ΧΑΤΖΟΠΟΥΛΟΣ ΓΕΡΑΣΙΜΟΣ

ΑΝΑΛΥΣΗ ΚΑΙ ΠΡΟΒΛΕΨΗ TIMESERIES

Σκοπός μου είναι η πλήρης ανάλυση για το US Mutual Fund JTUAX με την δημιουργία της κατάλληλης χρονοσειράς και του κατάλληλου προβλεπτικού μοντέλου στην γλωσσά R

Πιο αναλυτικά:

1. I constructed an appropriate time series model (AR, MA, ARMA).

2. Developed an appropriate regression model by eliminating potential autocorrelation or heteroscedasticity problems.

3. I assessed the goodness of fit of these models based on the AIC and BIC information criteria

4. I constructed forecasts of the analyzed series with various ways

5. Evaluation of my prediction based on specific criteria like the mean square prediction error and the Hit ratio

(The independent variables i used to refer to monthly returns for the variables)

.
.
.
.
.
.
.
.
.
.
.
.

FUND JTUAX

1. Καταρχήν πρέπει να βγάλουμε τις NA τιμές και να ξεκινήσουμε απτήν πρώτη ημερομηνία που ξεκινάει το fund μας το ίδιο πρέπει να κάνουμε και για τα factors πρέπει να σταθμίσουμε να ξεκινάνε απτήν ίδια ημερομηνία δηλαδή και ας έχει τιμές πιο πριν γιατί πρέπει να ξεκινά απτό ίδιο σημείο

```
factors<-data.frame(FACTORS[77:384,])
 jtuax<-data.frame(funds$JTUAX[78:385])
```

Εισαγωγή διαφορών πακέτων που θα χρειαστώ στην ανάλυση μου

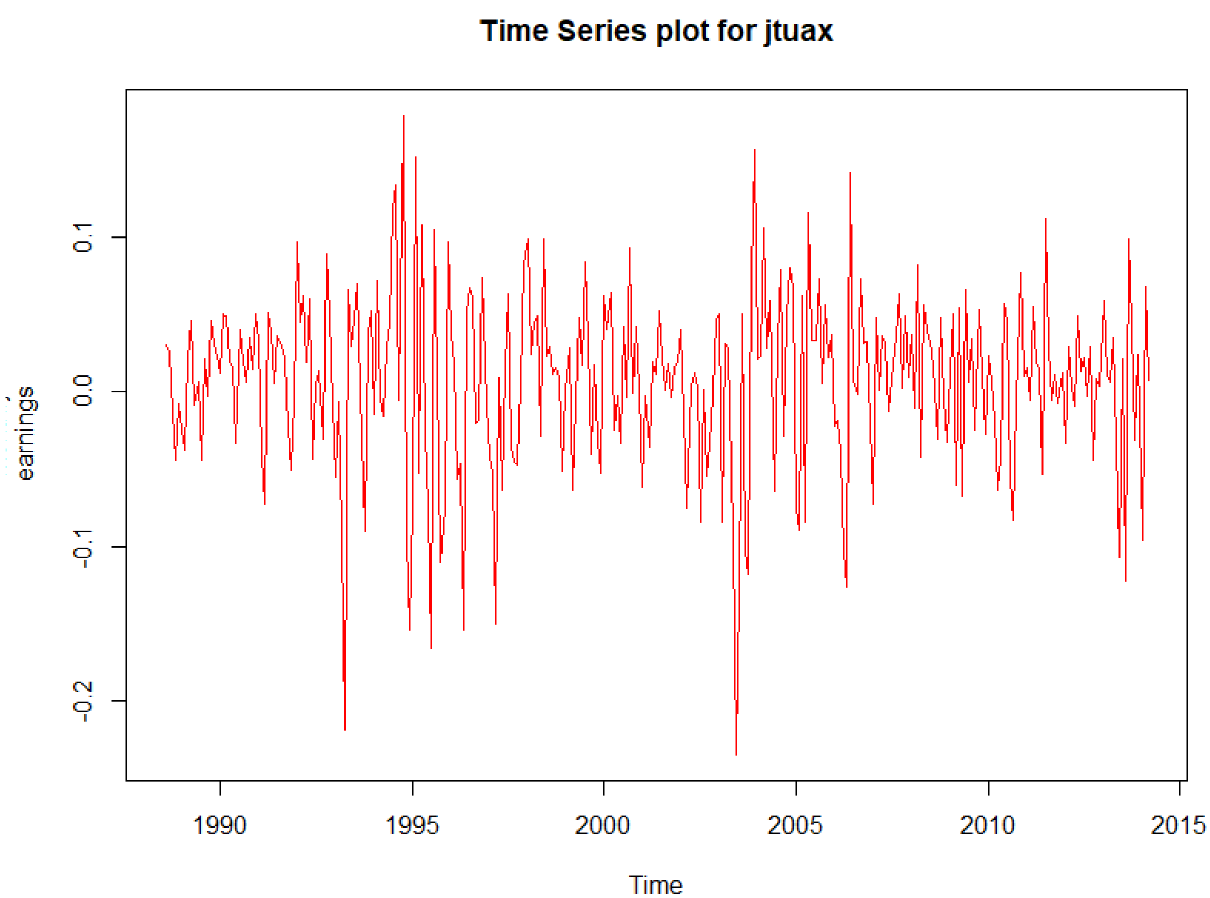
```
library(tseries)
library(Hmisc)
```

Δημιουργία time series για την jtuax με βήμα 12 δηλαδή την θέλω ανά μηνα(12 μήνες τον χρόνο)

```
j=ts(jtuax, frequency=12, start = c(1988,12)) #start date for my fund
```

Ααναπαριστώ γραφικά την χρονοσειρά που δημιούργησα για να βγάλω οπτικά κάποια πρώτα συμπεράσματα

```
plot(j, type="l", col='red', lwd=1, main="Time Series plot for jtuax", ylab="monthly  earnings")
```



Παρατηρώ τα εξής:

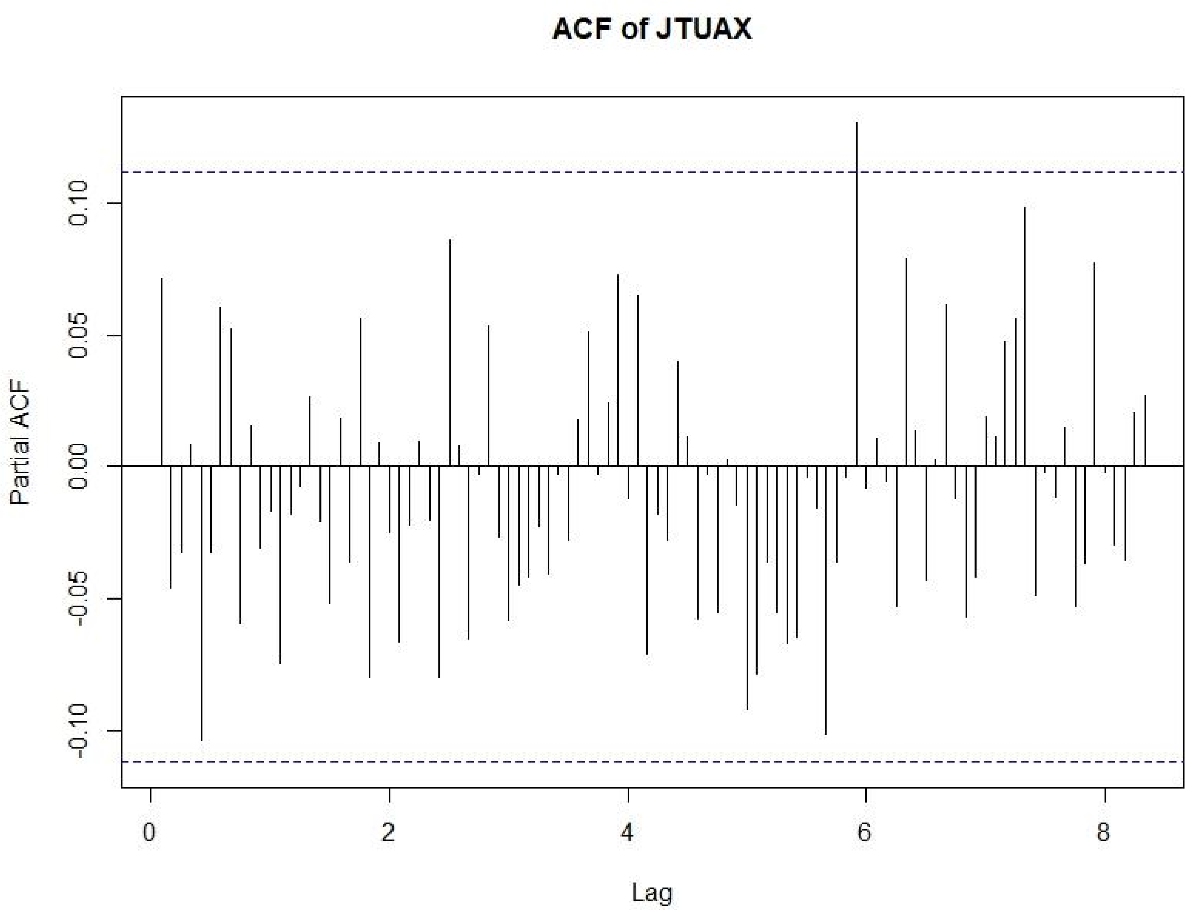
- η σειρά περιγράφεται από μοντέλο χωρίς σταθερά καθώς οι διακυμάνσεις των τιμών κυμαίνονται στο μηδέν
- χωρίς τάση καθώς οι τιμές δεν έχουν κάποια σταθερή αυξομείωση

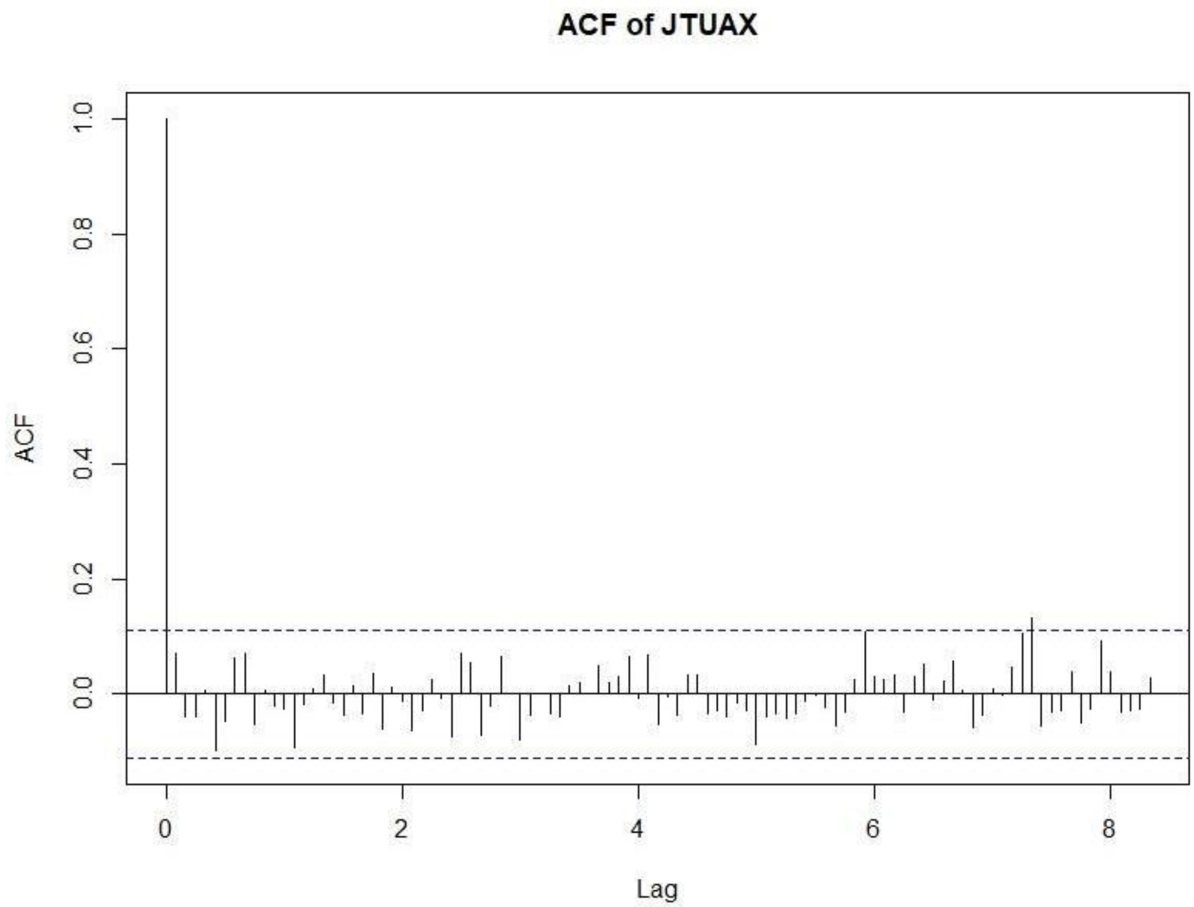
Πάω να κάνω ACF και PACF για να ελέγξω ενδεχόμενο πρόβλημα αυτοσυσχέτισης και για την χρονοσειρά που δημιούργησα

(θα μπορούσα να κάνω acf και pacf και για τις τιμές λογάριθμου(log) της jtuax η ακόμα και για τα difflog της αλλά κάτι τέτοιο δεν θα είχε μεγάλο νόημα γιατί έχουμε αρνητικές τιμές και θα μας έβγαζε πολλά να μην βοηθώντας μας)

```
pacf(j, 100, main="ACF of JTUAX")
```

```
acf(j, 100, main="ACF of JTUAX")
```





- Παρατηρώ ότι σχεδόν όλες οι τιμές είναι εντός ορίων όποτε έχω μια πρώτη ισχυρή ένδειξη ότι η σειρά είναι στάσιμη

- ακόμα παρατηρώ ότι δεν θα έχει θέμα aytocorellation(αυτοσυσχετιση) και δεν χρειάζεται AM() AR()μοντέλο για βελτιστοποίηση καθώς όλες οι τιμές στο acf μου είναι εντώς ορίων ,
- Η μόνη υποψία που έχω για θέμα autocorrelation είναι για lag 6 που βγαίνει ελαφρώς εκτός ορίων, μπορώ να κάνω box.test για να σιγουρευτώ ότι η τιμή αυτή που βγαίνει ελαφρώς εκτός ορίων είναι white noises(λευκός θόρυβος)

Box.test(j,lag=112,type="Ljung")

- p-value=0.8>a=0.05 άρα δεν απορρίπτω την Ho για α=5%,δεν έχω αυτοσυσχετιση

  οπότε επιβεβαίωσα ότι η τιμή lag που βγαίνει ελαφρώς εκτός είναι white

  noise(λευκός θόρυβος) και δεν δημιουργεί θέμα autocorrelation στην χρονοσειρά

- αυτό σημαίνει ότι έχω βάσιμες υποψίες ότι η σειρά μου είναι και στάσιμη

  σειρά(stationary) δηλαδή οι διακυμάνσεις των τιμών της χρονοσειράς δε

  διαφοροποιούνται με το χρόνο

Έλεγχος στασιμότητας(unit root testing)

Με το μάτι βλέπω ότι θα έχω lag=0 αλλά μπορώ να το ελέγξω και με την εντολή

m=ar(j)

m$order

0

Παρατηρώ ότι έχω lag=o όποτε ο έλεγχος μου γίνεται χωρίς σταθερά η τάση και lag=o

Έλεγχος στασιμότητας

m1=ur.df(j,type="none",lags=0)

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

Residuals:
     Min       1Q   Median       3Q      Max 
-0.22770 -0.02992  0.01182  0.04213  0.17968 

Coefficients:
        Estimate Std. Error t value Pr(>|t|)    
z.lag.1 -0.91789    0.05695  -16.12   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.05879 on 306 degrees of freedom
Multiple R-squared:  0.4592,    Adjusted R-squared:  0.4574 
F-statistic: 259.8 on 1 and 306 DF,  p-value: < 2.2e-16


Value of test-statistic is: -16.1177 

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

- Παρατηρώ R^2=0,4592=46% όπου είμαστε πολύ ικανοποιημένοι ειδικά επειδή

  έχουμε real-data.Η εκτίμηση Y_{t-1} για το β(καπελο=-0.9 και το stderror=0.056)

- Παίρνουμε την στατιστική υπόθεση H0:ρ=1(μη στάσιμη σειρά)

  H1:ρ<1 (στάσιμη σειρά)

Παρατηρώ ότι T-statistics=-16.17 < για όλα τα α=0.01,0.05,0.1 οπότε απορρίπτω την μηδενική υπόθεση άρα μπορούμε να πούμε ότι η σειρά μας είναι στάσιμη

Δεν χρειάζεται να μοντελοποιήσουμε τελικά την σειρά με (AR,MA,ARMA)

Επόμενο βήμα είναι να Εξερευνήσουμε τις γραμμικές σχέσεις μεταξύ των παραγόντων που επηρεάζουν το fund της JTUAX.

- **Φτιάχνω το μοντέλο πολλαπλής παλινδρόμησης**

jtuax<-lm(Y1~X1J+X2J+X3J+X4J+X5J+X6J)

Summary(jtuax)

```
Call:
lm(formula = JTUAX1 ~ X1J + X2J + X3J + X4J + X5J + X6J)

Residuals:
      Min        1Q    Median        3Q       Max
-0.040520 -0.006606 -0.000078  0.006504  0.060207

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0017147  0.0007482  -2.292 0.022610 *
X1J          1.0672728  0.0203264  52.507  < 2e-16 ***
X2J          0.7669400  0.0261949  29.278  < 2e-16 ***
X3J          0.1374486  0.0348832   3.940 0.000101 ***
X4J         -0.1504178  0.0361464  -4.161 4.13e-05 ***
X5J         -0.0808490  0.0477085  -1.695 0.091177 .
X6J          0.1098761  0.0156246   7.032 1.37e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01228 on 301 degrees of freedom
Multiple R-squared:  0.9569,    Adjusted R-squared:  0.956
F-statistic:  1113 on 6 and 301 DF,  p-value: < 2.2e-16
```

- Παρατηρώ ότι οι μεταβλητές X1J,X2J.X3J.X4J,X6J έχουν p-vaule<α όποτε είναι στατιστικά σημαντικές

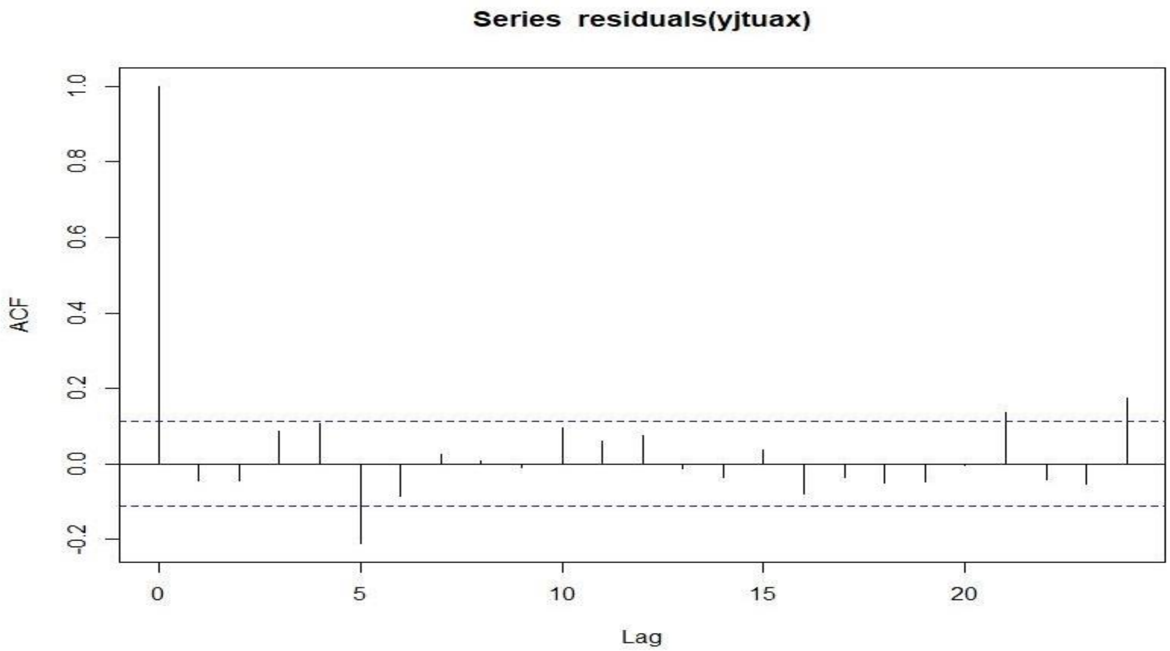
Ακόμα η X5J έχει p-value=0.09>0.05 **οπότε στην περίπτωση μόνο** που θα έχω ασυσχέτιστα κατάλοιπα, κανένα θέμα με την ετεροσκεδαστικότητα και το δείγμα μου θα ακλουθεί κανονική κατανομή (κανονικότητα) δεν θα είναι στατιστικός σημαντικός παράγοντας (δηλαδή δεν θα προσφέρει τίποτα ουσιαστικό στην εξήγηση του μοντέλου μας και θα μπορούσαμε να τον αφαιρέσουμε για α=5%, **<u>μόνο άμα ισχύουν οι παραπάνω προϋποθέσεις</u>**

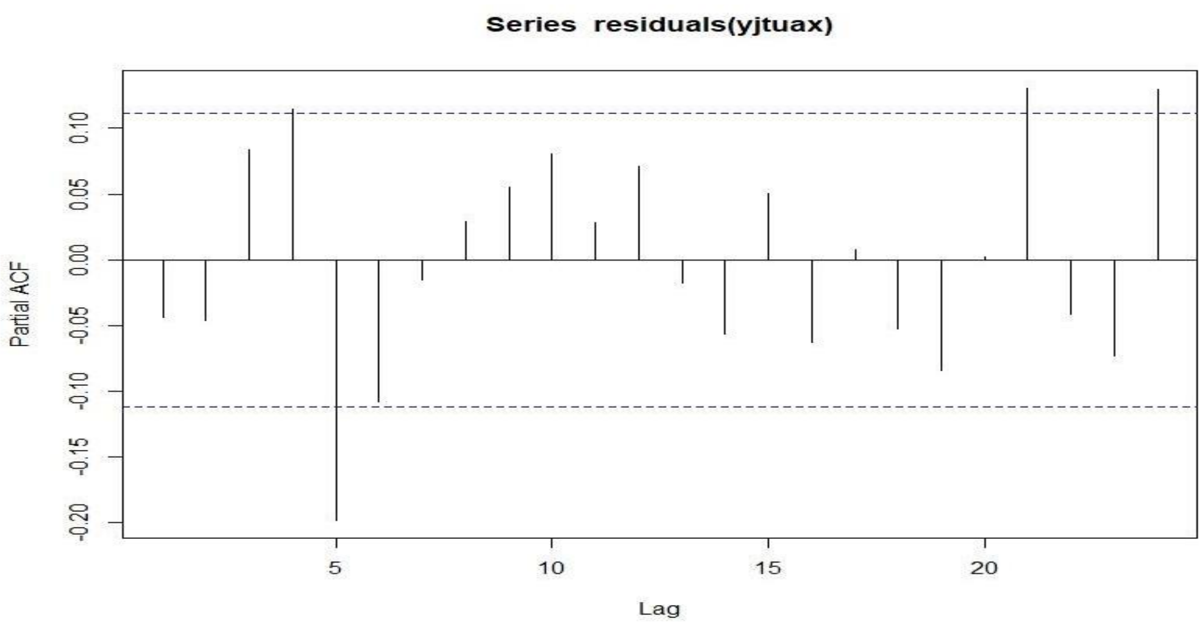
Ακόμα βλέπω R^2=0.956 όπου είναι πολύ κοντά στην μονάδα άρα έχουμε πολύ καλό συντελεστή συσχέτισης και αυτό σημαίνει ότι οι παράγοντες εξηγούν πολύ καλά το μοντέλο μας.

Κάνω acf και pacf στα residuals (κατάλοιπα)της jtuax για να βρω αν έχει θέμα aytocorrelation των καταλοίπων

acf(residuals(jtuax))

pacf(residuals(jtuax))



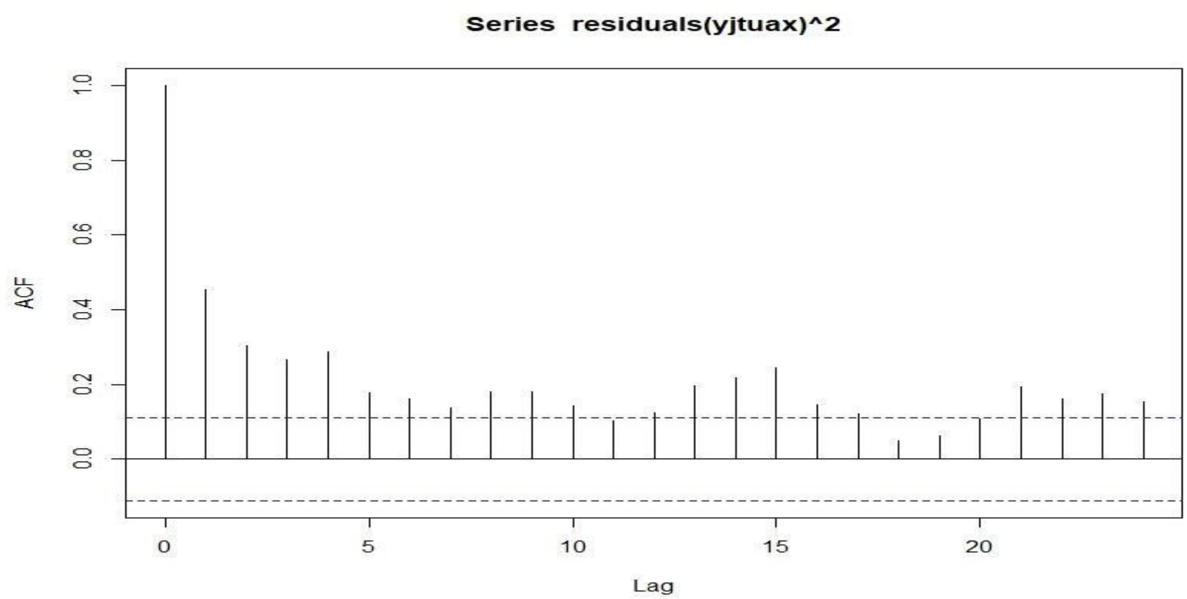


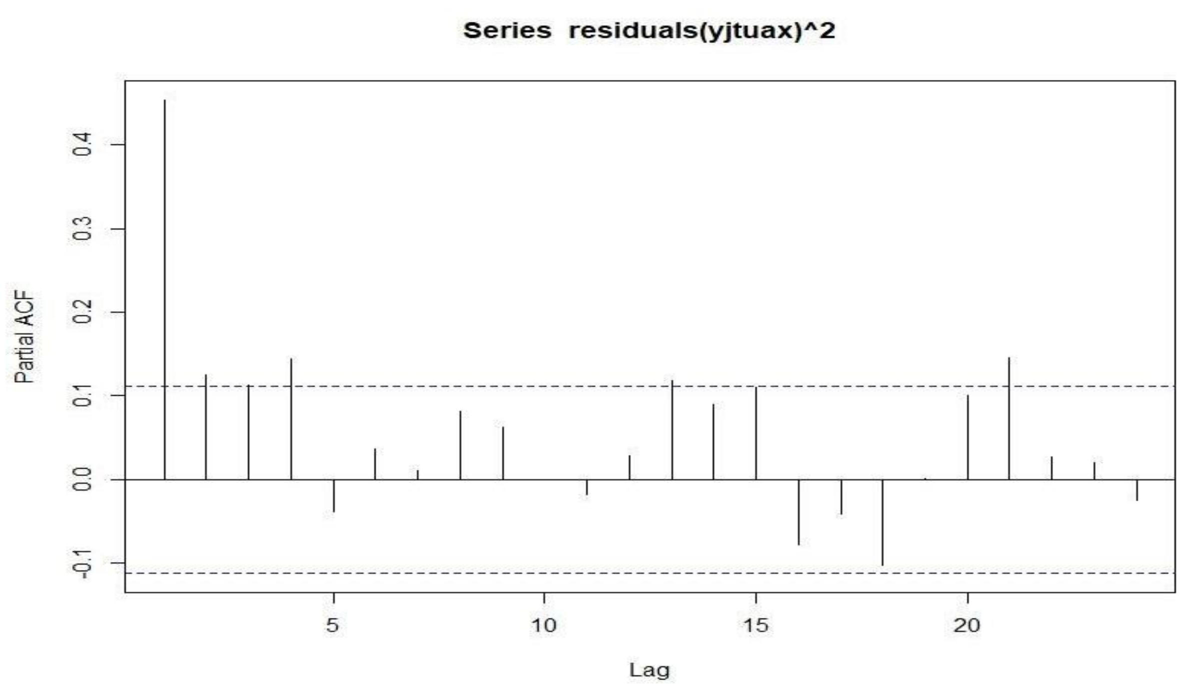
<u>Παρατηρώ ότι έχει θέμα σε lag(5) και σε acf και σε pacf</u>

Κάνω acf και pacf στα residuals^2 της ytuax για να βρω αν έχει και θέμα ετεροσκεδαστικότητας

acf(residuals(jtuax)^2)

pacf(residuals(jtuax)^2)





Παρατηρώ ότι για το acf είναι εκτός ορίων για lag1,2,3,4.. και για pacf ελαφρώς εκτός ορίων lag2,lag13..Έχω ενδείξεις για πρόβλημα ετεροσκεδαστικότητας των καταλοίπων (arch effects)

Ελενχω την κανονικότητα της σειράς

```
shapiro.test(yjtuax$residuals)
```

```
jarqueberaTest(yjtuax$residuals)
```

p-value<a και στους δύο έλεγχους άρα **έχω θέμα <u>κανονικότητας</u>**

Πάμε να διορθώσω το μοντέλο μας για το aytocorellation , θα δοκιμάσω να βάλω AR(1) και ας έχουμε θέμα στο lag 5 γιατί εμείς χρειαζόμαστε το πιο πρόσφατο κατάλοιπο(θα μπορούσα να ξεκινήσω και με το AR(5)) και άμα φτιαχτεί το θέμα του aytocorellation θα πάω να κάνω και ένα garch(1,1)για να λύσω το θέμα της ετεροσκεδαστικότητας.

Αυτό θα μπορούσα να το κάνω ξεχωριστά με τον ακόλουθο τρόπο δηλαδή πρώτα να κάνω AR(1) στα residuals με την εντολή

`jtuaxAR<-arima(residuals(yjtuax),order=c(1,0,0),include.mean = FALSE)` και μετά

για garch(1,1) με την εντολή

```
jtuaxGARCH=garchFit(~garch(1,1),data=residuals(jtuaxAR),cond.dist ='norm')
```

Καλύτερα όμως και από άποψη χρόνου αλλά και για να μας βοηθήσει καλυτέρα στο AIC και BIC θα τα κάνω όλα μαζί με τον ακόλουθο τρόπο

Δηλαδή ARMA(1,0) + GARCH(1,1)

```
X<-matrix(cbind(X1,X2,X3,X4,X5,X6),ncol=6)
```

```
spec<-ugarchspec(variance.model = list(model='sGARCH',garchOrder=c(1,1)),mean.model =
list(armaOrder=c(1,0),include.mean=TRUE,external.regressors=X),distribution.model = 'norm')
```

```
modelres<-ugarchfit(spec=spec,data=JTUAX1)
```

```
modelres
```

```
-----------------------------------
GARCH Model     : sGARCH(1,1)
Mean Model      : ARFIMA(1,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu     -0.001344    0.000541 -2.48486 0.012960
ar1     0.018522    0.066999  0.27645 0.782204
mxreg1  1.038454    0.016339 63.55676 0.000000
mxreg2  0.794345    0.022875 34.72534 0.000000
mxreg3  0.169021    0.028436  5.94402 0.000000
mxreg4 -0.044610    0.033965 -1.31338 0.189054
mxreg5 -0.105050    0.039066 -2.68908 0.007165
mxreg6  0.061385    0.014736  4.16571 0.000031
omega   0.000008    0.000003  2.65290 0.007980
alpha1  0.211182    0.049945  4.22827 0.000024
beta1   0.729988    0.035311 20.67317 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu     -0.001344    0.000619 -2.17040 0.029977
ar1     0.018522    0.076454  0.24226 0.808579
mxreg1  1.038454    0.021881 47.45919 0.000000
mxreg2  0.794345    0.028415 27.95522 0.000000
mxreg3  0.169021    0.036611  4.61665 0.000004
mxreg4 -0.044610    0.036137 -1.23446 0.217032
mxreg5 -0.105050    0.040754 -2.57765 0.009947
mxreg6  0.061385    0.015862  3.86995 0.000109
omega   0.000008    0.000004  2.00390 0.045081
alpha1  0.211182    0.040718  5.18641 0.000000
beta1   0.729988    0.054318 13.43912 0.000000

LogLikelihood : 978.55

Information Criteria
------------------------------------

Akaike       -6.2828
Bayes        -6.1496
Shibata      -6.2852
Hannan-Quinn -6.2295

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                   0.008114  0.9282
Lag[2*(p+q)+(p+q)-1][2]  0.018462  1.0000
Lag[4*(p+q)+(p+q)-1][5]  2.530117  0.5578
d.o.f=1
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                      2.204  0.1377
Lag[2*(p+q)+(p+q)-1][5]     2.846  0.4359
Lag[4*(p+q)+(p+q)-1][9]     3.980  0.5922
d.o.f=2

Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3] 0.0002491 0.500 2.000  0.9874
ARCH Lag[5] 0.8488902 1.440 1.667  0.7783
ARCH Lag[7] 1.4971165 2.315 1.543  0.8220
```

- Καταρχήν επιβεβαιώνουμε ότι το μοντέλο μας έτρεξε ARMA(1,0) και

  GARCH(1,1)

ΠΑΡΑΤΗΡΩ

Mu->p-value>a είναι στατιστικά σημαντικό

Weighted box of residuals

Όλα τα p-value>a=0.05 άρα με το μοντέλο AR(1) λύσαμε το πρόβλημα του aytocorellation

Weighted box of residuals^2

Όλα τα p-value>a=0.05 αρα και με το μοντελο GARCH(1,1) λύσαμε το πρόβλημα της ετεροσκεδαστικότητας

Και τα weighted arch lm tests όλα p -value>a

Το μόνο που μας μένει είναι να δούμε την κανονικότητα ,που αυτό θα το δούμε αν πάμε στην spec και αλλάξουμε το dist.con=norm σε =std.err και συγκρίνουμε τα aic και bic μας σε κάθε spec

Spec1<-ugarchspec(variance.model = list(model='sGARCH',garchOrder=c(1,1)),mean.model = list(armaOrder=c(1,0),include.mean=TRUE,external.regressors=X),distribution.model = ''std")

```
Information Criteria
------------------------------------

Akaike       -6.2762
Bayes        -6.1309
Shibata      -6.2791
Hannan-Quinn -6.2181

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
```

Ξέρω ότι όσο μικρότερο AIC,BIC τόσο το καλύτερο οπότε επειδή με distribution.model='norm' έχειμικρότερο AIC και BIC από το distribution.model='std' κρατάμε το 'norm' μοντέλο,

Λύσαμε και τα τρία θέματα που είχαμε(aytocorellation,ετεροσκεδαστικότητας,κανονικότητας)

Οφείλω όμως να ελέγξω και άλλα AR GARCH μοντέλα γιατί παρόλο που μου λύθηκαν όλα τα προβλήματα μπορεί να υπάρχει ακόμα καλύτερο μοντέλο από το ARMA(1,0) και GARCH(1,1) όπου αυτό θα το κρίνω με βάση το αντίστοιχο AIC και BIC(μικρότερο = καλύτερο) κάθε μοντέλου με τον ακόλουθο τρόπο.

.

.

.

.

.

.

- spec2<-ugarchspec(variance.model = list(model='sGARCH',garchOrder=c(2,1)),mean.model = list(armaOrder=c(1,0),include.mean=TRUE,external.regressors=X),distribution.model = 'norm')

- spec3<-ugarchspec(variance.model = list(model='sGARCH',garchOrder=c(2,2)),mean.model = list(armaOrder=c(1,0),include.mean=TRUE,external.regressors=X),distribution.model = 'norm')

- spec3<-ugarchspec(variance.model = list(model='sGARCH',garchOrder=c(3,2)),mean.model = list(armaOrder=c(2,0),include.mean=TRUE,external.regressors=X),distribution.model = 'norm')

.
.
.
.
.
.

#BEST MODEL BASED ON AIC IS THE GARCH(1,1) ARMA(1,0)

#BEST MODEL BASED ON BIC IS THE GARCH(1,2) ARMA(1,0)*(απειροελάχιστη διαφορά)

Οπότε καταλήγω ότι το GARCH(1,1) ARMA (1,0) ήταν το πιο κατάλληλο μοντέλο μας

Δεν εχω τελιωσει ακομα όμως στο μοντέλο μας παρατηρούμε ότι το XJ4->p-value=0.18>a=0.05 όποτε δεν είναι στατιστικά σημαντικό factor και δεν επηρεάζει στατιστικά σημαντικά την σειρά μας

Για επιβεβαίωση όμως μπορούμε να ελέγξουμε ποιο είναι το βέλτιστο μας μοντέλο με με χρήση των μεθόδων forward, backward και stepwise όπου με βάση το κριτήριο AIC και BIC μπορούμε να ελέγξουμε ποιο είναι το βέλτιστο μας μοντέλο μας από πλευράς επεξηγηματικών παραγόντων για την μεταβλητή μα

Μιας και έχουμε ειδή το γεμάτο μοντέλο θα μπορούσαμε να κάνουμε backward elimination

Όπου ξεκινά από το γεμάτο μοντέλο μας και αφαιρει κάθε φορά μια μεταβλητή και το ξανασυγκρίνει μέχρι να φτάσει στο μοντέλο y~1 δηλαδή να μην υπάρχει καμία επεξηγηματική μεταβλητή.

Όποιος συνδυασμός έχει το μικρότερο AIC σημαίνει ότι είναι και το καλύτερο μας μοντέλο για την εξήγηση την ανεξάρτητη μεταβλητή μας

```
stepBE <- step(fitall,scope=list(lower= ~1,upper= ~X1J+X2J+X3J+X4J+X5J+X6J,direction="backward"))
stepBE
```

Βλέπουμε ότι καταλήγουν στο ίδιο συμπέρασμα δηλαδή ότι το βέλτιστο μοντέλο είναι αυτό που δεν περιέχει την X4 επεξηγηματική μεταβλητή επειδή εκεί παρατηρούμε την μικρότερη τιμή AIC

Οπότε το μοντέλο μας γίνεται:

```
X1<-matrix(cbind(X1J,X2J,X3J,X5J,X6J),ncol=5)
```

<u>Έχω βγάλει το X4J και ‘έχω μειώσει τα ncol κατά ένα</u>

```
spec3<-ugarchspec(variance.model = list(model='sGARCH',garchOrder=c(1,1)),mean.model = list(armaOrder=c(1,0),include.mean=TRUE,external.regressors=X1),distribution.model='norm')
```

```
modelres1<-ugarchfit(spec=spec3,data=JTUAX1)
```

```
modelres1
```

Akaike -6.2837

Bayes -6.1626

Όπου είναι ακόμα μικρότερα τα AIC και BIC από όλα τα προηγούμενα

Οπότε ανακεφαλαιώνουμε ότι με **ARMA(1,0),GARCH(1,1)** και **χωρίς το X4J**
Έχουμε το «ιδανικό» μοντέλο**(modelres1)** που πληροί τις προϋποθέσεις aytocorellation,ετεροσκεδαστικότητας,κανονικότητας και έχει το μικρότερο δυνατόν AIC και BIC

ΑΣ ΠΕΡΑΣΟΥΜΕ ΣΤΗΝ ΠΡΟΒΛΕΨΗ

ΠΡΟΒΛΕΨΗ ΓΙΑ ΤΟ ΔΙΑΣΤΗΜΑ 8/2017 ΕΩΣ 7/2019 ΤΗΣ JTUAX

1)Πρώτος τρόπος για να κάνουμε πρόβλεψη, είναι μέσω της εντολής ugarchforecast, ως εξής :

```
data_for_forecast <- independentvariables[307,]
#SELECT THE DATA FROM 8/2017
```

```
pred <- ugarchforecast(modelres1,n.ahead = 24,external.forecasts=data_for_forecast)
pred
#PREDICTION ABOUT THE SERIES AND THE SIGMA 24 PERIODS AHEAD
ug_f <- pred@forecast$seriesFor
plot(ug_f,type="l")
#plot for volatility
```

```
##
## *       * ## *     GARCH Model Forecast      * ## *       * ## Model: sGARCH
## Horizon: 24 ## Roll Steps: 0
## Out of Sample: 0 ##
## 0-roll forecast [T0=1970-12-07 02:00:00]:



##          Series    Sigma
##  T+1     0.00103   0.01250
##  T+2     0.00103   0.01324
##  T+3     0.00103   0.01386
##  T+4     0.00103   0.01438
##  T+5     0.00103   0.01484
##  T+6     0.00103   0.01523
##  T+7     0.00103   0.01557
##  T+8     0.00103   0.01587
##  T+9     0.00103   0.01613
##  T+10    0.00103   0.01636
##  T+11    0.00103   0.01656
##  T+12    0.00103   0.01673
##  T+13    0.00103   0.01689
##  T+14    0.00103   0.01703
##  T+15    0.00103   0.01715
##  T+16    0.00103   0.01725
##  T+17    0.00103   0.01735
##  T+18    0.00103   0.01743
##  T+19    0.00103   0.01751
##  T+20    0.00103   0.01757
##  T+21    0.00103   0.01763
##  T+22    0.00103   0.01768
##  T+23    0.00103   0.01773
##  T+24    0.00103   0.01777
```

Το παραπάνω μας δίνει τις τιμές των Yt (σημειακή εκτίμηση) στην αριστερή στήλη (series) ,ενώ στην δεξιά στήλη έχουμε τις τιμές των δεσμευμένων διακυμάνσεων (sigma).

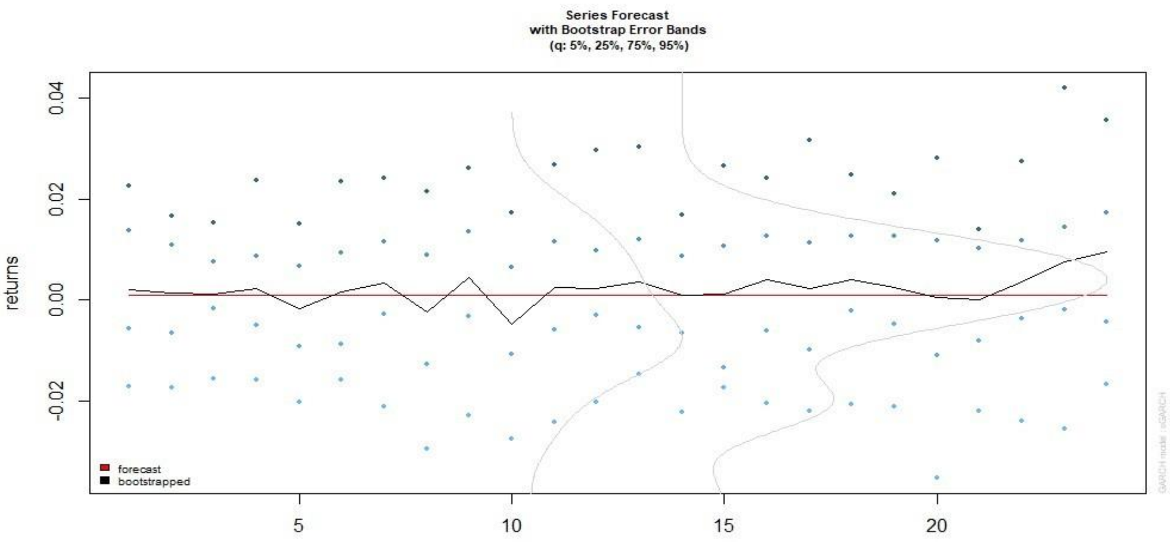
2)Εναλλακτικός τρόπος πρόβλεψης είναι μέσω του Garch Bootstrap :

#THE BOOTSTRAP METHOD IN THE rugarch PACKAGE IS BASED ON RESAMPLING STANDARDIZED RESIDUALS FROM THE EMPIRICAL DISTRIBUTION OF THE FITTED MODEL TO GENERATE FUTURE REALIZATIONS OF THE SERIES AND SIGMA (THE SQUARE ROOT OF THE VARIANCE)

```
2ndway <- ugarchboot(modelres1,method = c("Partial","Full")[1],n.ahead = 24,n.bootpred = 24)
2ndway
```

```
##
##
##
##
## *------------------------------------*
## *      GARCH Bootstrap Forecast      *
## *------------------------------------*
##
lytic]
## t+1   -0.017154  -0.001252  0.002354   0.009030  0.012386  0
.00103
## t+2   -0.021332  -0.008064  0.002507   0.009562  0.024638  0
.00103
## t+3   -0.021990  -0.012958  -0.002289  0.004855  0.054968  0
.00103
## t+4   -0.056254  -0.010721  -0.000064  0.013043  0.032601  0
.00103
## t+5   -0.036740  -0.006103  0.001205   0.005253  0.011194  0
.00103
## t+6   -0.025821  -0.015319  -0.005630  0.001359  0.018813  0
.00103
## t+7   -0.019915  -0.006846  0.002019   0.008541  0.022803  0
.00103
## t+8   -0.046613  -0.001332  0.002689   0.013316  0.025160  0
.00103
## t+9   -0.046189  -0.009957  -0.000503  0.011466  0.026719  0
.00103
## t+10  -0.041015  0.001453   0.009403   0.020464  0.042190  0
.00103
## ....................
##
## Sigma (summary):
```

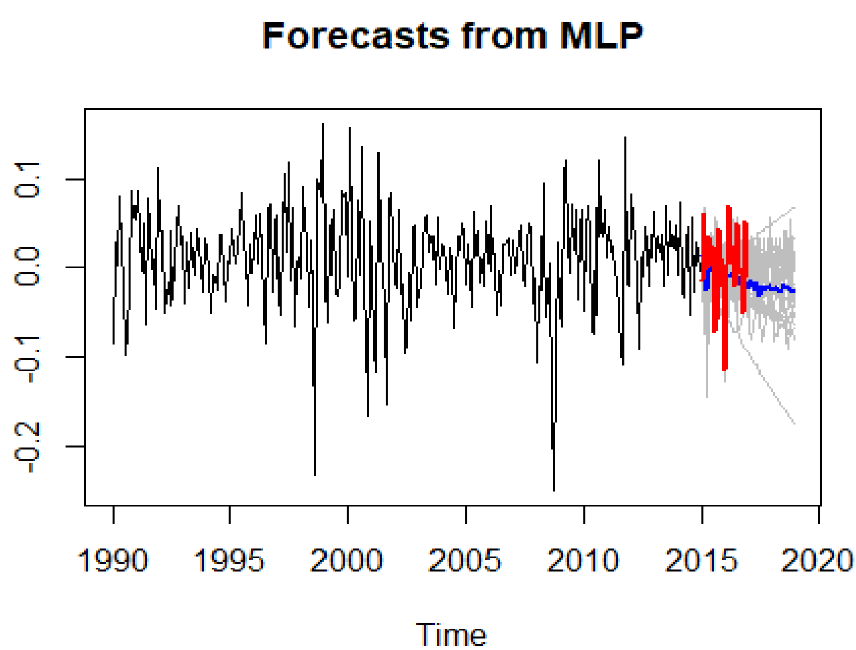
```
##           min     q0.25    mean     q0.75    max      forecast[analytic]
## t+1   0.012503 0.012503 0.012503 0.012503 0.012503            0.012503
## t+2   0.011476 0.011523 0.012113 0.012472 0.014958            0.013236
## t+3   0.010869 0.011729 0.013055 0.013879 0.016576            0.013855
## t+4   0.011301 0.011902 0.014155 0.014436 0.031407            0.014384
## t+5   0.010978 0.012430 0.015336 0.015695 0.039405            0.014839
## t+6   0.010507 0.011860 0.014543 0.014122 0.033293            0.015232
## t+7   0.010178 0.012479 0.014795 0.015663 0.027971            0.015573
## t+8   0.010002 0.012210 0.014067 0.014382 0.024077            0.015871
## t+9   0.010963 0.012067 0.014955 0.016660 0.031973            0.016131
## t+10  0.011224 0.011761 0.015374 0.016861 0.035802            0.016359
```



3)Ακόμα, μπορούμε να δούμε την πρόβλεψη με χρήση νευρωνικών δικτύων και συγκεκριμένα μέσω multiple layer perceptron (default mlp model has 5 hidden nodes) :

```
library(nnfor)
set.seed(1)
ynew <- read.csv2("C:/Users/GERRY /jtuax.txt",header=T,sep="",skip=24)
ynew2 <- ynew[1:307,]
df <- ts(ynew2,frequency = 12,start =c (1988,12))
kse <- df %>% window(end = c(2014,12))
kse %>%
  plot(
    main = "JTUAX PLOT",
    ylim = c(-0.3,0.2)
  )
kse1 <- df %>% window(start=2015)
kse1
fit1 <- kse %>% mlp()
pred1 <- fit1 %>% forecast(h=48)
pred1
pred1 %>% plot()
kse1 %>% lines(lwd=2,col="red")
#RED - REAL DATA
#BLUE - FORECAST
```

.
.
.
.



A) Κάνω evaluate την πρόβλεψη μας με βάση το hit ratio όπου στην ουσία μας λέει πόσο τις % από το προβλεπτικό μας μοντέλο ακολουθεί το ίδιο πρόσημο με το μοντέλο μας στην πραγματικότητα
Π.χ. για το πρώτο έτος πρόβλεψε το προβλεπτικό μας μοντέλο αύξηση όπως και έγινε και στην πραγματικότητα η ακόμα για το 13η έτος πρόβλεψε αύξηση ενώ στην πραγματικότητα ήρθε μείωση (όπου δεν είναι αρεστό φυσικά)
Στην ουσία συγκρίνεις τα πρόσημα του πραγματικών τιμών) και του y(predictions που βρήκα) και άμα έχουν διαφορετικά το σημειώνουμε και λέμε πόσες τις % παρατηρήσεις είχαν διαφορετικά πρόσημα στο σύνολο του δείγματος μας

Βλέπω ότι μόνο στους 4 μήνες από τους 24 είχα διαφορετικό πρόσημο
άρα

Hit ratio =20/24= 83,3% όπου οτιδήποτε πάνω από 80% θεωρείται άριστο

```
#hit ratio
Y_p <- Y[Y>0]
Y_p
#Y+
Y_n <- Y[Y<0]
Y_n
```

B) Να κάνω evaluate την πρόβλεψη με βάση το MSFE όπου στην ουσία μας λέει πόσο κοντά η πρόβλεψη μας ήρθε στην πραγματική μας τιμή, ο τύπος που μας το δίνει αυτό είναι $\Sigma \frac{(y-y_{p})^2}{24}$

```
#mean squared forecast error

i<-  1:24
Y<-predictionjtuax
Y_HAT<- pred@forecast@series
MSFE  <-(sum(Y[i]Y_hat[i])^2)/24
```

MSFE = 0.01331476 (αρκετά ικανοποιητικό)

ΕΥΧΑΡΙΣΤΩ