favar_sdfm

laura montaldo

2023-11-04

```
library(Metrics)
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
##
## Attaching package: 'forecast'
```

```
## The following object is masked from 'package:Metrics':
##
##     accuracy
```

```
library(readxl)
library(readr)
library(boot)
library(tsDyn)
```

```
##
## Attaching package: 'tsDyn'
```

```
## The following object is masked from 'package:Metrics':
##
##     mse
```

```
library(vars)
```

```
## Loading required package: MASS
```

```
## Loading required package: strucchange
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

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

```
## Loading required package: sandwich
```

```
## Loading required package: urca
```

```
## Loading required package: lmtest
```

```
library(repr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:MASS':
##
##     select

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(dfms)
library(xts)
```

```
##
## ########################## Warning from 'xts' package ##########################
## #                                                                              #
## # The dplyr lag() function breaks how base R's lag() function is supposed to  #
## # work, which breaks lag(my_xts). Calls to lag(my_xts) that you type or       #
## # source() into this session won't work correctly.                            #
## #                                                                              #
## # Use stats::lag() to make sure you're not using dplyr::lag(), or you can add #
## # conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop           #
## # dplyr from breaking base R's lag() function.                                #
## #                                                                              #
## # Code in packages is not affected. It's protected by R's namespace mechanism #
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning.  #
## #                                                                              #
## ################################################################################

##
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
library(vars)
library(fbi)
library(forecast)
library(OOS)
library(zoo)
```

```
cat("My Working directory is: ", getwd(), "\n")
```

```
## My Working directory is:  C:/Users/user/Desktop/Tesis_Maestria/src/stage2_models
```

```
df_train <- read_csv("../../data/prepro/sfr_train.csv", show_col_types = FALSE)
df_test <- read_csv("../../data/prepro/sfr_test.csv", show_col_types = FALSE)
slow <- read_csv("../../data/prepro/slow_columns.csv", show_col_types = FALSE)
fast <- read_csv("../../data/prepro/fast_columns.csv", show_col_types = FALSE)
descr <- read.table("../../data/prepro/descripciones.txt", header = TRUE, sep = "\t")
```

Dataframe a objeto xts

```
df=df_train
# Convert the date_column to Date type
df$index <- as.Date(df$index)
# Create an xts object with the date_column as the index
xts_object <- xts(df[, -which(colnames(df) == "index")], order.by = df$index)
cat("Rango de datos:", as.character(range(index(xts_object))), "\n")
```

```
## Rango de datos: 1960-01-01 2008-01-01
```

```
data_s = scale(xts_object, center = TRUE, scale = TRUE)
cat("Tamaño de mi muestra:", dim(data_s), "\n")
```

```
## Tamaño de mi muestra: 577 121
```

Paso 1: se extraen los componentes principales de X_t incluyendo Y_t y se determina la cantidad óptima de factores

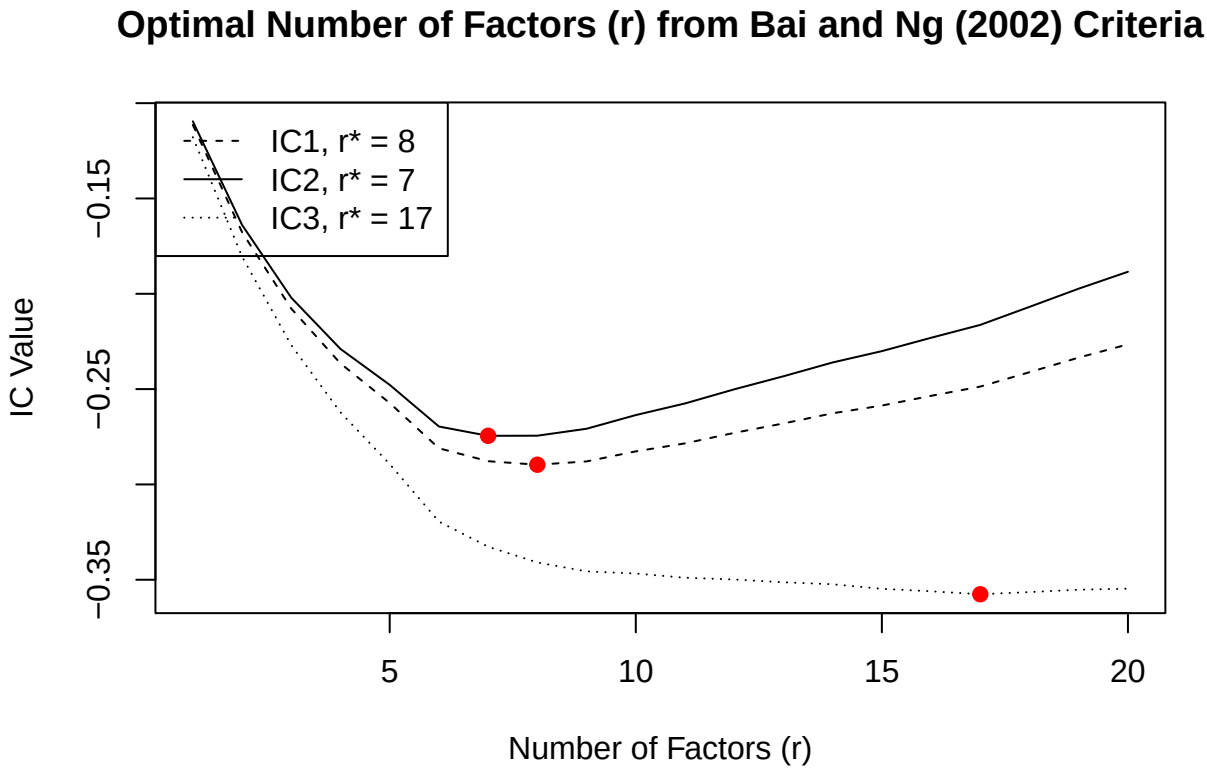
```
ics = ICr(data_s)
ic_p2_factors=ics$r.star[2]
print(ics)
```

```
## Optimal Number of Factors (r) from Bai and Ng (2002) Criteria
##
## IC1 IC2 IC3
##   8   7  17
```

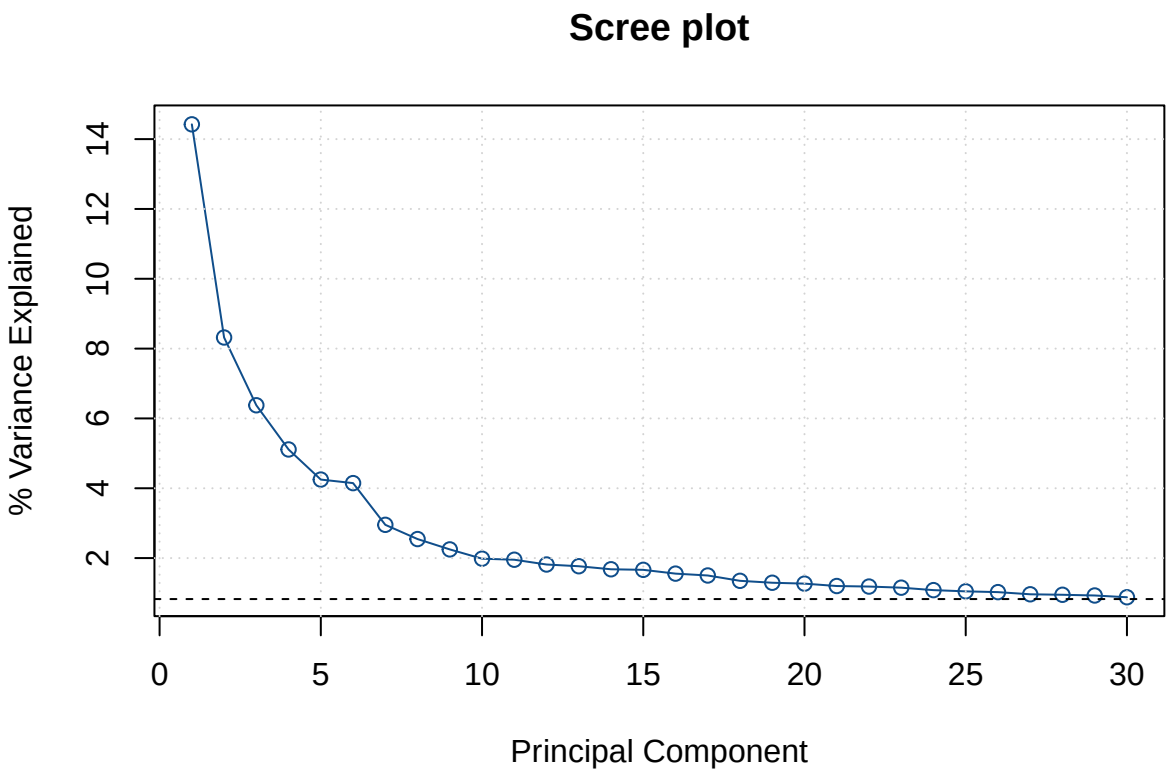
```
dim(ics$F_pca)
```

```
## [1] 577 121
```

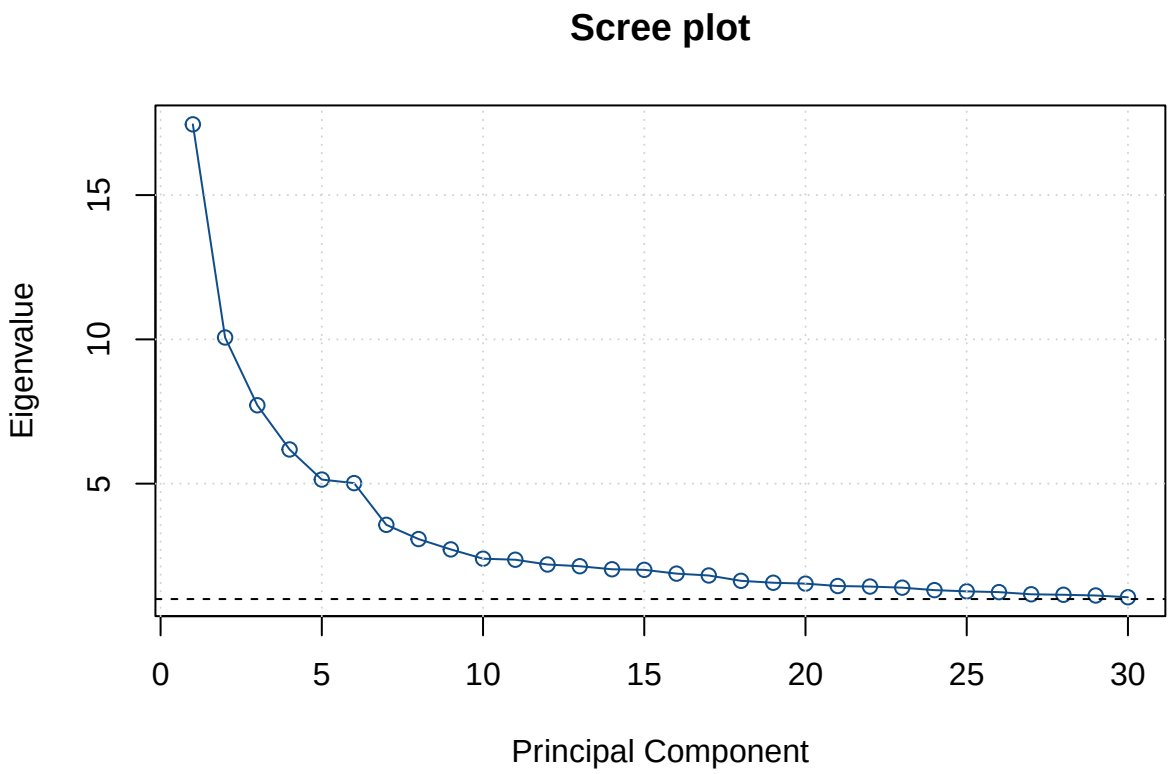
```
plot(ics)
```



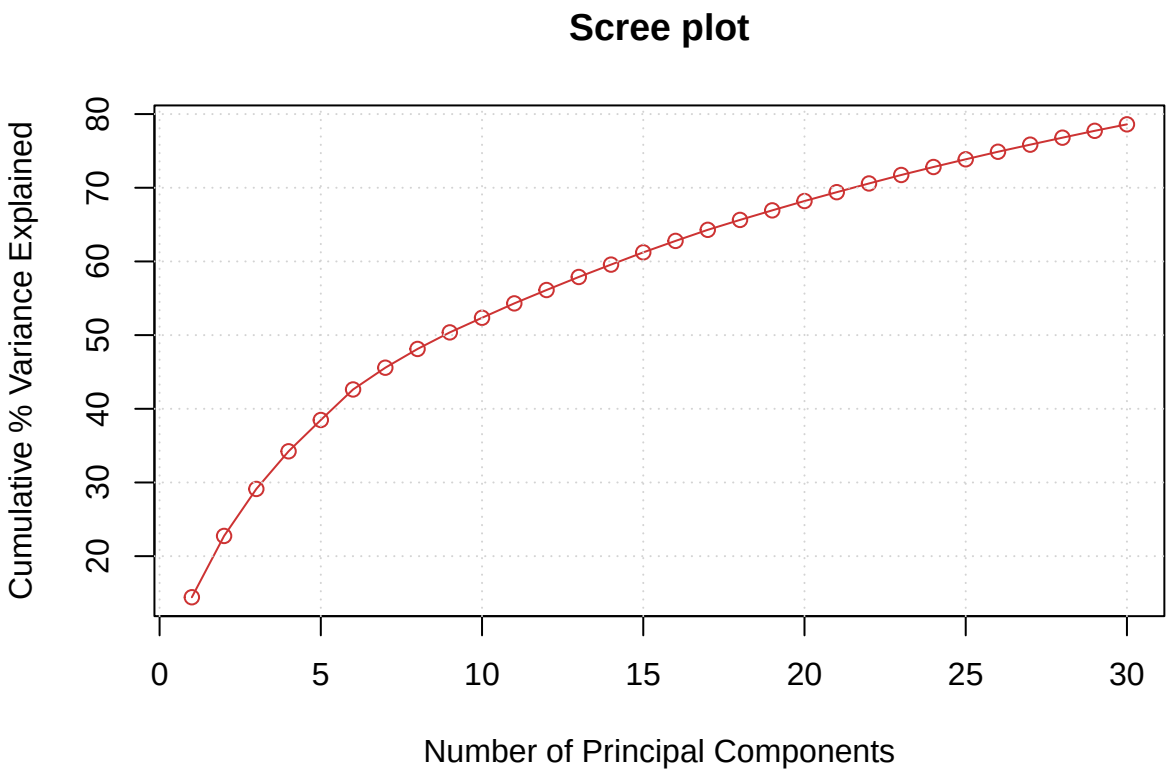

```r
screeplot(ics, main="Scree plot")
```




```
screeplot(ics, type="ev", main="Scree plot")
```




```
screeplot(ics,type="cum.pve", main="Scree plot")
```




```
# components all
C<-ics$F_pca[, 1:ic_p2_factors]
cat("Dimensiones de los factores comunes en filas y columnas:", dim(C))
```

```
## Dimensiones de los factores comunes en filas y columnas: 577 7
```

Paso 2: se extraen los componentes principales de las variables lentas

```
slow_vars <- unlist(slow$slow)
data_slow <- data_s[, slow_vars]
```

```
cat("Dimensiones de los factores de las variables lentas:", dim(data_slow))
```

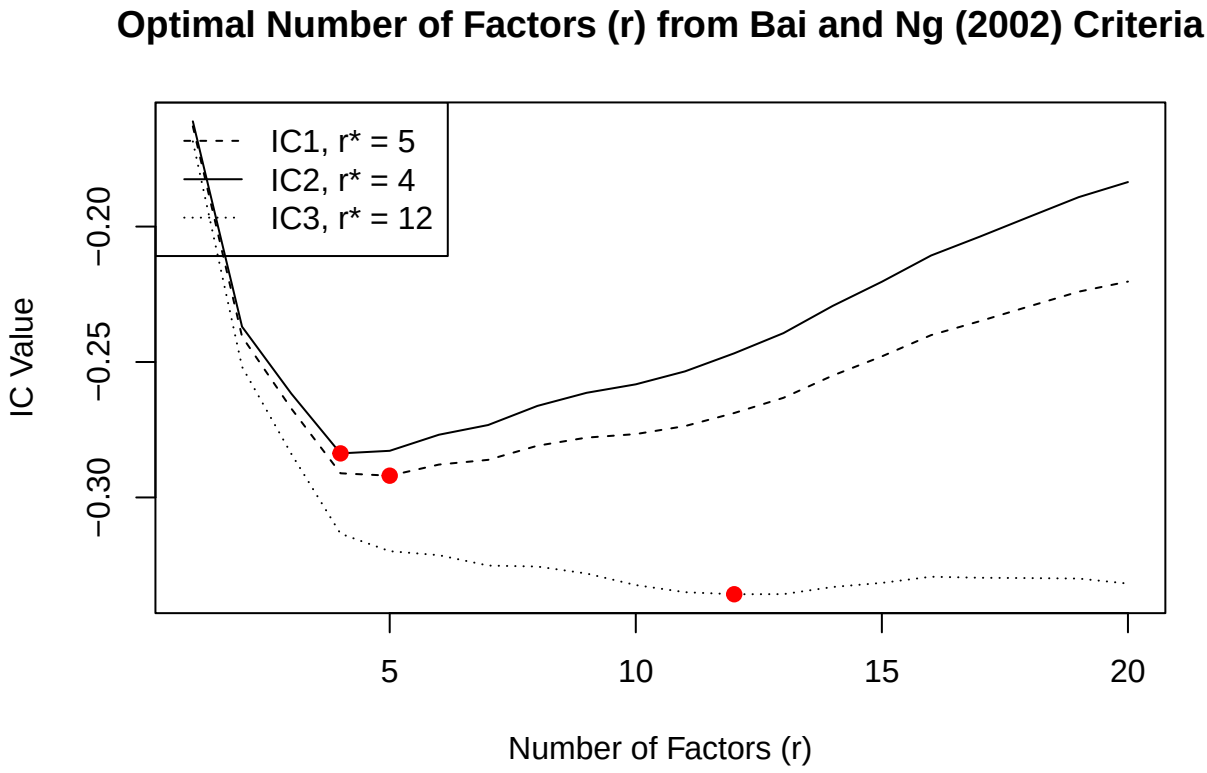
```
## Dimensiones de los factores de las variables lentas: 577 72
```

```
ics_slow = ICr(data_slow)
ic_p2_slow_factors=ics_slow$r.star[2]
```

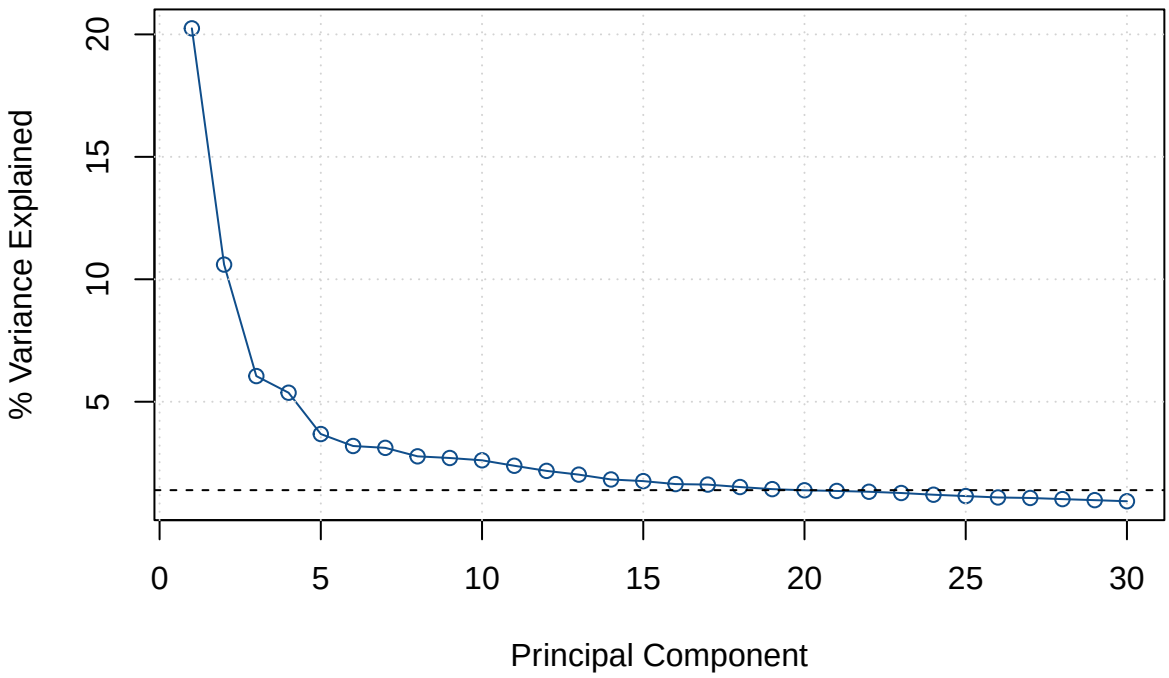
```
print(ics_slow)
```

```
## Optimal Number of Factors (r) from Bai and Ng (2002) Criteria
##
## IC1 IC2 IC3
##   5   4  12
```

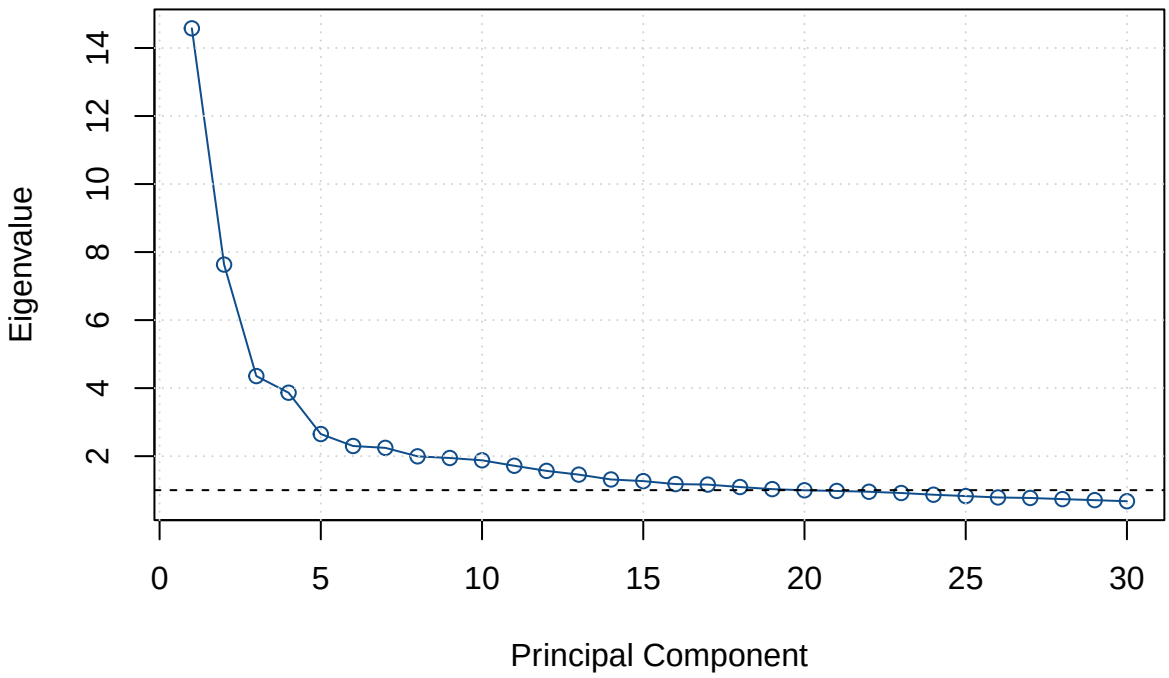
```
plot(ics_slow)
```



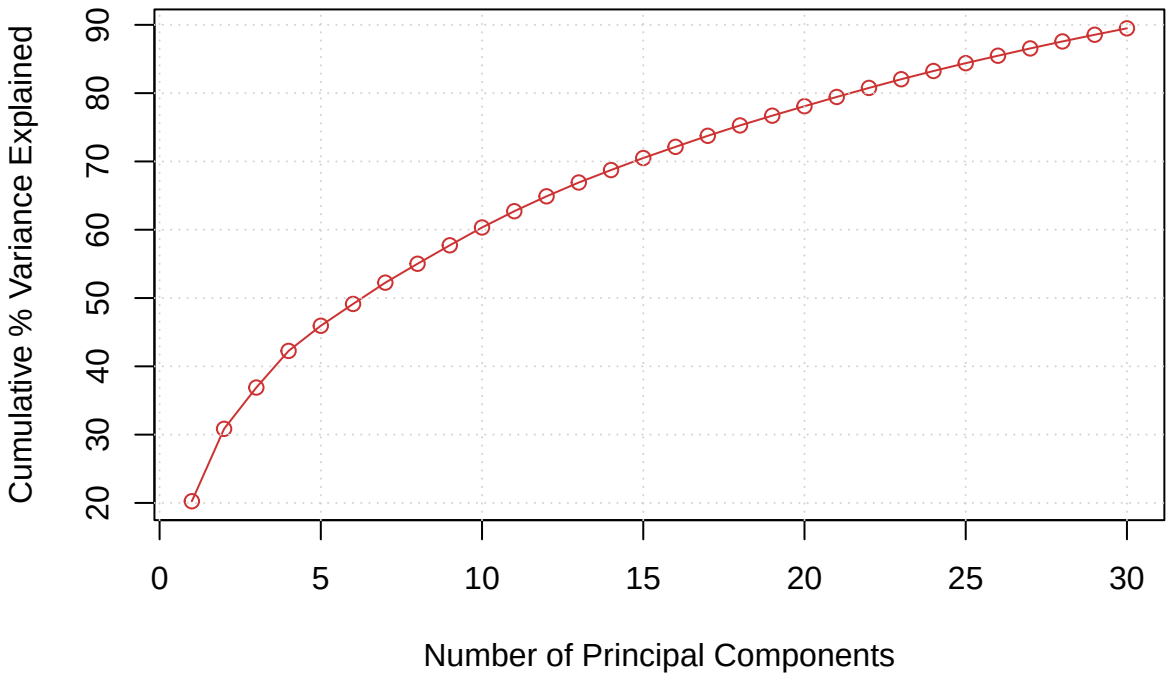

```
screeplot(ics_slow)
```




```
screeplot(ics_slow, type="ev")
```




```
screeplot(ics_slow,type="cum.pve")
```




```
F_slow<-ics$F_pca[,1:ic_p2_slow_factors]
dim(F_slow)
```

```
## [1] 577   4
```

```
dim(C)
```

```
## [1] 577   7
```

Step 3: Se limpian a los CP de los efectos de Y (FEDFUNDs)

```
fedfunds <- as.matrix(data_s[, "FEDFUNDS"])
reg <- lm(C ~ F_slow + fedfunds)
```

$\hat{F}_t$

```
F_hat <- C - data.matrix(data_s[, "FEDFUNDS"]) %*% reg$coefficients[nrow(reg$coefficients),]
dim(F_hat)
```

```
## [1] 577   7
```

Step 4: Estimo el FAVAR y las funciones de impulso-respuesta

```
data_var <- data.frame(F_hat, "FEDFUNDS" = data_s[, "FEDFUNDS"])
head(data_var)# common components
```

```
##                   PC1         PC2         PC3         PC4         PC5         PC6
```

```
## 1960-01-01 -7.4735326 -0.8138485 -2.6455136 -2.6365323 -5.1851192 -4.99703466
## 1960-02-01  0.2607595  2.8427705  0.7051430 -0.8325004  3.4964146 -1.86783561
## 1960-03-01  9.1145991  1.2271706  0.3890168 -1.7147965 -0.8151777 -2.89462563
## 1960-04-01 -1.8575241  0.4373753  1.6077299  3.7789511  2.5329622  0.09713551
## 1960-05-01  5.6978743  0.9456075 -1.8653719 -4.0243107 -1.9859044  0.80783573
## 1960-06-01  8.3183110 -2.8597768  0.1585260  0.4133536  0.2651923  0.98907113
##                  PC7    FEDFUNDS
## 1960-01-01  1.616118  0.01130991
## 1960-02-01 -4.871665 -0.04291257
## 1960-03-01 -3.234759 -0.34113620
## 1960-04-01 -7.915231  0.22819983
## 1960-05-01  3.575720 -0.17846876
## 1960-06-01 -1.816141 -1.42558579
```

```
var_select <- VARselect(data_var, lag.max = 15, type="none")
```

```
best_lag <- var_select$selection
best_lag[1]
```

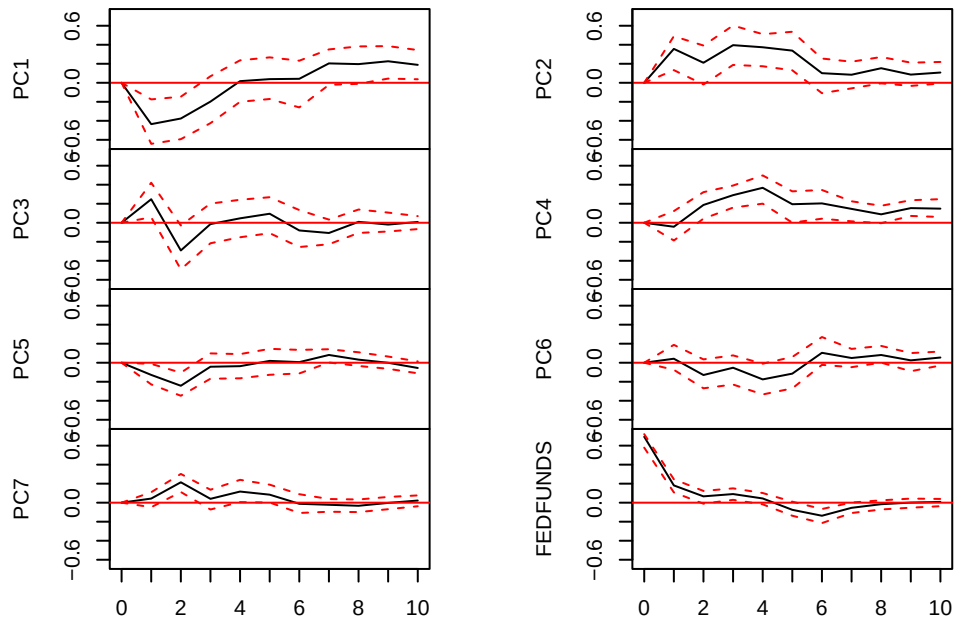
```
## AIC(n)
##      6
```

```
cat("Best # lags according to AIC:", best_lag[1], "\n")
```

```
## Best # lags according to AIC: 6
```

```
var = VAR(data_var, p =best_lag[1])
```

```
irf_results <- irf(var, impulse = "FEDFUNDS", n.ahead = 10)
plot(irf_results)
```




```
irf_point = irf(var, n.ahead = 48, impulse = "FEDFUNDS", response = "FEDFUNDS", boot = FALSE)
# Shock size of 25 basis points
impulse_sd = 0.25/sd(df$FEDFUNDS)
scale = impulse_sd/(irf_point$irf$FEDFUNDS[1]) # position of FYFF response at step 0
```

```
# Computing Loading Factors
matriz_s<- as.matrix(data_s)
matriz_fhat<- as.matrix(F_hat)
reg_loadings = lm(matriz_s ~ 0 + matriz_fhat + data_s[,"FEDFUNDS"])
loadings = reg_loadings$coefficients
#head(reg_loadings$coefficients)
#summary(reg_loadings)
Lamda_F=loadings[1:7,]
Lambda_ffr=loadings[nrow(loadings),]
```

```
#### BOOTSTRAPING ########
R = 500 # Number of simulations
nvars = dim(data_s)[2] # Number of variables
nsteps = 49 # numbers of steps
IRFs = array(c(0,0,0), dim = c(nsteps,nvars,R))
var = lineVar(data_var, lag = best_lag[1], include = "const")
for(j in 1:R){
    data_boot = VAR.boot(var, boot.scheme ="resample")
    var_boot = VAR(data_boot, lag = best_lag[1])
    irf1 = irf(var_boot, n.ahead = 48, impulse = "FEDFUNDS", boot = FALSE)
    for(i in 1:nvars){
        IRFs[,i,j] = (irf1$irf$FEDFUNDS %*% matrix(loadings[, i]))*scale
```

```
        }
} ## Boot simulations done

# Extract the quantiles of IRFs we are interested: 90% confidence intervals in BBE
Upper = array(c(0,0), dim = c(nsteps, nvars))
for(k in 1:nsteps){
    for(i in 1:nvars){
        Upper[k,i] = quantile(IRFs[k,i,], probs = c(0.95))[1]
        }
}
Lower = array(c(0,0), dim = c(nsteps, nvars))
for(k in 1:nsteps){
    for(i in 1:nvars){
        Lower[k,i] = quantile(IRFs[k,i,], probs = c(0.05))[1]
        }
}
IRF = array(c(0,0), dim = c(nsteps, nvars))
for(k in 1:nsteps){
    for(i in 1:nvars){
        IRF[k,i] = quantile(IRFs[k,i,], probs = c(0.5))[1]
        }
}
rm(var_boot)
rm(IRFs)
```

```
matching_names <- intersect(names(data_s), descr$fred)
filtered_descr <- descr %>%filter(fred %in% matching_names)
filtered_descr
```

```
##                fred                                      description
## 1               RPI                                Real Personal Income
## 2           W875RX1          Real personal income ex transfer receipts
## 3   DPCERA3M086SBEA             Real personal consumption expenditures
## 4          CMRMTSPLx              Real Manu. and Trade Industries Sales
## 5           RETAILx                     Retail and Food Services Sales
## 6            INDPRO                                          IP Index
## 7           IPFPNSS     IP: Final Products and Nonindustrial Supplies
## 8           IPFINAL                  IP: Final Products (Market Group)
## 9           IPCONGD                              IP: Consumer Goods
## 10         IPDCONGD                         IP: Durable Consumer Goods
## 11         IPNCONGD                       IP: Nondurable Consumer Goods
## 12          IPBUSEQ                             IP: Business Equipment
## 13            IPMAT                                      IP: Materials
## 14           IPDMAT                               IP: Durable Materials
## 15           IPNMAT                             IP: Nondurable Materials
## 16        IPMANSICS                              IP: Manufacturing (SIC)
## 17        IPB51222s                            IP: Residential Utilities
## 18          IPFUELS                                          IP: Fuels
## 19           CUMFNS                 Capacity Utilization: Manufacturing
## 20              HWI                 Help-Wanted Index for United States
## 21        HWIURATIO                 Ratio of Help Wanted/No. Unemployed
## 22          CLF16OV                              Civilian Labor Force
## 23           CE16OV                               Civilian Employment
## 24           UNRATE                          Civilian Unemployment Rate
```

```
## 25         UEMPMEAN          Average Duration of Unemployment (Weeks)
## 26          UEMPLT5          Civilians Unemployed - Less Than 5 Weeks
## 27        UEMP5TO14               Civilians Unemployed for 5-14 Weeks
## 28         UEMP15OV            Civilians Unemployed - 15 Weeks & Over
## 29        UEMP15T26              Civilians Unemployed for 15-26 Weeks
## 30         UEMP27OV        Civilians Unemployed for 27 Weeks and Over
## 31          CLAIMSx                                    Initial Claims
## 32           PAYEMS                      All Employees: Total nonfarm
## 33           USGOOD         All Employees: Goods-Producing Industries
## 34    CES1021000001         All Employees: Mining and Logging: Mining
## 35           USCONS                       All Employees: Construction
## 36           MANEMP                       All Employees: Manufacturing
## 37          DMANEMP                       All Employees: Durable goods
## 38         NDMANEMP                    All Employees: Nondurable goods
## 39           SRVPRD       All Employees: Service-Providing Industries
## 40            USTPU  All Employees: Trade, Transportation & Utilities
## 41         USWTRADE                     All Employees: Wholesale Trade
## 42          USTRADE                        All Employees: Retail Trade
## 43           USFIRE                 All Employees: Financial Activities
## 44           USGOVT                          All Employees: Government
## 45    CES0600000007                 Avg Weekly Hours : Goods-Producing
## 46          AWOTMAN         Avg Weekly Overtime Hours : Manufacturing
## 47           AWHMAN                   Avg Weekly Hours : Manufacturing
## 48            HOUST         Housing Starts: Total New Privately Owned
## 49            HOUSTW                                Housing Starts, West
## 50            PERMIT                 New Private Housing Permits (SAAR)
## 51         PERMITNE     New Private Housing Permits, Northeast (SAAR)
## 52          PERMITS          New Private Housing Permits, South (SAAR)
## 53          PERMITW           New Private Housing Permits, West (SAAR)
## 54          AMDMNOx                       New Orders for Durable Goods
## 55          AMDMUOx                   Un lled Orders for Durable Goods
## 56          BUSINVx                          Total Business Inventories
## 57         ISRATIOx        Total Business: Inventories to Sales Ratio
## 58             M1SL                                    M1 Money Stock
## 59             M2SL                                    M2 Money Stock
## 60           M2REAL                                 Real M2 Money Stock
## 61         TOTRESNS         Total Reserves of Depository Institutions
## 62        NONBORRES               Reserves Of Depository Institutions
## 63         BUSLOANS                   Commercial and Industrial Loans
## 64           REALLN         Real Estate Loans at All Commercial Banks
## 65         NONREVSL                          Total Nonrevolving Credit
## 66           CONSPI   Nonrevolving consumer credit to Personal Income
## 67         FEDFUNDS                       E?ective Federal Funds Rate
## 68            CP3Mx        3-Month AA Financial Commercial Paper Rate
## 69             TB3MS                              3-Month Treasury Bill:
## 70             TB6MS                              6-Month Treasury Bill:
## 71               GS1                                1-Year Treasury Rate
## 72               GS5                                5-Year Treasury Rate
## 73              GS10                               10-Year Treasury Rate
## 74               AAA         Moody s Seasoned Aaa Corporate Bond Yield
## 75               BAA         Moody s Seasoned Baa Corporate Bond Yield
## 76        COMPAPFFx           3-Month Commercial Paper Minus FEDFUNDS
## 77         TB3SMFFM                 3-Month Treasury C Minus FEDFUNDS
## 78         TB6SMFFM                 6-Month Treasury C Minus FEDFUNDS
```

```
## 79           T1YFFM                  1-Year Treasury C Minus FEDFUNDS
## 80           T5YFFM                  5-Year Treasury C Minus FEDFUNDS
## 81          T10YFFM                 10-Year Treasury C Minus FEDFUNDS
## 82           AAAFFM         Moody s Aaa Corporate Bond Minus FEDFUNDS
## 83           BAAFFM         Moody s Baa Corporate Bond Minus FEDFUNDS
## 84          EXSZUSx          Switzerland / U.S. Foreign Exchange Rate
## 85          EXJPUSx                Japan / U.S. Foreign Exchange Rate
## 86          EXUSUKx                U.S. / U.K. Foreign Exchange Rate
## 87          EXCAUSx              Canada / U.S. Foreign Exchange Rate
## 88       WPSFD49207                                PPI: Finished Goods
## 89       WPSFD49502                      PPI: Finished Consumer Goods
## 90          WPSID61                       PPI: Intermediate Materials
## 91          WPSID62                              PPI: Crude Materials
## 92        OILPRICEx                Crude Oil, spliced WTI and Cushing
## 93           PPICMM                   PPI: Metals and metal products:
## 94         CPIAUCSL                                    CPI : All Items
## 95         CPIAPPSL                                      CPI : Apparel
## 96         CPITRNSL                                CPI : Transportation
## 97         CPIMEDSL                                  CPI : Medical Care
## 98      CUSR0000SAC                                   CPI : Commodities
## 99      CUSR0000SAD                                      CPI : Durables
## 100     CUSR0000SAS                                      CPI : Services
## 101        CPIULFSL                           CPI : All Items Less Food
## 102   CUSR0000SA0L2                        CPI : All items less shelter
## 103   CUSR0000SA0L5                   CPI : All items less medical care
## 104           PCEPI                 Personal Cons. Expend.: Chain Index
## 105 DDURRG3M086SBEA                   Personal Cons. Exp: Durable goods
## 106 DNDGRG3M086SBEA                 Personal Cons. Exp: Nondurable goods
## 107 DSERRG3M086SBEA                         Personal Cons. Exp: Services
## 108   CES0600000008              Avg Hourly Earnings : Goods-Producing
## 109   CES2000000008                 Avg Hourly Earnings : Construction
## 110   CES3000000008                Avg Hourly Earnings : Manufacturing
## 111     DTCOLNVHFNM           Consumer Motor Vehicle Loans Outstanding
## 112        DTCTHFNM       Total Consumer Loans and Leases Outstanding
## 113          INVEST Securities in Bank Credit at All Commercial Banks
##       gsi.description                                  group slow_1_fast_0
## 1                  PI                    Output and Income             1
## 2   PI less transfers                    Output and Income             1
## 3    Real Consumption Consumption, Orders, and Inventories             1
## 4           M&T sales Consumption, Orders, and Inventories             0
## 5        Retail sales Consumption, Orders, and Inventories             0
## 6           IP: total                    Output and Income             1
## 7        IP: products                    Output and Income             1
## 8        IP: nal prod                    Output and Income             1
## 9        IP: cons gds                    Output and Income             1
## 10      IP: cons dble                    Output and Income             1
## 11   IP: cons nondble                    Output and Income             1
## 12       IP: bus eqpt                    Output and Income             1
## 13          IP: matls                    Output and Income             1
## 14     IP: dble matls                    Output and Income             1
## 15  IP: nondble matls                    Output and Income             1
## 16            IP: mfg                    Output and Income             1
## 17       IP: res util                    Output and Income             1
## 18          IP: fuels                    Output and Income             1
```

```
## 19           Cap util                    Output and Income            1
## 20    Help wanted indx                         Labor Market            1
## 21   Help wanted/unemp                         Labor Market            1
## 22       Emp CPS total                         Labor Market            1
## 23       Emp CPS nonag                         Labor Market            1
## 24             U: all                          Labor Market            1
## 25    U: mean duration                         Labor Market            1
## 26           U < 5 wks                         Labor Market            1
## 27          U 5-14 wks                         Labor Market            1
## 28           U 15+ wks                         Labor Market            1
## 29         U 15-26 wks                         Labor Market            1
## 30           U 27+ wks                         Labor Market            1
## 31           UI claims                         Labor Market            1
## 32          Emp: total                         Labor Market            1
## 33       Emp: gds prod                         Labor Market            1
## 34         Emp: mining                         Labor Market            1
## 35          Emp: const                         Labor Market            1
## 36            Emp: mfg                         Labor Market            1
## 37       Emp: dble gds                         Labor Market            1
## 38       Emp: nondbles                         Labor Market            1
## 39       Emp: services                         Labor Market            1
## 40            Emp: TTU                         Labor Market            1
## 41      Emp: wholesale                         Labor Market            1
## 42         Emp: retail                         Labor Market            1
## 43           Emp: FIRE                         Labor Market            1
## 44           Emp: Govt                         Labor Market            1
## 45             Avg hrs                         Labor Market            1
## 46       Overtime: mfg                         Labor Market            1
## 47        Avg hrs: mfg                         Labor Market            1
## 48     Starts: nonfarm                              Housing            0
## 49        Starts: West                              Housing            0
## 50           BP: total                              Housing            0
## 51              BP: NE                              Housing            0
## 52           BP: South                              Housing            0
## 53            BP: West                              Housing            0
## 54   Orders: dble gds Consumption, Orders, and Inventories            0
## 55   Unf orders: dble Consumption, Orders, and Inventories            0
## 56         M&T invent Consumption, Orders, and Inventories            0
## 57   M&T invent/sales Consumption, Orders, and Inventories            0
## 58                 M1                     Money and Credit            0
## 59                 M2                     Money and Credit            0
## 60          M2 (real)                     Money and Credit            0
## 61       Reserves tot                     Money and Credit            0
## 62    Reserves nonbor                     Money and Credit            0
## 63       C&I loan plus                     Money and Credit            0
## 64          DC&I loans                     Money and Credit            0
## 65         Cons credit                     Money and Credit            0
## 66         Inst cred/PI                     Money and Credit            0
## 67            Fed Funds          Interest and Exchange Rates            0
## 68          Comm paper          Interest and Exchange Rates            0
## 69         3 mo T-bill          Interest and Exchange Rates            0
## 70         6 mo T-bill          Interest and Exchange Rates            0
## 71         1 yr T-bond          Interest and Exchange Rates            0
## 72         5 yr T-bond          Interest and Exchange Rates            0
```

```
## 73       10 yr T-bond          Interest and Exchange Rates            0
## 74           Aaa bond          Interest and Exchange Rates            0
## 75           Baa bond          Interest and Exchange Rates            0
## 76       CP-FF spread          Interest and Exchange Rates            0
## 77     3 mo-FF spread          Interest and Exchange Rates            0
## 78     6 mo-FF spread          Interest and Exchange Rates            0
## 79     1 yr-FF spread          Interest and Exchange Rates            0
## 80     5 yr-FF spread          Interest and Exchange Rates            0
## 81    10 yr-FF spread          Interest and Exchange Rates            0
## 82      Aaa-FF spread          Interest and Exchange Rates            0
## 83      Baa-FF spread          Interest and Exchange Rates            0
## 84     Ex rate: Switz          Interest and Exchange Rates            0
## 85     Ex rate: Japan          Interest and Exchange Rates            0
## 86        Ex rate: UK          Interest and Exchange Rates            0
## 87    EX rate: Canada          Interest and Exchange Rates            0
## 88         PPI: n gds                               Prices            1
## 89      PPI: cons gds                               Prices            1
## 90     PPI: int matls                               Prices            1
## 91   PPI: crude matls                               Prices            1
## 92  Spot market price                               Prices            1
## 93    PPI: nonferrous                               Prices            1
## 94         CPI-U: all                               Prices            1
## 95     CPI-U: apparel                               Prices            1
## 96      CPI-U: transp                               Prices            1
## 97     CPI-U: medical                               Prices            1
## 98       CPI-U: comm.                               Prices            1
## 99       CPI-U: dbles                               Prices            1
## 100   CPI-U: services                               Prices            1
## 101    CPI-U: ex food                               Prices            1
## 102 CPI-U: ex shelter                               Prices            1
## 103     CPI-U: ex med                               Prices            1
## 104            PCE de                               Prices            1
## 105    PCE de : dlbes                               Prices            1
## 106  PCE de : nondble                               Prices            1
## 107  PCE de : service                               Prices            1
## 108        AHE: goods                          Labor Market            1
## 109        AHE: const                          Labor Market            1
## 110          AHE: mfg                          Labor Market            1
## 111              <NA>                    Money and Credit            0
## 112              <NA>                    Money and Credit            0
## 113              <NA>                    Money and Credit            0
##     drop_aggregate_1  X X.1 X.2
## 1                  0 NA  NA  NA
## 2                  0 NA  NA  NA
## 3                  0 NA  NA  NA
## 4                  0 NA  NA  NA
## 5                  0 NA  NA  NA
## 6                  1 NA  NA  NA
## 7                  0 NA  NA  NA
## 8                  0 NA  NA  NA
## 9                  0 NA  NA  NA
## 10                 0 NA  NA  NA
## 11                 0 NA  NA  NA
## 12                 0 NA  NA  NA
```

```
## 13                0 NA  NA  NA
## 14                0 NA  NA  NA
## 15                0 NA  NA  NA
## 16                0 NA  NA  NA
## 17                0 NA  NA  NA
## 18                0 NA  NA  NA
## 19                0 NA  NA  NA
## 20                0 NA  NA  NA
## 21                0 NA  NA  NA
## 22                0 NA  NA  NA
## 23                0 NA  NA  NA
## 24                0 NA  NA  NA
## 25                0 NA  NA  NA
## 26                0 NA  NA  NA
## 27                0 NA  NA  NA
## 28                0 NA  NA  NA
## 29                0 NA  NA  NA
## 30                0 NA  NA  NA
## 31                0 NA  NA  NA
## 32                1 NA  NA  NA
## 33                0 NA  NA  NA
## 34                0 NA  NA  NA
## 35                0 NA  NA  NA
## 36                0 NA  NA  NA
## 37                0 NA  NA  NA
## 38                0 NA  NA  NA
## 39                0 NA  NA  NA
## 40                0 NA  NA  NA
## 41                0 NA  NA  NA
## 42                0 NA  NA  NA
## 43                0 NA  NA  NA
## 44                0 NA  NA  NA
## 45                0 NA  NA  NA
## 46                0 NA  NA  NA
## 47                0 NA  NA  NA
## 48                1 NA  NA  NA
## 49                0 NA  NA  NA
## 50                0 NA  NA  NA
## 51                0 NA  NA  NA
## 52                0 NA  NA  NA
## 53                0 NA  NA  NA
## 54                0 NA  NA  NA
## 55                0 NA  NA  NA
## 56                0 NA  NA  NA
## 57                0 NA  NA  NA
## 58                0 NA  NA  NA
## 59                0 NA  NA  NA
## 60                0 NA  NA  NA
## 61                0 NA  NA  NA
## 62                0 NA  NA  NA
## 63                0 NA  NA  NA
## 64                0 NA  NA  NA
## 65                0 NA  NA  NA
## 66                0 NA  NA  NA
```

```
## 67                0 NA  NA  NA
## 68                0 NA  NA  NA
## 69                0 NA  NA  NA
## 70                0 NA  NA  NA
## 71                0 NA  NA  NA
## 72                0 NA  NA  NA
## 73                0 NA  NA  NA
## 74                0 NA  NA  NA
## 75                0 NA  NA  NA
## 76                0 NA  NA  NA
## 77                0 NA  NA  NA
## 78                0 NA  NA  NA
## 79                0 NA  NA  NA
## 80                0 NA  NA  NA
## 81                0 NA  NA  NA
## 82                0 NA  NA  NA
## 83                0 NA  NA  NA
## 84                0 NA  NA  NA
## 85                0 NA  NA  NA
## 86                0 NA  NA  NA
## 87                0 NA  NA  NA
## 88                0 NA  NA  NA
## 89                0 NA  NA  NA
## 90                0 NA  NA  NA
## 91                0 NA  NA  NA
## 92                0 NA  NA  NA
## 93                0 NA  NA  NA
## 94                1 NA  NA  NA
## 95                0 NA  NA  NA
## 96                0 NA  NA  NA
## 97                0 NA  NA  NA
## 98                0 NA  NA  NA
## 99                0 NA  NA  NA
## 100               0 NA  NA  NA
## 101               0 NA  NA  NA
## 102               0 NA  NA  NA
## 103               0 NA  NA  NA
## 104               0 NA  NA  NA
## 105               0 NA  NA  NA
## 106               0 NA  NA  NA
## 107               0 NA  NA  NA
## 108               0 NA  NA  NA
## 109               0 NA  NA  NA
## 110               0 NA  NA  NA
## 111               0 NA  NA  NA
## 112               0 NA  NA  NA
## 113               0 NA  NA  NA
```

```
names(data_s)
```

```
##   [1] "RPI"            "W875RX1"        "DPCERA3M086SBEA" "INDPRO"
##   [5] "IPFPNSS"        "IPFINAL"        "IPCONGD"         "IPDCONGD"
##   [9] "IPNCONGD"       "IPBUSEQ"        "IPMAT"           "IPDMAT"
##  [13] "IPNMAT"         "IPMANSICS"      "IPB51222s"       "IPFUELS"
##  [17] "CUMFNS"         "HWI"            "HWIURATIO"       "CLF16OV"
```

```
##  [21] "CE16OV"           "UNRATE"           "UEMPMEAN"         "UEMPLT5"
##  [25] "UEMP5TO14"        "UEMP15OV"         "UEMP15T26"        "UEMP27OV"
##  [29] "CLAIMSx"          "PAYEMS"           "USGOOD"           "CES1021000001"
##  [33] "USCONS"           "MANEMP"           "DMANEMP"          "NDMANEMP"
##  [37] "SRVPRD"           "USTPU"            "USWTRADE"         "USTRADE"
##  [41] "USFIRE"           "USGOVT"           "CES0600000007"    "AWOTMAN"
##  [45] "AWHMAN"           "S.P.500"          "S.P..indust"      "S.P.div.yield"
##  [49] "S.P.PE.ratio"     "WPSFD49207"       "WPSFD49502"       "WPSID61"
##  [53] "WPSID62"          "OILPRICEx"        "PPICMM"           "CPIAUCSL"
##  [57] "CPIAPPSL"         "CPITRNSL"         "CPIMEDSL"         "CUSR0000SAC"
##  [61] "CUSR0000SAD"      "CUSR0000SAS"      "CPIULFSL"         "CUSR0000SA0L2"
##  [65] "CUSR0000SA0L5"    "PCEPI"            "DDURRG3M086SBEA"  "DNDGRG3M086SBEA"
##  [69] "DSERRG3M086SBEA"  "CES0600000008"    "CES2000000008"    "CES3000000008"
##  [73] "CMRMTSPLx"        "RETAILx"          "HOUST"            "HOUSTW"
##  [77] "PERMIT"           "PERMITNE"         "PERMITS"          "PERMITW"
##  [81] "AMDMNOx"          "AMDMUOx"          "BUSINVx"          "ISRATIOx"
##  [85] "M1SL"             "M2SL"             "M2REAL"           "TOTRESNS"
##  [89] "NONBORRES"        "BUSLOANS"         "REALLN"           "NONREVSL"
##  [93] "CONSPI"           "S.P.500.1"        "S.P..indust.1"    "S.P.div.yield.1"
##  [97] "S.P.PE.ratio.1"   "CP3Mx"            "TB3MS"            "TB6MS"
## [101] "GS1"              "GS5"              "GS10"             "AAA"
## [105] "BAA"              "COMPAPFFx"        "TB3SMFFM"         "TB6SMFFM"
## [109] "T1YFFM"           "T5YFFM"           "T10YFFM"          "AAAFFM"
## [113] "BAAFFM"           "EXSZUSx"          "EXJPUSx"          "EXUSUKx"
## [117] "EXCAUSx"          "DTCOLNVHFNM"      "DTCTHFNM"         "INVEST"
## [121] "FEDFUNDS"
```

```
variables = c(grep("^FEDFUNDS$", colnames(data_s)),#Fed Funds Rate
              grep("^INDPRO$", colnames(data_s)), #IP Index
              grep("^CPIAUCSL$", colnames(data_s)), #CPI : All Items
            grep("^TB3MS$", colnames(data_s)), #3-Month Treasury Bill:
            grep("^GS5$", colnames(data_s)), #5-Year Treasury Rate
            grep("^M1SL$", colnames(data_s)),# M1 Money Stock
            grep("^M2SL$", colnames(data_s)), #M2 Money Stock
            grep("^EXJPUSx$", colnames(data_s)), #Japan / U.S. Foreign Exchange Rate
            grep("^CUSR0000SAC$", colnames(data_s)),#CPI : Commodities
            grep("^CUMFNS$", colnames(data_s)),#Capacity Utilization: Manufacturing
            grep("^DPCERA3M086SBEA$", colnames(data_s)),#Real personal consumption expenditures
            grep("^DDURRG3M086SBEA$", colnames(data_s)), #Personal Cons. Exp: Durable goods
            grep("^DNDGRG3M086SBEA$", colnames(data_s)), #Personal Cons. Exp: Nondurable goods
            grep("^UNRATE$", colnames(data_s)), #Civilian Unemployment Rate
            grep("^CE16OV$", colnames(data_s)), #Civilian Employment
            grep("^CES0600000008$", colnames(data_s)), #Avg Hourly Earnings : Goods-Producing
            grep("^HOUST$", colnames(data_s)),  #Housing Starts: Total New Privately Owned
            grep("^AMDMNOx$", colnames(data_s)), #New Orders for Durable Goods
            grep("^S.P.div.yield$", colnames(data_s))#S&P s Composite Common Stock: Dividend Yield
             )
length(variables)
```

```
## [1] 19
```

```
variable_names = c("Fed Funds Rate",
"IP Index",
"CPI",
"3-Month Treasury Bill",
```

```
"5-Year Treasury Rate",
"M1 Money Stock",
"M2 Money Stock",
"Exchange rate: Japan/U.S.",
"CPI : Commodities",
"Capacity Utilization",
"Real personal consump.",
"Durable goods consump.",
"Nondurable goods consump.",
"Unemployment Rate",
"Employment",
"Avg Hourly Earnings",
"Housing Starts",
"New Orders",
"Dividend Yield")
length(variable_names)
```

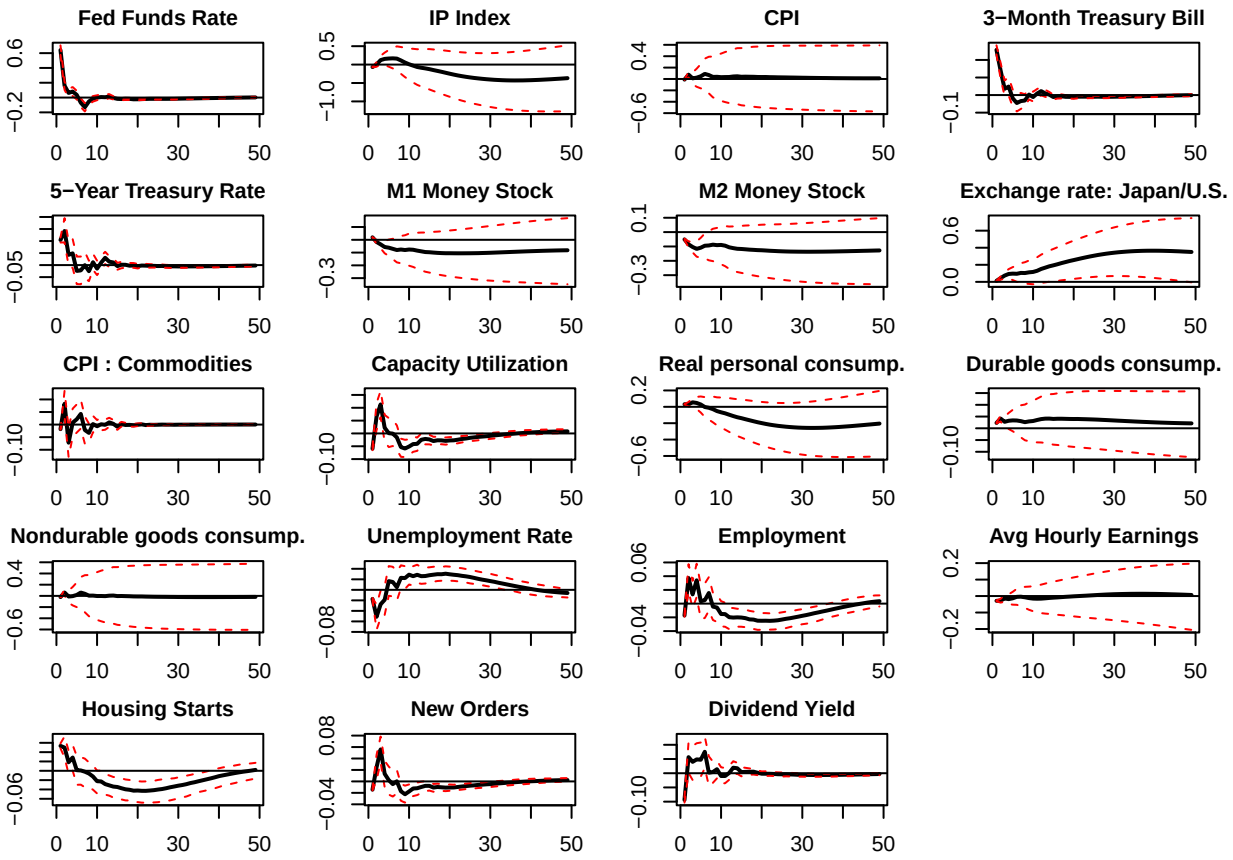
```
## [1] 19
```

```
transf_code = c(1,5,5,1,1,5,5,5,1,1,5,5,5,1,1,5,1,1,1)
length(transf_code)
```

```
## [1] 19
```

```
options(repr.plot.width=12, repr.plot.height=8)

par(mfrow=c(5,4),
   mar = c(2, 2, 2, 2))
for(i in variables){
    index = which(variables == i)
    if(transf_code[index] == 5){
        plot(cumsum(IRF[,i]), type ='l',lwd=2, main = variable_names[index],
             ylab= "", xlab="Steps", ylim=range(cumsum(Lower[,i]),cumsum(Upper[,i])),
             cex.main=1, cex.axis=1)
        lines(cumsum(Upper[,i]), lty=2, col="red")
        lines(cumsum(Lower[,i]), lty=2, col="red")
        abline(h=0)
    }
    else{
        plot(IRF[,i], type ='l',lwd=2, main = variable_names[index],
             ylab= "", xlab="Steps", ylim=range((Lower[,i]),(Upper[,i])),
             cex.main=1, cex.axis=1)
        lines((Upper[,i]), lty=2, col="red")
        lines((Lower[,i]), lty=2, col="red")
        abline(h=0)
    }
}
```



Regresión de cada factor sobre cada serie

```r
# Initialize an empty dataframe to store the results
results_df <- data.frame(Variable = character(0), Factor = character(0), R2 = numeric(0))

# Iterate through each factor in F_hat
for (factor_name in colnames(F_hat)) {
  # Extract the current factor
  current_factor <- F_hat[, factor_name]

  # Iterate through each column in data_s
  for (column_name in colnames(data_s)) {
    # Extract the current column
    current_column <- data_s[, column_name]

    # Perform linear regression
    regression_model <- lm(current_column ~ current_factor)

    # Calculate R-squared
    r_squared <- summary(regression_model)$r.squared

    # Add the results to the results dataframe
    results_df <- rbind(results_df, data.frame(Variable = column_name, Factor = factor_name, R2 = r_squa
  }
}
```

```r
results_df$R2 <- format(results_df$R2, scientific = FALSE)
results_df <- results_df[order(-as.numeric(results_df$R2)), ]

# Print the sorted results dataframe
print(head(results_df,20))
```

```
##            Variable Factor              R2
## 302     CUSR0000SAC    PC3 0.7851122457927
## 310 DNDGRG3M086SBEA    PC3 0.7752439243947
## 14        IPMANSICS    PC1 0.7651868584029
## 306   CUSR0000SA0L2    PC3 0.7475006892248
## 4             INDPRO    PC1 0.7473794536251
## 298        CPIAUCSL    PC3 0.7353152585852
## 31           USGOOD    PC1 0.7308305028622
## 17           CUMFNS    PC1 0.7194671091465
## 307   CUSR0000SA0L5    PC3 0.6841655116794
## 34            MANEMP    PC1 0.6721827240774
## 170    S.P.PE.ratio    PC2 0.6644681151674
## 218  S.P.PE.ratio.1    PC2 0.6644681151674
## 5            IPFPNSS    PC1 0.6630864490378
## 30            PAYEMS    PC1 0.6579379405403
## 169   S.P.div.yield    PC2 0.6514120447114
## 217 S.P.div.yield.1    PC2 0.6514120447114
## 35           DMANEMP    PC1 0.6242410032609
## 308            PCEPI    PC3 0.6206402467961
## 167          S.P.500    PC2 0.5951692531138
## 215        S.P.500.1    PC2 0.5951692531138
```

```r
average_r_squared_by_factor <- aggregate(R2 ~ Factor, data = results_df, FUN = function(x) mean(as.numer

# Print the average R-squared values by factor
print(average_r_squared_by_factor)
```

```
##   Factor         R2
## 1    PC1 0.14423198
## 2    PC2 0.08318371
## 3    PC3 0.06376903
## 4    PC4 0.05110978
## 5    PC5 0.04248469
## 6    PC6 0.04180347
## 7    PC7 0.02956900
```

Step 5: FEVD

```r
# Get the VAR point estimates
hor=60
irf_points = irf(var, n.ahead = hor,  boot = FALSE)
```

```r
# Get IRFs for all of X we are interested in, Dimensions: (hor, key_nvars)
# Find loadings
results = summary(reg_loadings) # the warning comes because of FEDFUNDS
key_nvars = length(variables)
key_nvars
```

```
## [1] 19
```

```
irf_X_pc1 = array(c(0,0), dim=c(hor+1, key_nvars))
irf_X_pc2 = array(c(0,0), dim=c(hor+1, key_nvars))
irf_X_pc3 = array(c(0,0), dim=c(hor+1, key_nvars))
irf_X_pc4 = array(c(0,0), dim=c(hor+1, key_nvars))
irf_X_pc5 = array(c(0,0), dim=c(hor+1, key_nvars))
irf_X_pc6 = array(c(0,0), dim=c(hor+1, key_nvars))
irf_X_pc7 = array(c(0,0), dim=c(hor+1, key_nvars))
irf_X_ffr = array(c(0,0), dim=c(hor+1, key_nvars))
for(i in 1:key_nvars){
    irf_X_pc1[,i] = irf_points$irf$PC1 %*% matrix(loadings[1:nrow(loadings), variables[i]])
    irf_X_pc2[,i] = irf_points$irf$PC2 %*% matrix(loadings[1:nrow(loadings), variables[i]])
    irf_X_pc3[,i] = irf_points$irf$PC3 %*% matrix(loadings[1:nrow(loadings), variables[i]])
    irf_X_pc4[,i] = irf_points$irf$PC3 %*% matrix(loadings[1:nrow(loadings), variables[i]])
    irf_X_pc5[,i] = irf_points$irf$PC3 %*% matrix(loadings[1:nrow(loadings), variables[i]])
    irf_X_pc6[,i] = irf_points$irf$PC3 %*% matrix(loadings[1:nrow(loadings), variables[i]])
    irf_X_pc7[,i] = irf_points$irf$PC3 %*% matrix(loadings[1:nrow(loadings), variables[i]])
    irf_X_ffr[,i] = (irf_points$irf$FEDFUNDS) %*% matrix(loadings[1:nrow(loadings), variables[i]])
    }
```

```
# Get the IRFs squared and accumulate them
psi2_pc1 = array(0, dim = key_nvars)
psi2_pc2 = array(0, dim = key_nvars)
psi2_pc3 = array(0, dim = key_nvars)
psi2_pc4 = array(0, dim = key_nvars)
psi2_pc5 = array(0, dim = key_nvars)
psi2_pc6 = array(0, dim = key_nvars)
psi2_pc7 = array(0, dim = key_nvars)
psi2_ffr = array(0, dim = key_nvars)

for(i in 1:key_nvars){
    for(j in 1:hor){
        psi2_pc1[i] = psi2_pc1[i] + irf_X_pc1[j,i]^2
        psi2_pc2[i] = psi2_pc2[i] + irf_X_pc2[j,i]^2
        psi2_pc3[i] = psi2_pc3[i] + irf_X_pc3[j,i]^2
        psi2_pc4[i] = psi2_pc4[i] + irf_X_pc4[j,i]^2
        psi2_pc5[i] = psi2_pc5[i] + irf_X_pc5[j,i]^2
        psi2_pc6[i] = psi2_pc6[i] + irf_X_pc6[j,i]^2
        psi2_pc7[i] = psi2_pc7[i] + irf_X_pc7[j,i]^2
        psi2_ffr[i] = psi2_ffr[i] + irf_X_ffr[j,i]^2
    }
}
```

```
var_total= array(0, dim = key_nvars)
var_fac= array(0, dim = key_nvars)
var_e= array(0, dim = key_nvars)

for(i in 1:key_nvars){
    var_fac[i] = psi2_pc1[i] + psi2_pc2[i] + psi2_pc3[i] + psi2_pc4[i]+psi2_pc5[i]+psi2_pc6[i]+psi2_pc7
    var_total[i] = psi2_pc1[i] + psi2_pc2[i] + psi2_pc3[i] +psi2_pc4[i]+psi2_pc5[i]+psi2_pc6[i]+psi2_pc
    var_e[i] = results[[variables[i]]]$sigma^2
}
```

```
table = data.frame("PC1" = round((psi2_pc1),3),
                   "PC2" = round((psi2_pc2),3),
```

```
                    "PC3" = round((psi2_pc3),3),
                    "PC4" = round((psi2_pc4),3),
                    "PC5" = round((psi2_pc5),3),
                    "PC6" = round((psi2_pc6),3),
                    "PC7" = round((psi2_pc7),3),
                    "FEDFUNDS" = round((psi2_ffr),3),
                   "Factor_Y_total" = round(var_fac,3) ,
                   "e" = round((var_e),3),
                   "Total" = round(var_total,3))
row.names(table) = variable_names
table
```

```
##                              PC1   PC2   PC3   PC4   PC5   PC6   PC7 FEDFUNDS
## Fed Funds Rate             0.076 0.125 0.014 0.014 0.014 0.014 0.014    0.565
## IP Index                   0.613 0.113 0.003 0.003 0.003 0.003 0.003    0.054
## CPI                        0.010 0.048 0.723 0.723 0.723 0.723 0.723    0.020
## 3-Month Treasury Bill      0.087 0.093 0.010 0.010 0.010 0.010 0.010    0.114
## 5-Year Treasury Rate       0.129 0.156 0.019 0.019 0.019 0.019 0.019    0.046
## M1 Money Stock             0.024 0.023 0.001 0.001 0.001 0.001 0.001    0.003
## M2 Money Stock             0.010 0.006 0.007 0.007 0.007 0.007 0.007    0.006
## Exchange rate: Japan/U.S.  0.020 0.014 0.009 0.009 0.009 0.009 0.009    0.007
## CPI : Commodities          0.007 0.047 0.773 0.773 0.773 0.773 0.773    0.020
## Capacity Utilization       0.571 0.112 0.003 0.003 0.003 0.003 0.003    0.054
## Real personal consump.     0.056 0.083 0.002 0.002 0.002 0.002 0.002    0.008
## Durable goods consump.     0.001 0.003 0.024 0.024 0.024 0.024 0.024    0.001
## Nondurable goods consump.  0.008 0.048 0.763 0.763 0.763 0.763 0.763    0.020
## Unemployment Rate          0.148 0.078 0.001 0.001 0.001 0.001 0.001    0.028
## Employment                 0.096 0.066 0.001 0.001 0.001 0.001 0.001    0.019
## Avg Hourly Earnings        0.009 0.001 0.001 0.001 0.001 0.001 0.001    0.001
## Housing Starts             0.163 0.138 0.004 0.004 0.004 0.004 0.004    0.056
## New Orders                 0.100 0.031 0.003 0.003 0.003 0.003 0.003    0.009
## Dividend Yield             0.047 0.733 0.027 0.027 0.027 0.027 0.027    0.029
##                            Factor_Y_total     e Total
## Fed Funds Rate                      0.837 0.000 0.837
## IP Index                            0.796 0.080 0.876
## CPI                                 3.692 0.243 3.935
## 3-Month Treasury Bill               0.342 0.358 0.700
## 5-Year Treasury Rate                0.428 0.164 0.591
## M1 Money Stock                      0.057 0.936 0.993
## M2 Money Stock                      0.057 0.938 0.995
## Exchange rate: Japan/U.S.           0.087 0.882 0.968
## CPI : Commodities                   3.940 0.198 4.138
## Capacity Utilization                0.752 0.100 0.852
## Real personal consump.              0.159 0.837 0.995
## Durable goods consump.              0.125 0.978 1.103
## Nondurable goods consump.           3.892 0.206 4.099
## Unemployment Rate                   0.260 0.632 0.892
## Employment                          0.185 0.696 0.881
## Avg Hourly Earnings                 0.018 0.972 0.990
## Housing Starts                      0.375 0.141 0.516
## New Orders                          0.156 0.845 1.001
## Dividend Yield                      0.943 0.091 1.034
```

```
r2 = array(0, dim = key_nvars)
for(i in 1:key_nvars){
    r2[i] = results[[variables[i]]]$r.squared
}

table2 = data.frame("Variables" = variable_names, "Contribution" = round((psi2_ffr/var_total),3), "R-squ
table2
```

```
##                    Variables Contribution R.squared
## 1             Fed Funds Rate        0.675     1.000
## 2                   IP Index        0.061     0.921
## 3                        CPI        0.005     0.760
## 4      3-Month Treasury Bill        0.163     0.646
## 5       5-Year Treasury Rate        0.078     0.838
## 6             M1 Money Stock        0.003     0.075
## 7             M2 Money Stock        0.006     0.073
## 8  Exchange rate: Japan/U.S.        0.007     0.129
## 9          CPI : Commodities        0.005     0.805
## 10      Capacity Utilization        0.063     0.901
## 11    Real personal consump.        0.008     0.173
## 12    Durable goods consump.        0.001     0.034
## 13 Nondurable goods consump.        0.005     0.796
## 14         Unemployment Rate        0.032     0.376
## 15                Employment        0.022     0.313
## 16       Avg Hourly Earnings        0.002     0.040
## 17            Housing Starts        0.108     0.861
## 18                New Orders        0.009     0.165
## 19            Dividend Yield        0.029     0.910
```

Predicciones de los factores y FFR

Model Performance

```
# Convert the date_column to Date type
df_test$index <- as.Date(df_test$index)
test_set=df_test[,2:ncol(df_test)]
actual_df=scale(test_set, center = TRUE, scale = TRUE)
# Create an xts object with the date_column as the index
xts_test <- xts(df_test[, -which(colnames(df_test) == "index")], order.by = df_test$index)
cat("Rango de datos:", as.character(range(index(xts_test))), "\n")
```

```
## Rango de datos: 2008-02-01 2020-02-01
```

```
data_test_s = scale(xts_test, center = TRUE, scale = TRUE)
cat("Tamaño de mi muestra de prueba:", dim(df_test), "\n")
```

```
## Tamaño de mi muestra de prueba: 145 122
```

```
predicciones=predict(var, n.ahead = dim(df_test)[1])
pred_F=predicciones[,1:7]
pred_FFR=predicciones[,ncol(predicciones)]
dim(pred_F)
```

```
## [1] 145   7
```

```r
length(pred_FFR)
```

```
## [1] 145
```

```r
F_part=pred_F%*%Lamda_F
dim(F_part)
```

```
## [1] 145 121
```

```r
Y_part=outer(pred_FFR, Lambda_ffr)
dim(Y_part)
```

```
## [1] 145 121
```

```r
X_pred=F_part+Y_part
X_forec=as.data.frame(X_pred)
dim(X_pred)
```

```
## [1] 145 121
```

```r
X_predictions <- as.data.frame(X_pred)
dim(X_predictions)
```

```
## [1] 145 121
```

```r
predictions_df <- as.data.frame(X_pred)
dim(predictions_df)
```

```
## [1] 145 121
```

```r
dim(actual_df)
```

```
## [1] 145 121
```

```r
sum(is.na(actual_df))
```

```
## [1] 0
```

```r
sum(is.na(predictions_df))
```

```
## [1] 0
```

```r
compute_accuracy_measures_df <- function(actual_df, predictions_df) {
  if(ncol(actual_df) != ncol(predictions_df) || nrow(actual_df) != nrow(predictions_df)) {
    stop("Dimensions of actual and predicted data must match.")
  }

  measures_list <- list()

  for(i in 1:ncol(actual_df)) {
    actual <- actual_df[, i]
    predicted <- predictions_df[, i]

    # MAE
    mae <- mean(abs(actual - predicted), na.rm = TRUE)

    # MSE
    mse <- mean((actual - predicted)^2, na.rm = TRUE)

    # MAPE
```

```
    mape <- mean(abs((actual - predicted) / actual) * 100, na.rm = TRUE)

    # RMSFE (Root Mean Squared Forecast Error)
    rmsfe <- sqrt(mse)

    # Store in list
    measures_list[[colnames(actual_df)[i]]] <- c(MAE = mae, MSE = mse, MAPE = mape, RMSFE = rmsfe)
  }

  # Convert the list to a dataframe
  results_df <- do.call(rbind, measures_list)
  return(results_df)
}

# Usage:
results_df <- compute_accuracy_measures_df(actual_df, predictions_df)

# View the results:
print(results_df)
```

```
##                       MAE       MSE      MAPE     RMSFE
## RPI             0.5618468 1.0004828  98.02350 1.0002414
## W875RX1         0.6642520 0.9609458 156.97945 0.9802784
## DPCERA3M086SBEA 0.7283913 0.9896897 104.38534 0.9948315
## INDPRO          0.6535803 0.9356030  96.56096 0.9672658
## IPFPNSS         0.7134571 0.9498812 513.33549 0.9746185
## IPFINAL         0.7371293 0.9702203 111.11108 0.9849976
## IPCONGD         0.7508497 1.0007265 108.16845 1.0003632
## IPDCONGD        0.7034448 0.9952014 105.67586 0.9975978
## IPNCONGD        0.7755643 0.9986313 118.28217 0.9993154
## IPBUSEQ         0.6580432 0.9182772 180.82942 0.9582678
## IPMAT           0.6185695 0.9438401 155.28916 0.9715143
## IPDMAT          0.6355494 0.9263535 108.66071 0.9624726
## IPNMAT          0.5846706 0.9500400 112.73065 0.9747000
## IPMANSICS       0.6206304 0.9121185 108.53528 0.9550489
## IPB51222s       0.7490610 0.9954915 102.12360 0.9977432
## IPFUELS         0.5945564 0.9922318 110.17600 0.9961083
## CUMFNS          0.6397009 0.9081204 106.68085 0.9529535
## HWI             0.7769355 0.9676485 109.18395 0.9836913
## HWIURATIO       0.7448308 0.9540082 121.16400 0.9767335
## CLF16OV         0.7958246 0.9966642 101.94263 0.9983307
## CE16OV          0.6975852 0.9159273 152.09592 0.9570409
## UNRATE          0.7389095 0.9276468 121.24935 0.9631442
## UEMPMEAN        0.7906770 0.9944448 102.98289 0.9972185
## UEMPLT5         0.7794981 0.9911097 100.23660 0.9955449
## UEMP5TO14       0.7505417 0.9767271 104.35726 0.9882950
## UEMP15OV        0.7243066 0.8898898 114.16768 0.9433397
## UEMP15T26       0.8063647 0.9584598 110.94469 0.9790096
## UEMP27OV        0.7172427 0.9134470 119.08047 0.9557442
## CLAIMSx         0.7107376 0.9657381 122.45505 0.9827198
## PAYEMS          0.6398495 0.8229099 107.60152 0.9071438
## USGOOD          0.6103140 0.8503756 119.78109 0.9221581
## CES1021000001   0.7862107 0.9670563 108.14013 0.9833902
## USCONS          0.6961875 0.9017496 108.66138 0.9496050
```

```
## MANEMP          0.5737396 0.8672230 112.10506 0.9312481
## DMANEMP         0.5698727 0.8667288 206.25692 0.9309827
## NDMANEMP        0.6586263 0.9247397 103.52224 0.9616339
## SRVPRD          0.6603295 0.8384021 133.44142 0.9156430
## USTPU           0.6385707 0.7818389 103.20207 0.8842165
## USWTRADE        0.6133453 0.7507438 111.49063 0.8664547
## USTRADE         0.7076643 0.8468747 109.09245 0.9202580
## USFIRE          0.6827171 0.8545981 102.36951 0.9244447
## USGOVT          0.5349203 1.0009624 487.43427 1.0004811
## CES0600000007   0.7328069 0.9549635 116.81466 0.9772223
## AWOTMAN         0.6843572 0.9830384 187.86572 0.9914830
## AWHMAN          0.7151563 0.9466323 160.88471 0.9729503
## S.P.500         0.6789562 1.0124350  97.19548 1.0061983
## S.P..indust     0.6921639 1.0076597 102.88473 1.0038225
## S.P.div.yield   0.6106931 1.0033801 139.08940 1.0016886
## S.P.PE.ratio    0.5807283 0.9533385 150.12285 0.9763906
## WPSFD49207      0.7459585 1.0018488 100.35930 1.0009240
## WPSFD49502      0.7433832 1.0026810 100.38459 1.0013396
## WPSID61         0.6874479 1.0072619 102.97461 1.0036244
## WPSID62         0.7534529 0.9963598 103.48901 0.9981782
## OILPRICEx       0.7707803 0.9962265 100.81154 0.9981115
## PPICMM          0.7719529 0.9969898 100.46279 0.9984938
## CPIAUCSL        0.6900578 1.0050928 104.37015 1.0025432
## CPIAPPSL        0.7701128 0.9915085 102.54656 0.9957452
## CPITRNSL        0.7031779 0.9998761 107.35668 0.9999381
## CPIMEDSL        0.7589266 0.9912921 101.27987 0.9956365
## CUSR0000SAC     0.7056888 1.0001457 113.64539 1.0000729
## CUSR0000SAD     0.6717970 0.9956913  99.53376 0.9978433
## CUSR0000SAS     0.7394475 0.9993883 100.32255 0.9996941
## CPIULFSL        0.6904599 1.0053194 100.60104 1.0026562
## CUSR0000SA0L2   0.6893045 1.0022537 109.56869 1.0011262
## CUSR0000SA0L5   0.6894633 1.0042186 112.34347 1.0021071
## PCEPI           0.7431406 1.0022619 101.50509 1.0011303
## DDURRG3M086SBEA 0.7739557 0.9951336  99.66004 0.9975638
## DNDGRG3M086SBEA 0.7085461 0.9982366 111.53742 0.9991179
## DSERRG3M086SBEA 0.7533352 0.9940354 103.10846 0.9970132
## CES0600000008   0.7566106 0.9933512 159.59242 0.9966701
## CES2000000008   0.6544714 0.9937314 220.84134 0.9968608
## CES3000000008   0.8115256 0.9928263 123.21243 0.9964067
## CMRMTSPLx       0.6986956 0.9361116 103.86839 0.9675286
## RETAILx         0.6690447 0.9660654 114.02829 0.9828863
## HOUST           0.9344915 1.1241834 148.74650 1.0602752
## HOUSTW          0.8882142 1.0848530 133.59885 1.0415628
## PERMIT          0.9198235 1.0914745 141.43140 1.0447366
## PERMITNE        0.8428695 1.1135936 116.19665 1.0552694
## PERMITS         0.8776308 1.0182284 154.03583 1.0090730
## PERMITW         0.8950875 1.0591890 261.37391 1.0291691
## AMDMNOx         0.6858900 0.9813545 122.16197 0.9906334
## AMDMUOx         0.6872679 0.9256854 140.37693 0.9621255
## BUSINVx         0.6478314 0.8174486 347.59419 0.9041287
## ISRATIOx        0.6762397 0.9826996 104.71562 0.9913120
## M1SL            0.7347551 0.9918470 101.06356 0.9959151
## M2SL            0.6767094 0.9928024 101.61691 0.9963947
## M2REAL          0.6711623 1.0182486 108.38954 1.0090831
```

```
## TOTRESNS        0.5807679 0.9936251 107.87569 0.9968075
## NONBORRES       0.6242403 0.9918978 107.08913 0.9959407
## BUSLOANS        0.6855052 0.9912161  99.82063 0.9955983
## REALLN          0.5656532 0.9934957 102.21174 0.9967426
## NONREVSL        0.4258184 0.9939795 123.38903 0.9969852
## CONSPI          0.5042471 1.0083480 143.84085 1.0041653
## S.P.500.1       0.6789562 1.0124350  97.19548 1.0061983
## S.P..indust.1   0.6921639 1.0076597 102.88473 1.0038225
## S.P.div.yield.1 0.6106931 1.0033801 139.08940 1.0016886
## S.P.PE.ratio.1  0.5807283 0.9533385 150.12285 0.9763906
## CP3Mx           0.4095981 0.9050329 119.38964 0.9513322
## TB3MS           0.4713963 0.8627254 124.84243 0.9288301
## TB6MS           0.5061056 0.8694881 156.16957 0.9324635
## GS1             0.5425923 0.8928955 123.78797 0.9449315
## GS5             0.7528440 0.9827239 102.59340 0.9913243
## GS10            0.7192136 0.9952342 107.66105 0.9976143
## AAA             0.6953148 1.0061838 135.85786 1.0030871
## BAA             0.6341287 0.9934204 396.57686 0.9967048
## COMPAPFFx       0.5859761 0.8852618 157.46378 0.9408835
## TB3SMFFM        0.4734592 0.9459452 137.38396 0.9725971
## TB6SMFFM        0.5156534 0.8888303 187.02056 0.9427780
## T1YFFM          0.5946082 0.8532528 117.31816 0.9237169
## T5YFFM          0.6861606 0.7644588 107.61625 0.8743334
## T10YFFM         0.6951674 0.7998536 192.78772 0.8943453
## AAAFFM          0.7162795 0.8186033 115.60422 0.9047670
## BAAFFM          0.6742173 0.7851883 142.90880 0.8861085
## EXSZUSx         0.7263824 0.9740320  99.51583 0.9869306
## EXJPUSx         0.7553119 0.9853552 101.61120 0.9926506
## EXUSUKx         0.7289075 0.9702972 178.97749 0.9850366
## EXCAUSx         0.7346908 0.9876587  99.87368 0.9938102
## DTCOLNVHFNM     0.4364153 0.9928699 114.73541 0.9964286
## DTCTHFNM        0.6760860 0.9911279  99.77615 0.9955541
## INVEST          0.5917606 0.9915638 101.25435 0.9957729
## FEDFUNDS        0.4325386 0.7971550 126.17372 0.8928354
```

```
averages <- colMeans(results_df, na.rm = TRUE)
# Print the averages:
print(averages)
```

```
##         MAE         MSE        MAPE       RMSFE
##   0.6816715   0.9577956 131.7726453   0.9780051
```