TIMESERIES ASSIGNMENT REPORT

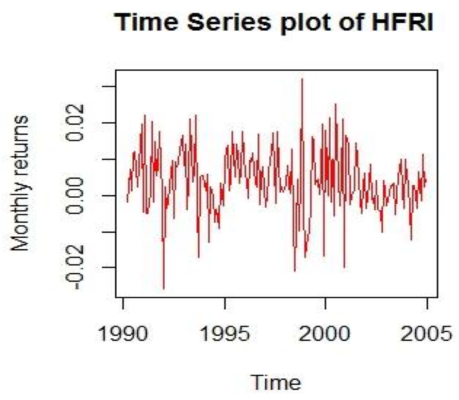
This report contains the description of the steps taken for the timeseries analysis of dependent variable y8. The implementation was done in R and the following libraries were used:

- library("xlsx")
- library(urca)
- library("rugarch")

Note: Rugarch package combines regression, ARMA and GARCH models in one step.

TIMESERIES

Create time series plot of y8.



Observations:

1. Mean value is around zero
2. There seems to be heteroscedasticity issue

TEST STATIONARITY:

In timeseries we first need to make sure that they are stationary. In order to use regression, ARCH and GARCH models to create a predictive model it is imperative that our timeseries are stationary. A way to turn non-stationary timeseries into stationary is to transform the data using logarithmic function. In R to check if timeseries are stationary we do a Dickey-fuller test:

Dickey fuller test without drift.

m=ar(j)

res<-ur.df(j,type="none",lag=m$order-1)

RESULTS:

```
###############################################
# Augmented Dickey-Fuller Test Unit Root Test #
###############################################

Test regression none


Call:
lm(formula = z.diff ~ z.lag.1 - 1)

Residuals:
      Min        1Q    Median        3Q       Max
-0.025788 -0.002705  0.002895  0.009073  0.031477

Coefficients:
        Estimate Std. Error t value Pr(>|t|)
z.lag.1  -0.8521     0.0748  -11.39   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '

Residual standard error: 0.009845 on 175 degrees of freedom
Multiple R-squared:  0.4258,    Adjusted R-squared:  0.4225
F-statistic: 129.8 on 1 and 175 DF,  p-value: < 2.2e-16


Value of test-statistic is: -11.3918

Critical values for test statistics:
      1pct  5pct 10pct
tau1 -2.58 -1.95 -1.62
```

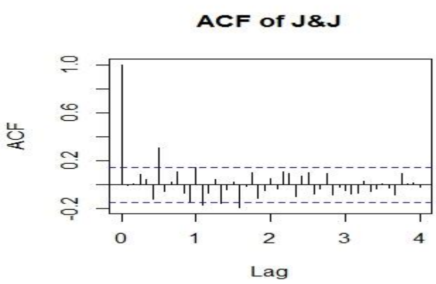
It seems that the y8 timeseries is stationary.

IDENTIFICATION STEP

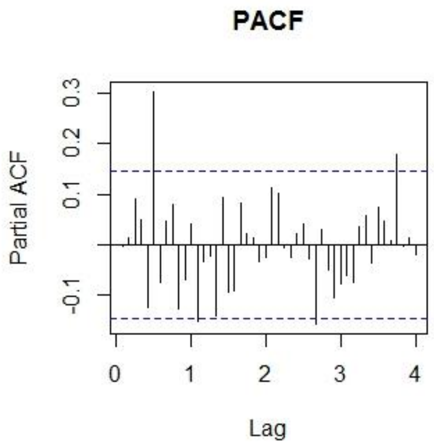
After checking stationarity we have to check possible autocorrelation issue:

***ACF** and **PACF** plots.*

ACF(J):



This is indicator of an MA(6) model to correct autocorrelation at lag6



Indicator of AR(6) model to correct partial autocorrelation

```
        Box-Ljung test

data:  j
X-squared = 34.206, df = 12, p-value = 0.0006259
```

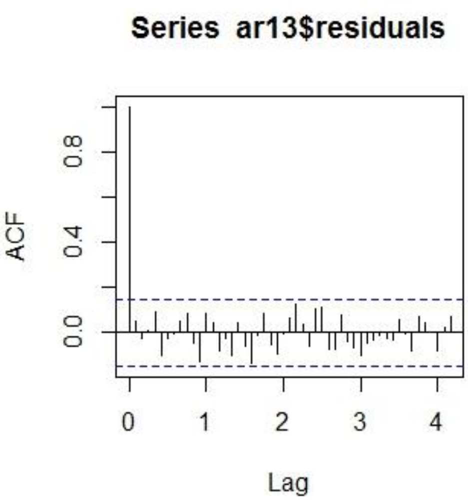
As p-value<0.05, there is autocorrelation

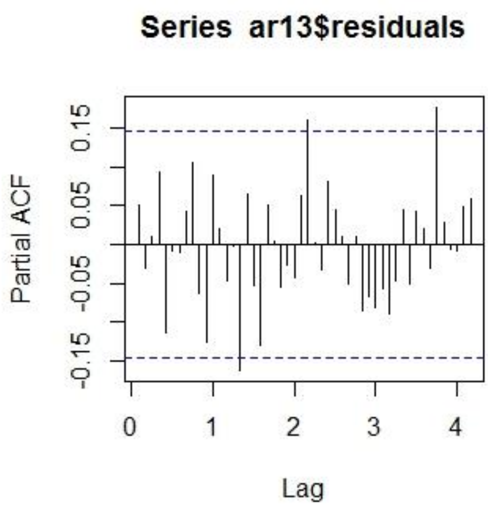
ESTIMATION OF ARMA MODEL

Three models tested:

1. AR(13)

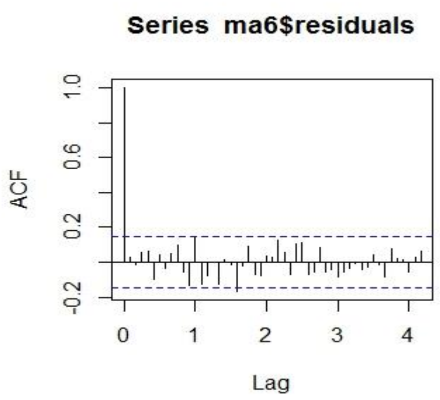
   with all coef fixed to 0 except for lag6, lag13

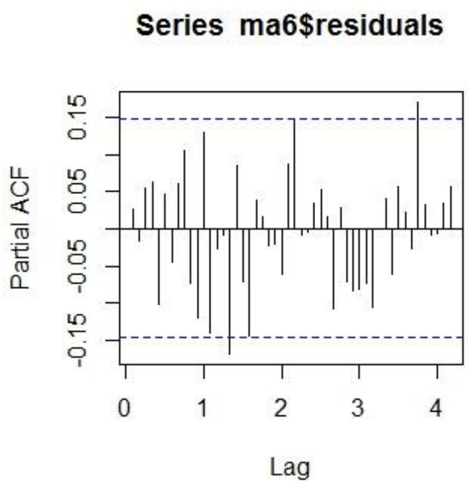




Comments:
It is a good fitting model. The issues in big lags are not a problem as they are not many. (AIC=-1174)

2. MA(6):





This is also a good fit(AIC=-1168)

3. ARMA(66)
   This is also a good fit with AIC=-1170 and because AR(13)had better AIC and was less complex I chose AR(13)

CHECKING HETEROSCEDASTICITY ISSUE:

To test heteroscedasticity, we made an ACF(j^2) and PACF(J^2) plot and a BOX.TEST and indeed the timeseries have heteroscedasticity issue. (see R code)

The model created using timeseries is the AR(13):

y_t=0.0039+0.2959y_t-6-0.1560y_t-13+et
AIC=-1174
BIC=-1161.774

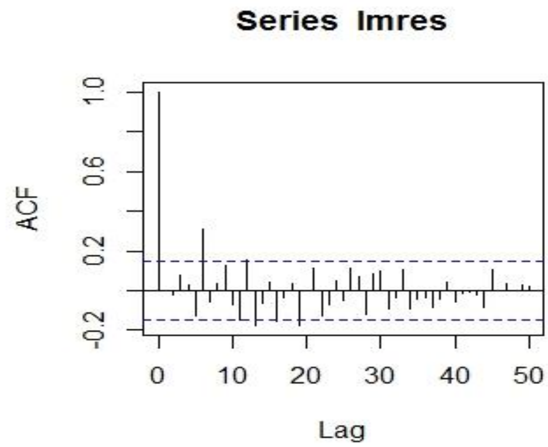
REGRESSION ANALYSIS:

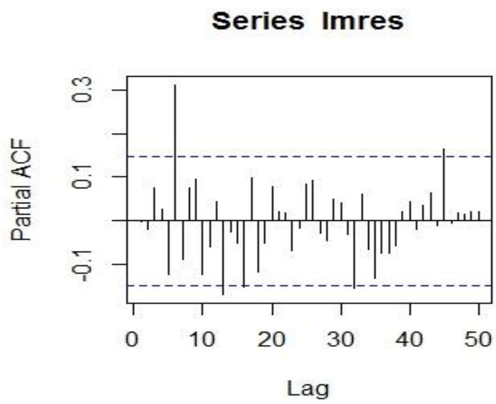
The regression model is first created using all independent variables and calculated A IC= -1152.753. The process to find a better model is to start removing independent v ariables x that are not significant. To do that we first need to check if the residuals of the regression model are:

- Independent
- With constant variance
- And normally distributed

If not, modelling the residuals using ARCH and GARCH models is imperative, so that we end up with a regression model where the residuals have no autocorrelation and heteroscedasticity issue. If residuals are not modelled we cannot decide which indep endent variable x of the regression is insignificant to remove it from the model.
The steps taken to fix residuals until the point where we can remove an independent variable x (if it is insignificant) are done repeatedly until we end up with only significa nt independent variables in our model.

Step 1: Testing autocorrelation of residuals of the regression by making an ACF and a PACF plot of the residuals.

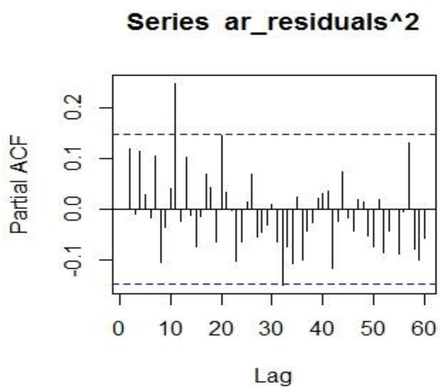


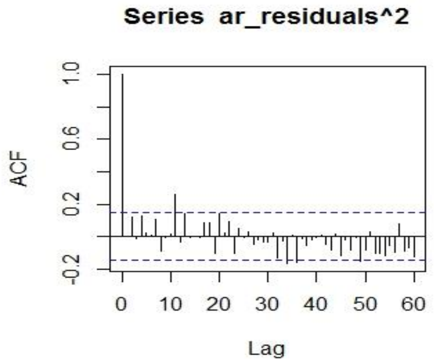


Observing that there is autocorrelation of the residuals at lag5 for ACD and lag 6 for PACF we do a Box.test to certify it. Indeed, there is autocorrelation issue. Thus, we need to use ARMA models to construct the one that solves the problem with the minimum AIC.

The models tried are AR(13) and ARMA(6,6). Both are fixing the problem but AR(13) has better AIC and it is a simpler model as only lags 6, 13 are included.

Step 2: Testing Heteroscedasticity of residuals of the AR model





It seems that there is no significant heteroscedasticity issue. To verify that we do a BOX.TEST on squared residual of AR.pvalue is less than 0.05 so Ho is rejected and so we have heteroscedasticity.

Fixing Heteroskedasticity

To fix heteroscedasticity I use rugarch library which combines ARMA model regression and GARCH. We try some variations of GARCH models which can be found in the code, and then we plot the residuals to check the autocorrelation. As ACF and PACF of residuals have no autocorrelation we do ACF and PACF on squared residuals to check that heteroscedasticity issue is also solved. Finally we plot the model to verify that the distribution of residuals are well fitted with normal distribution. If the underlying distribution in not normal we try to find an distribution that fits the data.

Step3: Normality test
After heteroscedasticity of the residuals is fixed, we check normality by doing a shapiro.test. If pvalue is less than 0.05 then the distribution of the residuals is normal. If the underlying distribution in not normal we try to find another distribution describing the residuals. For example, T-student distribution which is commonly used in finance. Now that the residuals are modelled, we can remove the variables of the regression model that are not significant and repeat the previous step.

After repeating the aforementioned process multiple times, we end up with our REGRESSION-ARMA-GARCH MODEL:

$Y_t = 0.020039*x_1+0.084141*x_2+ 1.037242*x_{14}+u_t$
$u_t = 0.354169*u_{t-6} - 0.094169*u_{t-13} + e_t$
$e_t \sim N(0, \sigma_t^2)$
$\sigma_t^2 = 0.000026 + 0.455543*e_{t-11}^2 + 0.166230*\sigma_{t-11}^2$

aic = -6.75
bic = -6.56

FORECASTS

After repeating the above process iteratively we end up with our REGRESSION-ARMA-GARCH MODEL.

To forecast yt+1 I also use rugarch library where I input my model and the number of periods I would like to forecast. To check the quality of my forecast I use MSE criteria and to check if my forecasts predict the direction of the true values, I use hit ratio. My forecasts on y8 had MSE= 0.0006993568 and hit ratio=75% which is satisfying.