Aplicación 2.1. Valoración de activos en el mercado de valores: el modelo APT (Arbitrage Pricing Theory)

J. Ramajo

2020

Definamos la rentabilidad (en porcentaje) de un activo como , donde *p*1 y *p*0 son, respectivamente, los precios de cotización del valor (título u obligación) al final y al principio de un período de tiempo y *d* es el dividendo cobrado (si lo hay) durante ese período. Denominemos por *Rf* al rendimiento de un activo libre de riesgo. Finalmente, denotemos por *Rm* a la rentabilidad que ofrece la cartera de mercado.

La ecuación de valoración básica del modelo APT puede definirse del siguiente modo:

donde las variables *F* representan distintos factores que pueden afectar a la rentabilidad observada del título.

En este ejercicio se va a estimar el modelo APT con datos de series temporales de carácter mensual para el período que va desde marzo de 1986 hasta marzo de 2018.

#  
library(car)

## Loading required package: carData

library(quantreg)

## Loading required package: SparseM

##   
## Attaching package: 'SparseM'

## The following object is masked from 'package:base':  
##   
## backsolve

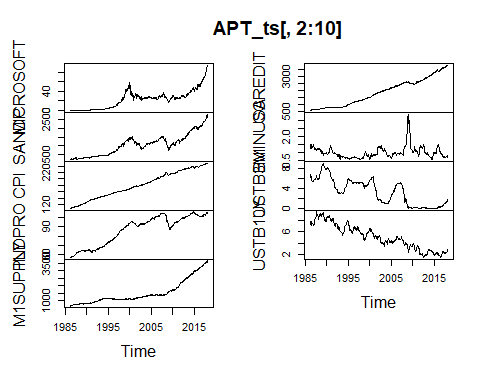
library(lmtest)

## Loading required package: zoo

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

library(sandwich)  
library(dynlm)  
library(moments)  
library(strucchange)  
#  
load("APT\_USA.RData")  
APT\_ts <- ts(macro, start=c(1986,3), end = c(2018,3), frequency = 12)  
plot(APT\_ts[,2:10])



#  
# X <- APT\_ts[,"X"]  
#  
S(lm\_msoft\_0 <- lm( ermsoft ~ ersandp + dprod + dcredit + dinflation + dmoney + dspread + rterm , data = APT\_ts ))

## Call: lm(formula = ermsoft ~ ersandp + dprod + dcredit + dinflation + dmoney +  
## dspread + rterm, data = APT\_ts)  
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.326002 0.475481 2.789 0.00556 \*\*   
## ersandp 1.280799 0.094354 13.574 < 2e-16 \*\*\*  
## dprod -0.303032 0.736881 -0.411 0.68113   
## dcredit -0.025364 0.027149 -0.934 0.35078   
## dinflation 2.194670 1.264299 1.736 0.08341 .   
## dmoney -0.006871 0.015568 -0.441 0.65919   
## dspread 2.260064 4.140284 0.546 0.58548   
## rterm 4.733069 1.715814 2.758 0.00609 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.845 on 375 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.3452  
## F-statistic: 28.24 on 7 and 375 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2674.67 2710.20

linearHypothesis(lm\_msoft\_0,c("dprod=0","dcredit=0","dmoney=0","dspread=0"))

## Linear hypothesis test  
##   
## Hypothesis:  
## dprod = 0  
## dcredit = 0  
## dmoney = 0  
## dspread = 0  
##   
## Model 1: restricted model  
## Model 2: ermsoft ~ ersandp + dprod + dcredit + dinflation + dmoney + dspread +   
## rterm  
##   
## Res.Df RSS Df Sum of Sq F Pr(>F)  
## 1 379 23180   
## 2 375 23078 4 101.88 0.4139 0.7986

S(lm\_msoft\_1 <- lm( ermsoft ~ ersandp + dinflation + rterm , data = APT\_ts ))

## Call: lm(formula = ermsoft ~ ersandp + dinflation + rterm, data = APT\_ts)  
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.0209 0.4010 2.546 0.01129 \*   
## ersandp 1.2663 0.0921 13.750 < 2e-16 \*\*\*  
## dinflation 2.1874 1.2082 1.810 0.07101 .   
## rterm 4.7388 1.7087 2.773 0.00582 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.821 on 379 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.3423  
## F-statistic: 65.75 on 3 and 379 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2668.36 2688.10

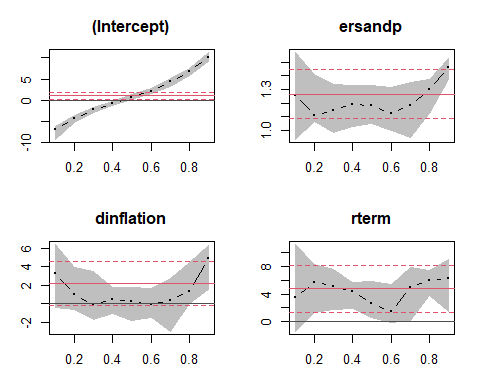
#  
# Regresiones cuantilíticas  
#  
S(qreg\_msoft\_1 <- rq( ermsoft ~ ersandp + dinflation + rterm , data = APT\_ts , tau=0.5 ))

##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = 0.5,   
## data = APT\_ts)  
##   
## tau: [1] 0.5  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) 0.69922 0.10649 1.33563  
## ersandp 1.18261 1.04725 1.33011  
## dinflation 0.22490 -1.81375 1.74272  
## rterm 2.69062 0.64957 5.77033

S(qreg\_msoft\_2 <- rq( ermsoft ~ ersandp + dinflation + rterm , data = APT\_ts , tau=seq(0.1,0.9,0.1)) )

##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.1  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) -6.93554 -9.22261 -6.30025  
## ersandp 1.25929 0.93517 1.56959  
## dinflation 3.23806 -0.35358 6.35754  
## rterm 3.47988 -1.34468 11.10778  
##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.2  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) -4.36624 -5.18185 -3.72122  
## ersandp 1.10747 1.06146 1.41223  
## dinflation 0.99008 -0.62408 3.93169  
## rterm 5.73882 1.29362 8.25746  
##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.3  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) -2.29585 -3.00542 -1.72686  
## ersandp 1.14891 0.98290 1.33385  
## dinflation -0.16247 -1.68276 3.48122  
## rterm 5.10024 1.73633 7.55012  
##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.4  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) -0.83051 -1.20180 -0.31972  
## ersandp 1.18902 1.03049 1.32768  
## dinflation 0.50790 -1.01487 1.75610  
## rterm 4.33329 1.92425 5.65259  
##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.5  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) 0.69922 0.10649 1.33563  
## ersandp 1.18261 1.04725 1.33011  
## dinflation 0.22490 -1.81375 1.74272  
## rterm 2.69062 0.64957 5.77033  
##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.6  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) 2.12678 1.57801 2.86407  
## ersandp 1.12560 0.99940 1.31542  
## dinflation -0.09797 -1.52715 1.61344  
## rterm 1.47236 -0.03570 5.32329  
##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.7  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) 4.32436 3.25631 5.12330  
## ersandp 1.18223 0.94388 1.35279  
## dinflation 0.38276 -2.94513 2.74575  
## rterm 4.93504 0.17177 7.86586  
##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.8  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) 6.75502 5.79420 7.40826   
## ersandp 1.29813 1.12571 1.37385   
## dinflation 1.33370 0.08681 4.47524   
## rterm 6.01196 3.81194 7.42345   
##   
## Call: rq(formula = ermsoft ~ ersandp + dinflation + rterm, tau = seq(0.1,   
## 0.9, 0.1), data = APT\_ts)  
##   
## tau: [1] 0.9  
##   
## Coefficients:  
## coefficients lower bd upper bd  
## (Intercept) 10.07336 9.19717 11.20272  
## ersandp 1.46360 1.37506 1.52635  
## dinflation 4.92445 1.54265 6.28507  
## rterm 6.21613 1.44595 9.03082

plot(summary(qreg\_msoft\_2), level=0.95)

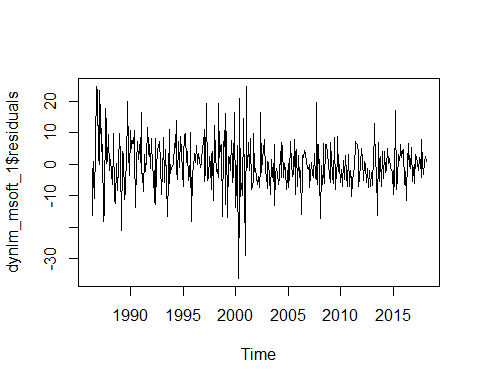


#   
# Heteroscedasticidad y autocorrelación

#  
  
S(dynlm\_msoft\_1 <- dynlm( ermsoft ~ ersandp + dinflation + rterm , data = APT\_ts ))

## Call: dynlm(formula = ermsoft ~ ersandp + dinflation + rterm, data = APT\_ts)  
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.0209 0.4010 2.546 0.01129 \*   
## ersandp 1.2663 0.0921 13.750 < 2e-16 \*\*\*  
## dinflation 2.1874 1.2082 1.810 0.07101 .   
## rterm 4.7388 1.7087 2.773 0.00582 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.821 on 379 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.3423  
## F-statistic: 65.75 on 3 and 379 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2668.36 2688.10

plot(dynlm\_msoft\_1$residuals)



bptest(formula(dynlm\_msoft\_1), data = APT\_ts , studentize = F)

##   
## Breusch-Pagan test  
##   
## data: formula(dynlm\_msoft\_1)  
## BP = 4.1765, df = 3, p-value = 0.243

bptest(formula(dynlm\_msoft\_1), data = APT\_ts , studentize = T)

##   
## studentized Breusch-Pagan test  
##   
## data: formula(dynlm\_msoft\_1)  
## BP = 2.0618, df = 3, p-value = 0.5597

dwtest(dynlm\_msoft\_1)

##   
## Durbin-Watson test  
##   
## data: dynlm\_msoft\_1  
## DW = 2.0837, p-value = 0.7902  
## alternative hypothesis: true autocorrelation is greater than 0

bgtest(dynlm\_msoft\_1,order = 10)

##   
## Breusch-Godfrey test for serial correlation of order up to 10  
##   
## data: dynlm\_msoft\_1  
## LM test = 4.6717, df = 10, p-value = 0.912

S(dynlm\_msoft\_1 , vcov. = vcovHC(dynlm\_msoft\_1,type="HC1")) # White

## Call: dynlm(formula = ermsoft ~ ersandp + dinflation + rterm, data = APT\_ts)  
## Standard errors computed by vcovHC(dynlm\_msoft\_1, type = "HC1")   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.02089 0.39622 2.577 0.01036 \*   
## ersandp 1.26628 0.08881 14.258 < 2e-16 \*\*\*  
## dinflation 2.18742 1.28377 1.704 0.08922 .   
## rterm 4.73880 1.71239 2.767 0.00593 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.821 on 379 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.3423  
## F-statistic: 67.95 on 3 and 379 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2668.36 2688.10

S(dynlm\_msoft\_1 , vcov. = vcovHAC(dynlm\_msoft\_1)) # HAC (automático)

## Call: dynlm(formula = ermsoft ~ ersandp + dinflation + rterm, data = APT\_ts)  
## Standard errors computed by vcovHAC(dynlm\_msoft\_1)   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.02089 0.39565 2.580 0.01025 \*   
## ersandp 1.26628 0.08902 14.225 < 2e-16 \*\*\*  
## dinflation 2.18742 1.28671 1.700 0.08995 .   
## rterm 4.73880 1.70843 2.774 0.00582 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.821 on 379 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.3423  
## F-statistic: 67.65 on 3 and 379 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2668.36 2688.10

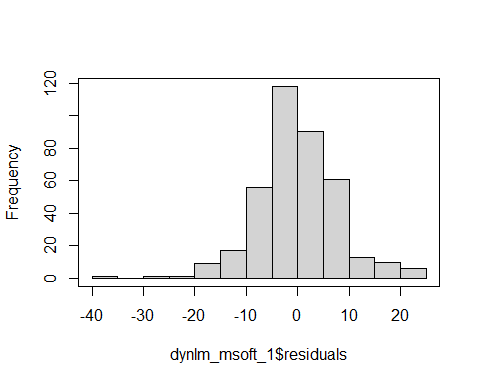
S(dynlm\_msoft\_1 , vcov. = NeweyWest(dynlm\_msoft\_1,lag = 6,adjust = T,prewhite = F)) # HAC (manual)

## Call: dynlm(formula = ermsoft ~ ersandp + dinflation + rterm, data = APT\_ts)  
## Standard errors computed by NeweyWest(dynlm\_msoft\_1, lag = 6, adjust = T, prewhite = F)   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.02089 0.38685 2.639 0.00866 \*\*   
## ersandp 1.26628 0.09918 12.767 < 2e-16 \*\*\*  
## dinflation 2.18742 1.30300 1.679 0.09402 .   
## rterm 4.73880 1.75694 2.697 0.00731 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.821 on 379 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.3423  
## F-statistic: 56.02 on 3 and 379 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2668.36 2688.10

#

# Normalidad de los errores

#  
  
hist(dynlm\_msoft\_1$residuals , main = "")  
box()



skewness(dynlm\_msoft\_1$residuals)

## [1] 0.006697427

kurtosis(dynlm\_msoft\_1$residuals)

## [1] 5.051381

jarque.test(lm\_msoft\_1$residuals)

##   
## Jarque-Bera Normality Test  
##   
## data: lm\_msoft\_1$residuals  
## JB = 67.158, p-value = 2.665e-15  
## alternative hypothesis: greater

agostino.test(dynlm\_msoft\_1$residuals)

##   
## D'Agostino skewness test  
##   
## data: dynlm\_msoft\_1$residuals  
## skew = 0.0066974, z = 0.0545108, p-value = 0.9565  
## alternative hypothesis: data have a skewness

anscombe.test(dynlm\_msoft\_1$residuals)

##   
## Anscombe-Glynn kurtosis test  
##   
## data: dynlm\_msoft\_1$residuals  
## kurt = 5.0514, z = 4.6693, p-value = 3.022e-06  
## alternative hypothesis: kurtosis is not equal to 3

#

# Variables ficticias

#  
# macro$Date = as.Date(macro$Date)  
# macro$APR00DUM = as.integer(macro$Date == as.Date("2000-04-01"))  
# macro$DEC00DUM = as.integer(macro$Date == as.Date("2000-12-01"))  
  
# require(lubridate)  
# macro$JANDUM = as.integer(month(macro$Date) == 1)  
# Añadir JANDUM a la regresión siguiente (+JANDUM)  
  
S(lm\_msoft\_1\_dummy <- lm( ermsoft ~ ersandp + dinflation + rterm + APR00DUM + DEC00DUM , data = APT\_ts ))

## Call: lm(formula = ermsoft ~ ersandp + dinflation + rterm + APR00DUM +  
## DEC00DUM, data = APT\_ts)  
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.20338 0.38369 3.136 0.001845 \*\*   
## ersandp 1.24275 0.08798 14.125 < 2e-16 \*\*\*  
## dinflation 1.44293 1.16084 1.243 0.214637   
## rterm 4.27217 1.63236 2.617 0.009223 \*\*   
## APR00DUM -37.20290 7.54143 -4.933 1.22e-06 \*\*\*  
## DEC00DUM -29.13635 7.47358 -3.899 0.000115 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.462 on 377 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.4044  
## F-statistic: 51.2 on 5 and 377 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2634.36 2662.00

jarque.test(lm\_msoft\_1\_dummy$residuals)

##   
## Jarque-Bera Normality Test  
##   
## data: lm\_msoft\_1\_dummy$residuals  
## JB = 30.456, p-value = 2.436e-07  
## alternative hypothesis: greater

agostino.test(lm\_msoft\_1\_dummy$residuals)

##   
## D'Agostino skewness test  
##   
## data: lm\_msoft\_1\_dummy$residuals  
## skew = 0.42404, z = 3.31944, p-value = 0.000902  
## alternative hypothesis: data have a skewness

anscombe.test(lm\_msoft\_1\_dummy$residuals)

##   
## Anscombe-Glynn kurtosis test  
##   
## data: lm\_msoft\_1\_dummy$residuals  
## kurt = 4.0905, z = 3.1949, p-value = 0.001399  
## alternative hypothesis: kurtosis is not equal to 3

#

# Multicolinealidad y especificación funcional

#   
cor(APT\_ts[-(1:2),c("ersandp", "dprod","dcredit","dinflation","dmoney","dspread","rterm")])

## ersandp dprod dcredit dinflation dmoney  
## ersandp 1.00000000 -0.01678420 0.047664705 -0.00228811 0.036726343  
## dprod -0.01678420 1.00000000 0.094273354 -0.14355079 -0.052514358  
## dcredit 0.04766470 0.09427335 1.000000000 -0.02460369 0.150165099  
## dinflation -0.00228811 -0.14355079 -0.024603694 1.00000000 -0.093571291  
## dmoney 0.03672634 -0.05251436 0.150165099 -0.09357129 1.000000000  
## dspread -0.17628529 -0.05275628 0.062818012 -0.22710010 0.170698675  
## rterm -0.04564889 -0.04375067 -0.004029469 0.04160626 0.003800624  
## dspread rterm  
## ersandp -0.17628529 -0.045648891  
## dprod -0.05275628 -0.043750669  
## dcredit 0.06281801 -0.004029469  
## dinflation -0.22710010 0.041606256  
## dmoney 0.17069868 0.003800624  
## dspread 1.00000000 -0.017622374  
## rterm -0.01762237 1.000000000

vif(lm\_msoft\_0)

## ersandp dprod dcredit dinflation dmoney dspread rterm   
## 1.045330 1.047018 1.039351 1.090154 1.062975 1.132181 1.005900

resettest(lm\_msoft\_0,power = 2)

##   
## RESET test  
##   
## data: lm\_msoft\_0  
## RESET = 1.5705, df1 = 1, df2 = 374, p-value = 0.2109

resettest(lm\_msoft\_0,power = 2:3)

##   
## RESET test  
##   
## data: lm\_msoft\_0  
## RESET = 1.2908, df1 = 2, df2 = 373, p-value = 0.2763

#

# Cambio estructural

#  
  
S(lm\_msoft\_1\_dummy <- lm( ermsoft ~ ersandp + dinflation + rterm + APR00DUM + DEC00DUM , data = APT\_ts ))

## Call: lm(formula = ermsoft ~ ersandp + dinflation + rterm + APR00DUM +  
## DEC00DUM, data = APT\_ts)  
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.20338 0.38369 3.136 0.001845 \*\*   
## ersandp 1.24275 0.08798 14.125 < 2e-16 \*\*\*  
## dinflation 1.44293 1.16084 1.243 0.214637   
## rterm 4.27217 1.63236 2.617 0.009223 \*\*   
## APR00DUM -37.20290 7.54143 -4.933 1.22e-06 \*\*\*  
## DEC00DUM -29.13635 7.47358 -3.899 0.000115 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.462 on 377 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.4044  
## F-statistic: 51.2 on 5 and 377 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2634.36 2662.00

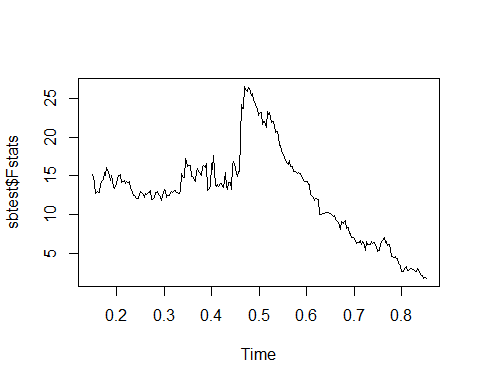
S(lm\_msoft\_1 <- lm( ermsoft ~ ersandp + dinflation + rterm , data = APT\_ts ))

## Call: lm(formula = ermsoft ~ ersandp + dinflation + rterm, data = APT\_ts)  
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.0209 0.4010 2.546 0.01129 \*   
## ersandp 1.2663 0.0921 13.750 < 2e-16 \*\*\*  
## dinflation 2.1874 1.2082 1.810 0.07101 .   
## rterm 4.7388 1.7087 2.773 0.00582 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard deviation: 7.821 on 379 degrees of freedom  
## (2 observations deleted due to missingness)  
## Multiple R-squared: 0.3423  
## F-statistic: 65.75 on 3 and 379 DF, p-value: < 2.2e-16   
## AIC BIC   
## 2668.36 2688.10

anova(lm\_msoft\_1, lm\_msoft\_1\_dummy)

## Analysis of Variance Table  
##   
## Model 1: ermsoft ~ ersandp + dinflation + rterm  
## Model 2: ermsoft ~ ersandp + dinflation + rterm + APR00DUM + DEC00DUM  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 379 23180   
## 2 377 20991 2 2189.2 19.66 7.559e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#  
sbtest = Fstats(formula(lm\_msoft\_1\_dummy),data = APT\_ts)  
plot(sbtest$Fstats) # Desde el 15% hasta el 85% de los datos



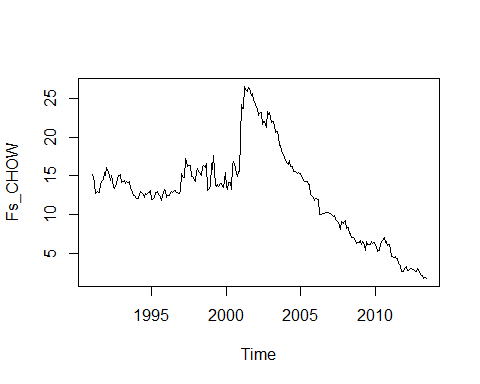
sbtest[["Fstats"]]

## Time Series:  
## Start = c(0, 58)   
## End = c(0, 327)   
## Frequency = 383   
## [1] 15.230218 14.926459 14.060668 12.700619 13.024027 12.847074 12.900004  
## [8] 14.044690 14.461736 14.583134 15.431237 15.040200 16.085147 15.449334  
## [15] 15.010218 14.364275 15.035150 13.631498 13.438098 13.640417 14.084027  
## [22] 15.024133 15.027487 15.175902 14.196067 14.303595 14.413801 14.078196  
## [29] 14.262718 14.093736 14.224163 13.562885 13.270058 12.538528 12.522698  
## [36] 12.352744 12.076621 12.058715 12.577509 13.028525 12.830584 12.560849  
## [43] 12.203075 12.677364 12.632226 12.778673 12.906037 13.057037 11.998765  
## [50] 12.009533 12.167736 12.871829 12.866377 13.009732 12.529840 12.292971  
## [57] 11.883952 12.522221 13.303841 13.078282 12.269628 12.501742 12.437770  
## [64] 12.927360 12.942188 12.911296 13.071517 13.106167 12.830149 12.840341  
## [71] 12.785493 13.159436 15.270016 14.979235 14.831373 17.202670 16.891203  
## [78] 16.251920 16.311435 16.285480 14.992420 14.990804 14.799167 14.265001  
## [85] 15.595324 15.911879 15.521777 15.323901 15.096503 16.207276 16.317718  
## [92] 16.131376 16.570201 13.110778 13.280138 13.681476 16.582334 15.884941  
## [99] 17.704279 13.815428 13.618864 13.898627 13.596031 14.048124 13.972621  
## [106] 13.668376 13.559170 15.477306 13.694140 13.301623 14.184123 14.184123  
## [113] 13.235077 16.487932 16.859456 16.546582 15.257385 14.930459 15.538147  
## [120] 15.538147 24.266337 23.649854 23.657191 26.562916 26.051620 25.891040  
## [127] 26.380012 26.244282 25.426562 25.678667 24.889624 24.659879 23.884785  
## [134] 23.766575 22.790996 23.110790 23.180541 21.630511 21.976037 22.044947  
## [141] 21.217357 23.322761 22.883755 23.158421 21.936487 21.982883 22.011921  
## [148] 21.561595 20.605213 20.771941 20.119681 18.947885 18.993894 17.889873  
## [155] 17.735807 17.533185 16.990428 16.547435 16.536913 16.971766 16.278779  
## [162] 16.226128 15.636155 15.522440 15.520383 15.357286 15.435228 15.362474  
## [169] 15.349080 14.783916 14.488504 14.236741 14.267478 14.229668 13.856174  
## [176] 13.931344 12.611001 12.361562 12.256491 11.787860 12.049922 12.117345  
## [183] 11.924354 9.976131 10.063372 10.061502 10.103818 10.175531 10.245715  
## [190] 10.284700 10.240400 10.110023 10.132825 10.038485 9.825805 9.922786  
## [197] 9.565544 9.243449 9.033067 8.610208 8.052556 9.180926 8.890138  
## [204] 9.141183 9.252352 8.228391 8.289140 7.625332 7.521740 7.051023  
## [211] 7.009234 6.949263 6.680514 6.321804 6.450136 6.442860 6.677955  
## [218] 6.173124 6.536852 6.137718 5.375458 6.532105 6.176879 6.100630  
## [225] 6.088645 6.503358 6.333941 6.375601 5.960213 5.893573 5.261671  
## [232] 5.379120 6.173143 6.501147 6.673092 7.077537 6.347912 6.528581  
## [239] 6.003607 6.137893 5.689007 4.609876 4.573790 4.445938 4.592796  
## [246] 4.288903 4.344708 3.649426 3.459371 2.664117 2.653897 2.648832  
## [253] 3.160002 3.251679 2.829707 2.831463 3.015851 3.007003 2.913481  
## [260] 2.946060 2.638081 2.712269 3.047379 2.938450 2.241001 2.130972  
## [267] 2.166131 1.771691 1.874242 1.793715

nFstats <- NROW(sbtest$Fstats) #dimensión del vector: T=383 ; 383x0.15=57.45 -> 57   
# Fstats comienzan en: Mayo 1986 (tener en cuenta los 2 NAs de dinflation) + 57 = 59 (Enero 1991)  
Fs\_CHOW <- ts(sbtest$Fstats, start=c(1991,1), frequency = 12)  
Fs\_CHOW

## Jan Feb Mar Apr May Jun Jul  
## 1991 15.230218 14.926459 14.060668 12.700619 13.024027 12.847074 12.900004  
## 1992 16.085147 15.449334 15.010218 14.364275 15.035150 13.631498 13.438098  
## 1993 14.196067 14.303595 14.413801 14.078196 14.262718 14.093736 14.224163  
## 1994 12.076621 12.058715 12.577509 13.028525 12.830584 12.560849 12.203075  
## 1995 11.998765 12.009533 12.167736 12.871829 12.866377 13.009732 12.529840  
## 1996 12.269628 12.501742 12.437770 12.927360 12.942188 12.911296 13.071517  
## 1997 15.270016 14.979235 14.831373 17.202670 16.891203 16.251920 16.311435  
## 1998 15.595324 15.911879 15.521777 15.323901 15.096503 16.207276 16.317718  
## 1999 16.582334 15.884941 17.704279 13.815428 13.618864 13.898627 13.596031  
## 2000 13.694140 13.301623 14.184123 14.184123 13.235077 16.487932 16.859456  
## 2001 24.266337 23.649854 23.657191 26.562916 26.051620 25.891040 26.380012  
## 2002 23.884785 23.766575 22.790996 23.110790 23.180541 21.630511 21.976037  
## 2003 21.936487 21.982883 22.011921 21.561595 20.605213 20.771941 20.119681  
## 2004 16.990428 16.547435 16.536913 16.971766 16.278779 16.226128 15.636155  
## 2005 15.349080 14.783916 14.488504 14.236741 14.267478 14.229668 13.856174  
## 2006 12.049922 12.117345 11.924354 9.976131 10.063372 10.061502 10.103818  
## 2007 10.132825 10.038485 9.825805 9.922786 9.565544 9.243449 9.033067  
## 2008 9.252352 8.228391 8.289140 7.625332 7.521740 7.051023 7.009234  
## 2009 6.677955 6.173124 6.536852 6.137718 5.375458 6.532105 6.176879  
## 2010 5.960213 5.893573 5.261671 5.379120 6.173143 6.501147 6.673092  
## 2011 5.689007 4.609876 4.573790 4.445938 4.592796 4.288903 4.344708  
## 2012 3.160002 3.251679 2.829707 2.831463 3.015851 3.007003 2.913481  
## 2013 2.241001 2.130972 2.166131 1.771691 1.874242 1.793715   
## Aug Sep Oct Nov Dec  
## 1991 14.044690 14.461736 14.583134 15.431237 15.040200  
## 1992 13.640417 14.084027 15.024133 15.027487 15.175902  
## 1993 13.562885 13.270058 12.538528 12.522698 12.352744  
## 1994 12.677364 12.632226 12.778673 12.906037 13.057037  
## 1995 12.292971 11.883952 12.522221 13.303841 13.078282  
## 1996 13.106167 12.830149 12.840341 12.785493 13.159436  
## 1997 16.285480 14.992420 14.990804 14.799167 14.265001  
## 1998 16.131376 16.570201 13.110778 13.280138 13.681476  
## 1999 14.048124 13.972621 13.668376 13.559170 15.477306  
## 2000 16.546582 15.257385 14.930459 15.538147 15.538147  
## 2001 26.244282 25.426562 25.678667 24.889624 24.659879  
## 2002 22.044947 21.217357 23.322761 22.883755 23.158421  
## 2003 18.947885 18.993894 17.889873 17.735807 17.533185  
## 2004 15.522440 15.520383 15.357286 15.435228 15.362474  
## 2005 13.931344 12.611001 12.361562 12.256491 11.787860  
## 2006 10.175531 10.245715 10.284700 10.240400 10.110023  
## 2007 8.610208 8.052556 9.180926 8.890138 9.141183  
## 2008 6.949263 6.680514 6.321804 6.450136 6.442860  
## 2009 6.100630 6.088645 6.503358 6.333941 6.375601  
## 2010 7.077537 6.347912 6.528581 6.003607 6.137893  
## 2011 3.649426 3.459371 2.664117 2.653897 2.648832  
## 2012 2.946060 2.638081 2.712269 3.047379 2.938450  
## 2013

plot(Fs\_CHOW)



# Localización temporal de los Fstats:   
enero96 = match(as.Date("1996-01-01"),macro$Date)  
chow = sbtest$Fstats[enero96-2-57]  
chow

## [1] 13.07828

1-pchisq(chow,sbtest$nreg)

## [1] 0.04180953

# Test SupWald  
sctest(sbtest)

##   
## supF test  
##   
## data: sbtest  
## sup.F = 26.563, p-value = 0.004481

bp = which.max(sbtest$Fstats)+59  
macro$Date[bp]

## [1] "2001-05-01"

# Test CUSUM  
plot(efp(lm\_msoft\_1\_dummy,data=APT\_ts))

