IE360 PROJECT

AHMET ÇAPAR-2019402183

BURAK CAYMAZ-2019402003

EDA BİTARAF-2019402024

YUSUF UTKU GÜLDÜR-2019402198

Table of Contents

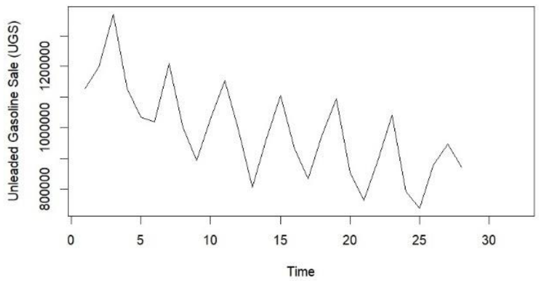
1. Introduction

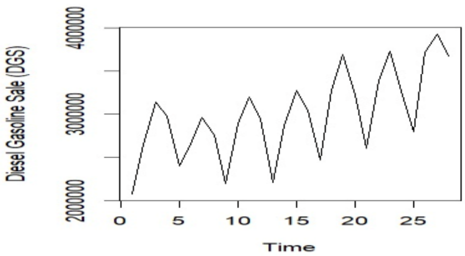
In this project, we made forecasts for a major distributor’s gasoline and diesel sales from previous years’ data. We used 2 models for that: (A) time series analysis and (B) regression.

2. Time Series Analysis

2.1 Stationarity

A stationary process is characterized by a series that appears flat without any trend, maintains a constant variance over time, exhibits a consistent autocorrelation structure, and lacks periodic fluctuations(seasonality).





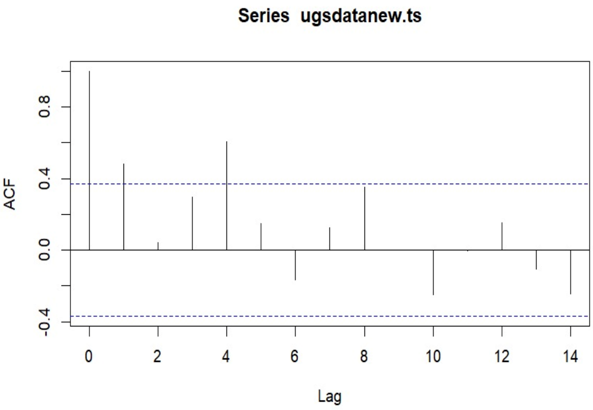
We can easily see that the mean is decreasing between periods. In the stationary model, mean should stay near constant level so we can say that the unleaded gasoline sale(UGS) is not stationary.

On the other hand, for the diesel gasoline sale(DGS) the mean is increasing between periods so DGS is not stationary as well.

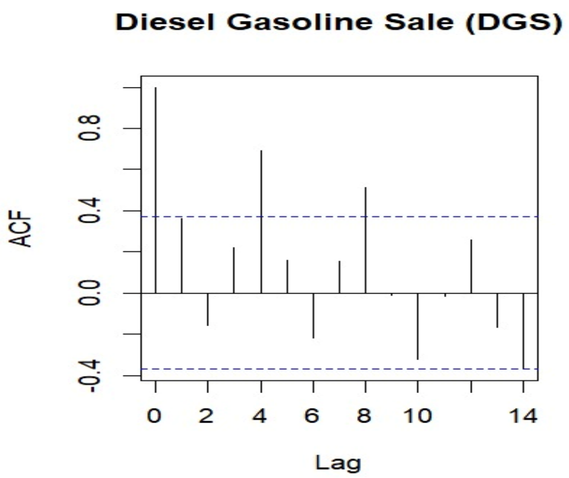
2.2 Autocorrelation Functions

In the presence of a trend in the data, the autocorrelations at short lags are typically positive and relatively high. This is due to the similarity in size between nearby observations in time. As the lag increases, the positive values of the autocorrelation slowly decrease.

On the other hand, in the case of seasonal data, the autocorrelations are larger for lags that correspond to multiples of the seasonal frequency. These specific lags exhibit stronger correlation compared to other lags.



For UGS, we can notice spikes at lag 4,8,12 from the Autocorrelation Function(ACF). This means that we should apply seasonal difference with S=4 and short lags are relatively high so we can say that we have a decreasing trend here. Also if we look at non-seasonal periods from ACF, we can see that it cut off after 1 lag and it makes us think MA(1) process.



In the case of DGS, we observe spikes in the autocorrelation function (ACF) at lags 4, 8, and 12. Consequently, applying a seasonal difference with a period of 4 (S=4) would be appropriate. Additionally, the presence of relatively high autocorrelations at short lags suggests the existence of an increasing trend, which aligns with the earlier observation that the mean of DGS increases over time.

2.3 Logarithmic Transformation

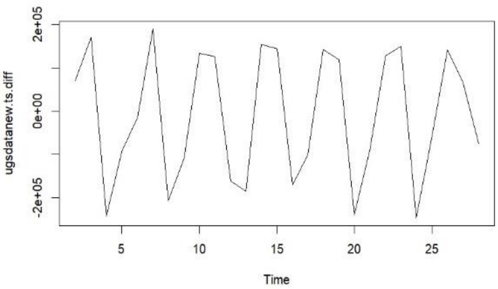
If the original continuous data deviates from a normal distribution, we can apply a log transformation to make it more closely resemble a normal distribution. This adjustment helps ensure that the statistical analysis conducted on the transformed data produces more reliable results by reducing or eliminating the skewness present in the original data. It's crucial to note that the log transformation is effective only when the original data follows or closely approximates a log-normal distribution. Otherwise, the log transformation may not be appropriate or effective. Since the variability is stable in both UGS and DGS, we don't need to use logarithmic transformation on our data.

2.4 Differencing

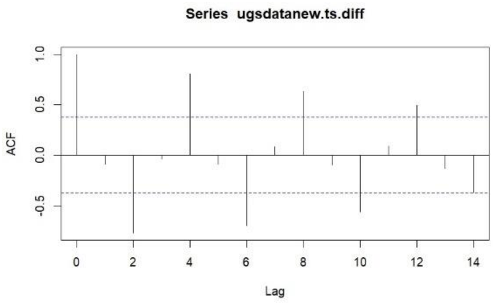
By employing differencing, we can stabilize the mean of a time series by eliminating or reducing changes in its level. As a result, differencing helps to eliminate or reduce both trend and seasonality in the time series. This part take up much space so seperating to UGS and DGS would be benefitial.

2.4.1 Differencing of UGS

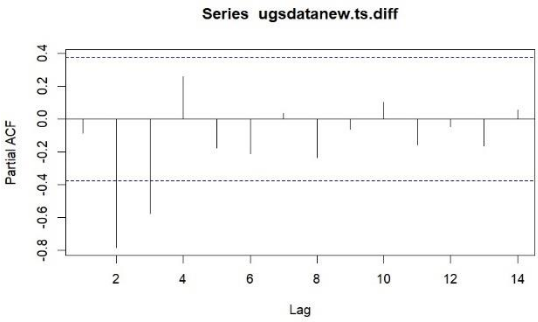
In the case of UGS, the time series is non-stationary, necessitating the initial step of taking regular differences.



After taking differences, the plot of the data seems stationary except seasonality(S=4). So the seasonal differencing should be applied to that data to make it stationary.

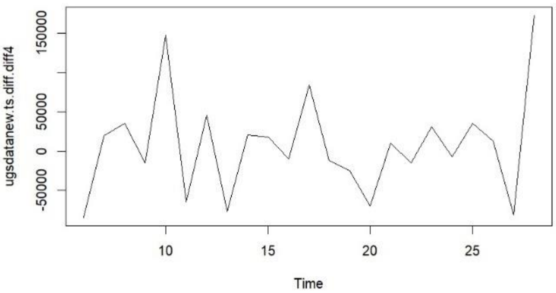


ACF has spikes at lag 4,8 and 12. This is an indicator of seasonality with a period of 4. Also at the non-seasonal points, ACF does not cut off, it dies out.

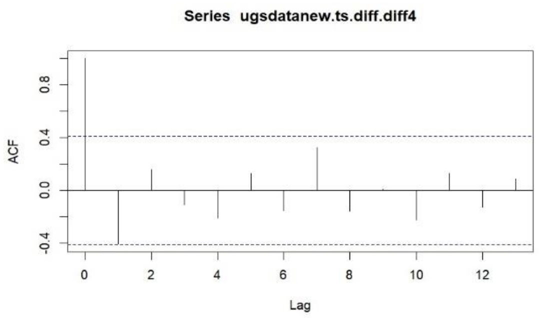


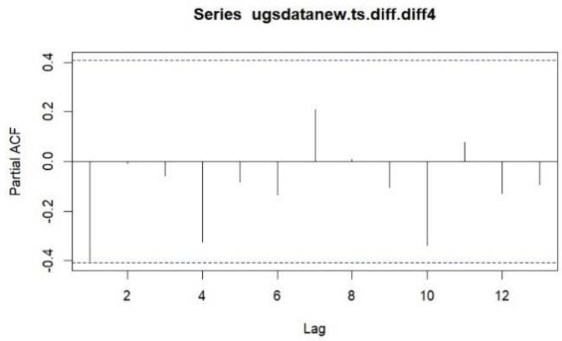
When we look at the Partial ACF, we can see that it cuts off at lag 3.

Subsequently, in order to obtain a stationary dataset, it is necessary to apply a seasonal difference with S=4.



After taking seasonal difference, the time series plot seems stationary now.



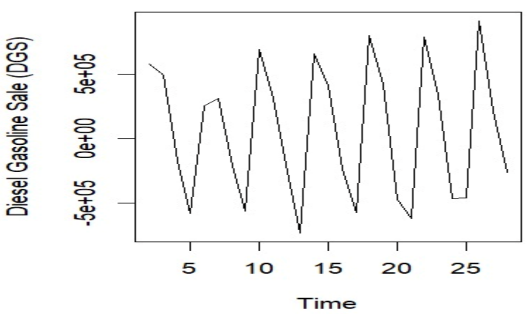


Upon examining the non-seasonal periods of the ACF and PACF, we observe that the ACF cuts off at lag 1, while the PACF dies out(almost). These observations indicate a regular Moving Average (MA) process of order 1.

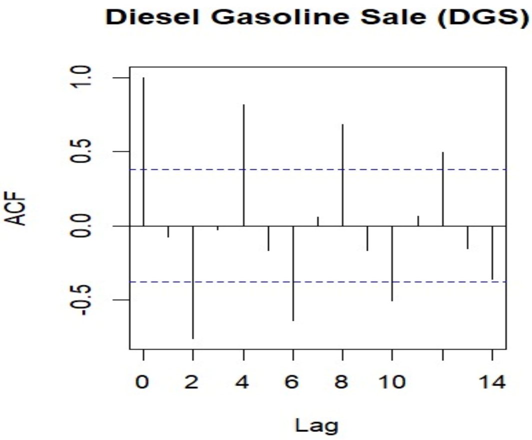
In contrast, when examining the seasonal periods, we notice that the ACF dies out(almost), and the PACF cuts off (almost) at a certain lag. These patterns suggest a possible seasonal Autoregressive (AR) process of order 1.

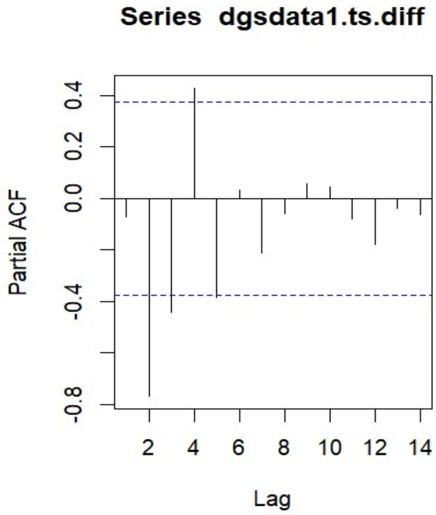
2.4.2 Differencing of DGS

For DGS, time series is non-stationary so in the first step we should take regular difference:



After that step we can see that data is stable except seasonality. A seasonal difference with period equal to 4 is needed.

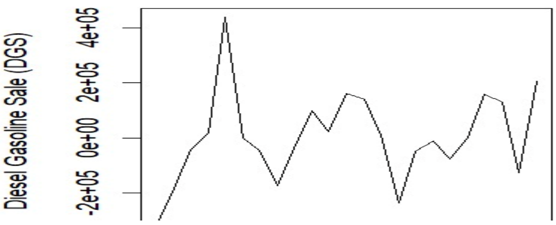




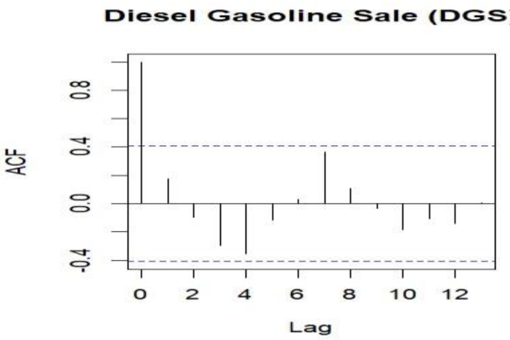
ACF values are high at lags 4,8 and 12 and it is a indicator of seasonality with s=4

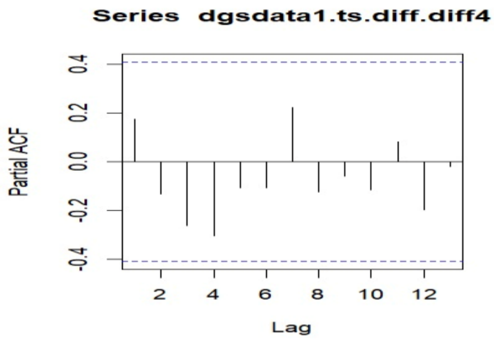
PACF of the first regular difference

After that, we need to take seasonal difference with s=4.



After the seasonal difference, time series data become stationary.





When we look at ACF and PACF for non-seasonal periods, we couldn't see any cut off, both died out. It means that the non-seasonal part should be (0,1,0) or we can use ARMA process.

From the seasonal parts, we can see cut off in ACF after lag 4, so we the seasonal part could be (0,1,1) or we can use MA(1).

2.5 Initial ARIMA

For UGS we can see from recent part there is cut off in ACF and die out in PACF in the non-seasonal points, so the non-seasonal part should be (0,1,1). In seasonal points there is cut off in PACF and die out in ACF, it means that seasonal part should be (1,1,0). So our initial ARIMA should be : **ARIMA (0,1,1) (1,1,0).**

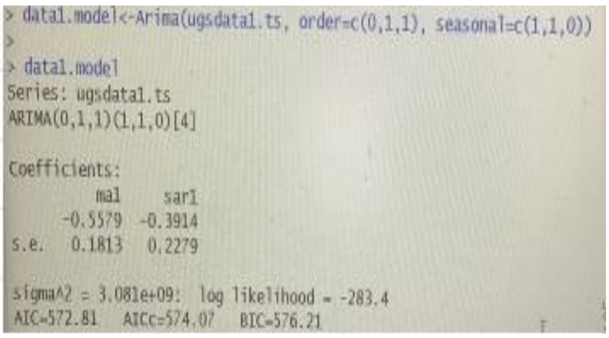
In the case of DGS, the absence of a cut-off point in the non-seasonal portion, as observed in the previous analysis, suggests that the non-seasonal component should be represented as (0,1,0). On the other hand, in the seasonal portion, we observe a cut-off in the ACF and a die out in the PACF, indicating that the seasonal component should be represented as (0,1,1). Consequently, the initial ARIMA model for DGS would be **ARIMA(0,1,0)(0,1,1).**

2.6 Neighborhood Search

After obtaining the initial ARIMA model from the previous analysis, it is essential to search for alternative models in the vicinity to identify the best fitting model. We can begin by exploring variations of the initial ARIMA model, modifying both the seasonal and non-seasonal components, and examining neighboring models. It would be beneficial to conduct this exploration separately for UGS and DGS.

2.6.1 Neighborhood Search for UGS

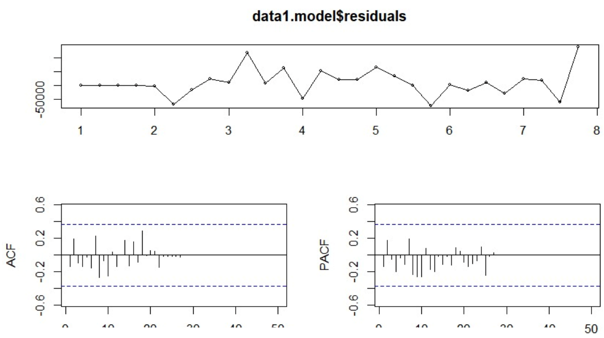
Our first candidate is, the initial ARIMA, ARIMA(0,1,1) (1,1,0):



```
> data1.model<-Arima(ugsdata1.ts, order=c(0,1,1), seasonal=c(1,1,0))
>
> data1.model
Series: ugsdata1.ts
ARIMA(0,1,1)(1,1,0)[4]

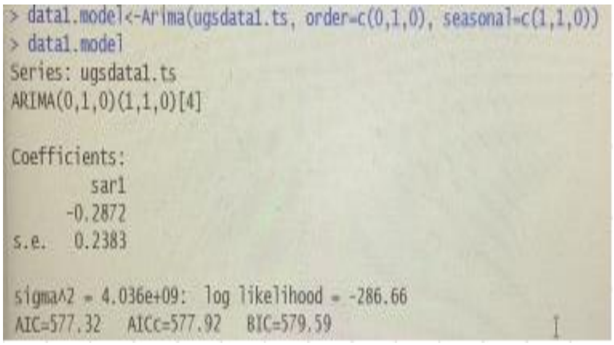
Coefficients:
         ma1     sar1
     -0.5579  -0.3914
s.e.  0.1813   0.2279

sigma^2 = 3.081e+09:  log likelihood = -283.4
AIC=572.81   AICc=574.07   BIC=576.21
```



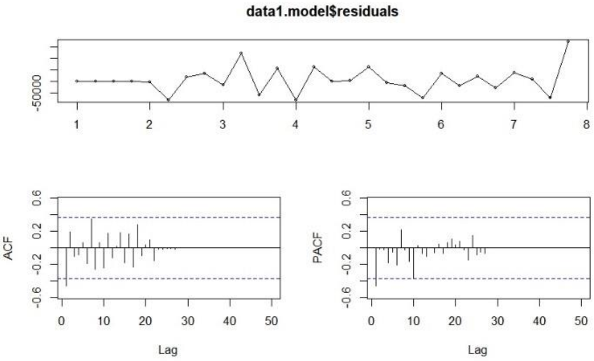
From the ACF and PACF perspective, this model seems good. However, we would still perform a neighborhood search to pursue a better model.

Our second candidate is, ARIMA(0,1,0) (1,1,0):


```
> data1.model<-Arima(ugsdata1.ts, order=c(0,1,0), seasonal=c(1,1,0))
> data1.model
Series: ugsdata1.ts
ARIMA(0,1,0)(1,1,0)[4]

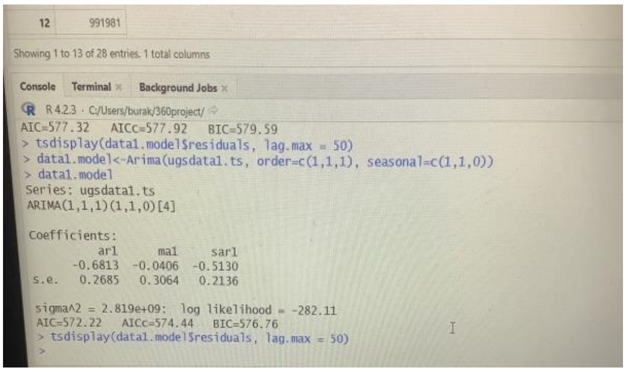
Coefficients:
         sar1
      -0.2872
s.e.   0.2383

sigma^2 = 4.036e+09:  log likelihood = -286.66
AIC=577.32   AICc=577.92   BIC=579.59
```



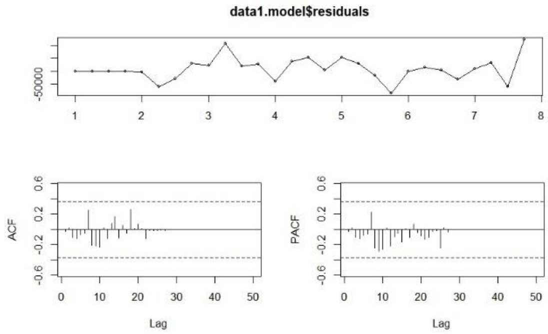
The high ACF value at lag 1 suggests put a regular MA(1) term into the model, which leads us back to the candidate 1.

Our third candidate is ARIMA(1,1,1) (1,1,0):


```
12        991981
Showing 1 to 13 of 28 entries, 1 total columns
Console   Terminal   Background Jobs
R 4.2.3 · C:/Users/burak/360project/
AIC=577.32   AICc=577.92   BIC=579.59
> tsdisplay(data1.model$residuals, lag.max = 50)
> data1.model<-Arima(ugsdata1.ts, order=c(1,1,1), seasonal=c(1,1,0))
> data1.model
Series: ugsdata1.ts
ARIMA(1,1,1)(1,1,0)[4]

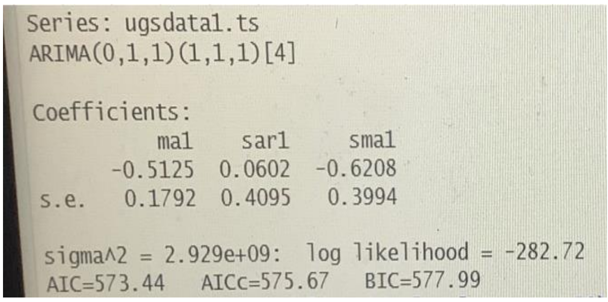
Coefficients:
          ar1      ma1     sar1
      -0.6813  -0.0406  -0.5130
s.e.   0.2685   0.3064   0.2136

sigma^2 = 2.819e+09:  log likelihood = -282.11
AIC=572.22   AICc=574.44   BIC=576.76
> tsdisplay(data1.model$residuals, lag.max = 50)
>
```



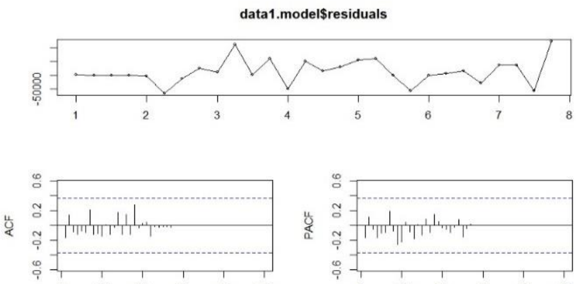
This model looks good when we look at ACF and PACF. But we should prefer first candidate because it has lower AIC_c and BIC.

Our fourth candidate is ARIMA(0,1,1) (1,1,1):


```
Series: ugsdata1.ts
ARIMA(0,1,1)(1,1,1)[4]

Coefficients:
          ma1    sar1     sma1
      -0.5125  0.0602  -0.6208
s.e.   0.1792  0.4095   0.3994

sigma^2 = 2.929e+09:  log likelihood = -282.72
AIC=573.44   AICc=575.67   BIC=577.99
```



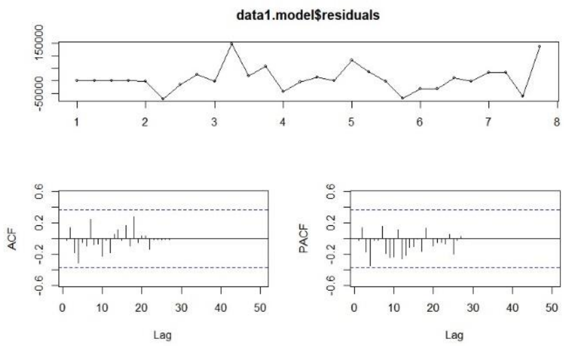
Nevertheless it seems good when we look at ACF and PACF, first constraint should be preferred because fourth candidate's AIC, AIC_c and BIC values are greater than first one.

Our fifth candidate is ARIMA(0,1,1) (0,1,0):

```
> data1.model
Series: ugsdata1.ts
ARIMA(0,1,1)(0,1,0)[4]

Coefficients:
          ma1
      -0.5810
s.e.   0.2207

sigma^2 = 3.386e+09:  log likelihood = -284.67
AIC=573.35   AICc=573.95   BIC=575.62
```



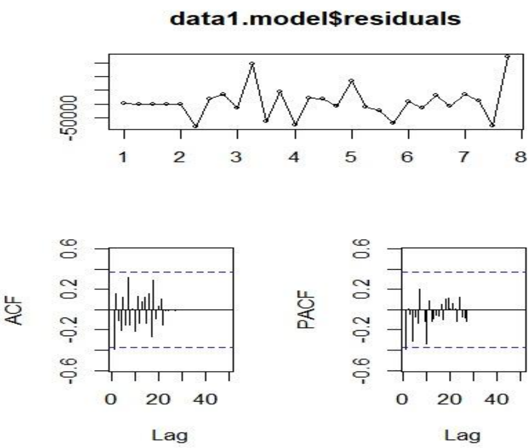
When we look at the seasonal periods in PACF, there is a significant peak at lag 4 and after that PACF dies out. This suggests us put a seasonal AR(1), which leads us to candidate 1.

Our sixth and final candidate is ARIMA(0,1,0) (0,1,0):

```
Series: dgsdata2.ts
ARIMA(1,1,1)(2,1,2)[4]

Coefficients:
         ar1      ma1     sar1    sar2    sma1     sma2
      0.6433  -1.0000  -0.7615  0.2359  0.4984  -0.4455
s.e.  0.1982   0.4338   1.3191  1.3174  1.2514   1.1697

sigma^2 = 1.91e+10:  log likelihood = -304.57
AIC=623.15   AICc=630.62   BIC=631.1
```



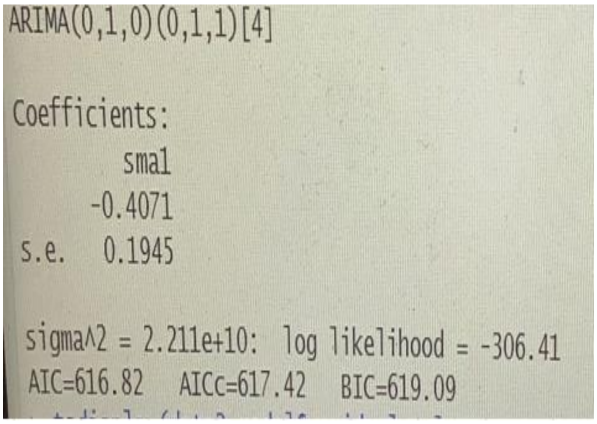
In the non-seasonal periods, we observe a strong correlation at lag 1 and a gradual decline in PACF, indicating a potential regular MA(1).

On the other hand, in the seasonal periods, ACF diminishes completely, and the PACF cuts off after lag 4. These patterns suggest the presence of a seasonal AR(1).

Considering these findings, we are led back to candidate 1 as a potential model.

2.6.2 Neighborhood Search for DGS

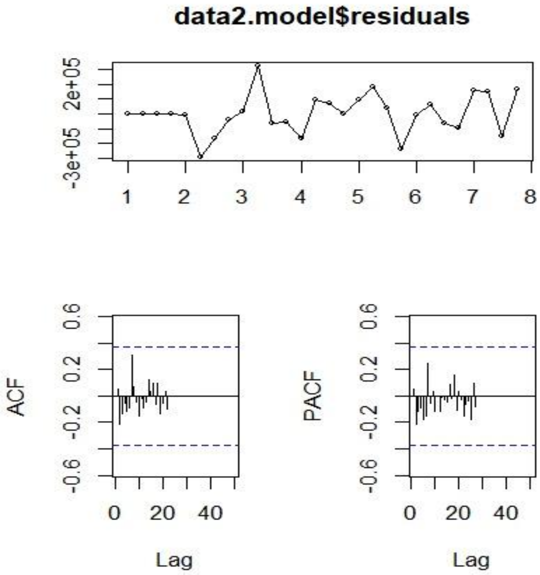
Our first candidate for DGS is, initial ARIMA, ARIMA(0,1,0) (0,1,1):



```
ARIMA(0,1,0)(0,1,1)[4]

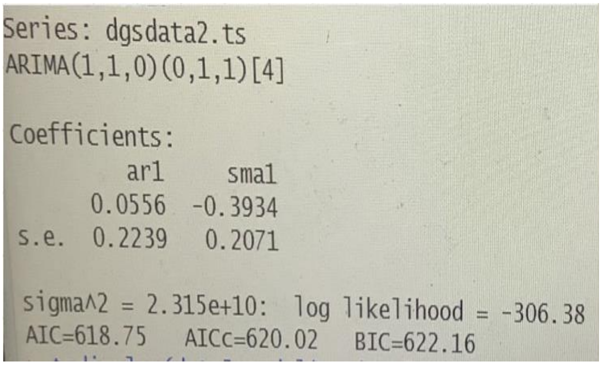
Coefficients:
         sma1
      -0.4071
s.e.   0.1945

sigma^2 = 2.211e+10:  log likelihood = -306.41
AIC=616.82   AICc=617.42   BIC=619.09
```



There are no significant lags in the ACF or PACF of this model. Hence, this model looks good.

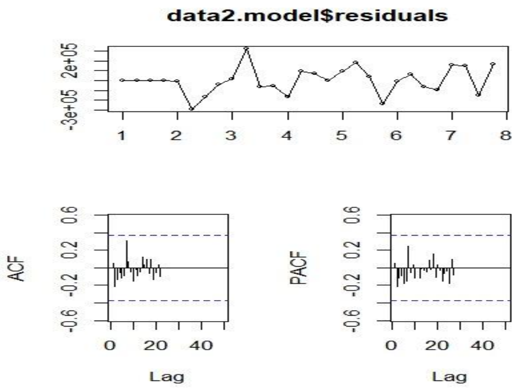
Our second candidate is ARIMA(1,1,0) (0,1,1):



```
Series: dgsdata2.ts
ARIMA(1,1,0)(0,1,1)[4]

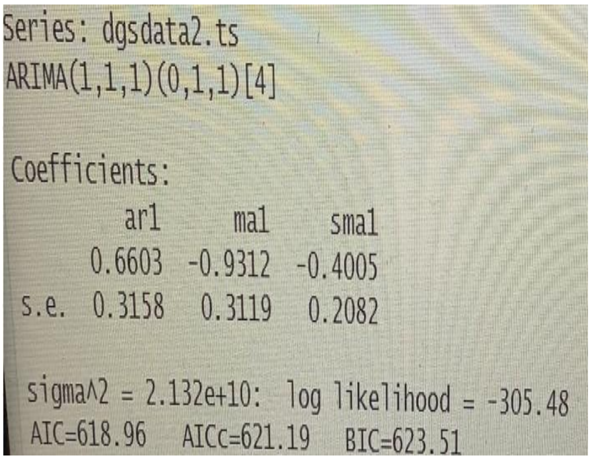
Coefficients:
         ar1     sma1
      0.0556  -0.3934
s.e.  0.2239   0.2071

sigma^2 = 2.315e+10:  log likelihood = -306.38
AIC=618.75   AICc=620.02   BIC=622.16
```



This model also has no significant lags in ACF and PACF but its AIC, AIC_c and BIC is greater than first candidate's so the first candidate should be preferred.

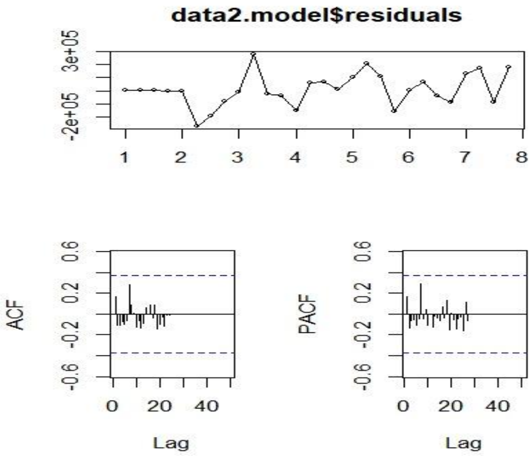
Our third candidate is ARIMA(1,1,1) (0,1,1):



```
Series: dgsdata2.ts
ARIMA(1,1,1)(0,1,1)[4]

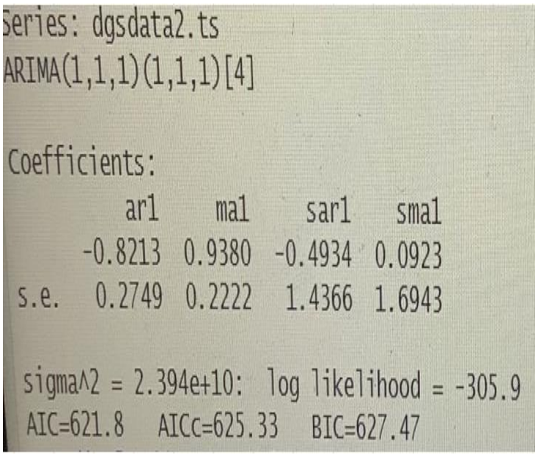
Coefficients:
          ar1      ma1     sma1
       0.6603  -0.9312  -0.4005
s.e.   0.3158   0.3119   0.2082

sigma^2 = 2.132e+10:  log likelihood = -305.48
AIC=618.96   AICc=621.19   BIC=623.51
```



This model also has no lags in ACF and PACF so it is a good model. Its AIC, AIC$_c$ and BIC is greater than first candidate's so the first candidate is more appropriate.

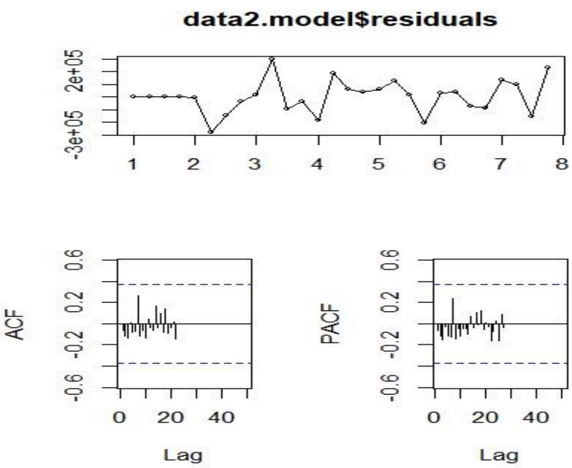
Our fourth candidate is ARIMA(1,1,1) (1,1,1):



```
Series: dgsdata2.ts
ARIMA(1,1,1)(1,1,1)[4]

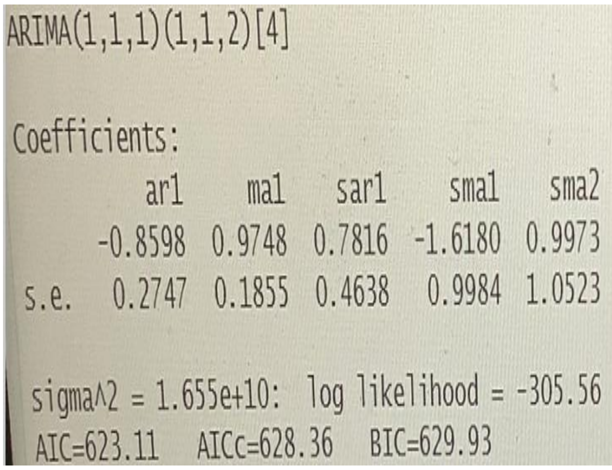
Coefficients:
          ar1     ma1     sar1    sma1
      -0.8213  0.9380  -0.4934  0.0923
s.e.   0.2749  0.2222   1.4366  1.6943

sigma^2 = 2.394e+10:  log likelihood = -305.9
AIC=621.8   AICc=625.33   BIC=627.47
```



The absence of lags in both the ACF and PACF indicates that this model is a good fit. However, when comparing the AIC, AICc, and Bayesian Information Criterion (BIC) values, we find that they are higher than those of the first candidate model. Therefore, the first candidate model is considered more suitable.

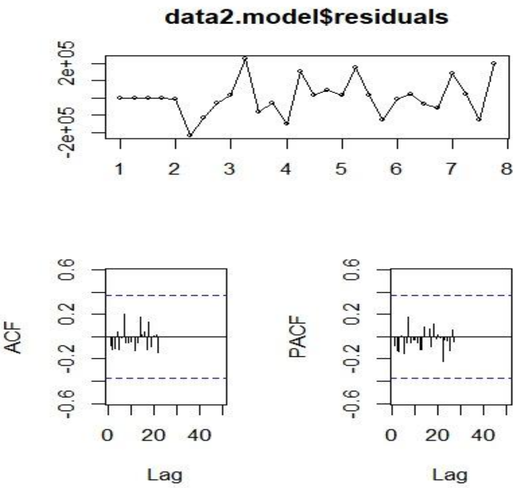
Our fifth candidate is ARIMA(1,1,1) (1,1,2):



```
ARIMA(1,1,1)(1,1,2)[4]

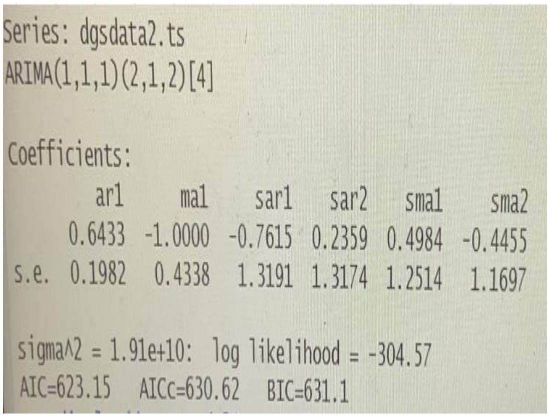
Coefficients:
          ar1     ma1    sar1     sma1    sma2
      -0.8598  0.9748  0.7816  -1.6180  0.9973
s.e.   0.2747  0.1855  0.4638   0.9984  1.0523

sigma^2 = 1.655e+10:  log likelihood = -305.56
AIC=623.11   AICc=628.36   BIC=629.93
```



The lack of lags in both the ACF an PACF suggests that this model is well-suited. Nonetheless, upon comparing the AIC, AIC_c, and BIC values, we observe that they are higher than those of the first candidate model. As a result, the first candidate model seems more appropriate.

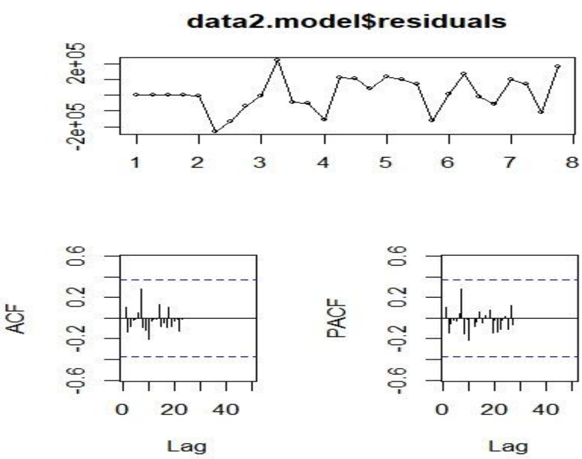
Our sixth candidate is ARIMA(1,1,1) (2,1,2):



```
Series: dgsdata2.ts
ARIMA(1,1,1)(2,1,2)[4]

Coefficients:
         ar1      ma1     sar1    sar2    sma1     sma2
      0.6433  -1.0000  -0.7615  0.2359  0.4984  -0.4455
s.e.  0.1982   0.4338   1.3191  1.3174  1.2514   1.1697

sigma^2 = 1.91e+10:  log likelihood = -304.57
AIC=623.15   AICc=630.62   BIC=631.1
```



Like the others, this model seems reasonable because there isn’t a specific behavior in ACF or PACF. However, we should select candidate 1 as it looks the best fit on initial ACF and PACF, also it has lowest AIC, AIC_c and BIC.

2.7 Best Model to Use

AIC (Akaike Information Criterion) assesses the balance between the model's goodness of fit and its simplicity when evaluating the amount of information lost. It takes into account both the risk of overfitting and the risk of underfitting. When comparing a group of candidate models, the preferred model is the one with the lowest AIC value. AIC rewards a good fit while also incorporating a penalty that increases with the number of estimated parameters. This penalty discourages overfitting, which is desirable because increasing the number of parameters in the model generally improves the fit. AIC is a first-order estimate, whereas AIC_c is a second-order estimate. BIC (Bayesian Information Criterion) is another criterion used for model selection among a finite set of models. It favors models with lower BIC values and is based, in part, on the likelihood function.

Additionally, the presence of cut-offs in the autocorrelation function (ACF) or partial autocorrelation function (PACF) indicates that a model may not accurately capture the underlying pattern. Therefore, it is preferable to choose a model with no cut-offs in the ACF and PACF for both seasonal and non-seasonal periods when selecting an appropriate model.

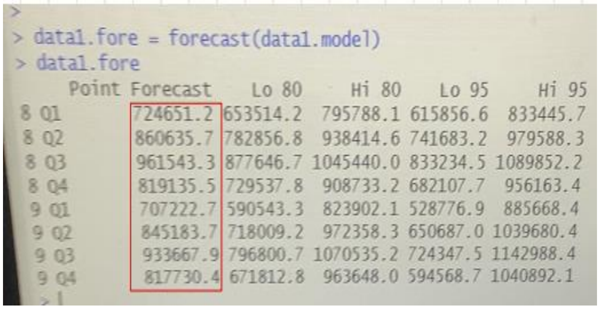
Consequently, we should select the model with the lowest AIC, AIC_c and BIC, and the model should not include cut off in ACF or PACF. So, for both UGS and DGS the first models are the most appropriate models.

For UGS we should use **ARIMA (0,1,1) (1,1,0).**

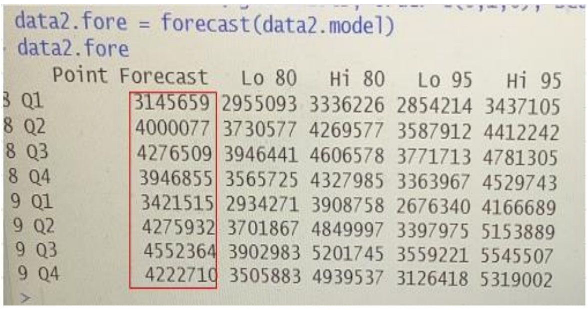
For DGS we should use **ARIMA(0,1,0)(0,1,1).**

2.8 Forecast

In the first method we made forecast with the Time Series Analysis.



```
>
> data1.fore = forecast(data1.model)
> data1.fore
     Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
8 Q1       724651.2 653514.2  795788.1 615856.6  833445.7
8 Q2       860635.7 782856.8  938414.6 741683.2  979588.3
8 Q3       961543.3 877646.7 1045440.0 833234.5 1089852.2
8 Q4       819135.5 729537.8  908733.2 682107.7  956163.4
9 Q1       707222.7 590543.3  823902.1 528776.9  885668.4
9 Q2       845183.7 718009.2  972358.3 650687.0 1039680.4
9 Q3       933667.9 796800.7 1070535.2 724347.5 1142988.4
9 Q4       817730.4 671812.8  963648.0 594568.7 1040892.1
```



```
data2.fore = forecast(data2.model)
data2.fore
     Point Forecast   Lo 80   Hi 80   Lo 95   Hi 95
8 Q1        3145659 2955093 3336226 2854214 3437105
8 Q2        4000077 3730577 4269577 3587912 4412242
8 Q3        4276509 3946441 4606578 3771713 4781305
8 Q4        3946855 3565725 4327985 3363967 4529743
9 Q1        3421515 2934271 3908758 2676340 4166689
9 Q2        4275932 3701867 4849997 3397975 5153889
9 Q3        4552364 3902983 5201745 3559221 5545507
9 Q4        4222710 3505883 4939537 3126418 5319002
>
```

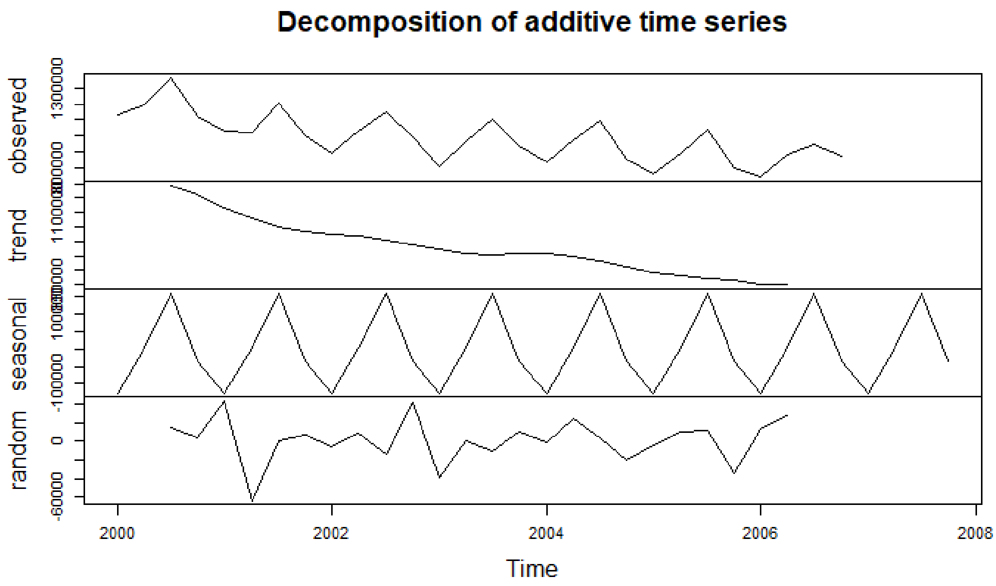
Periods/Product	**UGS**	**DGS**
2007 Q1	724651.2	3145659
2007 Q2	860635.7	4000077
2007 Q3	961543.3	4276509
2007 Q4	819135.5	3946855
2008 Q1	707222.7	3421515
2008 Q2	845183.7	4275932
2008 Q3	933667.9	4552364
2008 Q4	817730.4	4222710

3. REGRESSİON

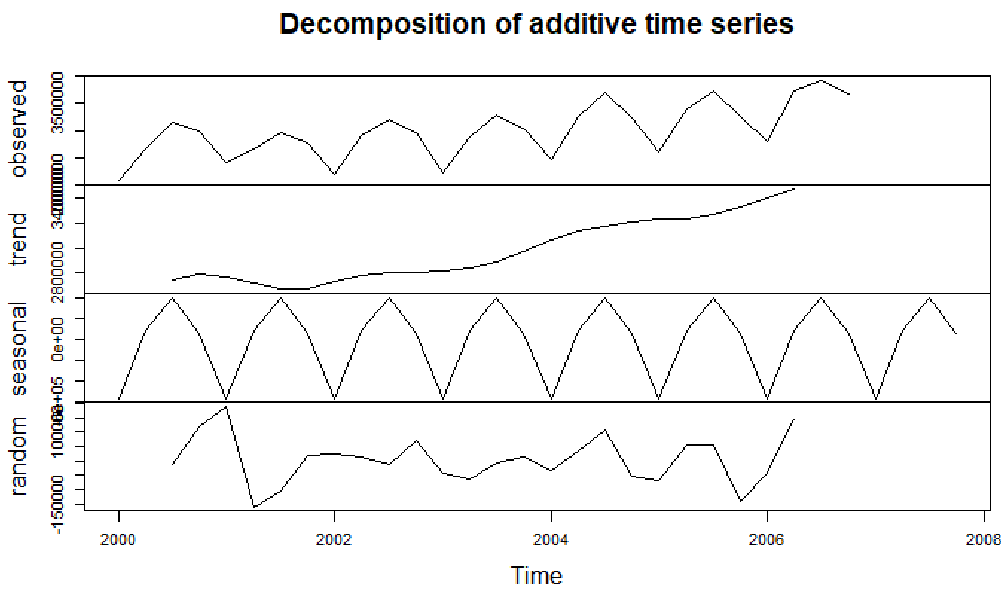
3.1 Preliminary Transformations

Time series decomposition will be used to decide on logarithmic transformation. if the seasonal component has varying variance, logarithm can be used.

Fort the UGS sales time series decomposition is as follows:

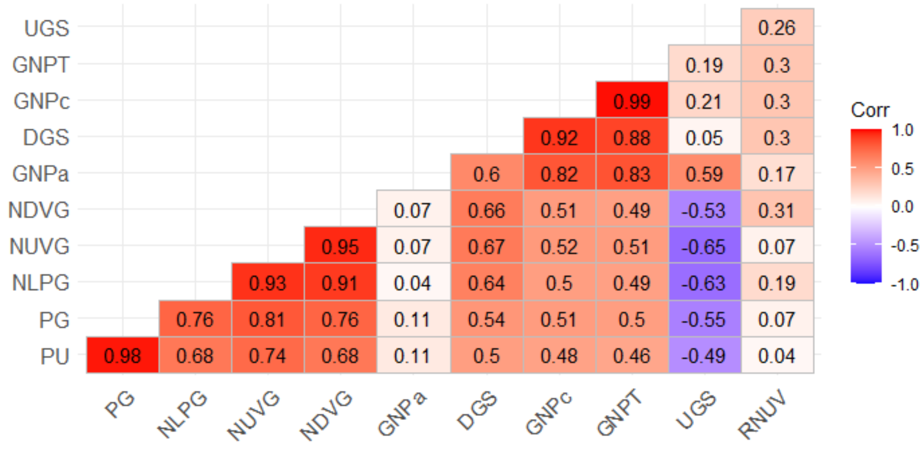


And for the DGS sales time series decomposition is as follows:



As can be seen from the plots the varience of seasonal component is constant over time hence logarithmic transformation is not needed.

Another transformation can be done on variables. Since there are too many variables in the data to omit the multicollinearity, the values that have highly correlated pairs in the data will be removed.



Due to high correlations between variables, PG, NUVG, NDVG,GNPc,GNPT are removed from the dataset.

3.2 Seasonality and trend related variables

3.2.1 Seasonality Variables

After doing colineraity checks by observing correlations among variables, data that represents seasonality and trend must be added.

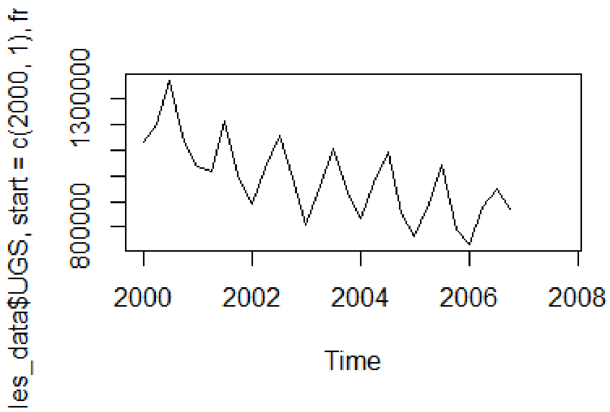
```
```{r}
seasonality_matrix = matrix(rep(diag(4), 8), ncol = 4, byrow = TRUE)
sales_data$Q1 = seasonality_matrix[,1]
sales_data$Q2 = seasonality_matrix[,2]
sales_data$Q3 = seasonality_matrix[,3]
sales_data$Q4 = seasonality_matrix[,4]

```
```

With the code in the figure above seasonality information added to the data for each quarter by assigning 1 to each row’s quarter.

3.2.2 Trend Variables

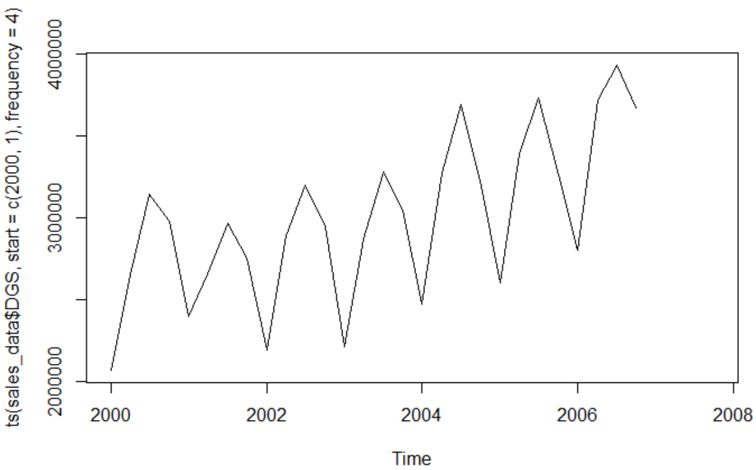
After adding seasonality, trend variables should also be added to the model. First, the trend component of the UGS sales will be added. From the plot given below one can see that, there is a diminishing trend. Until 2003 the trend is more steep, after that it slowly decreases. Hence 2 different trend variable will be added to the model.



The two trends are added with the given code below:

```{r}
sales_data$trend1_UGS = c(12:1,rep(0, 16),rep(NA,4))# until the end of 2003 the trend variable
                                                    will decrease from 12 to 1 then remain zero
sales_data$trend2_UGS = c(rep(0,12), c(16:1), rep(NA,4))
```

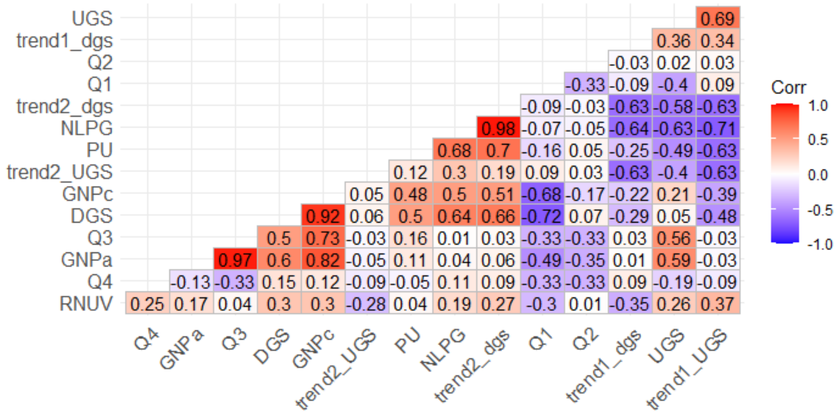
Lastly trend of DGS sale is examined



As can be seen from above plot there is a slow increasing trend until the end of 2003, then trend gets steeper. Hence 2 more trend variable for DGS sales will be added.

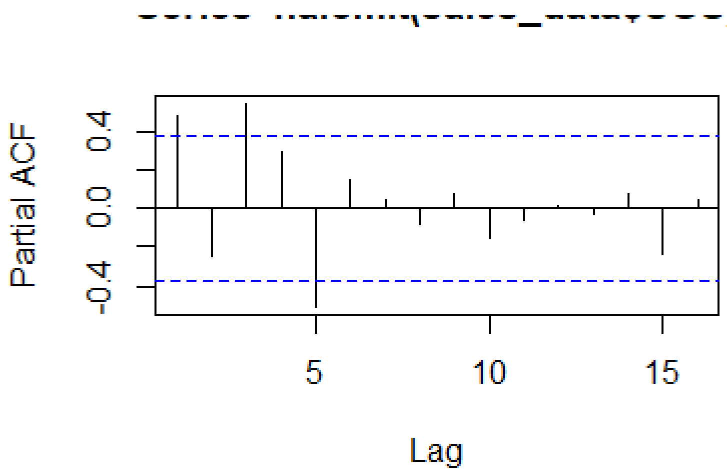
```{r}
sales_data$trend1_dgs = c(1:12, rep(0,16), rep(NA,4))   #trend that increases slowly until the
                                                         end of 2003
sales_data$trend2_dgs = c(rep(0,12), 1:16, rep(NA,4))   #trend for the steeper part of the data
```

After adding the trend and seasonality variables, again correlations will be examined to check multicollinearity.



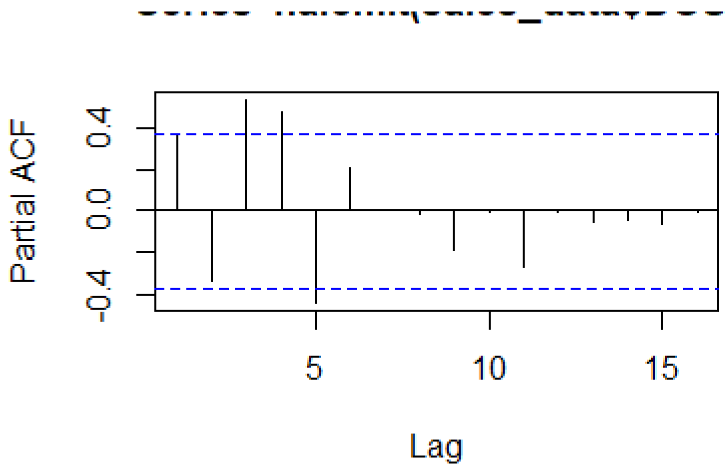
Due to high correlations some variables will be omitted since tehy are already represented by some other variables. Those variables that are omitted are; Q3, GNPc,trend2_dgs,trend1_ugs, trend1_dgs will be removed.

Also since the past values of the target variable can give information about the forecast, one can add lag variables to the model to include this information. To decide which lags one should add to the model pacf of the target value can be examined. Since there are significant jumps at lag1, lag3 and lag5 in below figure, 3 lagged variable will be added to the model for UGS sales data. This will help model to capture the autocorrelation structure of the data and improve the model's predictive performance



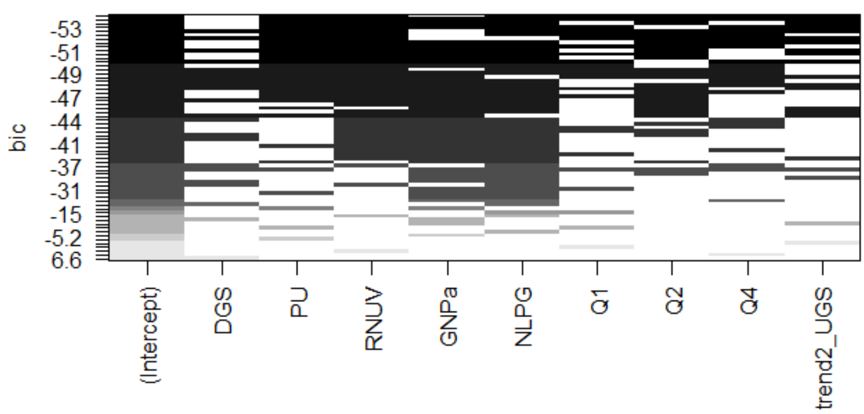
Pacf for ugs

For DGS sales there are significant jumps at lag3, lag4 and lag5, for the same reasons mentioned above these will be added to the model as lag variables.



Since new variable is added, correlations must be examined. When the correlations of lags examined with the given figüre below, one can see that lag variables are highly correlated and can misled the analysiss hence they are removed.

In addition to adding new variables, one can also delete the unnecessary columns. To see the effects of the variables to the regression, regsubsets() function will be used.

```
```{r}
leapsmodel=regsubsets(sales_data$UGS~  DGS + PU+RNUV+GNPa+NLPG+Q1+Q2+Q4+trend2_UGS,
data=sales_data,nbest=8)
plot(leapsmodel,scale="adjr2")
plot(leapsmodel,scale="r2")
plot(leapsmodel,scale="bic")
```
```



From above plot, we can see that all variables may have significant effect hence, we will not omit a variable and continue with building regression model

3.3Regression Models

While conducting the first regression model differencing will be applied since there is a trend in the residuals.By selecting arbitrary variable as a start regression model constructed as such:

```
```{r}
reg1_ugs = lm(diff(UGS)~  diff(DGS) + diff(PU)+diff(NLPG)+Q1[1:length(Q1)-1]+Q4[1:length(Q4)-1],
data=sales_data)
summary(reg1_ugs)
```
```

```
Call:
lm(formula = diff(UGS) ~ diff(DGS) + diff(PU) + diff(NLPG) + 
    Q1[1:length(Q1) - 1] + Q4[1:length(Q4) - 1], data = sales_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-101779  -26266    3446   18166   85852 

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)    
(Intercept)          -4.315e+04  1.666e+04  -2.590   0.0171 *  
diff(DGS)             4.483e-01  3.482e-02  12.876 1.96e-11 ***
diff(PU)              6.148e+02  3.108e+02   1.978   0.0612 .  
diff(NLPG)            1.413e-01  2.766e-01   0.511   0.6149    
Q1[1:length(Q1) - 1] -1.627e+05  3.177e+04  -5.123 4.49e-05 ***
Q4[1:length(Q4) - 1]  1.940e+05  3.263e+04   5.946 6.68e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 47650 on 21 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.921,	Adjusted R-squared:  0.9022 
F-statistic: 48.95 on 5 and 21 DF,  p-value: 7.226e-11
```

After conducting the model, residuals must be examined to see if there is a remaining trend in the model.

```
```{r}
plot(reg1_ugs$residuals)
acf(reg1_ugs$residuals)
```
```

```
```{r}
adf.test(reg2_ugs$residuals)
```

	Augmented Dickey-Fuller Test

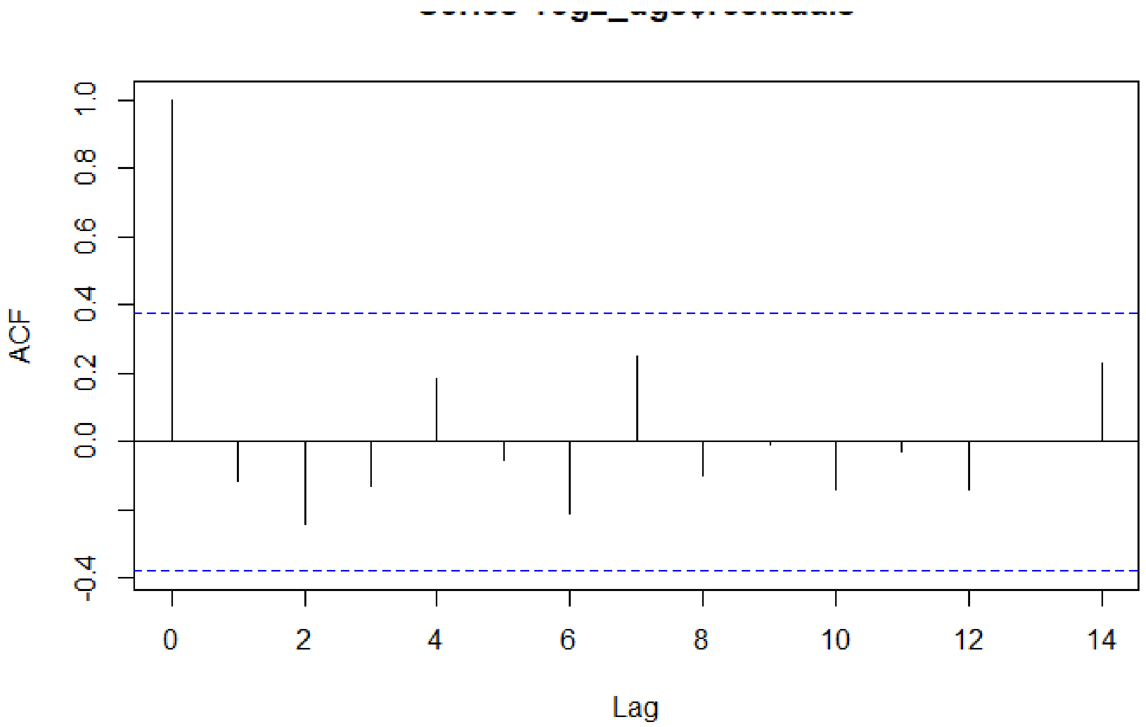
data: reg2_ugs$residuals
Dickey-Fuller = -4.0634, Lag order = 2, p-value = 0.0208
alternative hypothesis: stationary
```

Pvalue <0.05 hence residuals are stationary. Since this model passes the residual analysis now, Durbin-Watson test will used.

```{r}
dwtest(reg2_ugs)
```

```
	Durbin-Watson test

data:  reg2_ugs
DW = 2.083, p-value = 0.5694
alternative hypothesis: true autocorrelation is greater than 0
```



From acf function and the plot of the residuals, they seem stationary. To make sure adf.test will be used.

D value is close to two, p value isn't significant. Hence H0 cannot be rejected.

Again by looking at the coefficients of the variable new variables with most significant p value will be chosen.

The seconf model is as follows:

```
Call:
lm(formula = diff(UGS) ~ diff(DGS) + diff(PU) + Q1[1:length(Q1) -
    1] + Q4[1:length(Q4) - 1], data = sales_data)

Residuals:
    Min      1Q  Median      3Q     Max
-108369  -21088    5472   15904   83528

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)          -3.775e+04  1.266e+04  -2.983  0.00687 **
diff(DGS)             4.452e-01  3.370e-02  13.210 6.15e-12 ***
diff(PU)              5.803e+02  2.983e+02   1.946  0.06459 .
Q1[1:length(Q1) - 1] -1.638e+05  3.116e+04  -5.258 2.83e-05 ***
Q4[1:length(Q4) - 1]  1.902e+05  3.121e+04   6.093 3.92e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 46850 on 22 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:   0.92,     Adjusted R-squared:  0.9054
F-statistic: 63.24 on 4 and 22 DF,  p-value: 9.56e-12
```

Again residuals must be examined:

```
```{r}
plot(reg3_ugs$residuals)
adf.test(reg3_ugs$residuals)
```
```

```
Warning in adf.test(reg3_ugs$residuals) :
  p-value smaller than printed p-value

        Augmented Dickey-Fuller Test

data:  reg3_ugs$residuals
Dickey-Fuller = -4.4003, Lag order = 2, p-value = 0.01
alternative hypothesis: stationary
```

P value is significant, H0 will be rejected hence residuals are stationary.

Also for autocorrelations dwtest() will be constructed:

```
```{r}
dwtest(reg3_ugs)
```
```

```
        Durbin-Watson test

data:  reg3_ugs
DW = 2.07, p-value = 0.6195
alternative hypothesis: true autocorrelation is greater than 0
```

D value is around 2 and p value > 0.5 which indicates we cannot reject H0. Hence there is no enough evidence that autocorrelation is present in the model.

Only PU has a little less significant than others in the previous regression model hence one more regression model will be constructed without PU.

```
```{r}
reg4_ugs = lm(diff(UGS)~  diff(DGS) +Q1[1:length(Q1)-1]+Q4[1:length(Q4)-1], data=sales_data)
summary(reg4_ugs)
```
```

```
Call:
lm(formula = diff(UGS) ~ diff(DGS) + Q1[1:length(Q1) - 1] + Q4[1:length(Q4) -
    1], data = sales_data)

Residuals:
    Min      1Q  Median      3Q     Max
-126825  -19106    2006   20871   91284

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)          -4.074e+04  1.330e+04  -3.062  0.00552 **
diff(DGS)             4.485e-01  3.564e-02  12.584 8.52e-12 ***
Q1[1:length(Q1) - 1] -1.519e+05  3.235e+04  -4.696 9.93e-05 ***
Q4[1:length(Q4) - 1]  1.981e+05  3.276e+04   6.047 3.63e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 49600 on 23 degrees of freedom
  (4 observations deleted due to missingness)
Multiple R-squared:  0.9062,	Adjusted R-squared:  0.894
F-statistic: 74.09 on 3 and 23 DF,  p-value: 5.699e-12
```

Residual analysis fort he model:

```
```{r}
plot(reg4_ugs$residuals)
adf.test(reg4_ugs$residuals)
```
```

```
Warning in adf.test(reg4_ugs$residuals) :
  p-value smaller than printed p-value

	Augmented Dickey-Fuller Test

data:  reg4_ugs$residuals
Dickey-Fuller = -6.3943, Lag order = 2, p-value = 0.01
alternative hypothesis: stationary
```

DWtest fort he model:

```{r}
dwtest(reg4_ugs)
```

```
	Durbin-Watson test

data:  reg4_ugs
DW = 1.8853, p-value = 0.4542
alternative hypothesis: true autocorrelation is greater than 0
```

Among the models, reg3_ugs has the highest adjusted R^2 value and it has D around 2 hence it will be choosen as the best model

Forecast for the 2007 quarters will be found with the predict() function

```
#forecasts for 2007
forecasts <- predict(reg3_ugs, newdata =sales_data[28:32,c(2,3,4,8,10)] )
forecasts[1] = sales_data$UGS[28]+ 152412.04
tail_ugs_forecast = cumsum(forecasts)
tail_ugs_forecast
```

```
      1       2       3       4 
1024412 1107008 1269797 1064545 
```

Forecasts for DGS:

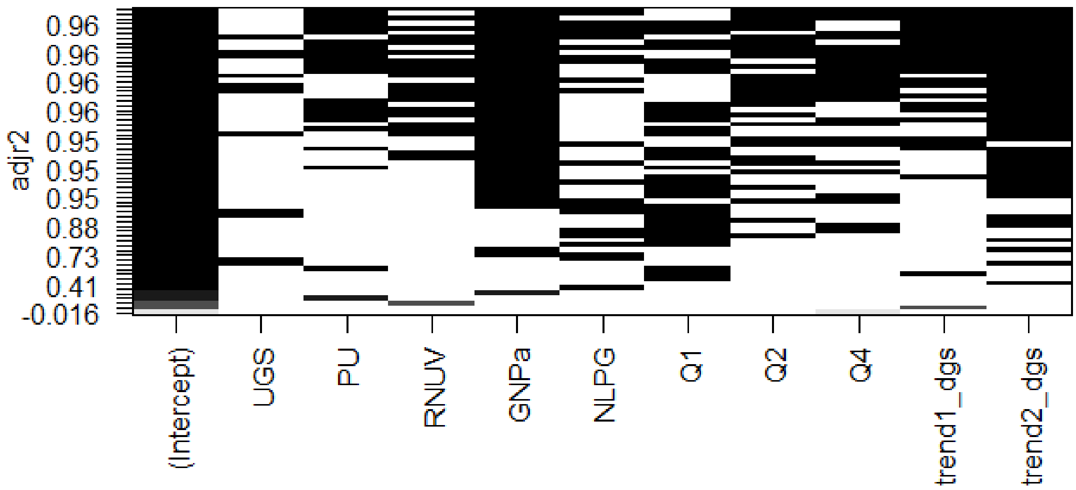
Similar procedures will be repeated for forecasting the DGS sales. The procedures for trend,seasonal, and lagged variables are done for both dgs and ugs hence multicollineraity operations will not be performed in this part.

However since trend variables for the dgs variable was omitted for ugs, trend variable must be added again.

```{r}
dgs_data = sales_data[1:28,]# to not include the forecasts we found for ugs
dgs_data$trend1_dgs = c(1:12, rep(0,16) )  #trend that increases slowly until the end of 2003
dgs_data$trend2_dgs = c(rep(0,12), 1:16)
```

First to have a starting point plots of regsubset() function for adjusted r^2 will be examined to select the variables for the first regression model.

```{r}
leapsmodel=regsubsets(dgs_data$DGS~  UGS + PU+RNUV+GNPa+NLPG+Q1+Q2+Q4+ trend1_dgs+trend2_dgs,
data=dgs_data,nbest=8)
plot(leapsmodel,scale="adjr2")
plot(leapsmodel,scale="r2")
plot(leapsmodel,scale="bic")
```



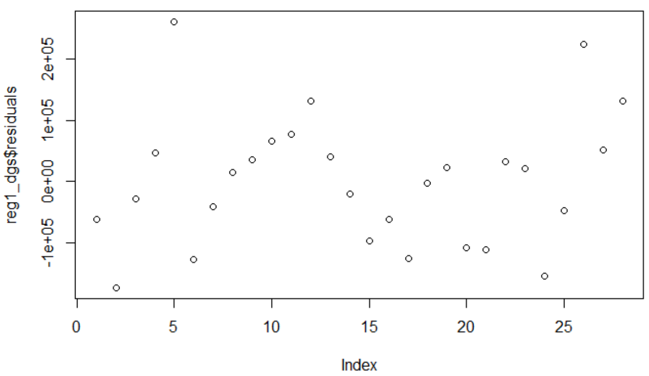
From above plot one can see that for the model with best adjusted r^2,all variables except ugs and Q1 is used.

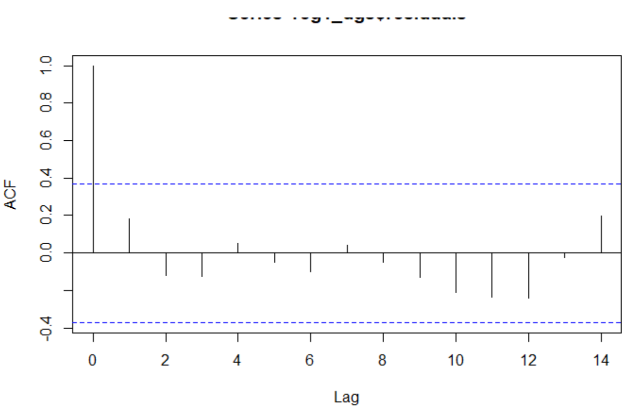
First we start with an arbitrary model and make the further analysis:

```
Call:
lm(formula = DGS ~ GNPa + NLPG + Q2 + Q4, data = dgs_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-172978  -70754    6260   48403  261335 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 9.716e+05  1.075e+05   9.041 4.94e-09 ***
GNPa        1.745e-01  1.082e-02  16.123 5.00e-14 ***
NLPG        1.042e+00  7.908e-02  13.177 3.34e-12 ***
Q2          5.622e+05  5.875e+04   9.568 1.75e-09 ***
Q4          3.994e+05  5.596e+04   7.138 2.86e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 115300 on 23 degrees of freedom
Multiple R-squared:  0.954,	Adjusted R-squared:  0.946 
F-statistic: 119.1 on 4 and 23 DF,  p-value: 5.063e-15
```

Residual Analysis fort he model:

```{r}
plot(reg1_dgs$residuals)
acf(reg1_dgs$residuals)
adf.test(reg1_dgs$residuals)
```





```
        Augmented Dickey-Fuller Test

data:  reg1_dgs$residuals
Dickey-Fuller = -1.7939, Lag order = 3, p-value = 0.6514
alternative hypothesis: stationary
```

Resulting p value is high difference should be taken.

The model with the difference operator:

```
Call:
lm(formula = diff(DGS) ~ diff(GNPa) + diff(NLPG) + Q2[1:length(Q2) -
    1] + Q4[1:length(Q4) - 1], data = dgs_data)

Residuals:
    Min      1Q  Median      3Q     Max
-381252  -53206    9360   80914  283378

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)            5.473e+05  5.112e+04  10.706 3.44e-10 ***
diff(GNPa)             1.992e-01  1.758e-02  11.329 1.19e-10 ***
diff(NLPG)             3.700e-01  8.520e-01   0.434    0.668
Q2[1:length(Q2) - 1] -1.216e+06  1.393e+05  -8.731 1.34e-08 ***
Q4[1:length(Q4) - 1] -8.661e+05  7.242e+04 -11.960 4.24e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 147500 on 22 degrees of freedom
Multiple R-squared:  0.9324,    Adjusted R-squared:  0.9201
F-statistic: 75.89 on 4 and 22 DF,  p-value: 1.51e-12
```

Residuals:

```{r}
plot(reg1_dgs_diff$residuals)
acf(reg1_dgs_diff$residuals)
adf.test(reg1_dgs_diff$residuals)
```

```
Warning in adf.test(reg1_dgs_diff$residuals) :
  p-value smaller than printed p-value

        Augmented Dickey-Fuller Test

data:  reg1_dgs_diff$residuals
Dickey-Fuller = -5.0714, Lag order = 2, p-value = 0.01
alternative hypothesis: stationary
```

After differencing residuals are stationary. Now, dwtest should be conducted to eliminate autocorrelation.

```{r}
dwtest(reg1_dgs_diff)
```

```
        Durbin-Watson test

data:  reg1_dgs_diff
DW = 2.7485, p-value = 0.9724
alternative hypothesis: true autocorrelation is greater than 0
```

D value higher than 2 indicates that there might be an autocorrelation. Hence lag variable will be added.

```
```{r}
reg1_dgs_diff_withlag = lm(diff(DGS)~ diff(lag(DGS))+ diff(GNPa)+ diff(NLPG)
+Q2[1:length(Q2)-1]+Q4[1:length(Q4)-1], data=dgs_data)
dwtest(reg1_dgs_diff_withlag)
```
```

```
	Durbin-Watson test

data:  reg1_dgs_diff_withlag
DW = 2.2696, p-value = 0.7757
alternative hypothesis: true autocorrelation is greater than 0
```

From the first model one can see that GNPa Q2 and Q4 has significant p values hence they will be used in the second model. Also first model excludes the trend term hence trend1_dgs and trend2_dgs will be added too.

Now, by taking the significance of the coefficients of the previous model, another model will constructed by omitting the trend2_dgs since its p value is the lowest

```
Call:
lm(formula = (DGS) ~ trend1_dgs + GNPa + Q2 + Q4, data = dgs_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-481326 -254542   57952  172555  555309 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.311e+06  1.324e+05  17.450 9.25e-15 ***
trend1_dgs  -4.054e+04  1.392e+04  -2.913 0.007836 ** 
GNPa         1.843e-01  2.702e-02   6.820 5.90e-07 ***
Q2           5.758e+05  1.468e+05   3.922 0.000683 ***
Q4           5.205e+05  1.396e+05   3.729 0.001100 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 288100 on 23 degrees of freedom
Multiple R-squared:  0.7125,	Adjusted R-squared:  0.6625 
F-statistic: 14.25 on 4 and 23 DF,  p-value: 5.476e-06
```

Residual analysis:

```{r}
plot(reg3_dgs$residuals)
acf(reg3_dgs$residuals)
adf.test(reg3_dgs$residuals)
```

```
        Augmented Dickey-Fuller Test

data:  reg3_dgs$residuals
Dickey-Fuller = -2.1295, Lag order = 3, p-value = 0.523
alternative hypothesis: stationary
```

Due to high p value differencing is necessary. The model after difference is taken.

```
Call:
lm(formula = diff(DGS) ~ diff(trend1_dgs) + diff(GNPa) + Q2[1:length(Q2) -
    1] + Q4[1:length(Q4) - 1], data = dgs_data)

Residuals:
    Min      1Q  Median      3Q     Max
-390270  -57701   -2679   74448  287719

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)          5.478e+05  4.904e+04  11.170 1.55e-10 ***
diff(trend1_dgs)     1.178e+04  1.252e+04   0.941    0.357
diff(GNPa)           1.961e-01  1.585e-02  12.373 2.20e-11 ***
Q2[1:length(Q2) - 1] -1.195e+06  1.291e+05  -9.257 4.83e-09 ***
Q4[1:length(Q4) - 1] -8.429e+05  7.591e+04 -11.103 1.74e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 145200 on 22 degrees of freedom
Multiple R-squared:  0.9345,	Adjusted R-squared:  0.9226
F-statistic: 78.45 on 4 and 22 DF,  p-value: 1.077e-12
```

Residual analysis after differencig:

```{r}
plot(reg3_dgs_diff$residuals)
acf(reg3_dgs_diff$residuals)
adf.test(reg3_dgs_diff$residuals)
```

```
Warning in adf.test(reg3_dgs_diff$residuals) :
  p-value smaller than printed p-value

        Augmented Dickey-Fuller Test

data:  reg3_dgs_diff$residuals
Dickey-Fuller = -4.7812, Lag order = 2, p-value = 0.01
alternative hypothesis: stationary
```

P value is smaller then 0.05 now residuals are stationary

DW test for the moel

```{r}
dwtest(reg3_dgs_diff)
```

```
	Durbin-Watson test

data:  reg3_dgs_diff
DW = 2.7439, p-value = 0.9819
alternative hypothesis: true autocorrelation is greater than 0
```

D higher then 2 hence there may be positive autocorrelation.Lag variables will be added.After adding the lag dwtest is conducted again and the value of d didn't changed significantly hence this model won't be chosen.

The model that gives the best D value and the R^2 was first model after taking the difference.

Forecasts for DGS in 2007 are as follows:

```
      1       2       3       4
2730405 3364263 3761148 3428365
```

4.Comparision:

For UGS with Time Series Analysis 2006 forecasts based on 2000-2005 data and the real sales are:

Periods/Values	Sales	Forecast	(Sales-Forecast)^2
Q1	736580	709633.2	726130030.24
Q2	877614	843214.8	1183304960.64
Q3	946783	982115.1	1248357290.41
Q4	872000	737254.9	18156241974.01
Total			21,314,034,255.3

For DGS with Time Series Analysis 2006 forecasts based on 2000-2005 data and the real sales are:

Periods/Values	Sales	Forecast	(Sales-Forecast)^2
Q1	2800111	2645215	23992770816
Q2	3717347	3407978	95709178161
Q3	3932606	3773550	25298811136
Q4	3671000	3344973	106293604729
Total			251,294,364,842

Comparision fort the regression model for UGS:

Here is the forecast for 2006 with the model constructed

```
       1        2        3        4
708970.3 789585.4 928523.0 742619.5
```

The real values fort he 2006

```
[1] 736580 877614 946783 872000
```

Comparision fort the regression model for DGS:

Here is the forecast for 2006 with the model constructed

```
      1       2       3       4
2730405 3364263 3761148 3428365
```

The real values for the 2006

```
[1] 2800111 3717347 3932606 3671000
```

Bye using the models and taking the first five years as the training data, the 2006 values are estimated and compared with the real values. After comparasion the regression models are better for estimating sales values.