IBEX Forecasting

Forecasting Time Series

HOMEWORK # 3

LOUIS DUBAERE
JUAN PABLO GARCÍA GÓMEZ
FRANZ-ANTON GRAF BASSELET VON LA ROSEE
SIYAO LU
ANDREW MARTINEZ
STEPHANIE NJERENGA
TOMAS TELLO GARCÍA

GROUP 01 - F

FTS Homework 3 - IBEX Forecasting

MBD O1 - Group F

February, 2019

Content

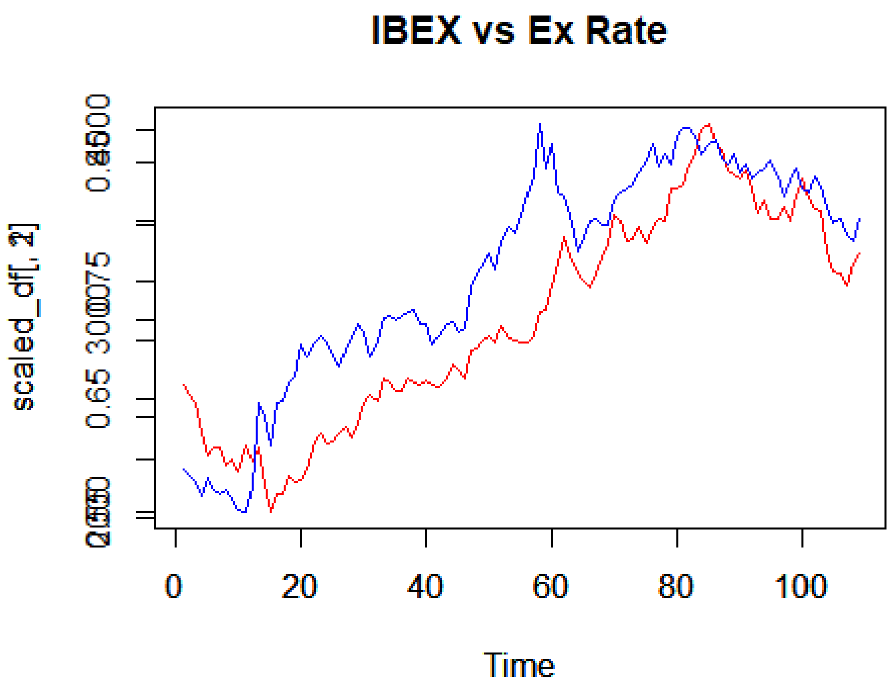
The following document illustrates the analysis for forecasting the future value of the IBEX stock-market index, based on explanatory variables whose future value can be predicted with reasonable accuracy. The variables considered in the analysis are:

- IBEX (dependent variable)
- Interest Rates
- Short Term Rates (90-day MIBOR)
- Long Term Rates (10-year bond rates)

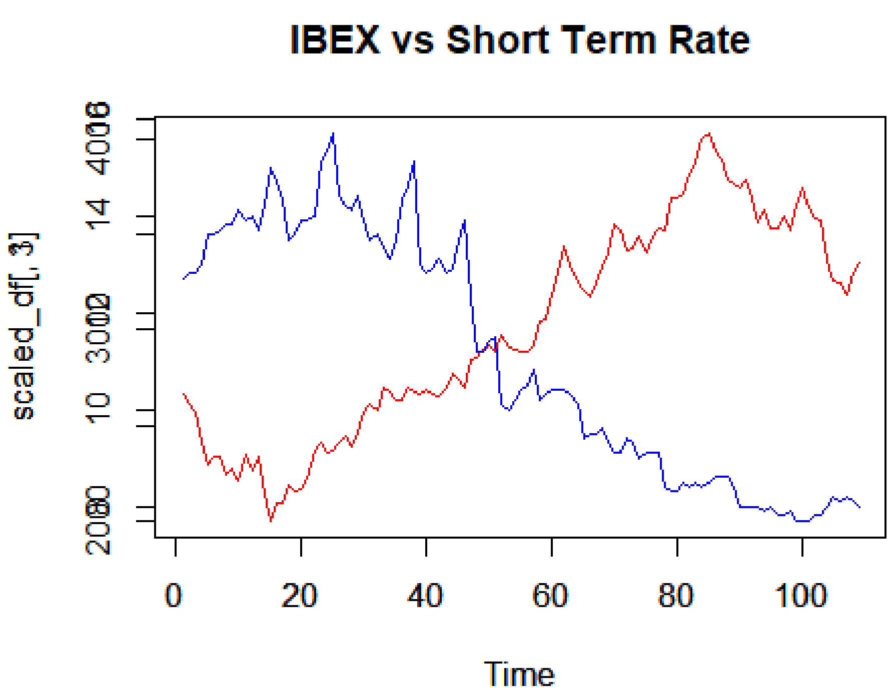
Exploratory analysis

As an initial start point, we will scale the variables, plot the different time series and start by analyzing a possible correlation of the explanatory variables with IBEX (our independent variable).

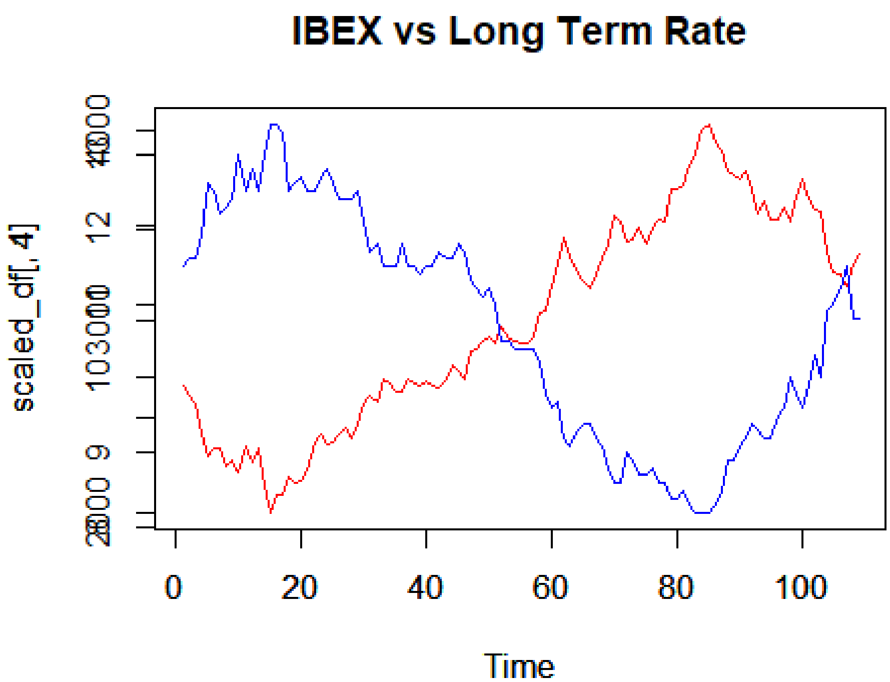
```
scaled_df<- scale(df)
scaled_df<- subset(df, select=-c(Week))

ts.plot(scaled_df[,1], col='red', main="IBEX vs Ex Rate")
par(new=TRUE)
ts.plot(scaled_df[,2], col='blue')
```



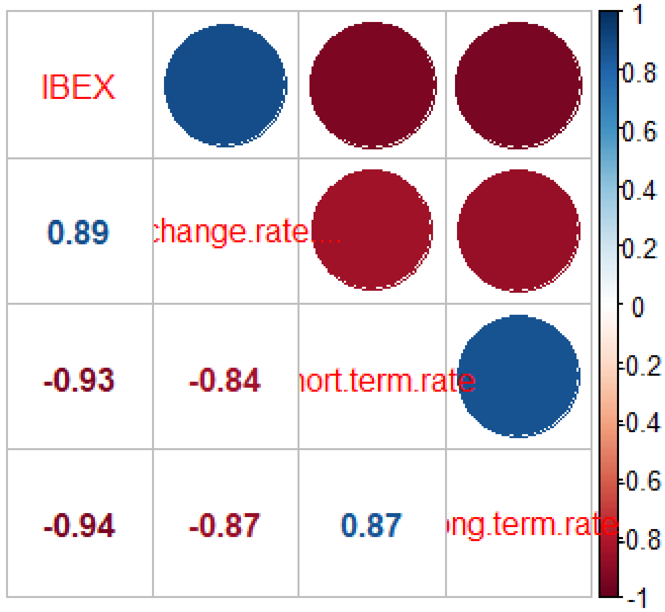

```r
ts.plot(scaled_df[,1], col='red', main="IBEX vs Short Term Rate")
par(new=TRUE)
ts.plot(scaled_df[,3], col='blue')
```



```r
ts.plot(scaled_df[,1], col='red', main="IBEX vs Long Term Rate")
par(new=TRUE)
ts.plot(scaled_df[,4], col='blue')
```



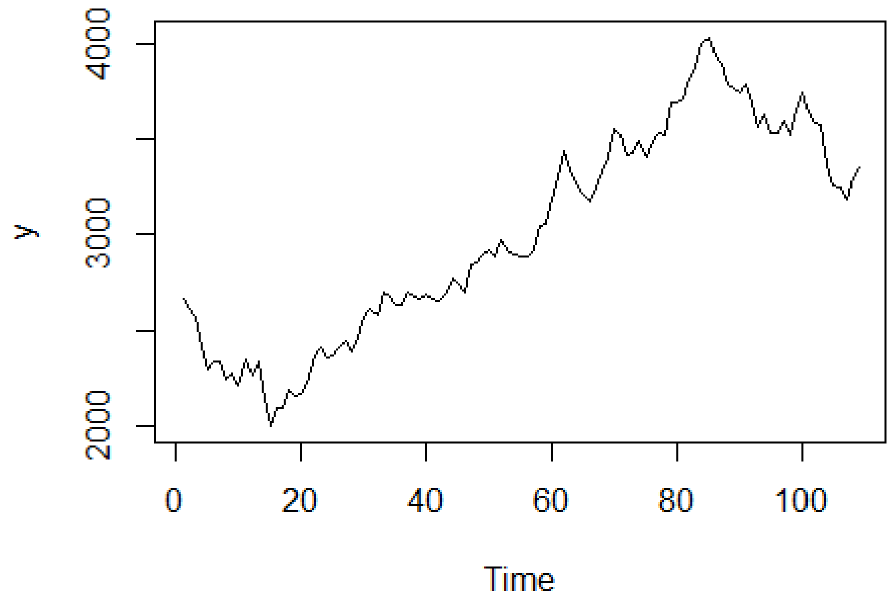

```r
cor_df<-cor(scaled_df)
corrplot.mixed(cor_df)
```



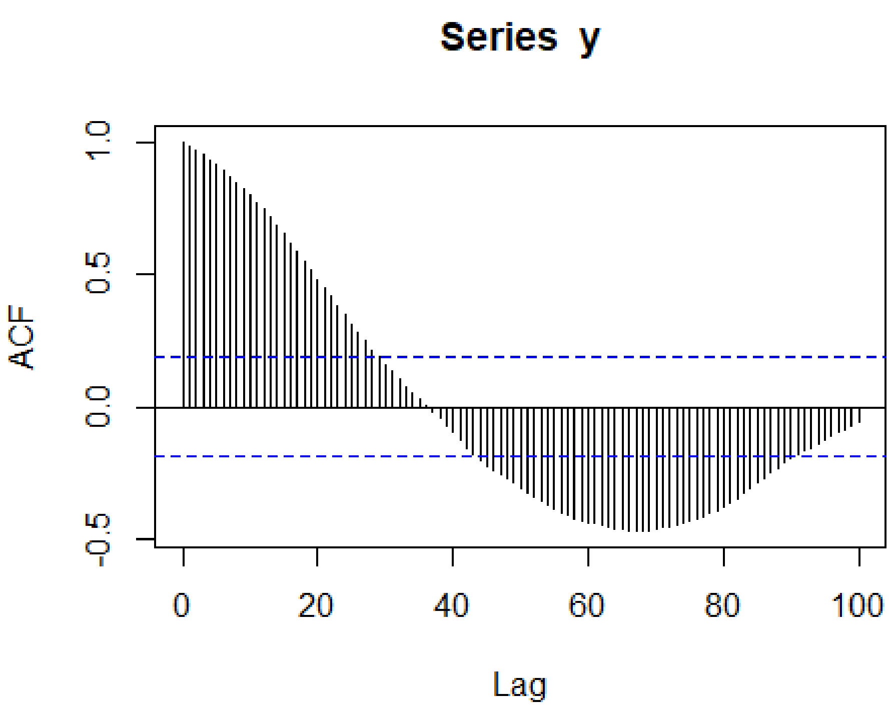
Best time series model for variable IBEX

As an initial step, we will find the best Forecasting Model for IBEX as a stand-alone variable:

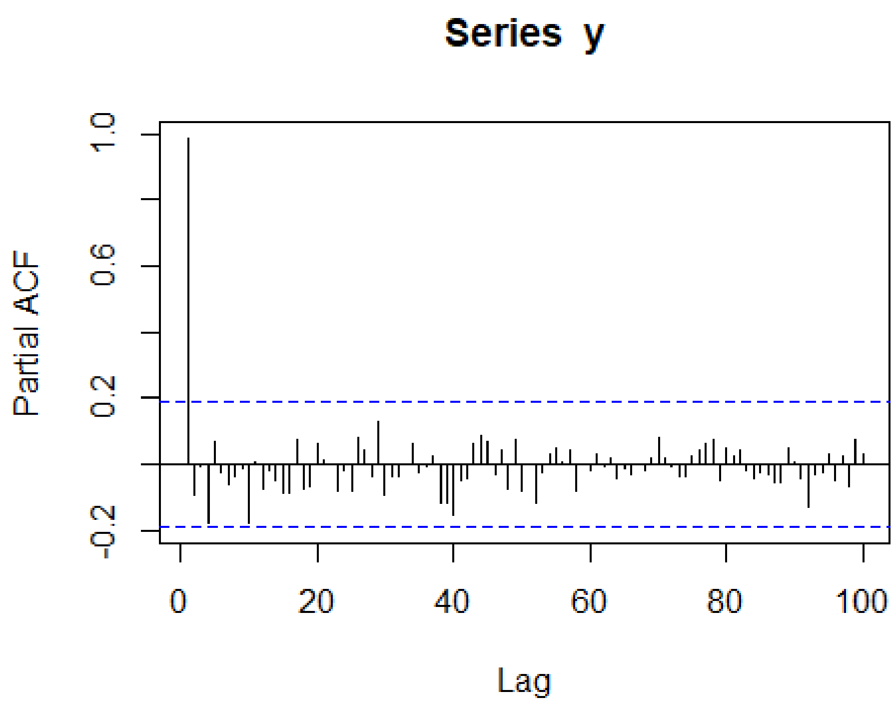
```
y<-scaled_df$IBEX
ts.plot(y)
```




```
nlags <- 100
acf(y,nlags)
```




```
pacf(y, nlags)
```

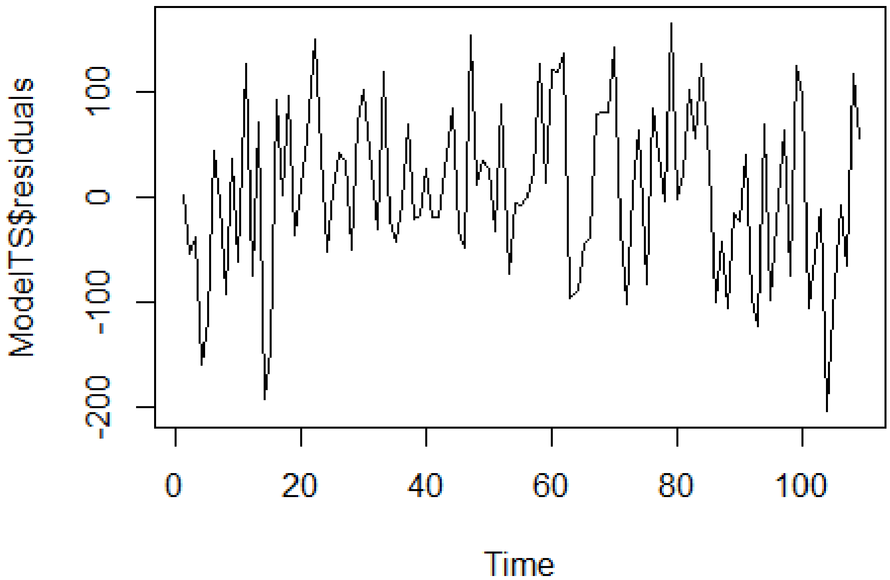


In the series plot we can see the data isn't stationary in the mean. In the ACF plot, we can see a possible cycle in the data. Considering the data isn't stationary in the mean, with the Augmented Dickey Fuller Test we will analyze if we need differences in the data.

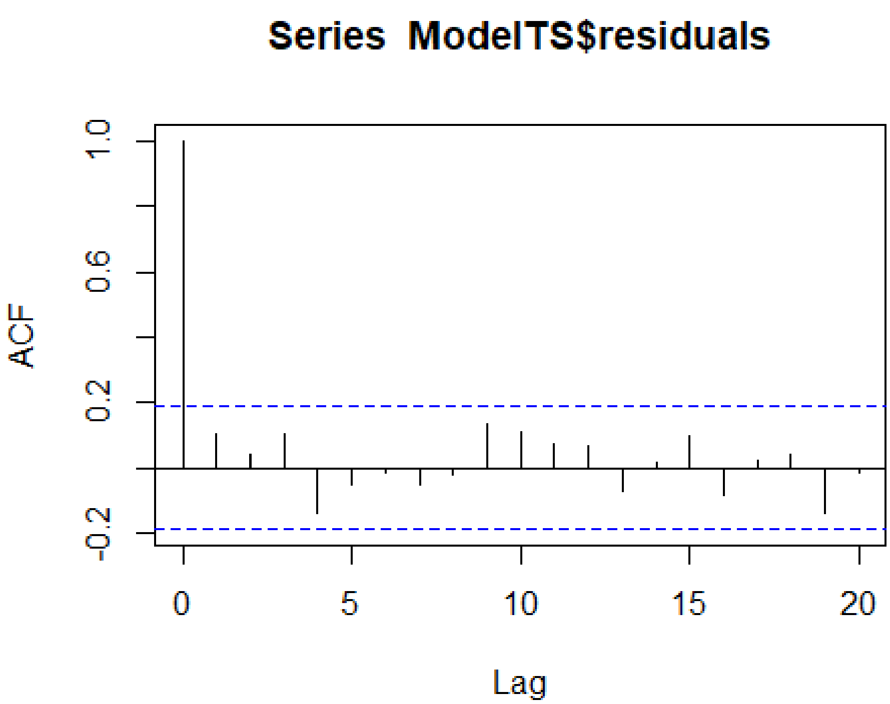
```
ndiffs(y,alpha=0.05, test=c("adf"))

## [1] 1
```

We need to apply one difference to the series.

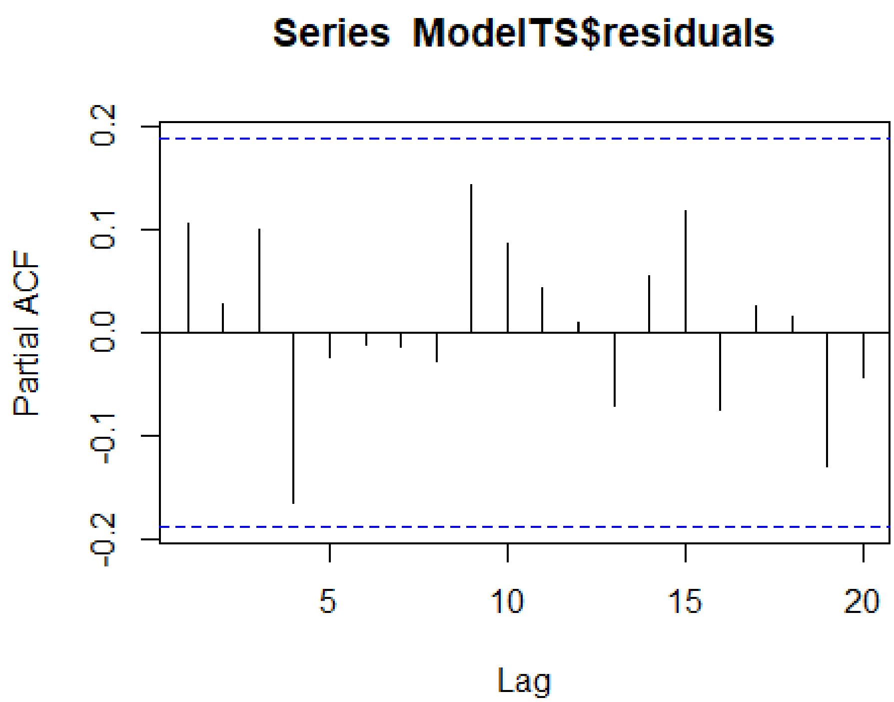
```
ModelTS<-arima(y, order = c(0,1,0))
plot(ModelTS$residuals)
```




```r
acf(ModelTS$residuals)
```




```r
pacf(ModelTS$residuals)
```




```
shapiro.test(ModelTS$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  ModelTS$residuals
## W = 0.98799, p-value = 0.4433

Box.test(ModelTS$residuals, lag = 50,type = "Ljung")

##
##  Box-Ljung test
##
## data:  ModelTS$residuals
## X-squared = 50.362, df = 50, p-value = 0.4591
```

If we apply one difference to the data, the residuals appear to be stationary. We can see:

- Constant mean
- Constant Variance
- No lags out of limits in the ACF nor PACF

With the Shapiro test we fail to reject the null hypothesis and conclude the data is Normally Distributed with an 95% confidence.

With the Box Ljung test we fail to reject the null hypothesis and conclude the data is not correlated.

With this considerations we may conclude the data is WN, SWN and GWN.

The best TS Model is ARIMA(0,1,0)

Regression Model

In the correlation plot from the beginning, we can see there is a correlation between all the different variables. Multicollinearity happens when we see such strong correlation between explanatory variables that adding variables to our model doesn't improve the final result. We will analyze if this is the case:

We will start by analyzing which is the best linear model with stepwise regression. As we can see in the following tables, the stepwise regression includes all the explanatory variables concluding that all of them are actually relevant and we do not have multicollinearity.

```
model <- IBEX~.
fit<- lm(model ,scaled_df)
stepw_mod <- ols_step_both_p(fit, pent = 0.05, prem = 0.3, details = F)
```

```
## Stepwise Selection Method
## ---------------------------
##
## Candidate Terms:
##
## 1. Exchange.rate....
## 2. Short.term.rate
## 3. Long.term.rate
##
## We are selecting variables based on p value...
##
## Variables Entered/Removed:
##
## - Long.term.rate added
## - Short.term.rate added
## - Exchange.rate.... added
##
##
## Final Model Output
## ------------------
##
##                           Model Summary
## -----------------------------------------------------------------
## R                       0.973       RMSE                 129.323
## R-Squared               0.947       Coef. Var              4.306
## Adj. R-Squared          0.946       MSE                16724.467
## Pred R-Squared          0.943       MAE                  102.455
```

```
## ---------------------------------------------------------------
##  RMSE: Root Mean Square Error
##  MSE: Mean Square Error
##  MAE: Mean Absolute Error
##
##                                    ANOVA
## --------------------------------------------------------------------
-----
##                      Sum of
##                     Squares         DF     Mean Square       F
Sig.
## --------------------------------------------------------------------
-----
## Regression    31412217.386          3    10470739.129    626.073
0.0000
## Residual       1756069.036        105       16724.467
## Total         33168286.422        108
## --------------------------------------------------------------------
-----
##
##                                         Parameter Estimates
## --------------------------------------------------------------------
------------------------------
##              model        Beta    Std. Error    Std. Beta      t
Sig        lower       upper
## --------------------------------------------------------------------
------------------------------
##       (Intercept)    5231.677       376.906                 13.881
0.000    4484.341    5979.013
##    Long.term.rate    -172.164        18.923       -0.476    -9.098
0.000    -209.685    -134.642
##   Short.term.rate     -88.705        10.505       -0.408    -8.444
0.000    -109.535     -67.874
## Exchange.rate....     783.344       288.440        0.132     2.716
0.008     211.421    1355.266
## --------------------------------------------------------------------
------------------------------

IBEX <- scaled_df$IBEX
Dep_Variables <- as.data.frame(subset(scaled_df, select = -c(IBEX)))
Ex_rate <- scaled_df$Exchange.rate....
Short_Rate <- scaled_df$Short.term.rate
Long_Rate <- scaled_df$Long.term.rate

Modellr=lm(scaled_df$IBEX~Ex_rate+Short_Rate+Long_Rate)
summary(Modellr)

##
## Call:
## lm(formula = scaled_df$IBEX ~ Ex_rate + Short_Rate + Long_Rate)
```

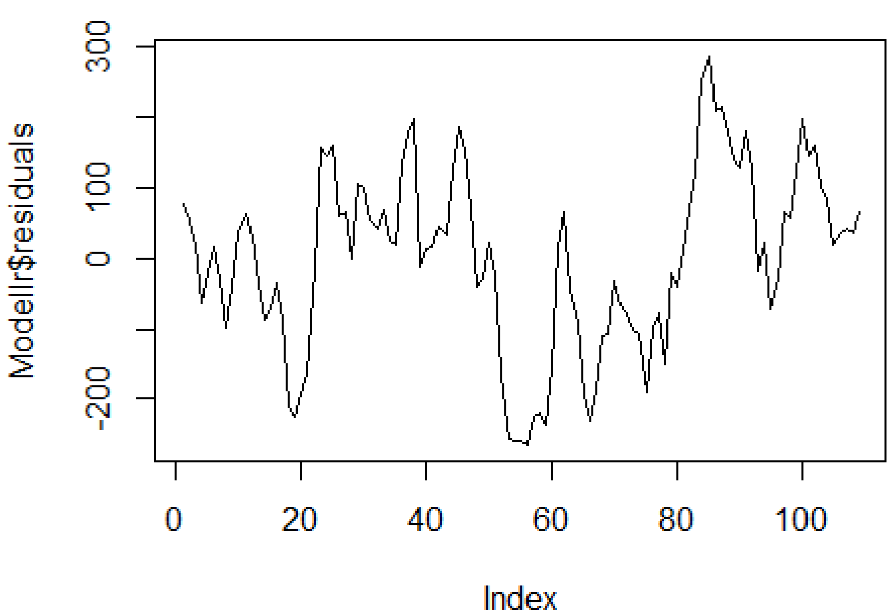
```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -264.47  -78.77   16.29   76.58  285.68
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5231.68     376.91  13.881  < 2e-16 ***
## Ex_rate       783.34     288.44   2.716  0.00773 **
## Short_Rate    -88.70      10.51  -8.444 1.84e-13 ***
## Long_Rate    -172.16      18.92  -9.098 6.45e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 129.3 on 105 degrees of freedom
## Multiple R-squared:  0.9471, Adjusted R-squared:  0.9455
## F-statistic: 626.1 on 3 and 105 DF,  p-value: < 2.2e-16
```

Our final linear model for IBEX includes the three explanatory variables as stated in the previous coefficients table. IBEX is a function of Ex_Rate, Short_Rate and Long_Rate.

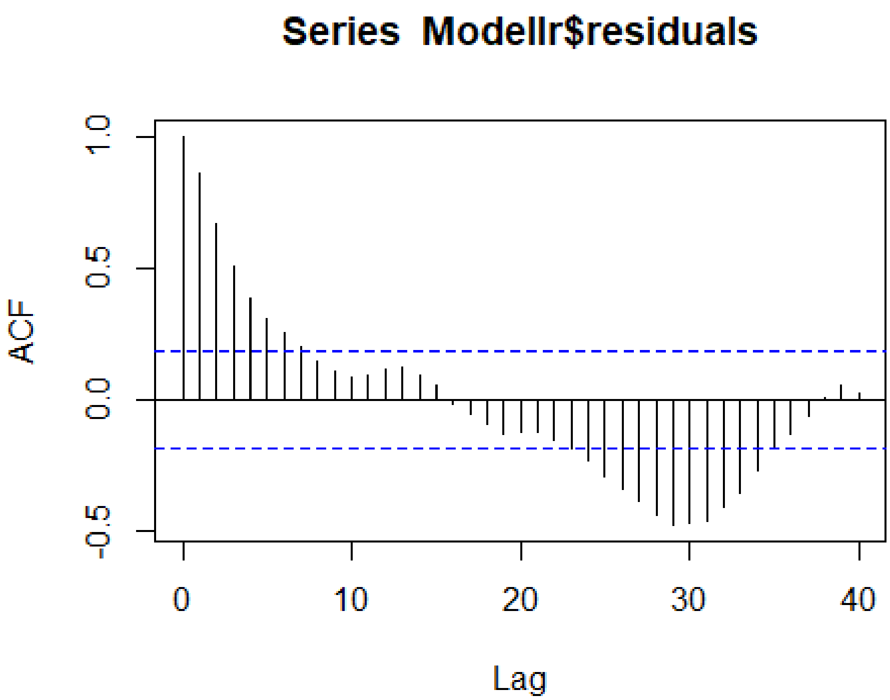
Regression model WITH Time Series Errors

Now that we have a *linear model*, we will see if the residuals are White Noise

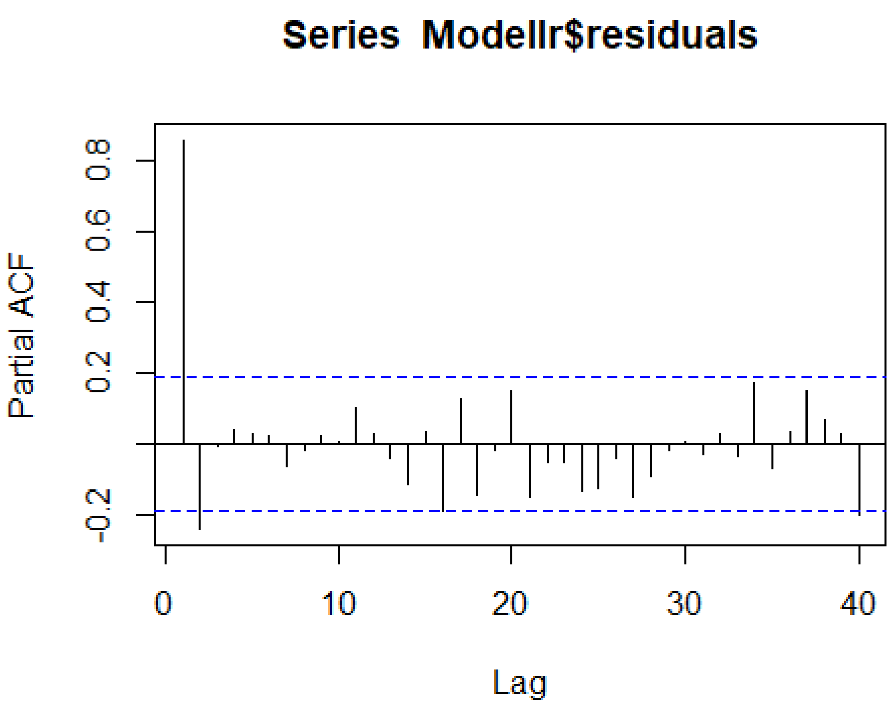
```
plot(Modellr$residuals, type='l')
```



```r
acf(Modellr$residuals, lag=40)
```

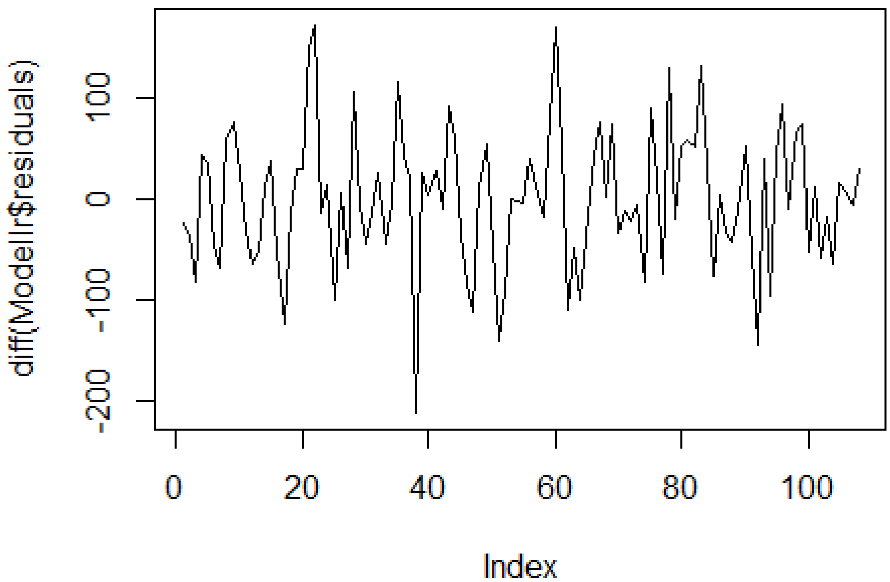



```r
pacf(Modellr$residuals, lag=40)
```

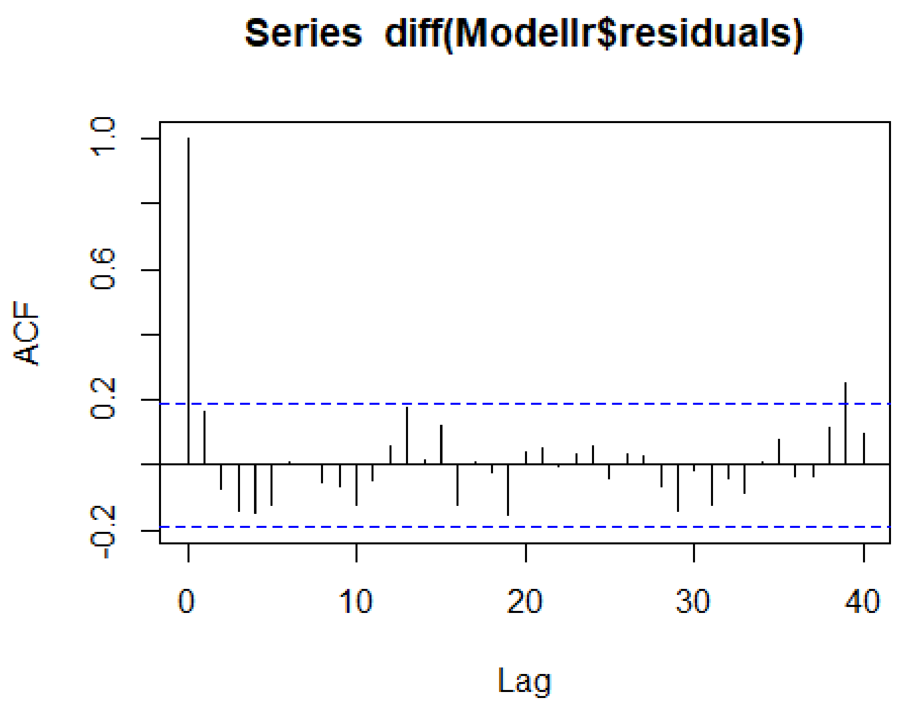


We can see the residuals are not White Noise so we will apply one difference to the model and plot the ACF and PACF.

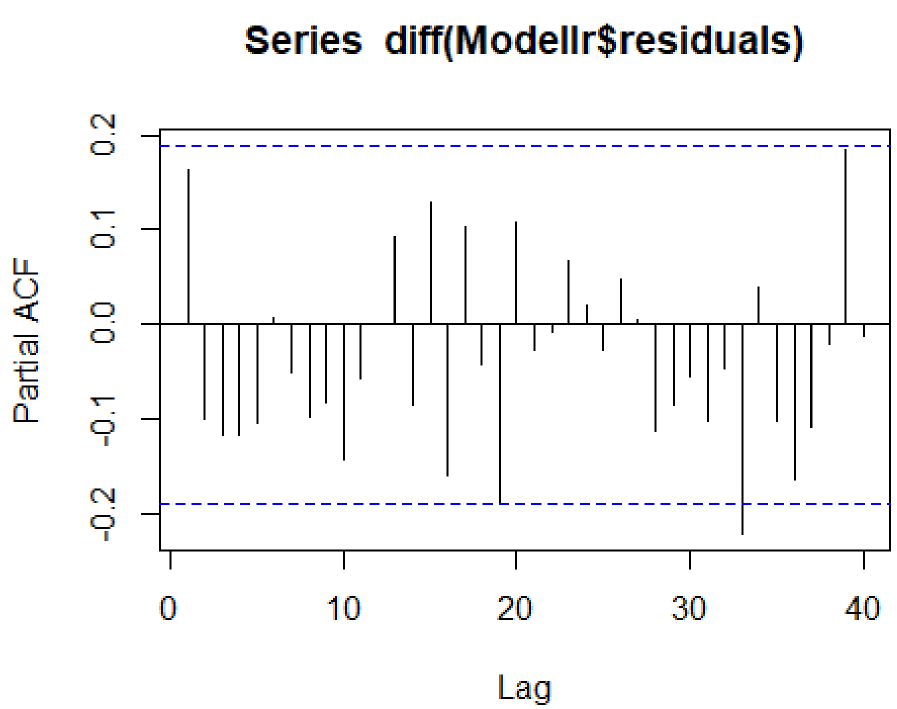
```
plot(diff(Modellr$residuals), type='l')
```




```
acf(diff(Modellr$residuals), lag=40)
```

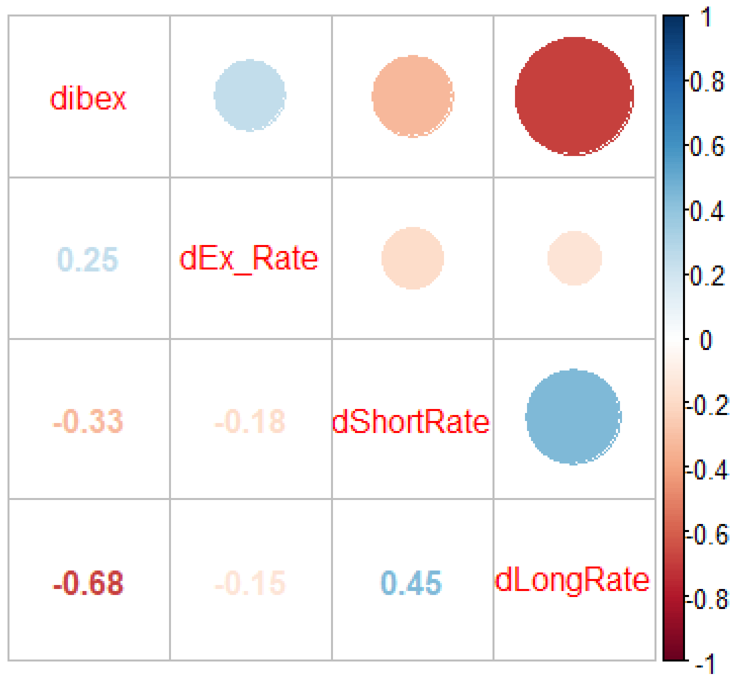



```
pacf(diff(Modellr$residuals), lag=40)
```



We can see that by applying one difference to the linear model (dependent and independent variables), the residuals appear to be white noise.

```
#These are the new time series we have to work with:
dibex <- diff(IBEX)
dEx_Rate <- diff(Ex_rate)
dShortRate <- diff(Short_Rate)
dLongRate <- diff(Long_Rate)

ddf <- cbind(dibex,dEx_Rate,dShortRate,dLongRate) #differenced complete
matrix to see correlations
Dexpl <- cbind(dEx_Rate,dShortRate,dLongRate) #differenced dependent
variables only to run the lm

cddf <- cor(ddf)
corrplot.mixed(cddf)
```



After applying one difference to the different variables, the correlations are smaller than those from the original data.

We will now build our linear regression model with these variables.

```
Modellrts <- lm(dibex~dEx_Rate+dShortRate+dLongRate)
summary(Modellrts)

##
## Call:
```

```
## lm(formula = dibex ~ dEx_Rate + dShortRate + dLongRate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -150.683  -34.686    2.572   36.021  167.939
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.5640     5.6745   0.628   0.5313
## dEx_Rate     767.1516   360.1982   2.130   0.0355 *
## dShortRate    -0.9214    14.4686  -0.064   0.9493
## dLongRate   -200.5001    23.8518  -8.406 2.37e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 58.32 on 104 degrees of freedom
## Multiple R-squared:  0.4903, Adjusted R-squared:  0.4755
## F-statistic: 33.34 on 3 and 104 DF,  p-value: 3.511e-15
```

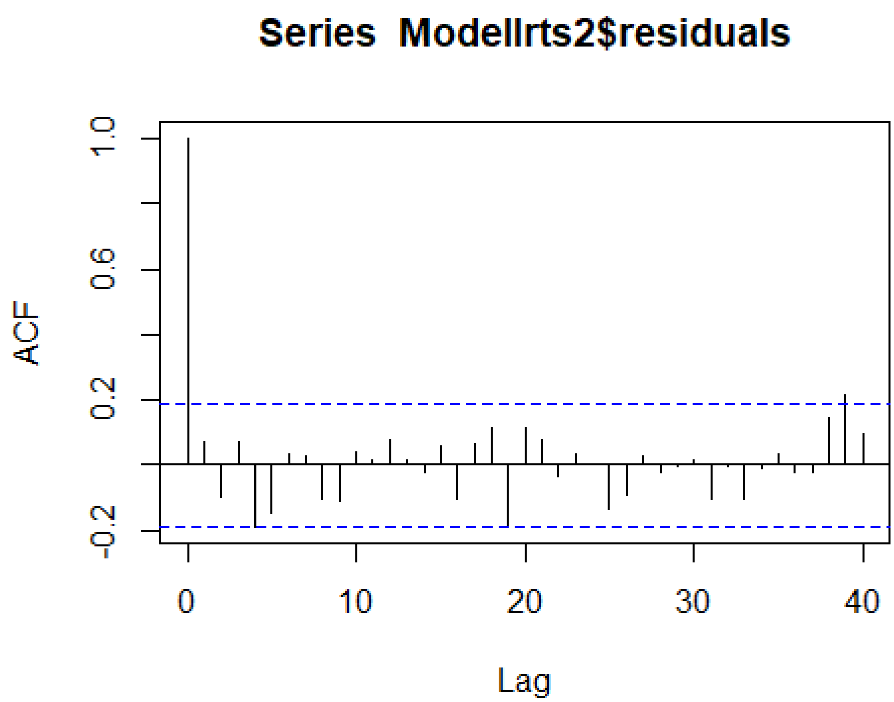
In the previous model we can see dShortRate is not significant in the model. We will compute a new linear model removing this variable.

```
Modellrts2<- lm(dibex~dEx_Rate+dLongRate)
summary(Modellrts2)
```

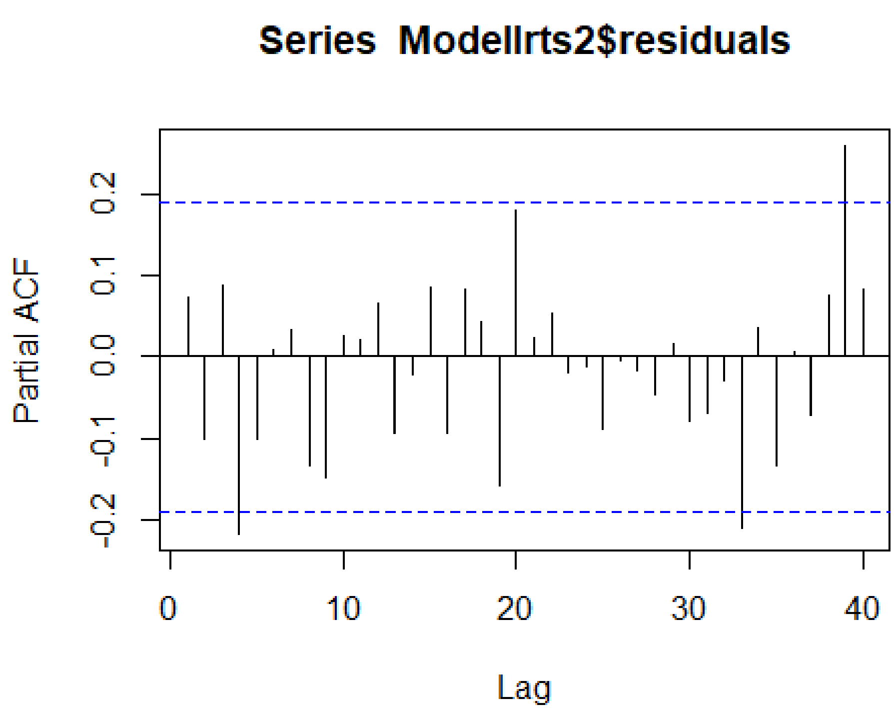
```
##
## Call:
## lm(formula = dibex ~ dEx_Rate + dLongRate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -150.790  -34.078    2.372   35.934  168.101
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.594      5.628   0.639   0.5245
## dEx_Rate     770.234    355.234   2.168   0.0324 *
## dLongRate   -201.156     21.414  -9.394 1.41e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 58.04 on 105 degrees of freedom
## Multiple R-squared:  0.4902, Adjusted R-squared:  0.4805
## F-statistic: 50.49 on 2 and 105 DF,  p-value: 4.335e-16
```

Now we will look to the ACF and PACF of the residuals.

```
acf(Modellrts2$residuals,lag=40)
```




```r
pacf(Modellrts2$residuals,lag=40)
```




```r
Box.test(Modellrts2$residuals)
```

```
##
##  Box-Pierce test
##
## data:  Modellrts2$residuals
## X-squared = 0.58416, df = 1, p-value = 0.4447
```

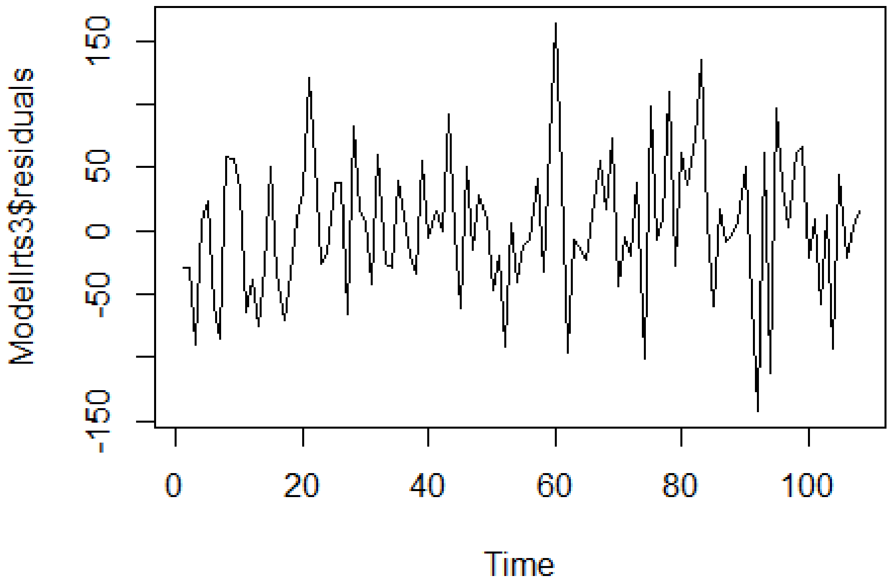
With the Box test we fail to reject the Null Hypothesis and conclude the data isn't correlated, so we may have WN and work with this first model. ARIMA(0,1,0)

However, in the PACF we can see a lag in 4. We will apply a ARIMA(4,0,0) to the transformed model (ARIMA(4,1,0) to the original data)

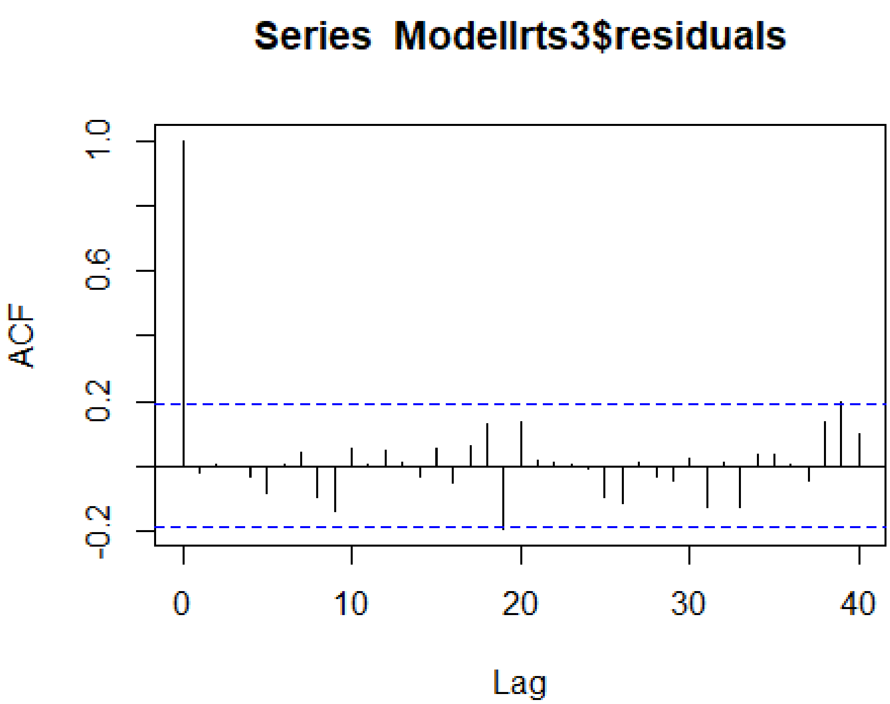
```
Dexpl2 <- cbind(dEx_Rate,dLongRate)
Modellrts3<-arima(dibex,order=c(4,0,0),xreg=Dexpl2,include.mean=F)
Modellrts3

##
## Call:
## arima(x = dibex, order = c(4, 0, 0), xreg = Dexpl2, include.mean = F)
##
## Coefficients:
##          ar1      ar2     ar3      ar4   dEx_Rate  dLongRate
##       0.1443  -0.1283  0.1276  -0.2341  1000.6455  -185.0093
## s.e.  0.0977   0.0941  0.0958   0.0957   323.1217    20.6746
##
## sigma^2 estimated as 3033:  log likelihood = -586.33,  aic = 1186.66
```

We can see that ar4 is significant while ar1, ar2, and ar3 aren't. In this new model we will test if the residuals are WN.

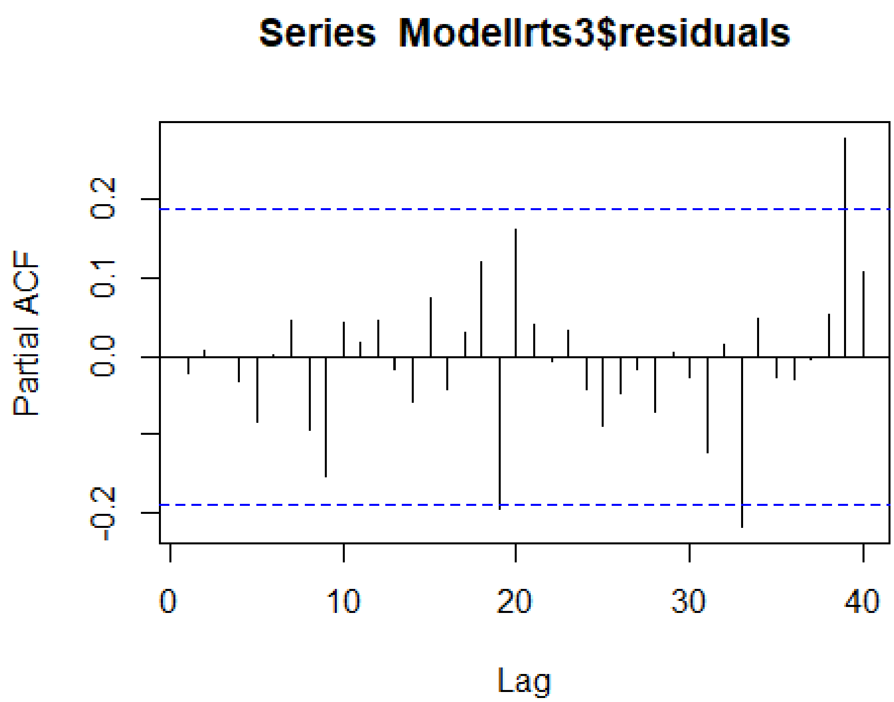
```
plot(Modellrts3$residuals)
```




```
acf(Modellrts3$residuals,lag=40)
```




```
pacf(Modellrts3$residuals,lag=40)
```



We now can see constant mean, constant variance and no lags out of limits for the ACF and PACF, therefore the residuals appear to be *WN*. (We only see lags out of bounds in the PACF in 33 and 39. However the will not be considered because they add complexity to the model and may not be relevant to exlain Yt)

With the Shapiro test we fail to reject the null hypothesis and conclude the residuals are normally distributed.

With the Box test we fail to reject the null hypothesis and conclude the residuals are uncorrelated.

Taking this into consideration, we have WN, SWN and GWN residuals.

```
shapiro.test(Modellrts3$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  Modellrts3$residuals
## W = 0.99569, p-value = 0.9846

Box.test(Modellrts3$residuals)

##
##  Box-Pierce test
##
## data:  Modellrts3$residuals
## X-squared = 0.049185, df = 1, p-value = 0.8245
```

Model comparison:

Now we have four models:

- TS only model –> ModelTS –> ARIMA(0,1,0)
- Linear Regression only model –> Modellr –> IBEX = 5231.68 + 783.34ExRate - 88.7ShortRate - 172.16LongRate
- Linear Regression with TS Errors –> Modellrts3 –> IBEX ~ f(Ex_Rate, LongRate) ARIMA(4,1,0)
- Linear Regression with TS Errors –> IBEX ~ f(Ex_Rate, LongRate) ARIMA(0,1,0)

We can see that the LR with TS errors may be different from the two previous models in the number of lags if we consider lag # 4 from the Autoregressive Part of the ARIMA, and different number of regressors from the linear regression model (we don't include Short Term Rate).

Now we will see which model is the best:

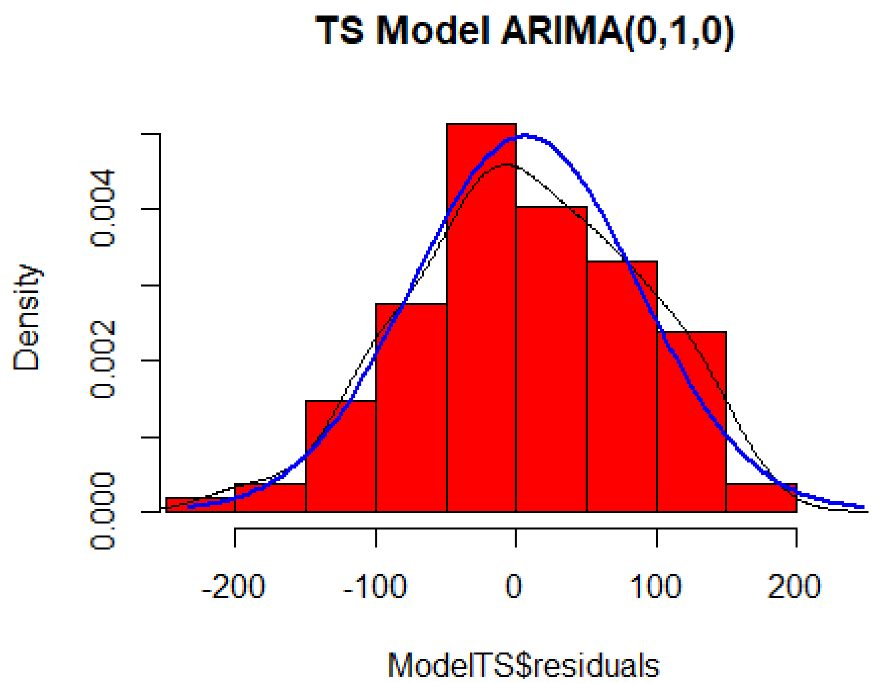
```
summary(ModelTS)

##
## Call:
## arima(x = y, order = c(0, 1, 0))
##
##
## sigma^2 estimated as 6467:  log likelihood = -627.07,  aic = 1256.13
##
## Training set error measures:
##                    ME    RMSE      MAE       MPE     MAPE      MASE
## Training set 6.382239 80.0477 64.91435 0.1737769 2.220795 0.9911989
##                   ACF1
## Training set 0.1047383

summary(Modellr)

##
## Call:
## lm(formula = scaled_df$IBEX ~ Ex_rate + Short_Rate + Long_Rate)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -264.47  -78.77   16.29   76.58  285.68
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5231.68     376.91  13.881  < 2e-16 ***
## Ex_rate       783.34     288.44   2.716  0.00773 **
## Short_Rate    -88.70      10.51  -8.444 1.84e-13 ***
## Long_Rate    -172.16      18.92  -9.098 6.45e-15 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 129.3 on 105 degrees of freedom
## Multiple R-squared:  0.9471, Adjusted R-squared:  0.9455
## F-statistic: 626.1 on 3 and 105 DF,  p-value: < 2.2e-16
```

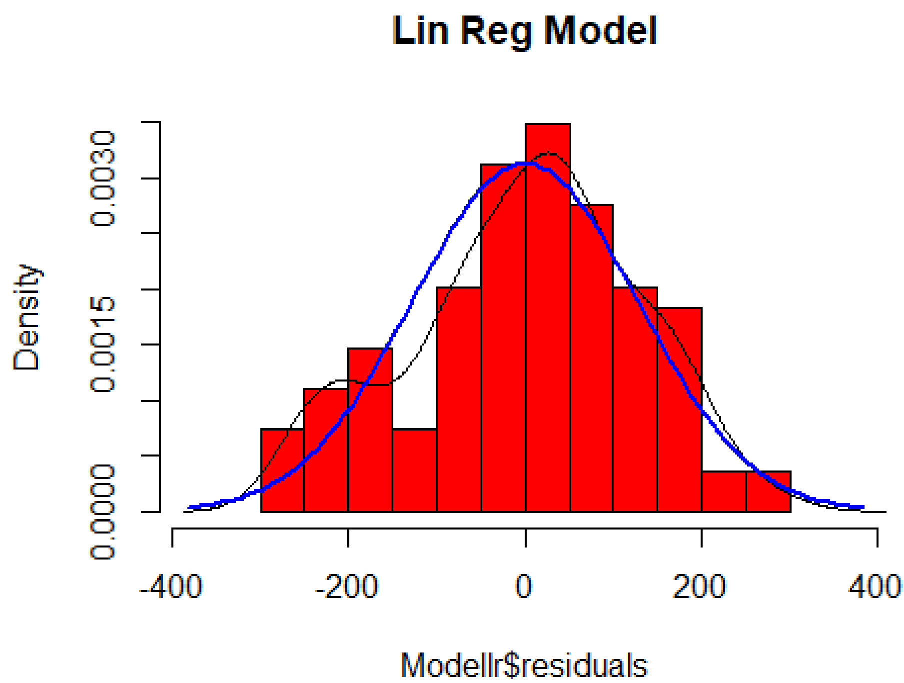
```
summary(Modellrts3)
```

```
##
## Call:
## arima(x = dibex, order = c(4, 0, 0), xreg = Dexpl2, include.mean = F)
##
## Coefficients:
##          ar1      ar2     ar3      ar4   dEx_Rate  dLongRate
##       0.1443  -0.1283  0.1276  -0.2341  1000.6455  -185.0093
## s.e.  0.0977   0.0941  0.0958   0.0957   323.1217    20.6746
##
## sigma^2 estimated as 3033:  log likelihood = -586.33,  aic = 1186.66
##
## Training set error measures:
##                    ME     RMSE      MAE  MPE MAPE      MASE
ACF1
## Training set 3.474125 55.07672 43.35776 -Inf  Inf 0.4944346 -
0.02134058
```

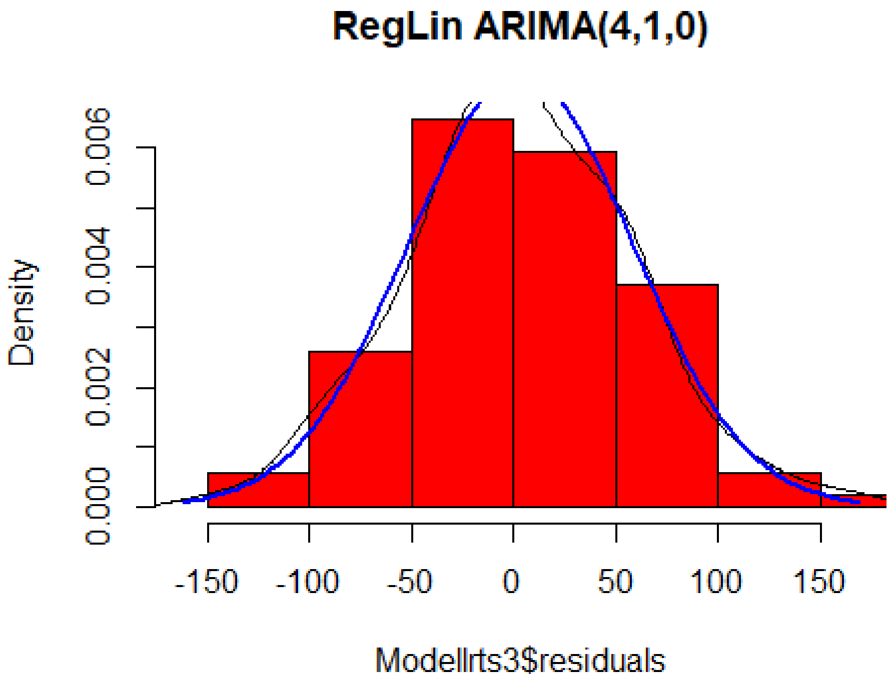
```
# Histogram Model TS
hist(ModelTS$residuals,prob=T,xlim=c(mean(ModelTS$residuals)-
3*sd(ModelTS$residuals),mean(ModelTS$residuals)+3*sd(ModelTS$residuals)),
col="red",main = "TS Model ARIMA(0,1,0)")
lines(density(ModelTS$residuals))
mu<-mean(ModelTS$residuals)
sigma<-sd(ModelTS$residuals)
x<-seq(mu-3*sigma,mu+3*sigma,length=100)
yy<-dnorm(x,mu,sigma)
lines(x,yy,lwd=2,col="blue")
```



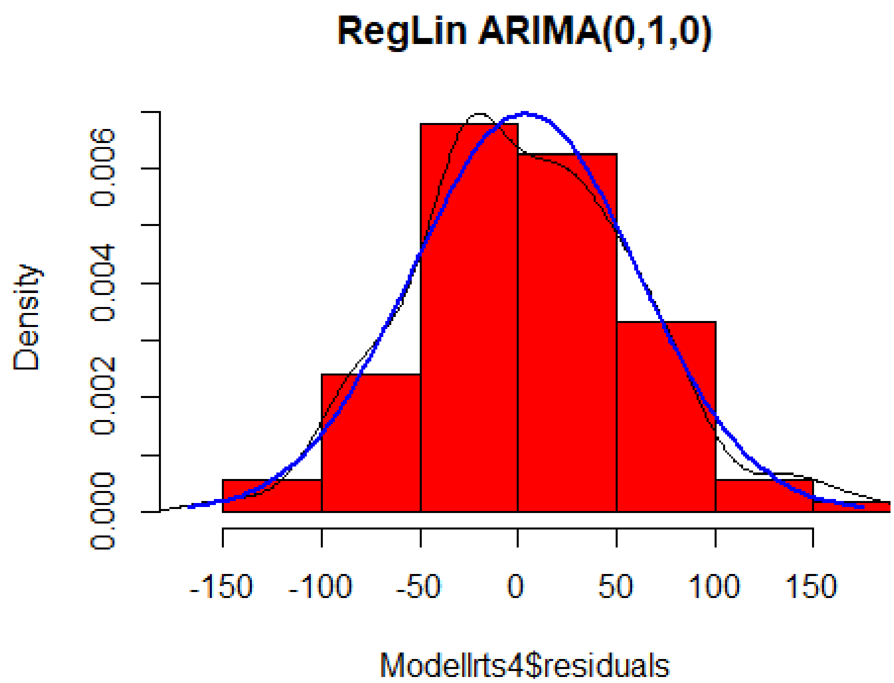

```
shapiro.test(ModelTS$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  ModelTS$residuals
## W = 0.98799, p-value = 0.4433

#Histogram model Lin Reg
hist(Modellr$residuals, prob=T, xlim = c(mean(Modellr$residuals)-
3*sd(Modellr$residuals),mean(Modellr$residuals)+3*sd(Modellr$residuals)),
col = 'red', main = "Lin Reg Model")
lines(density(Modellr$residuals))
mu<-mean(Modellr$residuals)
sigma<-sd(Modellr$residuals)
x<-seq(mu-3*sigma,mu+3*sigma,length=100)
yy<-dnorm(x,mu,sigma)
lines(x,yy,lwd=2,col="blue")
```




```
shapiro.test(Modellr$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  Modellr$residuals
## W = 0.97898, p-value = 0.08226

#Histogram Model TS+LinReg
hist(Modellrts3$residuals, prob=T, xlim = c(mean(Modellrts3$residuals)-
3*sd(Modellrts3$residuals),mean(Modellrts3$residuals)+3*sd(Modellrts3$res
iduals)), col = 'red', main = "RegLin ARIMA(4,1,0)")
lines(density(Modellrts3$residuals))
mu<-mean(Modellrts3$residuals)
sigma<-sd(Modellrts3$residuals)
x<-seq(mu-3*sigma,mu+3*sigma,length=100)
yy<-dnorm(x,mu,sigma)
lines(x,yy,lwd=2,col="blue")
```




```
shapiro.test(Modellrts3$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  Modellrts3$residuals
## W = 0.99569, p-value = 0.9846

expl<-cbind(Ex_rate,Long_Rate)
Modellrts4 <- arima(y, order=c(0,1,0),xreg = expl,include.mean=F)
hist(Modellrts4$residuals, prob=T, xlim = c(mean(Modellrts4$residuals)-
3*sd(Modellrts4$residuals),mean(Modellrts4$residuals)+3*sd(Modellrts4$res
iduals)), col = 'red', main = "RegLin ARIMA(0,1,0)")
lines(density(Modellrts4$residuals))
mu<-mean(Modellrts4$residuals)
sigma<-sd(Modellrts4$residuals)
x<-seq(mu-3*sigma,mu+3*sigma,length=100)
yy<-dnorm(x,mu,sigma)
lines(x,yy,lwd=2,col="blue")
```




```
shapiro.test(Modellrts4$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  Modellrts4$residuals
## W = 0.99003, p-value = 0.6071

summary(Modellrts4)

##
## Call:
## arima(x = y, order = c(0, 1, 0), xreg = expl, include.mean = F)
##
## Coefficients:
##        Ex_rate  Long_Rate
##       797.6573  -201.2459
## s.e.  348.3714    21.1547
##
## sigma^2 estimated as 3288:  log likelihood = -590.54,  aic = 1187.09
##
## Training set error measures:
##                    ME   RMSE      MAE        MPE     MAPE      MASE
## Training set 3.548023 57.082 45.76891 0.07312952 1.553525 0.6988608
##                    ACF1
## Training set 0.07486168
```

The four previous models show the following RMSE (Root Mean Squared Error):

- ARIMA(0,1,0) –> RMSE = 80.05
- LM –> Residual Standard Error = 129.3
- LM+ARIMA(4,1,0) –> RMSE = 55.08
- LM+ARIMA(0,1,0) –> RMSE = 57.082

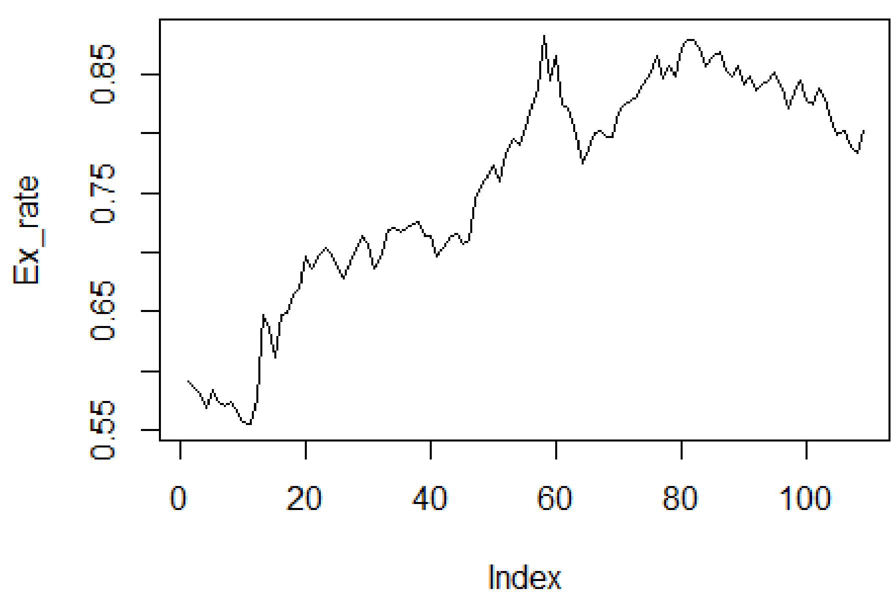
We can see the best model is the LM+ARIMA(4,1,0) considering that it has the smallest Root Mean Squared Error. However, the LM+ARIMA(0,1,0) also shows a small RMSE, similar to the LM+ARIMA(4,1,0). Maybe for simplicity, using the LM+ARIMA(0,1,0) is the best solution at the end.

We will continue our analysis with comparing these two models.

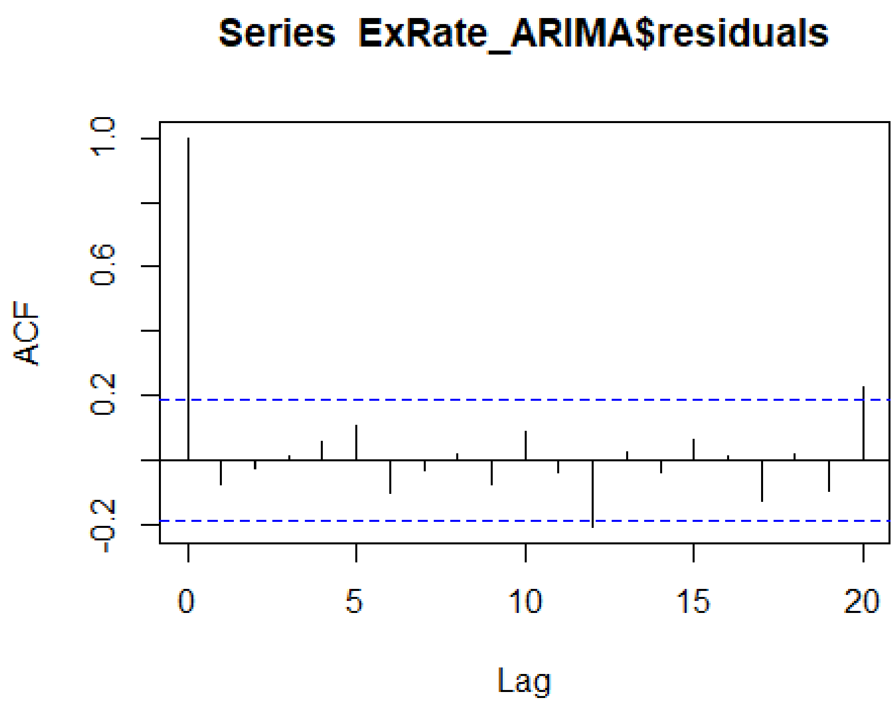
Forecasting Ex_Rate and Long Term Rate

The first step to forecast IBEX is forecasting the explanatory variables Ex_Rate and Long Term Rate. We can see that both time series with one difference are white noise.The best prediction for the residuals of these models is the mean. We will predict one step ahead of these two explanatory variables and undo the difference to predict the original series.

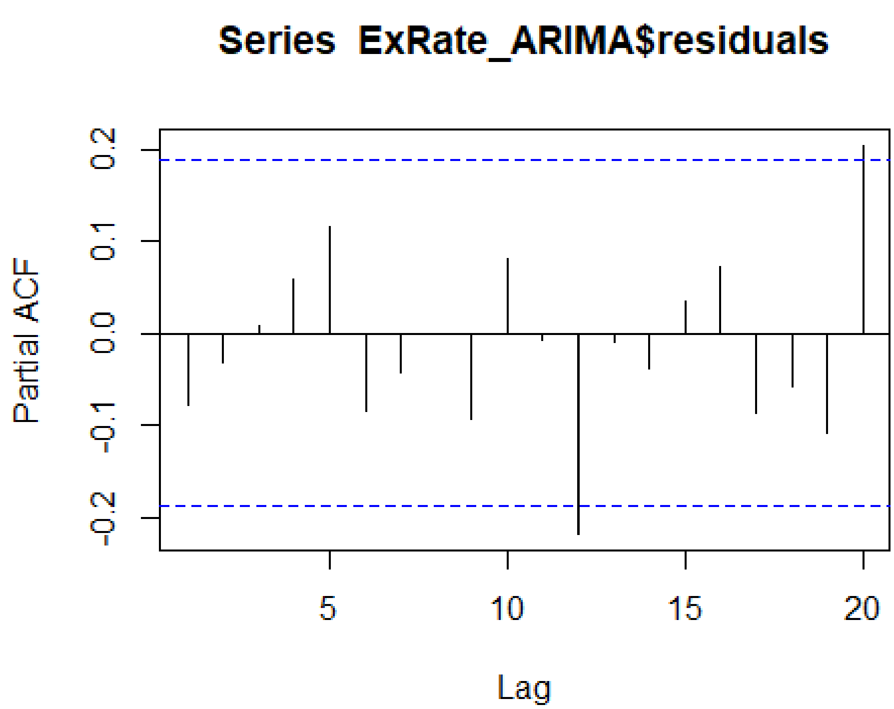
```
plot(Ex_rate, type = 'l')
```



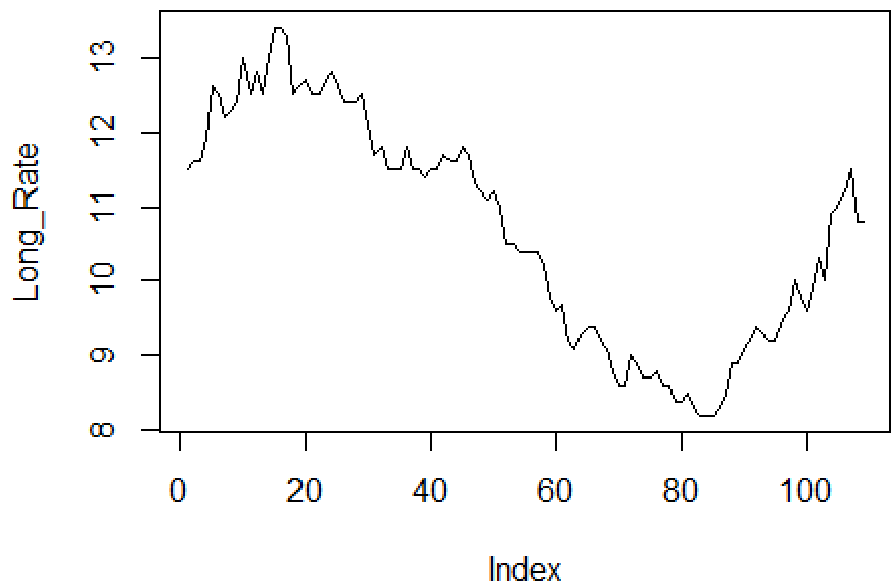

```
ExRate_ARIMA <- Arima(Ex_rate, order = c(0,1,0))
acf(ExRate_ARIMA$residuals)
```



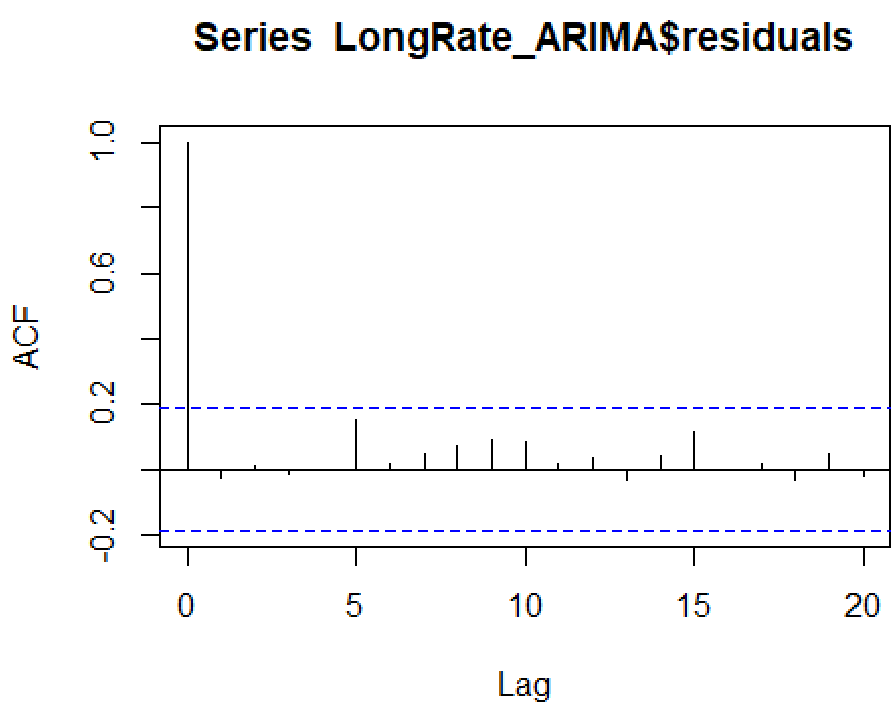

```r
pacf(ExRate_ARIMA$residuals)
```



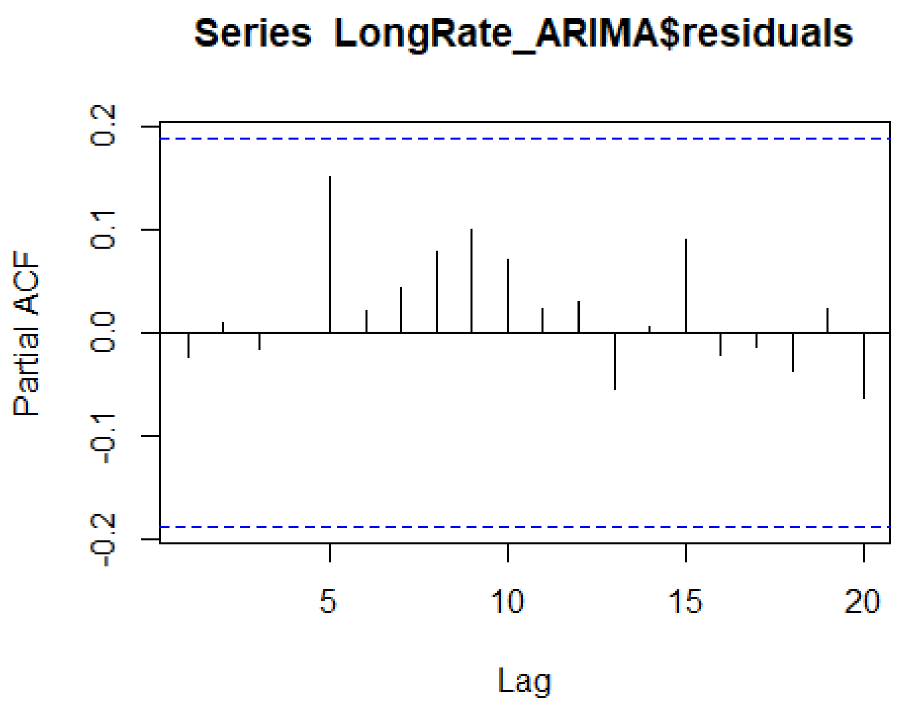

```r
plot(Long_Rate, type = 'l')
```




```
LongRate_ARIMA <- Arima(Long_Rate, order = c(0,1,0))
acf(LongRate_ARIMA$residuals)
```




```
pacf(LongRate_ARIMA$residuals)
```




```
ExRate_prediction <- predict(ExRate_ARIMA, n.ahead = 1)
LongRate_prediction<-predict(LongRate_ARIMA, n.ahead = 1)

ExRate_prediction$pred

## Time Series:
## Start = 110
## End = 110
## Frequency = 1
## [1] 0.803

LongRate_prediction$pred

## Time Series:
## Start = 110
## End = 110
## Frequency = 1
## [1] 10.8
```

Now, we will predict the IBEX in both models considering the one point prediction of the explanatory variables.

```
newdExRate <- ExRate_prediction$pred
newdLongRate <- LongRate_prediction$pred
lrpredictors <- as.matrix(cbind(newdExRate,newdLongRate))

expl<-cbind(Ex_rate,Long_Rate)
```

```r
LR_ARIMA_4_1_0 <- arima(y, order=c(4,1,0),xreg = expl,include.mean=F)
LR_ARIMA_4_1_0

##
## Call:
## arima(x = y, order = c(4, 1, 0), xreg = expl, include.mean = F)
##
## Coefficients:
##          ar1      ar2     ar3      ar4    Ex_rate  Long_Rate
##       0.1444  -0.1283  0.1276  -0.2342  1000.4160  -185.0082
## s.e.  0.0977   0.0941  0.0958   0.0957   323.1235    20.6746
##
## sigma^2 estimated as 3033:  log likelihood = -586.33,  aic = 1186.66

LR_ARIMA_0_1_0 <- arima(y, order=c(0,1,0),xreg = expl,include.mean=F)
LR_ARIMA_0_1_0

##
## Call:
## arima(x = y, order = c(0, 1, 0), xreg = expl, include.mean = F)
##
## Coefficients:
##        Ex_rate  Long_Rate
##       797.6573  -201.2459
## s.e.  348.3714    21.1547
##
## sigma^2 estimated as 3288:  log likelihood = -590.54,  aic = 1187.09

LRARIMA410_predict<-predict(LR_ARIMA_4_1_0,newxreg=lrpredictors, n.ahead
= 1)
LRARIMA410_predict

## $pred
## Time Series:
## Start = 110
## End = 110
## Frequency = 1
## [1] 3356.882
##
## $se
## Time Series:
## Start = 110
## End = 110
## Frequency = 1
## [1] 55.07672

LRARIMA010_predict<-predict(LR_ARIMA_0_1_0,newxreg=lrpredictors, n.ahead
= 1)
LRARIMA010_predict

## $pred
## Time Series:
```

```
## Start = 110
## End = 110
## Frequency = 1
## [1] 3357
##
## $se
## Time Series:
## Start = 110
## End = 110
## Frequency = 1
## [1] 57.34402
```

We see the following one point predictions:

- LR_ARIMA(0,1,0) –> Pred = 3357.00, S.E. = +- 57.34
- LR_ARIMA(4,1,0) –> Pred = 3356.88, S.E. = +- 55.08

The predictions are very close (considering they are only one point prediction) and the standard error is slightly smaller in the LR_ARIMA(4,1,0). However, the LR_ARIMA(0,1,0) model may be simpler and does not rely on data from 4 periods in the past to predict the next period.

Final IBEX Prediction

The previous prediction was carried out forecasting the independent variables. However, the company Analistas Cuantitativos de Inversiones S.A. wants to forecast the IBEX based in the following forecast for the variables:

- Long Term Rate: 10.76%
- Short Term Rates: 7.6%
- Exchange rate: 0.781 ???/$

```
newdExRate <- 0.781
newdLongRate <- 10.76
lrpredictors <- as.matrix(cbind(newdExRate,newdLongRate))

LRARIMA410_predict<-predict(LR_ARIMA_4_1_0,newxreg=lrpredictors, n.ahead
= 1)
LRARIMA410_predict

## $pred
## Time Series:
## Start = 110
## End = 110
## Frequency = 1
## [1] 3342.273
##
## $se
## Time Series:
## Start = 110
```

```
## End = 110
## Frequency = 1
## [1] 55.07672
```

```
LRARIMA010_predict<-predict(LR_ARIMA_0_1_0,newxreg=lrpredictors, n.ahead
= 1)
LRARIMA010_predict
```

```
## $pred
## Time Series:
## Start = 110
## End = 110
## Frequency = 1
## [1] 3347.501
##
## $se
## Time Series:
## Start = 110
## End = 110
## Frequency = 1
## [1] 57.34402
```

With these new variable forecasts we see the following point predictions:

- LR_ARIMA(0,1,0) –> Pred = 3347.501, S.E. = +- 57.34
- LR_ARIMA(4,1,0) –> Pred = 3342.273, S.E. = +- 55.08