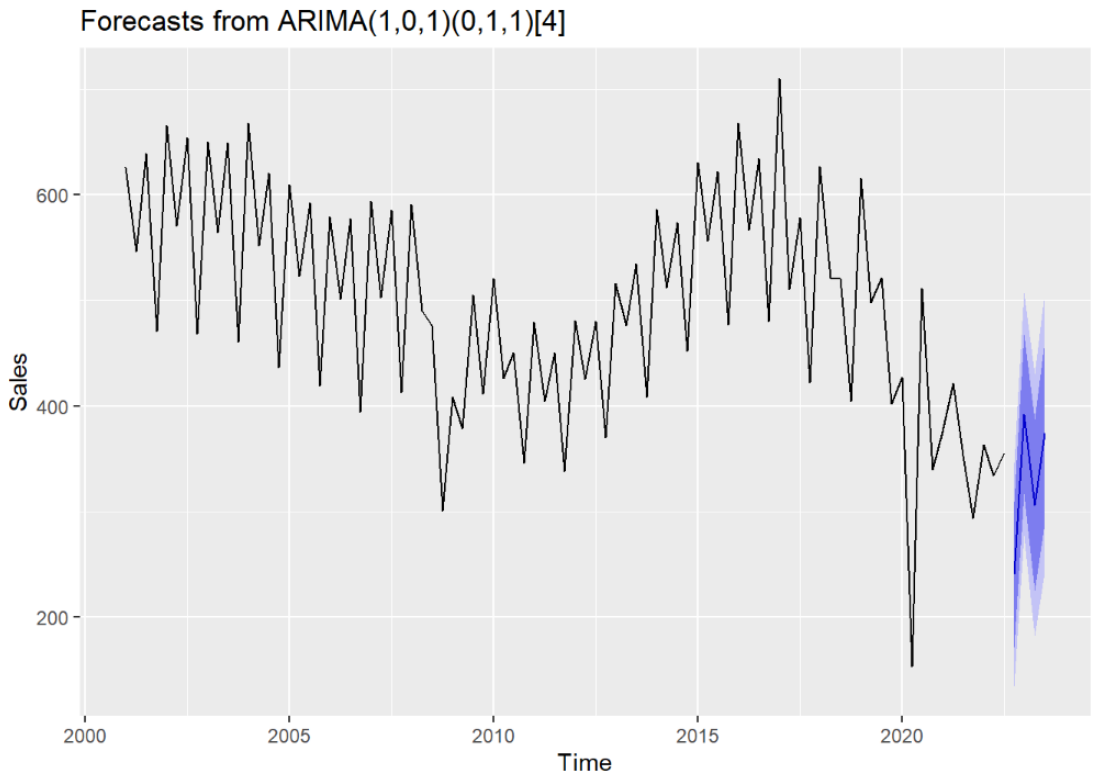
- Delta Xt represents the differenced value of the time series at time (t).
- B is the backshift operator, which shifts the time series backward by one time period.
- c is the constant term or intercept of the model.
- Zt represents the error term, assumed to follow an independent and identically distributed process with mean zero and variance (sigma^2).

FORECASTING:

By using the ARIMA(1,0,1)(0,1,1) model we will forecasting in the R using forecast package:

Below are the forecasted output and plot:

```
##          Qtr1     Qtr2     Qtr3     Qtr4
## 2022                           240.2288
## 2023 392.2740 305.7431 374.2759
```



The shaded areas give uncertainty bounds (80% and 95%) around the forecasted estimates.

The forecast for new car registrations in England, represented in thousands, is as follows:

- Q1 2022: The number of new car registrations in Q1 2022 is projected to be 240.2288 thousand.
- Q2 2023: For Q2 2023, the forecast indicates an increase in new car registrations to 392.2740 thousand.
- Q3 2023: The forecast for Q3 2023 suggests a decrease in new car registrations to 305.7431 thousand.
- Q4 2023: Finally, in Q4 2023, there is an expected rise in new car registrations to 374.2759 thousand.

CONCLUSION:

To summarize, our analysis of the dataset on new car registrations in England has provided valuable insights for the car industry. By delving into the data and applying statistical techniques, we have uncovered important trends and seasonal patterns that drive car registration numbers. These findings enable us to make accurate forecasts for future registration figures.

Our forecast for Q1 2022 suggests a projected number of 240.2288 thousand new car registrations, indicating a steady demand. Looking ahead to Q2 2023, we anticipate a significant increase in registrations to 392.2740 thousand, indicating a potential surge in consumer interest. However, the forecast for Q3 2023 indicates a slight dip to 305.7431 thousand registrations, suggesting a temporary slowdown. Lastly, our forecast for Q4 2023 predicts a rebound in registrations to 374.2759 thousand, indicating a potential resurgence in the market.

By leveraging these forecasts, stakeholders in the car industry can make informed decisions about production, marketing, and inventory management. This enables them to meet consumer demand effectively and stay ahead of the competition. Our analysis provides valuable insights that can drive strategic planning and foster success in the dynamic car market.

APPENDIX: R Code

```
data = read.csv('eng_car_reg.csv', header=TRUE) #reading the data
ts_data = ts(data$no_new_regs,frequency = 4,start = c(2001,1)) #setting data as time series data
ts_data
plot(ts_data,main="Time plot", xlab="year", ylab="New car registrations") # time plot
acf(ts_data, lag.max = 100) #ACF plot
pacf(ts_data)   #PACF plot
Seasonal_diff=diff(ts_data,lag=4)  # performing seasonal difference with period 4
plot(Seasonal_diff,main="Time plot")
acf(Seasonal_diff,ylim=c(-1,1))
pacf(Seasonal_diff,ylim=c(-1,1))
library(tseries)
adf_test <- adf.test(Seasonal_diff)  #checking stationarity using ADF test function by importing tseries
package
adf_test
if ((adf_test$p.value)<0.05)
  print(paste("as p-value ",adf_test$p.value," is less than 0.05, hence rejecting the null hypothesis. Therefore,
it is stationary time series " ))else
    print("Non stationary time series")
#Fitting model
arima_model1=arima(ts_data,order=c(0,0,1),seasonal=list(order=c(0,1,1),period=4),method="ML")
arima_model1
library(forecast)
resid_m1=resid(arima_model1)  #checking residuals using forecast package
checkresiduals(arima_model1)
#Fitting model
arima_model2=arima(ts_data,order=c(1,0,1),seasonal=list(order=c(0,1,1),period=4),method="ML")
arima_model2
resid_m2=resid(arima_model2)
checkresiduals(arima_model2) # checking residuals
#ljung box test for ARIMA(1,0,1)(0,1,1)[4]
LB_test <- function(resid, max.k, p, d, q, P, D, Q, m) {
  lb_result <- list()
  df <- list()
  p_value <- list()

  for (i in 1:max.k) {
    lb_result[[i]] <- Box.test(resid, lag = i, type = "Ljung-Box", fitdf = (p + d + q + P + D + Q))
    df[[i]] <- lb_result[[i]]$parameter
    p_value[[i]] <- lb_result[[i]]$p.value
  }
  df <- as.vector(unlist(df))
  p_value <- as.vector(unlist(p_value))
  test_output <- data.frame(df, p_value)
  names(test_output) <- c("deg_freedom", "LB_p_value")
```

```
  return(test_output)
}
ARIMA_LB <- LB_test(resid_m2, max.k = 20, p = 1, d = 0, q = 1, P = 0, D = 1, Q = 1, m = 4)
plot(ARIMA_LB$deg_freedom, ARIMA_LB$LB_p_value, xlab = "Degrees of Freedom", ylab = "P-value",
     main = "Ljung-Box Test P-values for ARIMA(1,0,1)(0,1,1)[4] ", ylim = c(0, 1))
abline(h = 0.05, col = "blue", lty = 2)
#fitting model
arima_model3=arima(ts_data,order=c(2,0,1),seasonal=list(order=c(0,1,1),period=4),method="ML")
arima_model3
library(forecast)
checkresiduals(arima_model3) #checking residuals
resid_m3=residuals(arima_model3)
#ljung box test for ARIMA(2,0,1)(0,1,1)[4]
LB_test <- function(resid, max.k, p, d, q, P, D, Q, m) {
  lb_result <- list()
  df <- list()
  p_value <- list()

  for (i in 1:max.k) {
    lb_result[[i]] <- Box.test(resid, lag = i, type = "Ljung-Box", fitdf = (p + d + q + P + D + Q))
    df[[i]] <- lb_result[[i]]$parameter
    p_value[[i]] <- lb_result[[i]]$p.value
  }

  df <- as.vector(unlist(df))
  p_value <- as.vector(unlist(p_value))
  test_output <- data.frame(df, p_value)
  names(test_output) <- c("deg_freedom", "LB_p_value")

  return(test_output)
}

ARIMA_LB <- LB_test(resid_m3, max.k = 20, p = 2, d = 0, q = 1, P = 0, D = 1, Q = 1, m = 4)

plot(ARIMA_LB$deg_freedom, ARIMA_LB$LB_p_value, xlab = "Degrees of Freedom", ylab = "P-value",
     main = "Ljung-Box Test P-values for ARIMA(2,0,1)(0,1,1)[4]", ylim = c(0, 1))
abline(h = 0.05, col = "blue", lty = 2)
# hypothesis testing ARMA(2,1)
# Coefficients from the ARIMA(2,0,1)(0,1,1)[4] model
ar_coef = 0.0151
ar_coef_se = 0.1799



ma_coef = -0.529
ma_coef_se = 0.159

# Calculate the test statistic for the AR coefficient
```

```
test_stat_ar = abs(ar_coef / ar_coef_se)
test_stat_ar

# Calculate the test statistic for the MA coefficient

test_stat_ma = abs(ma_coef / ma_coef_se)
test_stat_ma
# hypothesis testing ARMA(1,1)

# Coefficients from the ARIMA(1,0,1)(0,1,1)[4] model
ar_coef = 0.9845
ar_coef_se = 0.0548



ma_coef <- -0.5401
ma_coef_se <- 0.0969

# Calculate the test statistic for the AR coefficient

test_stat_ar = abs(ar_coef / ar_coef_se)
test_stat_ar

# Calculate the test statistic for the MA coefficient

test_stat_ma = abs(ma_coef / ma_coef_se)
test_stat_ma
#forecasting using forecast library
library(forecast)
forecast_res=forecast(arima_model2,h=4)
print(forecast_res$mean)
autoplot(forecast_res,ylab="Sales")
```