FINANCIAL ECONOMETRICS- R_LAB

NGUYEN NGOC PHUNG

5/6/2021

Contents

REGRESSION MODEL AND TESTINGS

IMPORT THE LIBRARIES

```
library(psych)
library(readxl)
library(car)
library(tseries)
```

```r
library(lmtest)
library(MASS)
library(foreign)
library(sandwich)
```

UKHP DATASET

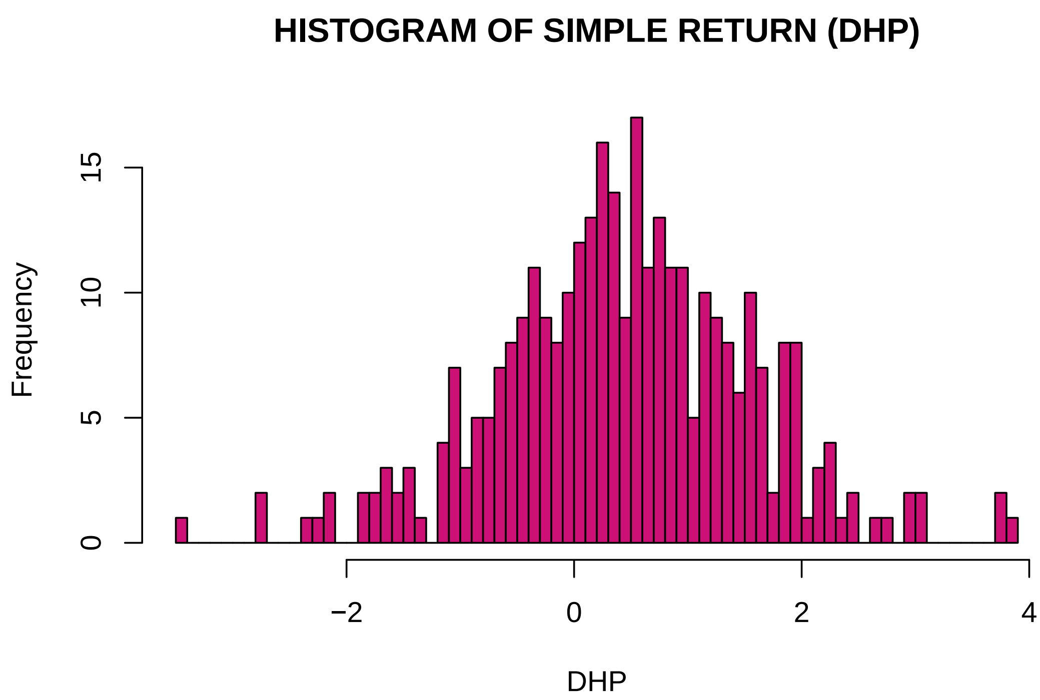
UKHP DATASET AND RETURNS

```r
HP <- read_excel("UKHP.xls")


Z<-ts(HP$`Average House Price`)
DHP<-100*diff(Z)/lag(Z,-1)
DHP[1:20]
```

```
##  [1]  0.8389504256 -1.1289220193  1.4833261822  1.3195330084  1.3269075933
##  [6] -1.0275463501 -0.9174932632 -1.4456792164  0.3889743215 -0.8390243074
## [11]  0.1623230805 -2.1709901200 -0.1944398560  0.2456052867  0.0005801514
## [16]  1.4136151132  0.3660040378 -0.3377067136 -0.7873699395 -2.3221311513
```

HISTOGRAM OF RETURNS TIME SERIES (DHP)

```r
hist(DHP, breaks = 70, col='deeppink3', main='HISTOGRAM OF SIMPLE RETURN (DHP)')
```



DESCRITIVE STATISTICS OF RETURNS (DHP)

```
summary(DHP)
```

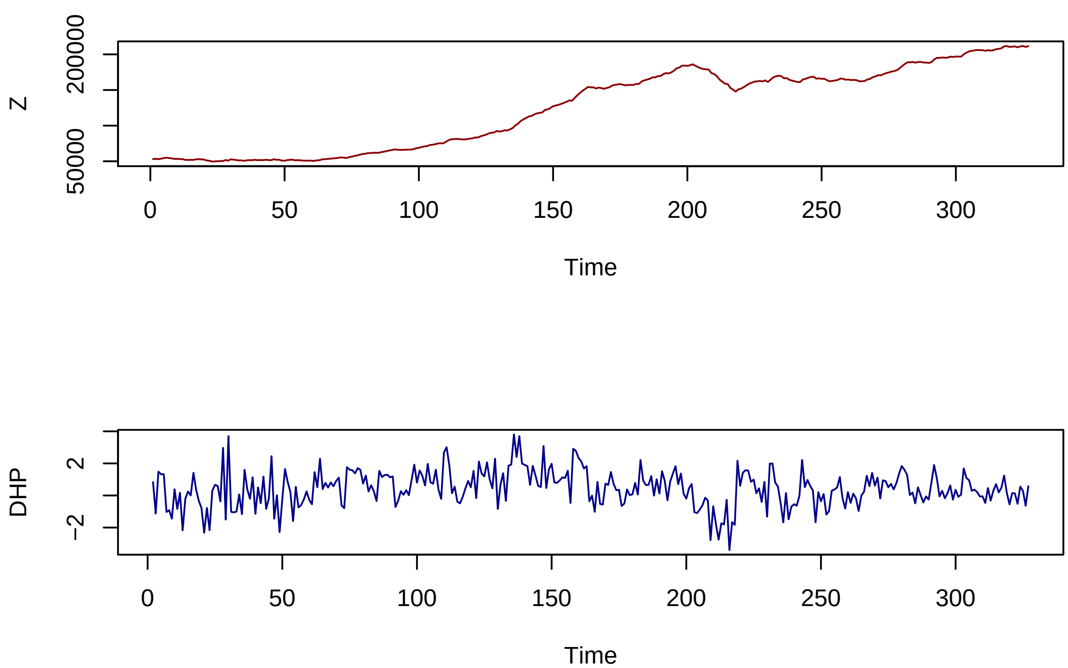
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3.4047 -0.2557  0.4483  0.4315  1.1522  3.8022
```

```
describe(DHP)
```

```
##    vars   n mean   sd median trimmed  mad  min max range  skew kurtosis   se
## X1    1 326 0.43 1.12   0.45    0.44 1.04 -3.4 3.8  7.21 -0.08     0.59 0.06
```

PLOTTING OF HOUSE PRICE AND RETURNS TIME SERIES

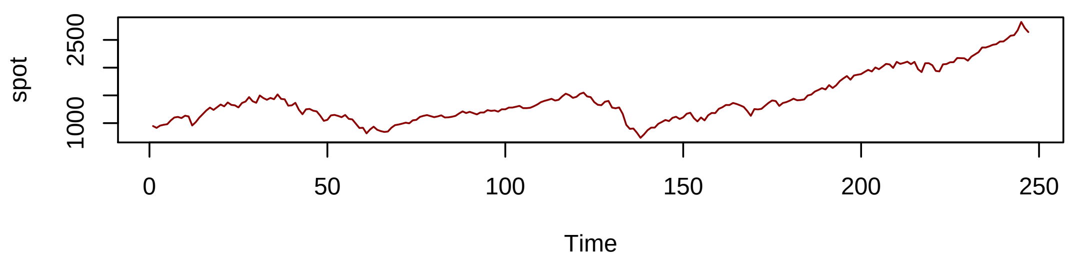
```
layout(matrix(c(1,2,1,2),2,2))
plot(Z,type='l',col='darkred')
plot(DHP,type='l',col='darkblue')
```

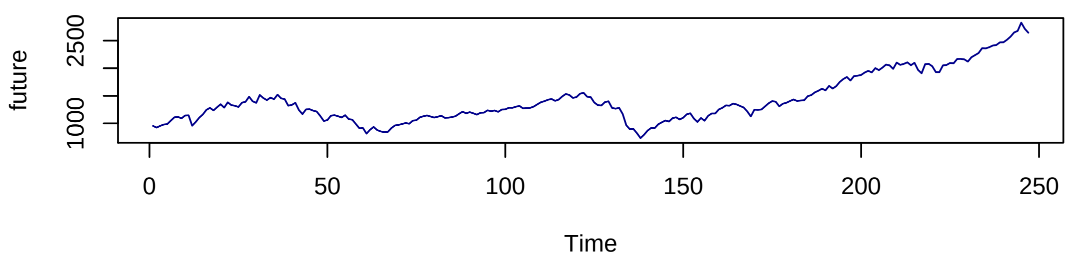


SANDPHEDGE DATASET

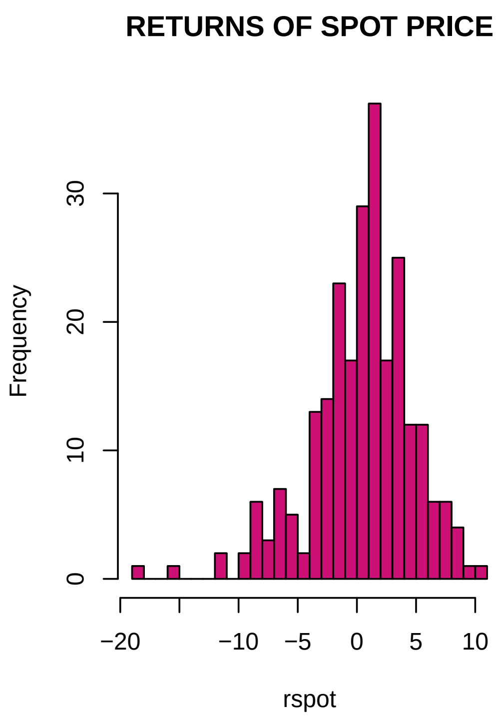
SANDPHEDGE DATASET AND RETURNS

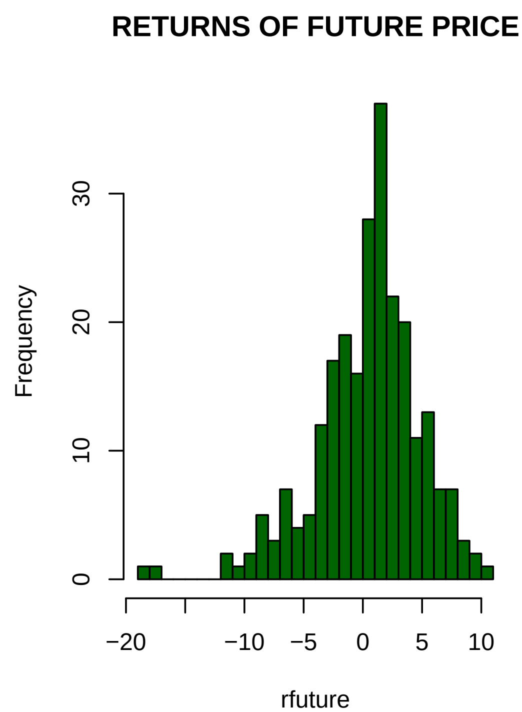
```r
# The dataset include the spot and future price @-@
SandPhedge <- read_excel('SandPhedge.xls')


spot<-SandPhedge$Spot ; future<- SandPhedge$Futures

spot<-ts(spot) ; future<-ts(future)

# Plotting of spot and future price 8-*
layout(matrix(c(1,2,1,2),2,2))
plot(spot,type='l',col='darkred')
plot(future,type='l',col='darkblue')
```






```r
# Now, we calculate the log-returns of both spot and future prices @-@
rspot <- 100*diff(log(spot))
rfuture <- 100*diff(log(future))

# Histogram of both returns time series
layout(matrix(c(1,1,2,2),2,2))
hist(rspot,breaks=30, col='deeppink3',main='RETURNS OF SPOT PRICE')
hist(rfuture,breaks=30, col='darkgreen',main='RETURNS OF FUTURE PRICE')
```



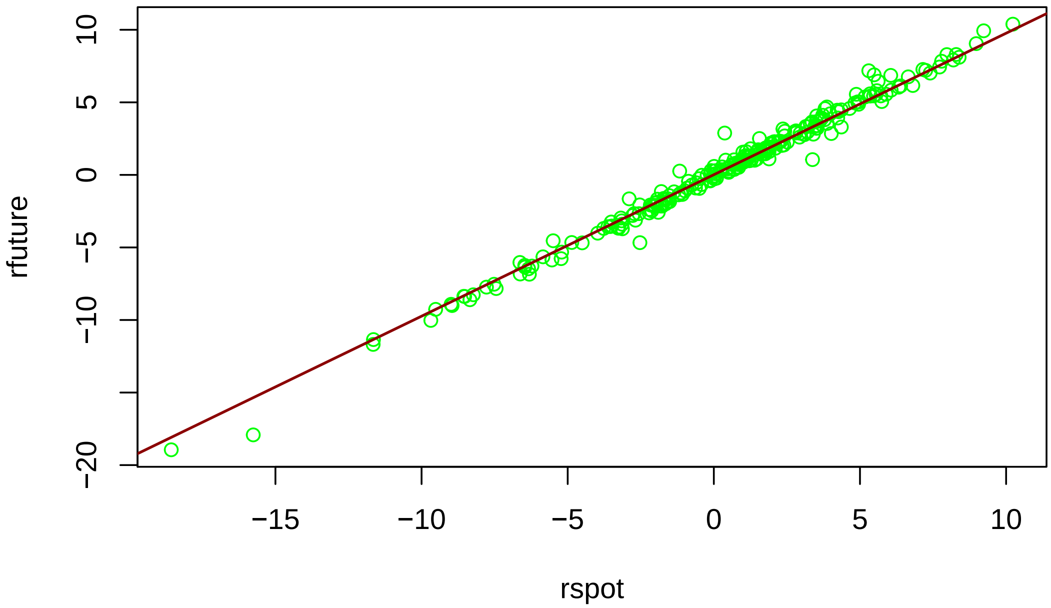


SIMPLE REGRESSION MODEL

```
# We concern about whether the return of future price explain the changes of return in spot price or no

ReturnReg<-lm(rspot~rfuture)
summary(ReturnReg)
```

```
##
## Call:
## lm(formula = rspot ~ rfuture)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.45284 -0.16401  0.00236  0.23692  2.33789
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.013077   0.029473   0.444    0.658
## rfuture     0.975077   0.006654 146.543   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4602 on 244 degrees of freedom
## Multiple R-squared:  0.9888, Adjusted R-squared:  0.9887
## F-statistic: 2.147e+04 on 1 and 244 DF,  p-value: < 2.2e-16
```

```r
# Plotting of regression model @-@
plot(rspot, rfuture,col='green')
abline(ReturnReg,col='darkred',lwd=1.5)
```



GOODNESS-OF-FIT STATISTIC - DEVIANCE AND LOG-LIKELIHOOD

```r
# in order to assess how good our model fit the actual values, we calculate the deviance (SSR)
deviance(ReturnReg)
```

```
## [1] 51.68444
```

```r
logLik(ReturnReg)  # return the logLik objects s.a the degree of freedom
```

```
## 'log Lik.' -157.1574 (df=3)
```

BREUSCH-PAGAN TEST

```r
bptest(ReturnReg)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  ReturnReg
## BP = 0.070052, df = 1, p-value = 0.7913
```

BREUSCH-GODFREY TEST

```
bgtest(ReturnReg)
```

```
##
##  Breusch-Godfrey test for serial correlation of order up to 1
##
## data:  ReturnReg
## LM test = 59.696, df = 1, p-value = 1.107e-14
```

DURBIN-WATSON TEST FOR AUTO-CORRELATED ERRORS

```
dwtest(ReturnReg)
```

```
##
##  Durbin-Watson test
##
## data:  ReturnReg
## DW = 2.9694, p-value = 1
## alternative hypothesis: true autocorrelation is greater than 0
```

```
durbinWatsonTest(ReturnReg,max.lag=2)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      -0.4860789      2.969363   0.000
##    2      -0.2293546      2.431056   0.002
##  Alternative hypothesis: rho[lag] != 0
```

NON-CONSTANT VARIANCE SCORE TEST

```
ncvTest(ReturnReg)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.3534588, Df = 1, p = 0.55216
```

AUGMENTED DICKEY-FULLER TEST FOR RETURNS (SPOT & FUTURE) AND PRICE (SPOT & FUTURE)

```
# returns
adf.test(na.omit(rspot))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(rspot)
## Dickey-Fuller = -5.6811, Lag order = 6, p-value = 0.01
## alternative hypothesis: stationary
```

```r
adf.test(na.omit(rfuture))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  na.omit(rfuture)
## Dickey-Fuller = -5.7291, Lag order = 6, p-value = 0.01
## alternative hypothesis: stationary
```

```r
# prices
adf.test(spot)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  spot
## Dickey-Fuller = -0.88531, Lag order = 6, p-value = 0.953
## alternative hypothesis: stationary
```

```r
adf.test(future)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  future
## Dickey-Fuller = -0.9341, Lag order = 6, p-value = 0.9476
## alternative hypothesis: stationary
```

MACRO DATASET

MACRO DATASET AND RETURNS

```r
macro <- read_excel('macro.xls')


spread<-ts(macro$BMINUSA)
credit<-ts(macro$CCREDIT)
prod<-ts(macro$INDPRO)
msoft<-ts(macro$MICROSOFT)
sandp<-ts(macro$SANDP)
money<-ts(macro$M1SUPPLY)
cpi<-ts(macro$CPI)

dspread<-diff(spread)
dcredit<-diff(credit)
dprod<-diff(prod)
dmoney<-diff(money)

rmsoft<-100*diff(log(msoft))
rsandp<-100*diff(log(sandp))
```

```r
term<-ts(macro$USTB10Y)-ts(macro$USTB3M)
inflation<-100*diff(log(cpi))
dinflation<-diff(inflation)
mustb3m<-ts(macro$USTB3M)/12
rterm<-diff(term)
ermsoft<-rmsoft - mustb3m
ersandp<-rsandp - mustb3m
```

MULTIPLE REGRESSION MODEL

```r
msoftreg <- lm(ermsoft[2:325]~ersandp[2:325] + dprod[2:325] + dcredit[2:325] +
                 dinflation[2:325] + dmoney[2:325] + dspread[2:325] + rterm[2:325])
summary(msoftreg)
```

```
##
## Call:
## lm(formula = ermsoft[2:325] ~ ersandp[2:325] + dprod[2:325] +
##     dcredit[2:325] + dinflation[2:325] + dmoney[2:325] + dspread[2:325] +
##     rterm[2:325])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -36.924  -4.671  -0.384   4.574  24.382
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        1.38426    0.53388   2.593  0.00996 **
## ersandp[2:325]     1.32625    0.10302  12.873  < 2e-16 ***
## dprod[2:325]      -0.91455    0.82171  -1.113  0.26656
## dcredit[2:325]    -0.01683    0.03488  -0.482  0.62986
## dinflation[2:325] -3.14294    1.38715  -2.266  0.02414 *
## dmoney[2:325]     -0.01194    0.02317  -0.516  0.60652
## dspread[2:325]     0.72326    4.38207   0.165  0.86901
## rterm[2:325]       4.75868    1.86769   2.548  0.01131 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.202 on 316 degrees of freedom
## Multiple R-squared:  0.3608, Adjusted R-squared:  0.3466
## F-statistic: 25.48 on 7 and 316 DF,  p-value: < 2.2e-16
```

NEWEY-WEST TEST FOR HETEROSCEDASTICITY AND AUTOCORRELATION

```r
# Simply stated, this function return teh HAC Covariance matric Estimation
NeweyWest(msoftreg)
```

```
##                    (Intercept) ersandp[2:325]  dprod[2:325] dcredit[2:325]
## (Intercept)        0.305520096   5.178095e-03 -0.0158598406   -9.926349e-03
```

```
## ersandp[2:325]     0.005178095   1.110974e-02 -0.0150081345  -5.396904e-04
## dprod[2:325]      -0.015859841  -1.500813e-02  0.2711811688  -2.192278e-04
## dcredit[2:325]    -0.009926349  -5.396904e-04 -0.0002192278   7.688322e-04
## dinflation[2:325] -0.072675457  -6.173293e-02  0.1534943327   8.265343e-03
## dmoney[2:325]     -0.001846171   2.103895e-05  0.0019349167   2.355234e-06
## dspread[2:325]     0.317658305   4.140168e-03  0.0055105657  -1.437012e-02
## rterm[2:325]       0.120233907   5.964648e-02 -0.1669457032  -1.309566e-02
##                   dinflation[2:325] dmoney[2:325] dspread[2:325] rterm[2:325]
## (Intercept)            -0.072675457 -1.846171e-03    0.317658305   0.12023391
## ersandp[2:325]         -0.061732930  2.103895e-05    0.004140168   0.05964648
## dprod[2:325]            0.153494333  1.934917e-03    0.005510566  -0.16694570
## dcredit[2:325]          0.008265343  2.355234e-06   -0.014370121  -0.01309566
## dinflation[2:325]       1.801701889 -2.244382e-03    0.116674663  -0.90465206
## dmoney[2:325]          -0.002244382  3.158284e-04   -0.026330103  -0.01021057
## dspread[2:325]          0.116674663 -2.633010e-02    7.288247914   0.83785431
## rterm[2:325]           -0.904652057 -1.021057e-02    0.837854309   3.03324948
```

```r
bwNeweyWest(msoftreg)
```

```
## [1] 7.695043
```

```r
NeweyWest(msoftreg, lag = 4, prewhite = FALSE)
```

```
##                     (Intercept) ersandp[2:325] dprod[2:325] dcredit[2:325]
## (Intercept)         0.293262795   7.658101e-03 -0.039611730  -8.789173e-03
## ersandp[2:325]      0.007658101   1.142868e-02 -0.012309860  -6.050553e-04
## dprod[2:325]       -0.039611730  -1.230986e-02  0.325312764  -1.624426e-03
## dcredit[2:325]     -0.008789173  -6.050553e-04 -0.001624426   7.790433e-04
## dinflation[2:325]  -0.028438008  -5.576858e-02  0.164841144   3.318870e-03
## dmoney[2:325]      -0.001499844  -5.774214e-05  0.002728501  -2.556316e-05
## dspread[2:325]      0.241051820   5.773703e-02  0.019427240  -1.446388e-02
## rterm[2:325]        0.101233416   4.471238e-02 -0.195826416  -1.046251e-02
##                   dinflation[2:325] dmoney[2:325] dspread[2:325] rterm[2:325]
## (Intercept)            -0.028438008 -1.499844e-03     0.24105182  0.101233416
## ersandp[2:325]         -0.055768584 -5.774214e-05     0.05773703  0.044712382
## dprod[2:325]            0.164841144  2.728501e-03     0.01942724 -0.195826416
## dcredit[2:325]          0.003318870 -2.556316e-05    -0.01446388 -0.010462512
## dinflation[2:325]       1.886506252 -1.597368e-03    -0.12534855 -0.379002137
## dmoney[2:325]          -0.001597368  3.224004e-04    -0.02163564 -0.009477304
## dspread[2:325]         -0.125348548 -2.163564e-02     7.39953192  0.706771660
## rterm[2:325]           -0.379002137 -9.477304e-03     0.70677166  2.946691601
```

ARIMA MODELS

IMPORT LIBRARIES

```r
library(psych)
library(lmtest)
library(forecast)
library(readxl)
library(tseries)
```

IMPORT AND VIEW THE DATASET

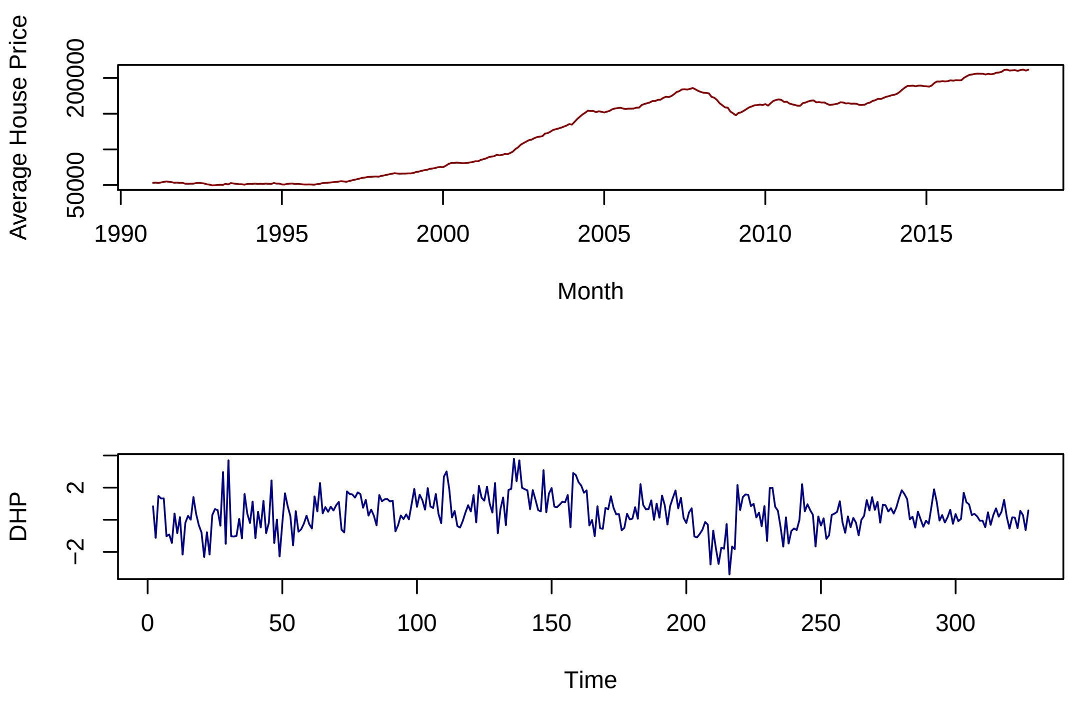
```
HP <- read_excel("UKHP.xls")
```

EXTRACT THE HOUSE PRICE TIME SERIES AND ESTIMATE THE SIMPLE RETURNS

```
Z<-HP$`Average House Price`
Z<-ts(HP$`Average House Price`)
DHP<-100*diff(Z)/lag(Z,-1)
```

PLOTTING OF HP AND DHP

```
layout(matrix(c(1,2,1,2),2,2))
plot(HP,type='l',col='darkred')
plot(DHP,type='l',col='darkblue')
```



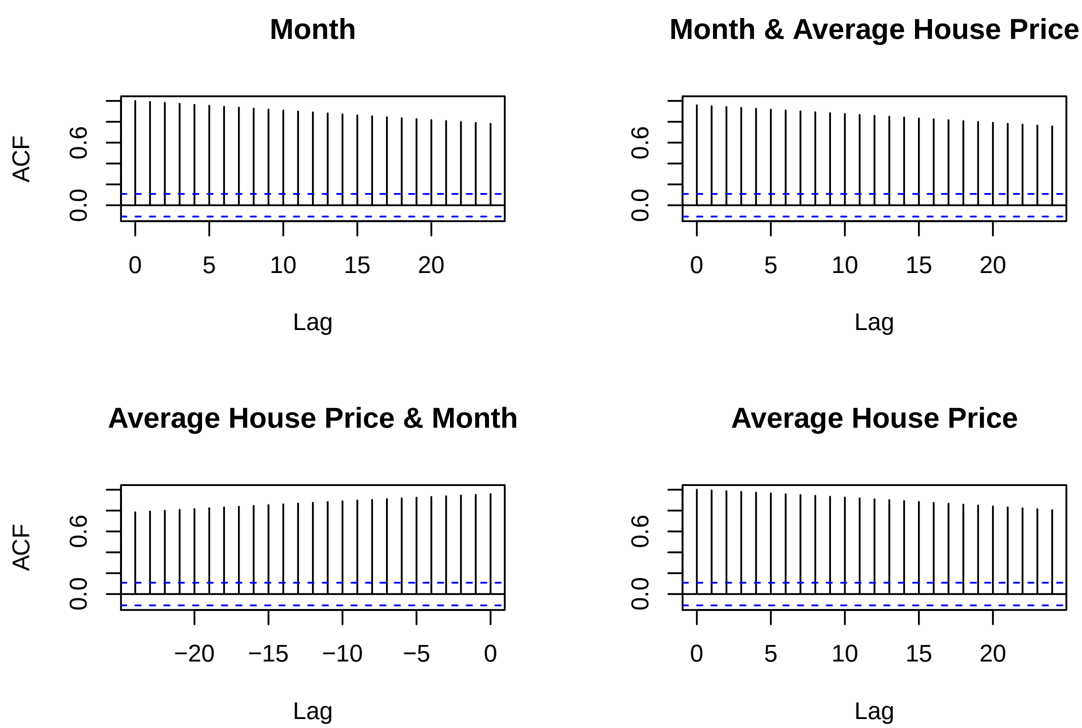
DESCRIPTIVE STATISTICS OF RETURNS

```r
summary(DHP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -3.4047 -0.2557  0.4483  0.4315  1.1522  3.8022
```

ACF AND PACF PLOTS OF HOUSE PRICE (HP)

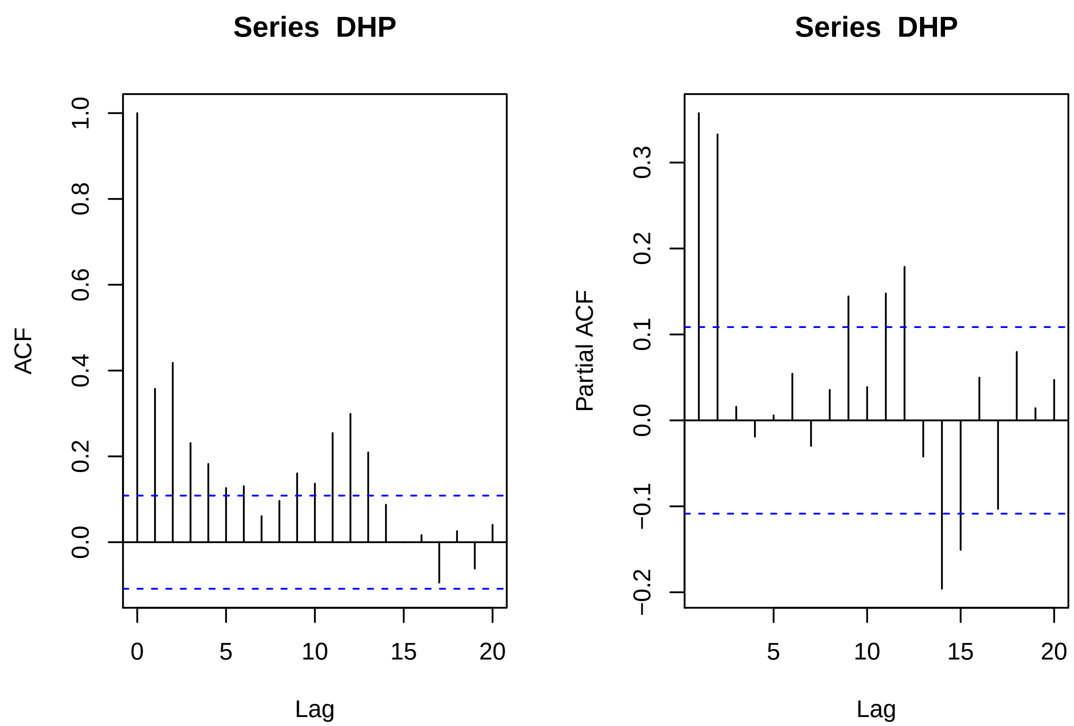
```r
acf(HP,lag=24)
```




```r
pacf(HP,lag=24)
```

ACF AND PACF PLOTS OF SIMPLE RETURNS (DHP)

```
layout(matrix(c(1,1,2,2),2,2))
acf(DHP,lag=20)
pacf(DHP,lag=20)
```



AUGMENTED DICKEY-FULLER TEST

```
# In order to test whether the DHP time series is stationary or not(unit root), we perform the ADF test
adf.test(DHP)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  DHP
## Dickey-Fuller = -5.1732, Lag order = 6, p-value = 0.01
## alternative hypothesis: stationary
```

FIND THE COEFFICIENT MATRIX OF ARIMA MODEL BY coeftest() FUNCTION

```
# Setting the order: ARIMA(0,1,2), we can estimate the coefficient matrix (including estimates, SEs, te
arima11<-arima(DHP,order=c(0,1,2))
coeftest(arima11)
```

```
##
## z test of coefficients:
##
##      Estimate Std. Error z value Pr(>|z|)
## ma1 -0.758418   0.052550 -14.432   <2e-16 ***
```

```
## ma2 -0.067894   0.057246  -1.186   0.2356
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

FIND AIC (INFORMATION CRITERIA) OF ARIMA(1,0,2) AND ARIMA(1,0,1)

```r
# ARIMA(1,0,2)
arima11<-arima(DHP,order=c(1,0,2))
AIC(arima11)
```

```
## [1] 925.6487
```

```r
# ARIMA(1,0,1)
arima11<-arima(DHP,order=c(1,0,1))
AIC(arima11)
```

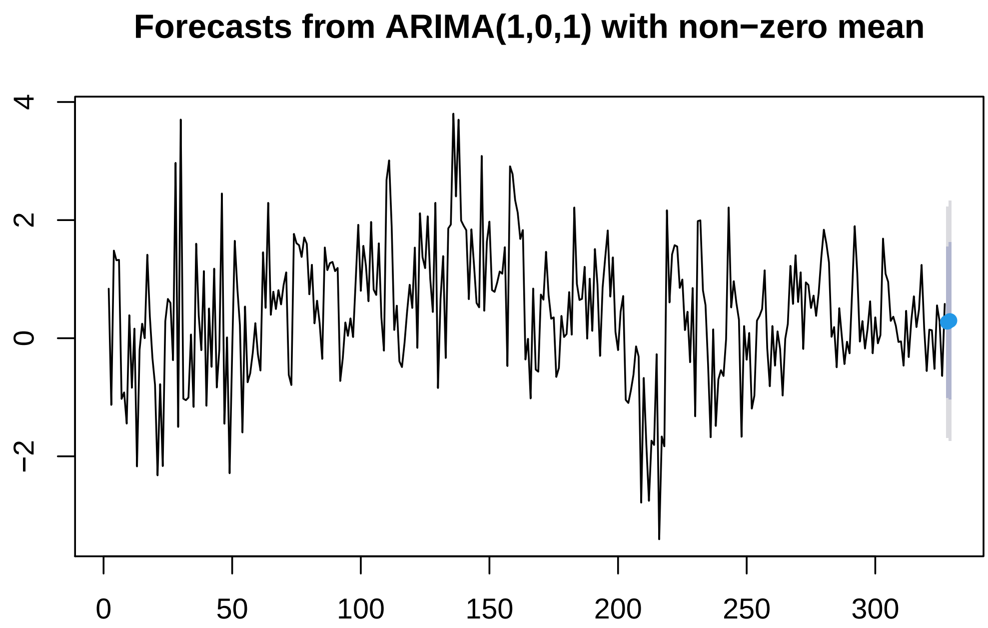
```
## [1] 933.4199
```

FORECASTING IN TIME SERIES

```r
# forecasting for ARIMA(1,0,1) over 2 periods (h=2)
fcast <- forecast(arima11, h=2)
summary(fcast)
```

```
##
## Forecast method: ARIMA(1,0,1) with non-zero mean
##
## Model Information:
##
## Call:
## arima(x = DHP, order = c(1, 0, 1))
##
## Coefficients:
##          ar1      ma1  intercept
##       0.8224  -0.5417     0.4286
## s.e.  0.0596   0.0877     0.1414
##
## sigma^2 estimated as 0.9999:  log likelihood = -462.71,  aic = 933.42
##
## Error measures:
##                        ME      RMSE       MAE      MPE    MAPE      MASE
## Training set 5.715104e-06 0.9999633 0.7567761 86.34112 206.557 0.8067148
##                     ACF1
## Training set -0.05935702
##
## Forecasts:
##     Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## 328      0.2706600 -1.010844 1.552164 -1.689232 2.230552
## 329      0.2987119 -1.032304 1.629727 -1.736901 2.334324
```

```r
# Plotting
plot(fcast)
```

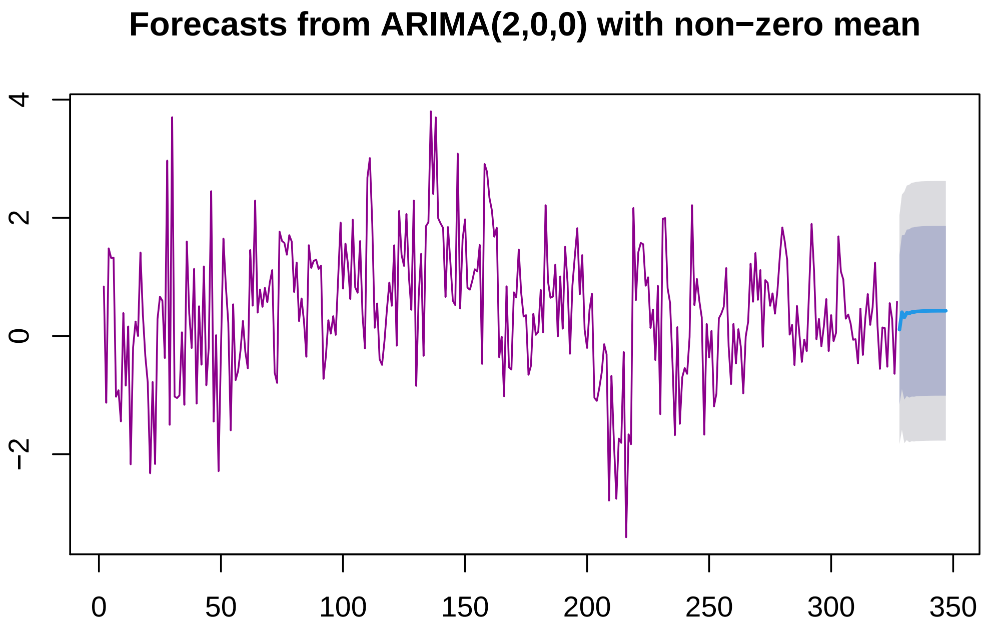


FINDING THE BEST ARIMA MODEL ACCORDING TO AIC, BIC VALUE

```r
# Instead of finding the best ARIMA model for time series, we can use the auto.arima() function to find

auto.arima(DHP, max.order = 5) #example with max order =5
```

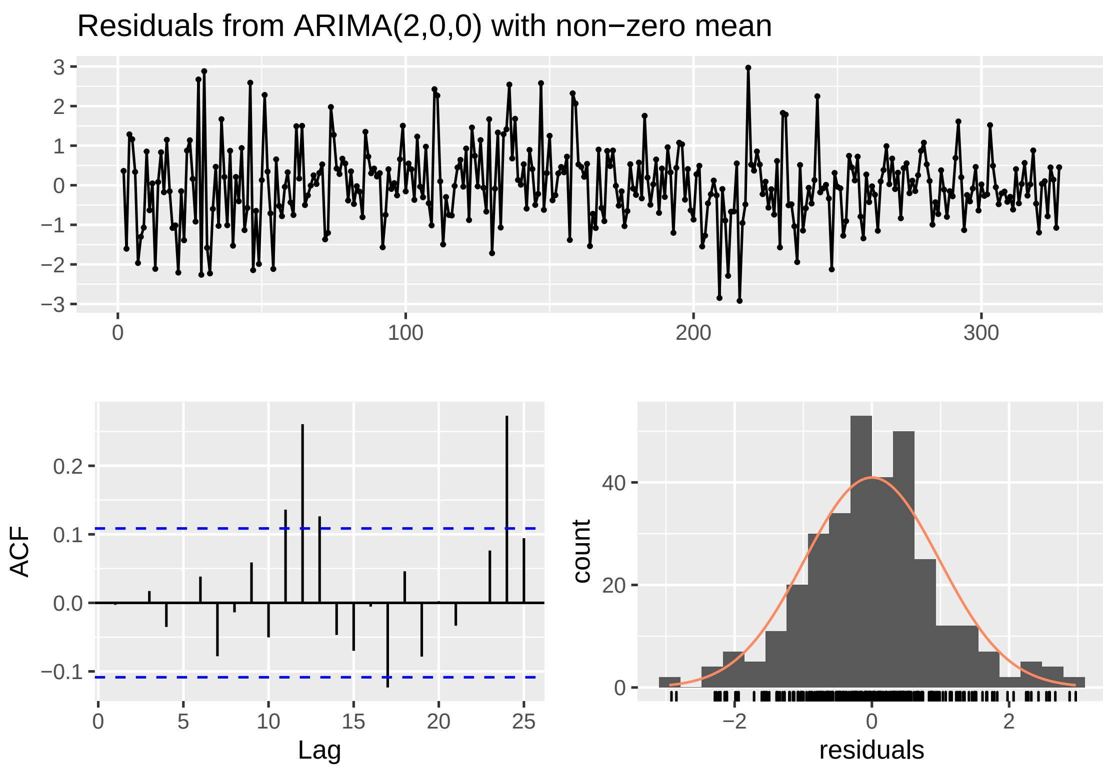
```
## Series: DHP
## ARIMA(2,0,0) with non-zero mean
##
## Coefficients:
##          ar1     ar2    mean
##       0.2361  0.3340  0.4275
## s.e.  0.0521  0.0523  0.1259
##
## sigma^2 estimated as 0.9756:  log likelihood=-457.23
## AIC=922.46   AICc=922.58   BIC=937.61
```

```r
# we find optimal ARIMA model for returns (DHP) using max arima order =10
fit<-auto.arima(DHP, max.order = 10)

# forecasting of 'fit' over 20 periods (extra)
plot(forecast(fit,h=20),col='darkmagenta')
```



LJUNG-BOX TEST TO CHECK IF THE RESIDUALS SERIES IS WHITE NOISE PROCESS

```
checkresiduals(fit)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,0,0) with non-zero mean
## Q* = 5.1323, df = 7, p-value = 0.6438
##
## Model df: 3.   Total lags used: 10
```