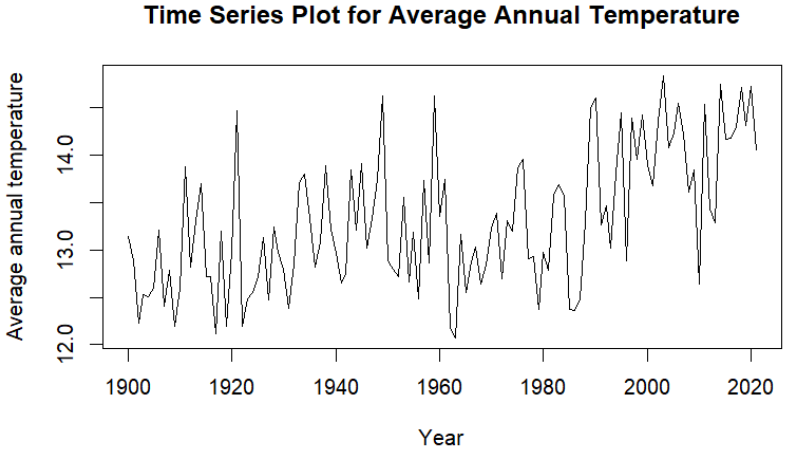
QUESTION 1

INTRODUCTION

This report analyses the temperature data from the Midlands region of England. It covers the yearly average temperatures recorded from 1900 to 2021 in degree Celsius. The objective of this analysis is to model the temperature time series and provide insights into its behavior over time.

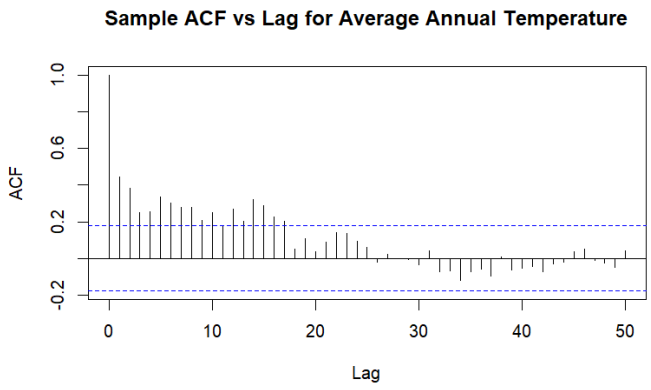
DATA EXPLORATION

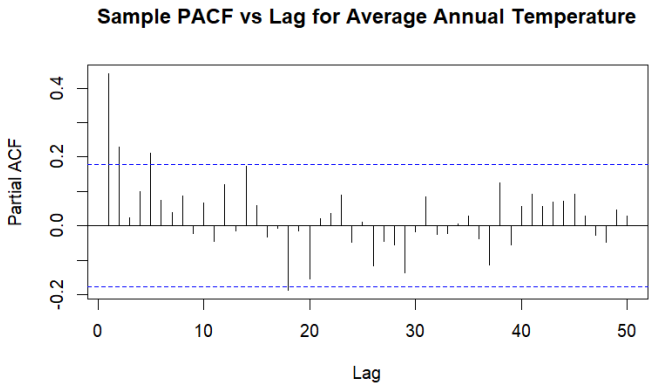
We begin by loading and visualizing the data using a time series plot. The Autocorrelation (ACF) and Partial Autocorrelation (PACF) are also plotted to understand the temporal dependencies present in the data.



The timeseries plot of the data does not appear to be stationary. It shows variation over time. We can see that the mean of the series is higher in the later years (for example 1990 - 2020), than in the beginning of the plot(1900 - 1990). There is an upward trend in the plot.

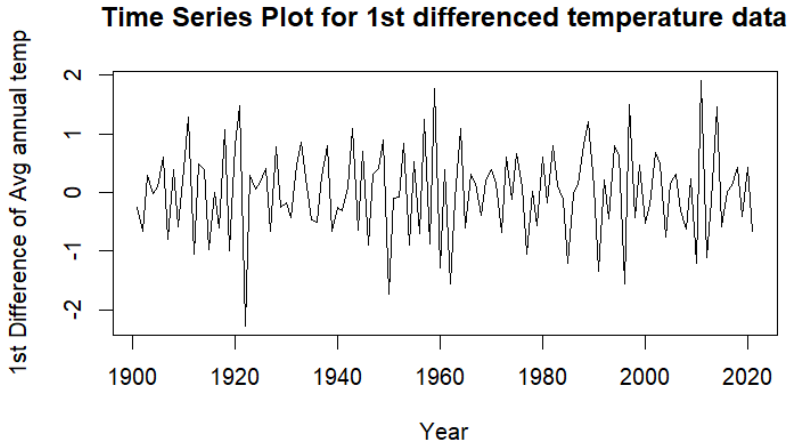
The sample ACF plot shows that the ACF value decreases initially, however, it does not decline to zero very rapidly. The sample PACF plot does not provide any extra information, compared to the timeseries and sample ACF plot.



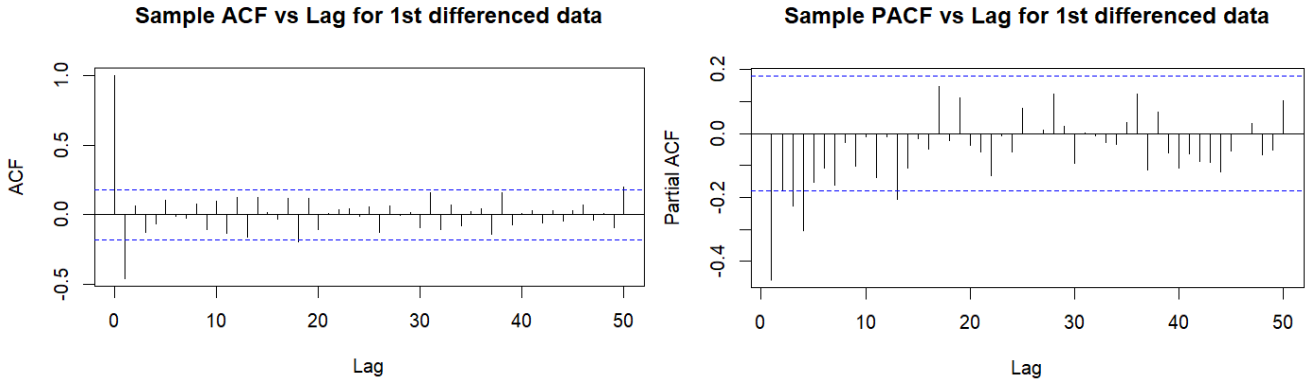


REMOVAL OF NON-STATIONARITY

To address the non-stationarity observed in the data, we take first difference of the time series. This helps to achieve (weak) stationarity in the data. The timeseries plot shows a constant mean at 0 and appears to have constant variability over time.



The sample ACF has a sudden drop to 0 after lag 1. This shows that the timeseries has an MA component of order 1. The sample PACF shows a gradual decline to 0. Therefore, we assume that there is no AR component present.



ARIMA MODEL FOR THE TIMESERIES

As described above, the ACF and PACF plots suggest that the first differenced data has an MA(1) component and no AR component. Therefore, we fit an MA(1) model to the differenced time series.

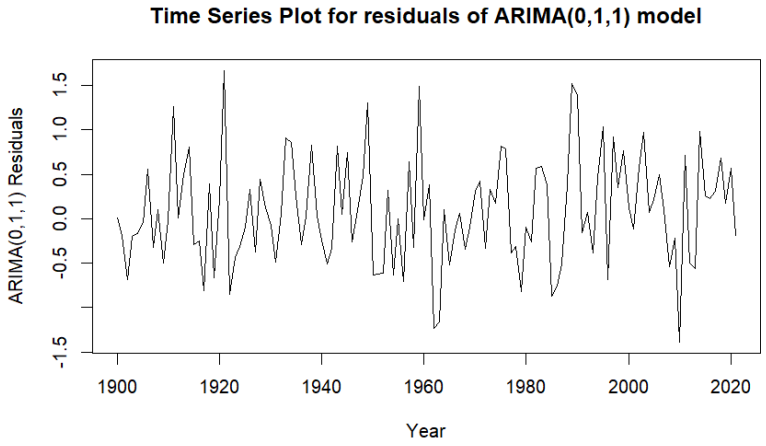
The parameter estimates for θ1 and sigma^2 are -0.8495 and 0.3654 respectively.

We get a negative log likelihood value of -111.43 and an AIC value of 226.86.

```
Call:
arima(x = cet_temp, order = c(0, 1, 1), method = "ML")

Coefficients:
          ma1
      -0.8495
s.e.   0.0480

sigma^2 estimated as 0.3654:  log likelihood = -111.43,
aic = 226.86
```

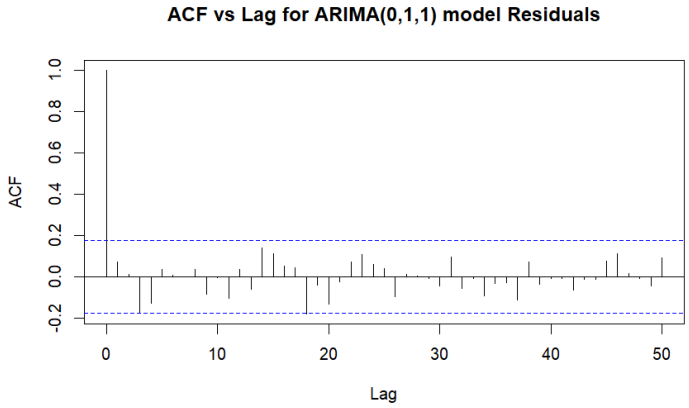
MODEL EVALUATION

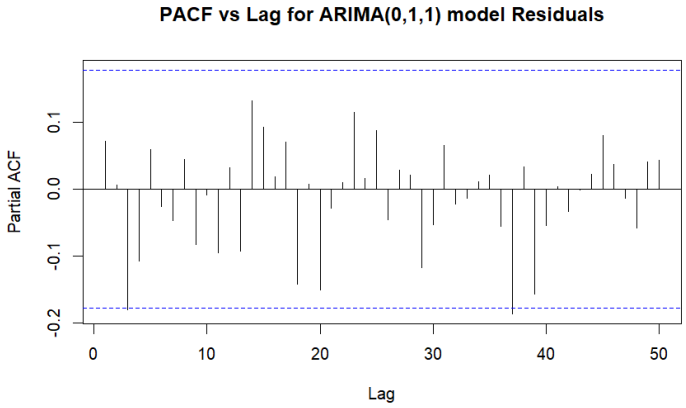
We assess the goodness of fit of the above ARIMA(0,1,1) model by observing the residuals of the model.



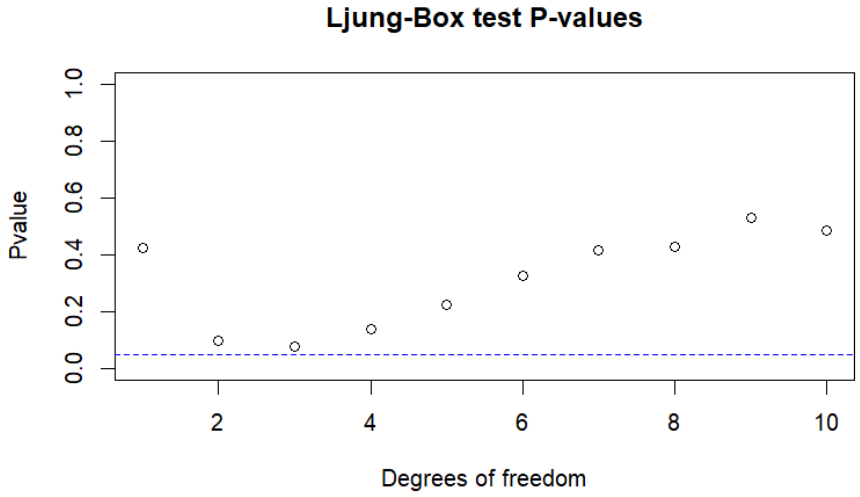
The model residuals appear to be white noise. The timeseries plot shows constant mean 0 and a constant variance for the residuals.

The ACF is also 0 (or close to 0) for lag >= 1. Both ACF and PACF have values in the range (-2/√n, 2/√n) This suggests that the residuals are independent.





The Ljung-Box test is applied to see how well the model fits the data. The p-values are plotted against the degrees of freedom. All the p-values are greater than 0.05 (5% significance). Therefore, we conclude that the model ARIMA(0, 1, 1) is a good fit for the data.



OTHER MODELS

In order to check if addition of further parameters improves the fit, we consider some other ARIMA models.

The models ARIMA(0, 1, 2) and ARIMA(1, 1, 1) are fitted to the data to check if they better describes the data.

```
Call:
arima(x = cet_temp, order = c(0, 1, 2), method = "ML")

Coefficients:
          ma1      ma2
      -0.7726  -0.0847
s.e.   0.0848   0.0807

sigma^2 estimated as 0.3622:  log likelihood = -110.89,
aic = 227.77
```

There is a slight increase in AIC from 226.86 to 227.77 for ARIMA(0, 1, 2) compared to the ARIMA(0,1,1) model. Therefore, we prefer the model MA(0,1,1) over MA(0,1,2).

The following hypothesis test is performed.

H0 : θ2 = 0 versus H1 : θ2 != 0

The test statistic is |− 0.0847/0.0807| = 1.05 < 2. Therefore we retain H0 at 5% significance level and conclude that the ARIMA(0, 1, 1) model is preferred over ARIMA(0, 1, 2).

Next, we consider the model ARIMA(1, 1, 1).

```
Call:
arima(x = cet_temp, order = c(1, 1, 1), method = "ML")

Coefficients:
          ar1      ma1
      0.1137  -0.8749
s.e.  0.1026   0.0454

sigma^2 estimated as 0.3618:  log likelihood = -110.81,
aic = 227.63
```

There is an increase in AIC from 226.86 to 227.63 for ARIMA(1, 1, 1) compared to the ARIMA(0, 1, 1) model. Therefore, we prefer the model ARIMA(0, 1, 1) over ARIMA(1, 1, 1).

We perform the following hypothesis test.

H0 : φ1 = 0 versus H1 : φ1 != 0

The test statistic is |0.1137/0.1026| = 1.11 < 2. Therefore we retain H0 at 5% significance level and conclude that the ARIMA(0, 1, 1) model is the most appropriate model for the data

EQUATION FOR THE FINAL MODEL

Let Xt is the original timeseries, Wt is the 1st differenced timeseries and Zt is white noise for the ARIMA(0, 1, 1) model. The value for θ1 was found to be -0.8495 as described earlier.

$$W_t = (1 + \theta_1 B) Z_t$$
$$(1 - B) X_t = (1 + \theta_1 B) Z_t$$
$$(1 - B) X_t = (1 - 0.8495 B) Z_t$$
$$X_t - X_{t-1} = Z_t - 0.8495 Z_{t-1}$$

CONCLUSION

The ARIMA (0, 1, 1) model provides a suitable description of the annual mean temperature in the Midlands region of England. The model captures the temporal dependencies present in the data and adequately accounts for the observed trend in the data.

APPENDIX: R CODE

```
# Read the data

cet <- read.csv('cet_temp.csv')


# Convert it into a timeseries with

  # start = 1900

  # frequency = 1 (yearly data)

cet_temp <- ts(cet$avg_annual_temp_C, start = 1900, frequency = 1)


# Plot the timeseries

ts.plot(cet_temp, gpars = list(main = "Time Series Plot for Average
Annual Temperature", xlab = "Year", ylab = "Average annual
temperature"))


# Plot ACF vs Lag

acf(cet_temp, lag.max = 50, main = "Sample ACF vs Lag for Average
Annual Temperature")


# Plot PACF vs Lag

pacf(cet_temp, lag.max = 50,main = "Sample PACF vs Lag for Average
Annual Temperature")

# Take first difference of the timeseries ( Wt = (1-B)Xt )

temp_diff <- diff(cet_temp)


# Plot the timeseries

ts.plot(temp_diff, gpars = list(main = "Time Series Plot for 1st
differenced temperature data", xlab = "Year", ylab = "1st Difference
of Avg annual temp"))
```

```
# Plot the ACF vs Lag

acf(temp_diff, lag.max = 50,main = "Sample ACF vs Lag for 1st
differenced data")


# Plot the PACF vs Lag

pacf(temp_diff, lag.max = 50,main = "Sample PACF vs Lag for 1st
differenced data")


#Code to fit MA(1) model to the first difference of the average annual
temperature

#p = 0 Order of the AR part of the model

#d = 1 Order of differencing (First differenced data)

#q = 1 Order of the MA part of the model


model.MA1<-arima(cet_temp, order=c(0,1,1), method="ML")

model.MA1


# Extract the residuals of the model

resid.MA1<-residuals(model.MA1)

# Plot the residuals

ts.plot(resid.MA1, gpars = list(main = "Time Series Plot for residuals
of ARIMA(0,1,1) model", xlab = "Year", ylab = "ARIMA(0,1,1)
Residuals"))


# Plot ACF for the residuals

acf(resid.MA1, lag.max = 50, main = "ACF vs Lag for ARIMA(0,1,1) model
Residuals")


# Plot PACF for the residuals

pacf(resid.MA1, lag.max = 50, main = "PACF vs Lag for ARIMA(0,1,1)
model Residuals")
```

```
#Function to produce P-values for the Ljung-Box test for different
lags

#where an ARMA(p,q) model has been fitted.

#Note that k must be > p+q

#Number of degrees of freedom for the test = k-p-q


#Arguments for the function "LB_test"

#resid = residuals from a fitted ARMA(p,q) model.


#max.k = the maximum value of k at which we perform the test

#Note that the minimum k is set at p+q+1 (corresponding to a test with
one degree of freedom)


#p = Order of the AR part of the model

#q = Order of the MA part of the model


#The function returns a table with one column showing the number of
degrees of freedom for the test and the other the associated P-value.


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)
}

#Since p+q=1, we run the following command to perform the first ten
#Ljung-Box tests for the model residuals (max.k=11)
MA1.LB<-LB_test(resid.MA1,max.k=11,p=0,q=1)
#To see the table of P-values
MA1.LB
#To produce a plot of the P-values against the degrees of freedom and
#add a blue dashed line at 0.05, we run the commands
plot(MA1.LB$deg_freedom,MA1.LB$LB_p_value,xlab="Degrees of
freedom",ylab="Pvalue",main="Ljung-Box test P-values",ylim=c(0,1))
abline(h=0.05,col="blue",lty=2)

#Code to fit MA(2) model to the 1st difference of the average annual
temperature
#p = 0 Order of the AR part of the model
#d = 1 Order of differencing (First differenced data)
#q = 2 Order of the MA part of the model

model.MA2<-arima(cet_temp, order=c(0,1,2), method="ML")
model.MA2

#Code to fit ARIMA(1, 1, 1) model to the average annual temperature
#p = 1 Order of the AR part of the model
#d = 1 Order of differencing (First differenced data)
#q = 1 Order of the MA part of the model

model.ARMA11<-arima(cet_temp, order=c(1,1,1), method="ML")
model.ARMA11
```

QUESTION 2

EXECUTIVE SUMMARY

This report presents the findings of a comprehensive time series analysis aimed at forecasting the monthly average house prices in the East Midlands region for the first six months of 2020.

Key Findings

1. **Data Analysis**
   The average house price data spanning from January 2010 to December 2019 was analyzed to uncover underlying patterns and trends. The analysis revealed a clear upward trend in prices and a seasonality with a 12 month period.

2. **Model Selection**
   Various time series models were explored and after rigorous evaluation, ARIMA(1,1,2)x$(0, 1, 1)_{12}$ was identified as the most suitable model for capturing the data dynamics effectively.

3. **Model Evaluation**
   The selected model demonstrated a strong fit to the data, with white noise residuals indicating randomness. The visual inspection of the timeseries plot, ACF plot and PACF plot of the model residuals along with the Ljung-Box test, affirmed the adequacy of the model.

4. **Forecasting**
   Utilizing the ARIMA(1,1,2)x$(0, 1, 1)_{12}$ model, the average house prices for January 2020 to June 2020 were forecasted. The predicted values exhibit a consistent trend, with prediction intervals providing insights into the uncertainty surrounding the forecasts at 80% and 95% confidence levels.

By leveraging advanced time series analysis techniques, reliable forecasts for the monthly average house prices in the East Midlands region for the first half of 2020 have been provided. These forecasts can serve as valuable insights for stakeholders in the real estate industry, policymakers, and potential homebuyers, aiding informed decision-making in a dynamic market environment.

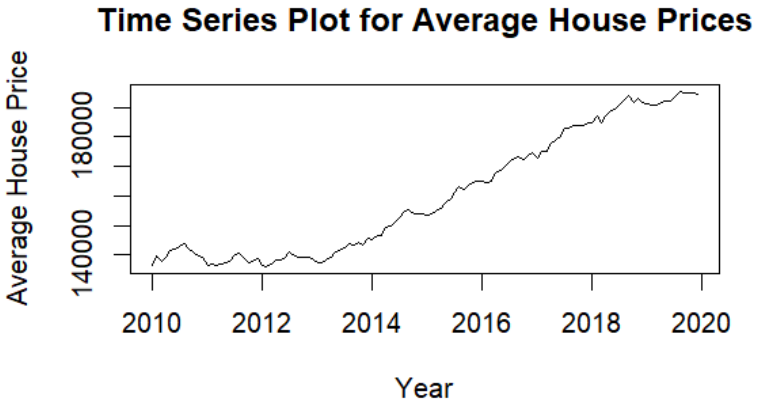
INTRODUCTION

This report analyses the monthly average house prices in the East Midlands region from January 2010 to December 2020. The objective is to model the timeseries for the house prices, provide insights into the behavior over time and forecast the mean house prices for the first half of 2020.

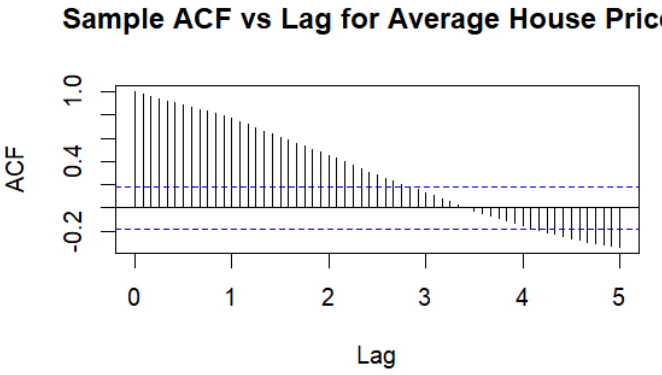
DATA EXPLORATION

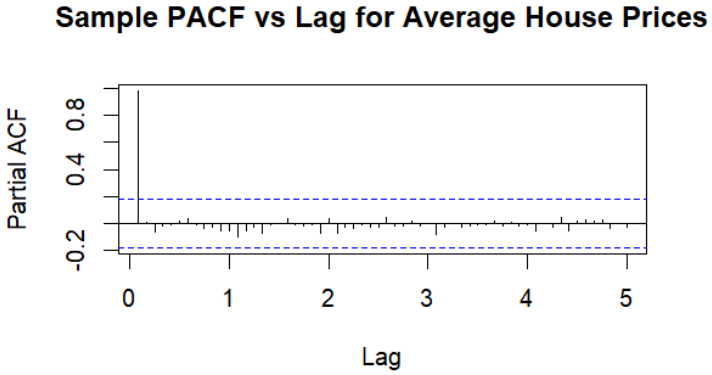
We begin by loading and visualizing the data using a time series plot. The Autocorrelation (ACF) and Partial Autocorrelation (PACF) are also plotted to understand the temporal dependencies present in the data.

The timeseries plot does not appear to be stationary. We can clearly see that the mean of the data is higher in the later years, than in the beginning of the plot. This shows an upward trend in the timeseries.



The sample ACF is decreasing gradually. However, it does not become zero. Instead it keeps on decreasing until the ACF values eventually become less than $-2/\sqrt{n}$. This verifies the non-stationarity of the data.

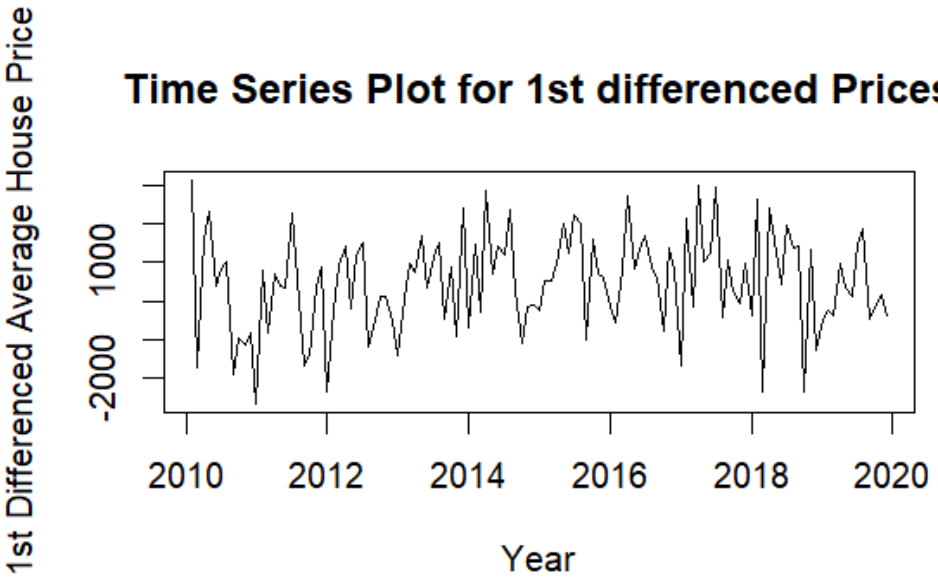




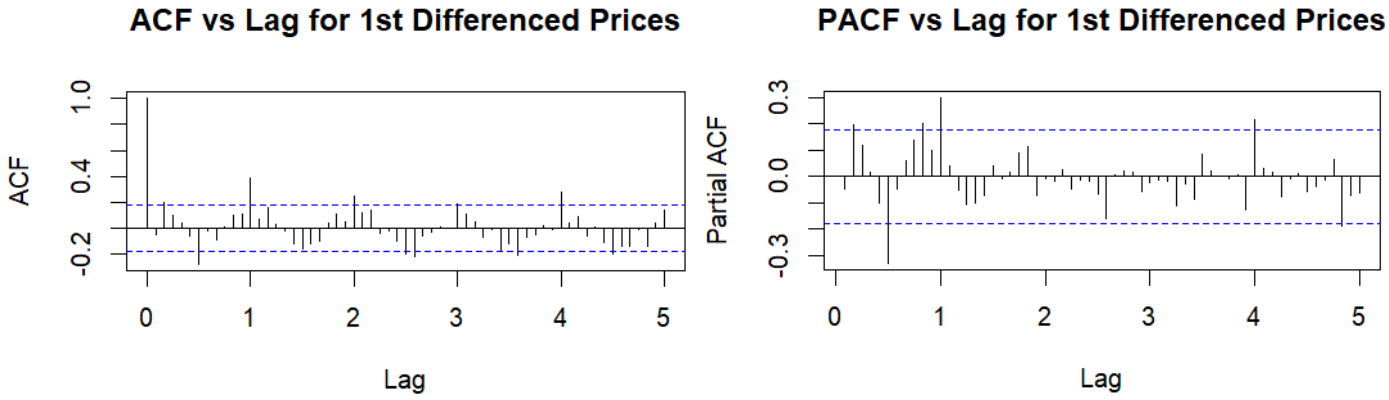
REMOVAL OF TREND

To address the non-stationarity in the data, we take first difference of the time series in order to remove the upward trend.

The first differenced timeseries appears to have some seasonality. Since, the time series consists of monthly data, we can see that a similar pattern is repeated after every 12 months. Therefore, we assume that a seasonality with period 12 exists in the first differenced timeseries.



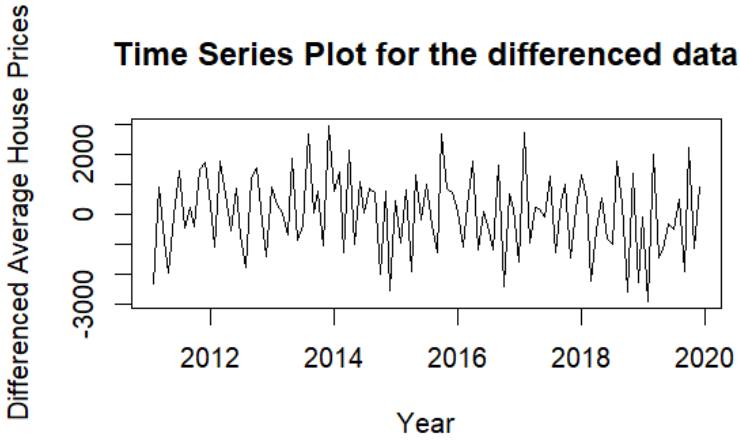
The ACF and PACF plots also follow a cyclic pattern affirming the seasonality present in the data.



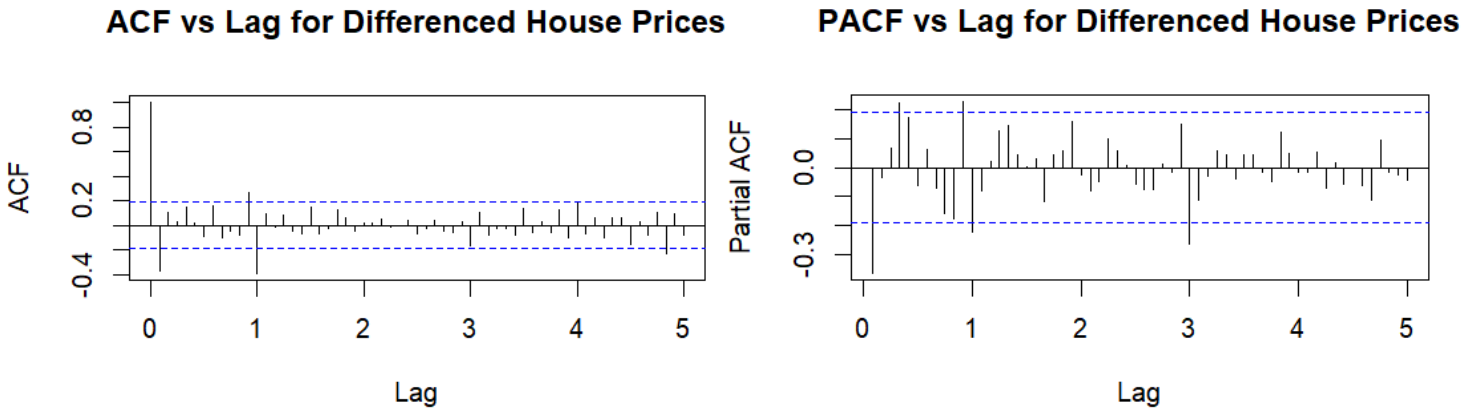
REMOVAL OF SEASONALITY

Since, the timeseries has seasonality with a period 12, we take a difference with lag 12 in order to remove the seasonality.

The timeseries plot of the differenced data shows that it is (weakly stationary). The mean appears to have an almost constant mean 0 and a constant variability over time.



From the ACF and PACF plots, we can see that the ACF and PACF values decay to zero quickly. This shows the stationary behavior of the differenced timeseries.



SELECTION AND EVALUATION OF MODEL

From the above ACF plot, we observe that the ACF value suddenly declines to 0 after lag 1. Therefore, we assume that the non-seasonal part of the timeseries has a Moving Average component of order 1.

We can also see a spike at lag of 1 year (12 months). This means that the seasonal part of the timeseries also contains an MA component. Since the ACF drops to almost zero (within the range $(-2/\sqrt{n}, 2/\sqrt{n})$) after one spike at 1, we assume that the seasonal part has an MA component of order 1.

MODEL 1: ARIMA (0, 1, 1) × $(0, 1, 1)_{12}$

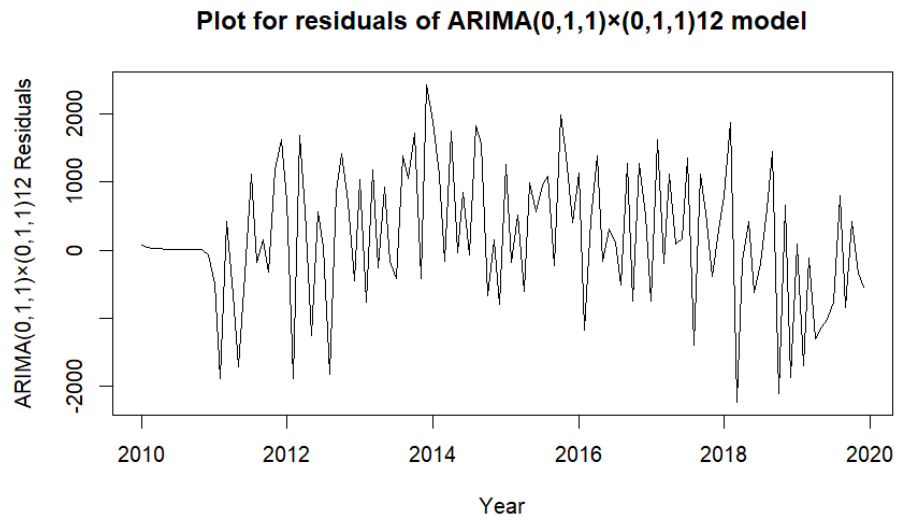
We begin by fitting an ARIMA (0, 1, 1) × $(0, 1, 1)_{12}$ model on the average prices data. Here the order of differencing is 1 for both seasonal and non-seasonal parts in order to remove the trend and seasonality and make the timeseries stationary.

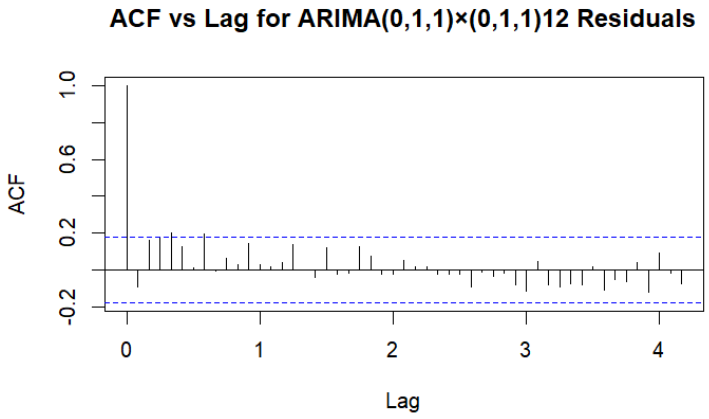
The estimates for θ1(non-seasonal), Θ1(seasonal part) and sigma^2 are -0.1808, -0.6961 and 1131137 respectively.

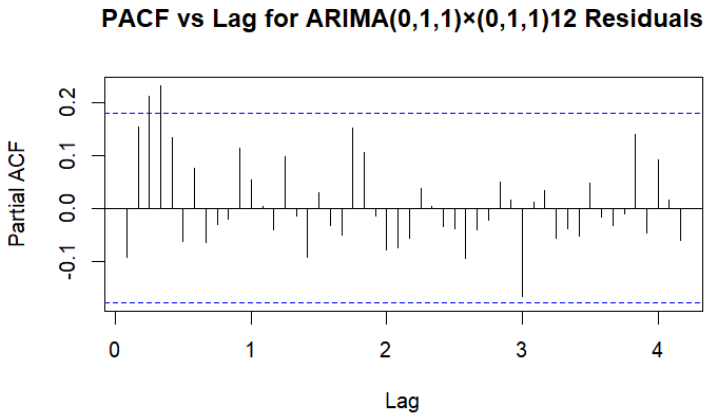
We have a negative log likelihood value of -901.54 and an AIC value of 1809.08

```
Call:
arima(x = prices, order = c(0, 1, 1), seasonal =
list(order = c(0, 1, 1), period = 12),
    method = "ML")

Coefficients:
          ma1     sma1
      -0.1808  -0.6961
s.e.   0.0814   0.1350

sigma^2 estimated as 1131137:  log likelihood = -901.54,
aic = 1809.08
```

We plot the timeseries, ACF and PACF for the model residuals to check if they are white noise.







The timeseries plot shows that the residuals are not white noise. We can see that the mean is high in the middle years and less in the start and end years. This shows that the residuals still have some pattern and are not independent.

The ACF values are in the range ($-2/\sqrt{n}$, $2/\sqrt{n}$). Therefore, we will not add any more MA component for now.

The PACF plot has significant spikes at lag 3 and 4. After lag 4 it is within the range ($-2/\sqrt{n}$, $2/\sqrt{n}$). We can try adding a non-seasonal AR component of order 4. There are no significant spikes after seasonal lags. Therefore, we assume that there is no AR component in the seasonal part of the timeseries.

MODEL 2: ARIMA (4, 1, 1) × (0, 1, 1)$_{12}$

We fit an ARIMA (4, 1, 1) × (0, 1, 1)$_{12}$ model on the data. The AIC value for the model is 1794.92.

```
Call:
arima(x = prices, order = c(4, 1, 1), seasonal =
list(order = c(0, 1, 1), period = 12),
    method = "ML")

Coefficients:
         ar1     ar2     ar3     ar4      ma1     sma1
      0.1710  0.2498  0.2201  0.1750  -0.5282  -0.8325
s.e.  0.2078  0.1114  0.1061  0.1294   0.1972   0.1429

sigma^2 estimated as 869901:  log likelihood = -890.46,
aic = 1794.92
```

Perform a hypothesis test for φ4

H0 : φ4 = 0 versus H1 : φ4 != 0

The test statistic is |0.1750/0.1294| = 1.35 < 2. Therefore we retain H0 at 5% significance level and conclude that φ4 = 0.

Therefore, we try fitting the model ARIMA (3, 1, 1) × (0, 1, 1)$_{12}$

MODEL 3: ARIMA (3, 1, 1) × (0, 1, 1)$_{12}$

The AIC is slightly decreased to 1794.4. However, for the following hypothesis test:

```
Call:
arima(x = prices, order = c(3, 1, 1), seasonal =
list(order = c(0, 1, 1), period = 12),
    method = "ML")

Coefficients:
         ar1     ar2     ar3      ma1     sma1
      0.3700  0.3296  0.2042  -0.7155  -0.8570
s.e.  0.1515  0.1023  0.1147   0.1286   0.1597

sigma^2 estimated as 873017:  log likelihood = -891.2,
aic = 1794.4
```

H0 : φ3 = 0 versus H1 : φ3 != 0

The test statistic is |0.2042/0.1147| = 1.78 < 2. Therefore we retain H0 at 5% significance level and conclude that φ3 = 0.

MODEL 4: ARIMA (2, 1, 1) × (0, 1, 1)$_{12}$

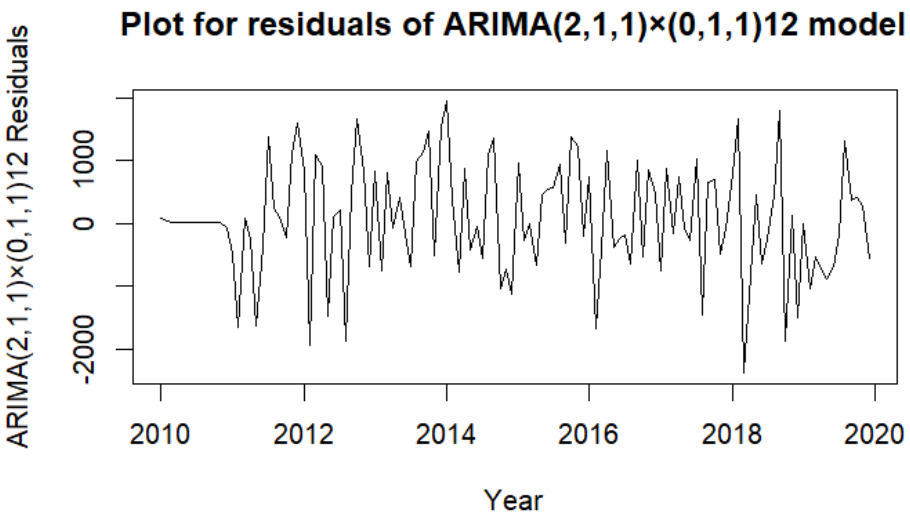
The AIC has increased slightly. However, an unnecessary parameter has been removed. Therefore, we prefer this ARIMA (2, 1, 1) × (0, 1, 1)12 model over the previous models.

```
Call:
arima(x = prices, order = c(2, 1, 1), seasonal =
list(order = c(0, 1, 1), period = 12),
    method = "ML")

Coefficients:
         ar1     ar2      ma1     sma1
      0.5362  0.4184  -0.8323  -0.8786
s.e.  0.1092  0.0935   0.0761   0.1949

sigma^2 estimated as 888206:  log likelihood = -892.62,
aic = 1795.24
```

The estimates for φ1, φ2, θ1(non-seasonal) and Θ1(seasonal part) are 0.5362, 0.4184, -0.8323 and -0.8786 respectively.

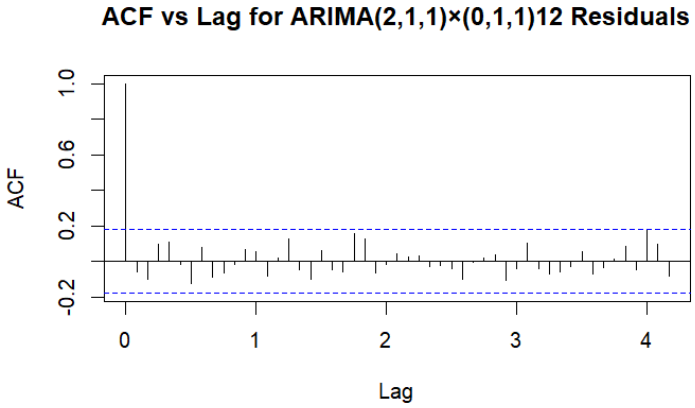
The sigma^2 estimate is 888206. The log likelihood is -892.62 and aic is 1795.24.

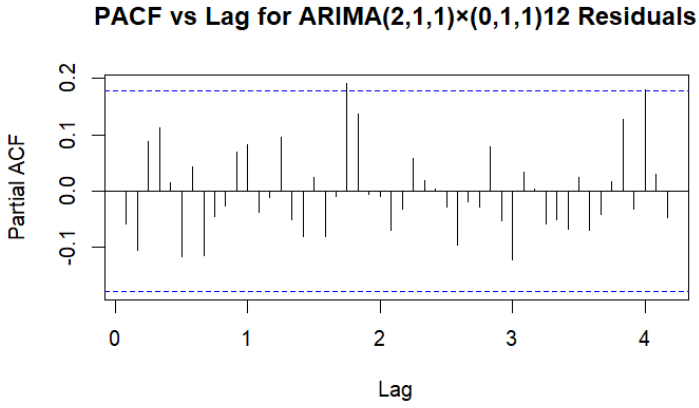
We assess the goodness of fit of the model by plotting the model residuals.



The timeseries plot shows that the residuals have a constant mean 0 and a constant variance. Therefore, the model residuals are stationary.

The ACF plot drops to 0 (or close to 0) for lag >= 1. This suggests that the residuals are independent.

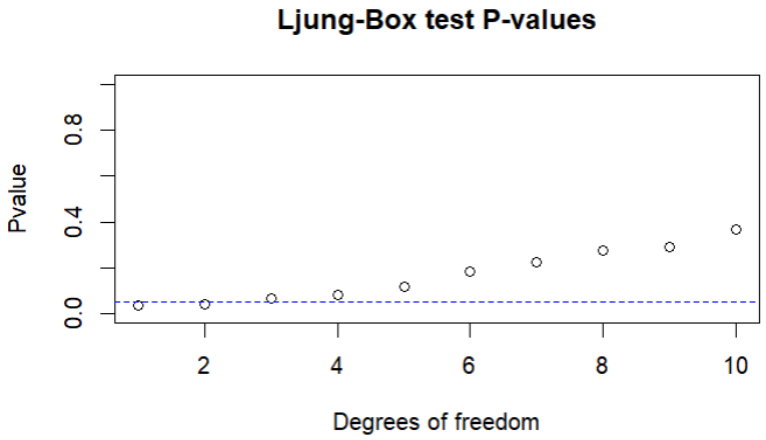




Next, we performing the Ljung-Box test to see how well the model fits the data. The plot of the p-values against the degrees of freedom is as follows.

We can see that the first 2 p-values are less than 0.05.

A slight change in the order might improve the model fit.



OTHER MODELS

The following models can be tried in order to improve the model fit.

Model 5: ARIMA $(2, 1, 2) \times (0, 1, 1)_{12}$,

Model 6: ARIMA $(1, 1, 2) \times (0, 1, 1)_{12}$,

Model 7: ARIMA $(2, 1, 1) \times (1, 1, 1)_{12}$,

Model 8: ARIMA $(2, 1, 1) \times (0, 1, 2)_{12}$

```
Call:
arima(x = prices, order = c(2, 1, 2), seasonal =
list(order = c(0, 1, 1), period = 12),
    method = "ML")

Coefficients:
         ar1     ar2      ma1     ma2     sma1
      0.7424  0.1849  -1.0930  0.2986  -0.8505
s.e.  0.2277  0.2798   0.2386  0.3284   0.1625

sigma^2 estimated as 873389:  log likelihood = -891.06,
aic = 1794.13
```

```
Call:
arima(x = prices, order = c(1, 1, 2), seasonal =
list(order = c(0, 1, 1), period = 12),
    method = "ML")

Coefficients:
        ar1      ma1     ma2     sma1
      0.855  -1.2235  0.5234  -0.8109
s.e.  0.093   0.0996  0.0901   0.1337

sigma^2 estimated as 874922:  log likelihood = -890.44,
aic = 1790.89
```

```
Call:
arima(x = prices, order = c(2, 1, 1), seasonal =
list(order = c(1, 1, 1), period = 12),
    method = "ML")

Coefficients:
         ar1     ar2      ma1     sar1     sma1
      0.5403  0.4163  -0.8349  0.0411  -0.9453
s.e.  0.1144  0.0941   0.0801  0.1571   0.6106

sigma^2 estimated as 852077:  log likelihood = -892.58,
aic = 1797.16
```

```
Call:
arima(x = prices, order = c(2, 1, 1), seasonal =
list(order = c(0, 1, 2), period = 12),
    method = "ML")

Coefficients:
         ar1     ar2      ma1     sma1     sma2
      0.5414  0.4160  -0.8356  -0.9183  -0.0473
s.e.  0.1154  0.0941   0.0810   0.9043   0.1883

sigma^2 estimated as 836839:  log likelihood = -892.57,
aic = 1797.14
```

Among these models ARIMA $(1, 1, 2) \times (0, 1, 1)_{12}$ has the lowest AIC of 1790.89. Therefore we prefer this model.

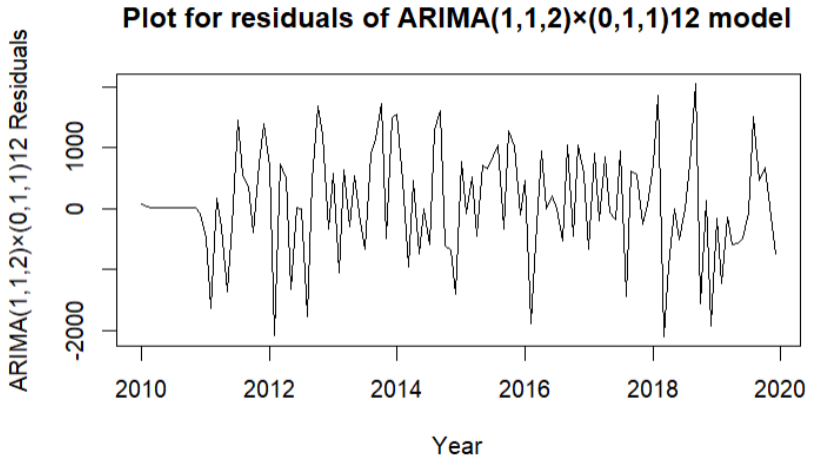
The estimates for φ1, θ1, θ2(non-seasonal) and Θ1(seasonal part) are 0.855, -1.2235, 0.5234 and -0.8109 respectively. The sigma^2 estimate is 874922. The log likelihood is -890.44.

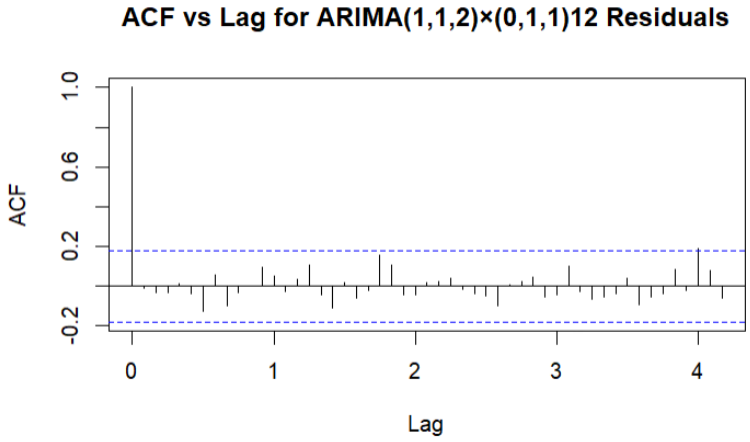
We plot the model residuals to check the goodness of fit.

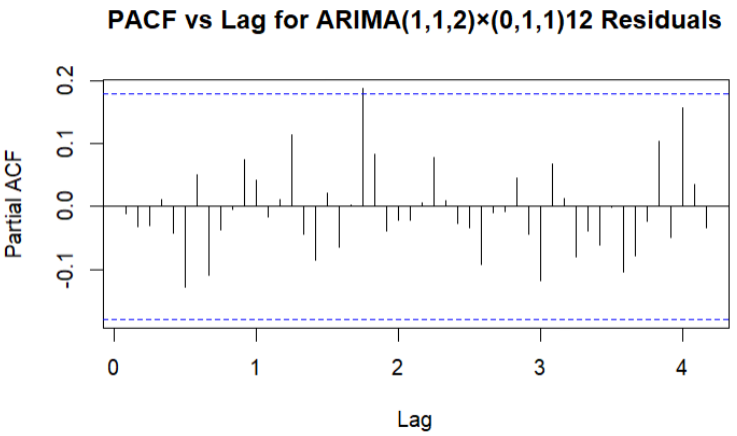
The residuals appear to be white noise.

The timeseries plot shows a constant mean 0 and a constant variance over time.

The ACF is also 0 (or close to 0) for lag>= 1. This suggests that the residuals are independent.



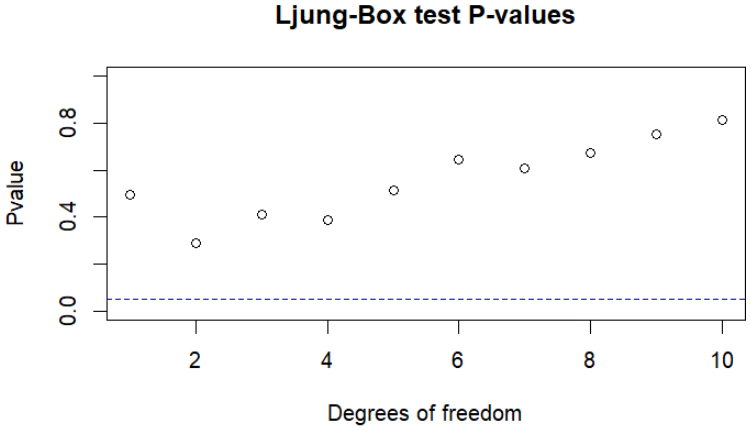




Next we produce a plot of the Ljung-Box test p-values to see how well the model fits the data.

All the p-values are greater than 0.05. This suggests that the ARIMA$(1,1,2)\times(0,1,1)_{12}$ is a good fit for the data.

Therefore, we choose the ARIMA$(1,1,2)\times(0,1,1)_{12}$ model as the most appropriate model for the Average House Prices data.



EQUATION FOR THE FINAL MODEL

Let X_t is the house prices timeseries and Zt is white noise. The equation for ARIMA(1,1,2)×(0,1,1)$_{12}$ is as follows.

$$\phi_p(B)\Phi_P(B^h)(1-B)^d(1-B^h)^D X_t = \theta_q(B)\Theta_Q(B^h)Z_t$$

Here,

$$\phi_p(B) = 1 - \phi_1 B - \phi_2 B^2 - \ldots - \phi_p B^p$$
$$\Phi_P(B^h) = 1 - \Phi_1 B^h - \Phi_2 B^{2h} - \ldots - \Phi_P B^{Ph}$$
$$\theta_q(B) = 1 + \theta_1 B + \theta_2 B^2 + \ldots + \theta_q B^q$$
$$\Theta_Q(B) = 1 + \Theta_1 B^h + \Theta_2 B^{2h} + \ldots + \Theta_Q B^{Qh}$$

We know that p = 1, d = 1, q = 2, P = 0, D = 1, Q = 1 and h = 12 for ARIMA(1,1,2)×(0,1,1)$_{12}$

The equation becomes,

$$\phi_1(B)\Phi_0(B^{12})(1-B)^1(1-B^{12})^1 X_t = \theta_2(B)\Theta_1(B^{12})Z_t$$
$$(1-\phi_1 B)(1)(1-B)(1-B^{12})X_t = (1+\theta_1 B+\theta_2 B^2)(1+\Theta_1 B^{12})Z_t$$

$$X_t - (1+\phi_1)X_{t-1} + \phi_1 X_{t-2} - X_{t-12} + (1+\phi_1)X_{t-13} + \phi_1 X_{t-14} =$$
$$Z_t + \theta_1 Z_{t-1} + \theta_2 Z_{t-2} + \Theta_1 Z_{t-12} + \theta_1\Theta_1 Z_{t-13} + \theta_2\Theta_1 Z_{t-14}$$

The estimates of the parameters are φ1 = 0.855, θ1 = -1.2235, θ2 = 0.5234 and Θ1 = -0.8109

Putting values in the equation,

$$X_t - 1.855X_{t-1} + 0.855X_{t-2} - X_{t-12} + 1.855X_{t-13} + 0.855X_{t-14} =$$
$$Z_t - 1.2235Z_{t-1} + 0.5234Z_{t-2} - 0.8109Z_{t-12} + 0.99Z_{t-13} - 0.424Z_{t-14}$$

PRICE FORECAST FOR JANUARY 2020 – JUNE 2020

The ARIMA(1,1,2)×(0,1,1)$_{12}$ was used to forecast the average house prices for the next 6 months. The predicted prices can be seen in the table below.

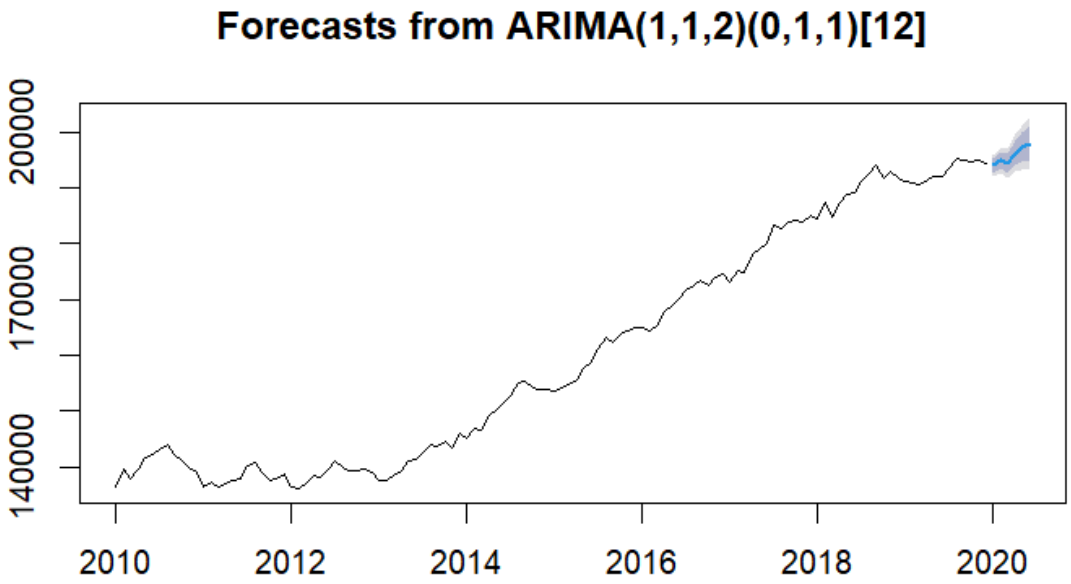
	Point Forecast <dbl>	Lo 80 <dbl>	Hi 80 <dbl>	Lo 95 <dbl>	Hi 95 <dbl>
Jan 2020	193930.5	192727.4	195133.6	192090.5	195770.5
Feb 2020	194836.4	193413.9	196258.9	192660.8	197012.0
Mar 2020	194401.0	192656.7	196145.3	191733.3	197068.7
Apr 2020	196204.1	194073.5	198334.7	192945.6	199462.5
May 2020	197202.1	194649.2	199754.9	193297.8	201106.3
Jun 2020	197933.9	194940.8	200926.9	193356.4	202511.4

The Point Forecast column contains the forecasted prices for Jan 2020 – June 2020.

The lower and upper bounds for the prediction intervals of these values at 80% and 95% confidence levels are also shown in the table. These confidence levels mean that there is 80% and 95% probability respectively that the house prices will fall within these bounds.

The 95% prediction interval has a wider range compared to the 80% interval, indicating higher uncertainty but with greater confidence.

The following plot shows the forecasted house prices for the first six months of 2020 along with the known prices for Jan 2010 - Dec 2019.



The black line represents the known prices, while blue line represents the forecasted prices. The grey shade on the forecasted values shows that there is some uncertainty in the predicted values.

The ligher shade of grey represents the 95% confidence level, while dark grey represents the 80% confidence level. It can clearly be seen that the 95% prediction interval is wider than the 80% interval.

CONCLUSION

The ARIMA$(1,1,2)\times(0,1,1)_{12}$ model provides reliable forecasts for monthly average house prices in the East Midlands region for the first half of 2020. The model captures the temporal dependencies present in the data and adequately accounts for the trend and seasonality in the timeseries. These forecasts can assist the local government agencies and stakeholders in making informed decisions.

APPENDIX: R CODE

```
# Read the data

prices <- read.csv('em_house_prices.csv')


#Convert it into timeseries with

  # start = 2010

  # frequency = 12 (monthly data)

prices <- ts(prices$average_price_gbp, start = 2010, frequency = 12)


# Plot the timeseries for average house prices

ts.plot(prices, gpars = list(main = "Time Series Plot for Average
House Prices", xlab = "Year", ylab = "Average House Price"))


# Plot ACF vs Lag

acf(prices, lag.max = 60, main = "Sample ACF vs Lag for Average House
Prices")


#Plot PACF vs Lag

pacf(prices, lag.max = 60, main = "Sample PACF vs Lag for Average
House Prices")


# Take first difference of the timeseries ( Wt = (1-B)Xt )

prices_diff <- diff(prices)


# Plot the timeseries for 1st differenced data

ts.plot(prices_diff, gpars = list(main = "Time Series Plot for 1st
differenced Prices", xlab = "Year", ylab = "1st Differenced Average
House Price"))


# Plot ACF vs Lag for 1st differenced data

acf(prices_diff, lag.max = 60, main = "ACF vs Lag for 1st Differenced
Prices")
```

```
# Plot PACF vs Lag for 1st differenced data

pacf(prices_diff, lag.max = 60, main = "PACF vs Lag for 1st
Differenced Prices")


# Take seasonal difference for lag = 12 (Yt = Wt - Wt-12)

prices_diff2 <- diff(prices_diff, lag = 12)


# Plot the timeseries for the differenced data

ts.plot(prices_diff2, gpars = list(main = "Time Series Plot for the
differenced data", xlab = "Year", ylab = "Differenced Average House
Prices"))


# Plot ACF vs Lag for the differenced data

acf(prices_diff2, lag.max =60, main = "ACF vs Lag for Differenced
House Prices")


# Plot PACF vs Lag for the differenced data

pacf(prices_diff2, lag.max = 60, main = "PACF vs Lag for Differenced
House Prices")


#Code to fit ARIMA (0, 1, 1) × (0, 1, 1)12 model to the Average House
Prices data


#p = 0 Order of the AR component of the non-seasonal part

#d = 1 Order of differencing for the non-seasonal part

#q = 1 Order of the MA component of the non-seasonal part


#P = 0 Order of the AR component of the seasonal part

#D = 1 Order of differencing for the seasonal part

#Q = 1 Order of the MA component of the seasonal part
```

```
model.SARMA0101<-arima(prices,order=c(0,1,1), seasonal = list(order =
c(0,1,1), period = 12), method="ML")

model.SARMA0101


# Extract the residuals of the model

resid.SARMA0101<-residuals(model.SARMA0101)

# Plot the residuals

ts.plot(resid.SARMA0101, gpars = list(main = "Plot for residuals of
ARIMA(0,1,1)×(0,1,1)12 model", xlab = "Year", ylab =
"ARIMA(0,1,1)×(0,1,1)12 Residuals"))


# Plot ACF for the residuals

acf(resid.SARMA0101, lag.max = 50, main = "ACF vs Lag for
ARIMA(0,1,1)×(0,1,1)12 Residuals")


# Plot PACF for the residuals

pacf(resid.SARMA0101, lag.max = 50, main = "PACF vs Lag for
ARIMA(0,1,1)×(0,1,1)12 Residuals")


#Code to fit ARIMA (4, 1, 1) × (0, 1, 1)12 model to the average house
prices data


#p = 4 Order of the AR component of the non-seasonal part

#d = 1 Order of differencing for the non-seasonal part

#q = 1 Order of the MA component of the non-seasonal part


#P = 0 Order of the AR component of the seasonal part

#D = 1 Order of differencing for the seasonal part

#Q = 1 Order of the MA component of the seasonal part


model.SARMA4101<-arima(prices,order=c(4,1,1), seasonal = list(order =
c(0,1,1), period = 12), method="ML")

model.SARMA4101
```

```
#Code to fit ARIMA (3, 1, 1) × (0, 1, 1)12 model to the average house
prices data


#p = 3 Order of the AR component of the non-seasonal part

#d = 1 Order of differencing for the non-seasonal part

#q = 1 Order of the MA component of the non-seasonal part


#P = 0 Order of the AR component of the seasonal part

#D = 1 Order of differencing for the seasonal part

#Q = 1 Order of the MA component of the seasonal part


model.SARMA3101<-arima(prices,order=c(3,1,1), seasonal = list(order =
c(0,1,1), period = 12), method="ML")

model.SARMA3101


#Code to fit ARIMA (2, 1, 1) × (0, 1, 1)12 model to the average house
prices data


#p = 2 Order of the AR component of the non-seasonal part

#d = 1 Order of differencing for the non-seasonal part

#q = 1 Order of the MA component of the non-seasonal part


#P = 0 Order of the AR component of the seasonal part

#D = 1 Order of differencing for the seasonal part

#Q = 1 Order of the MA component of the seasonal part


model.SARMA2101<-arima(prices,order=c(2,1,1), seasonal = list(order =
c(0,1,1), period = 12), method="ML")

model.SARMA2101


# Extract the residuals of the model

resid.SARMA2101<-residuals(model.SARMA2101)
```

```
# Plot the residuals

ts.plot(resid.SARMA2101, gpars = list(main = "Plot for residuals of
ARIMA(2,1,1)×(0,1,1)12 model", xlab = "Year", ylab =
"ARIMA(2,1,1)×(0,1,1)12 Residuals"))


# Plot ACF for the residuals

acf(resid.SARMA2101, lag.max = 50, main = "ACF vs Lag for
ARIMA(2,1,1)×(0,1,1)12 Residuals")


# Plot PACF for the residuals

pacf(resid.SARMA2101, lag.max = 50, main = "PACF vs Lag for
ARIMA(2,1,1)×(0,1,1)12 Residuals")


#Function to produce P-values for the Ljung-Box test for different
lags

#where an ARIMA(p,d,q)x(P,D,Q)_h model has been fitted.

#Note that k must be > p+q+P+Q

#Number of degrees of freedom for the test = k-p-q-P-Q


#Arguments for the function "LB_test"

#resid = residuals from a fitted ARIMA(p,d,q)x(P,D,Q)_h model


#max.k = the maximum value of k at which we perform the test

#Note that the minimum k is set at p+q+P+Q+1 (corresponding to a test
with one degree

#of freedom)


#p = Order of the non-seasonal AR part of the model

#q = Order of the non-seasonal MA part of the model

#P = Order of the seasonal AR part of the model

#Q = Order of the seasonal MA part of the model
```

```
#The function returns a table with one column showing the number of
degrees
#of freedom for the test and the other the associated P-value.


LB_test_SARIMA<-function(resid,max.k,p,q,P,Q){
  lb_result<-list()
  df<-list()
  p_value<-list()
  for(i in (p+q+P+Q+1):max.k){
    lb_result[[i]]<-Box.test(resid,lag=i,type=c("Ljung-
Box"),fitdf=(p+q+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)
}


#Ljung-Box tests for the model residuals (max.k=14)
MA1.LB<-LB_test_SARIMA(resid.SARMA2101,max.k=14,p=2,q=1, P=0, Q=1)
#To see the table of P-values, type
MA1.LB
#To produce a plot of the P-values against the degrees of freedom and
#add a blue dashed line at 0.05, we run the commands
plot(MA1.LB$deg_freedom,MA1.LB$LB_p_value,xlab="Degrees of
freedom",ylab="Pvalue",main="Ljung-Box test P-values",ylim=c(0,1))
abline(h=0.05,col="blue",lty=2)
```

```
# ARIMA (2, 1, 2) × (0, 1, 1)12

model.SARMA2201<-arima(prices,order=c(2,1,2), seasonal = list(order =
c(0,1,1), period = 12), method="ML")

model.SARMA2201


# ARIMA (1, 1, 2) × (0, 1, 1)12

model.SARMA1201<-arima(prices,order=c(1,1,2), seasonal = list(order =
c(0,1,1), period = 12), method="ML")

model.SARMA1201


# ARIMA (2, 1, 1) × (1, 1, 1)12

model.SARMA2111<-arima(prices,order=c(2,1,1), seasonal = list(order =
c(1,1,1), period = 12), method="ML")

model.SARMA2111


# ARIMA (2, 1, 1) × (0, 1, 2)12

model.SARMA2102<-arima(prices,order=c(2,1,1), seasonal = list(order =
c(0,1,2), period = 12), method="ML")

model.SARMA2102


# Extract the residuals of the model

resid.SARMA1201<-residuals(model.SARMA1201)

# Plot the residuals

ts.plot(resid.SARMA1201, gpars = list(main = "Plot for residuals of
ARIMA(1,1,2)×(0,1,1)12 model", xlab = "Year", ylab =
"ARIMA(1,1,2)×(0,1,1)12 Residuals"))


# Plot ACF for the residuals

acf(resid.SARMA1201, lag.max = 50, main = "ACF vs Lag for
ARIMA(1,1,2)×(0,1,1)12 Residuals")


# Plot PACF for the residuals
```

```
pacf(resid.SARMA1201, lag.max = 50, main = "PACF vs Lag for
ARIMA(1,1,2)×(0,1,1)12 Residuals")


#Ljung-Box tests for the model residuals (max.k=14)

MA1.LB<-LB_test_SARIMA(resid.SARMA1201,max.k=14,p=1,q=2, P=0, Q=1)

#To see the table of P-values, type

MA1.LB

#To produce a plot of the P-values against the degrees of freedom and

#add a blue dashed line at 0.05, we run the commands

plot(MA1.LB$deg_freedom,MA1.LB$LB_p_value,xlab="Degrees of
freedom",ylab="Pvalue",main="Ljung-Box test P-values",ylim=c(0,1))

abline(h=0.05,col="blue",lty=2)


#install.packages("forecast")

library(forecast)


# Forecast the values for next 6 months

forecast_values <- forecast(model.SARMA1201, h = 6)


# Plot the forecasted values

plot(forecast_values)


# Print the forecasted values

print(forecast_values)
```