DATOS MACROECONOMICOS DE SUDÁFRICA

By: Ael-Dev

Diccionario de variables:

- GDP (PIB)
- INF (tasa de inflación)
- INT (tasa de interés)

Leer dataset

```
library(readr)

data <- read_csv('https://raw.githubusercontent.com/ecabestadistica/series-temporales-multivariantes/master/2.%20Casos%20de%20estudio%20en%20R/B.%20Datos%20macroecon%C3%B3micos%20de%20Sud%C3%A1frica/data_sa.csv',
                 col_types = cols(
                                  dates = col_double(),
                                  GDP = col_double(),
                                  U = col_double()
                                  )
                 )
```

```
## Warning: The following named parsers don't match the column names: dates, GDP,
## U
```

```
head(data)
```

Date <chr>	gdp <dbl>	inf <dbl>	int <dbl>
1981/02	982319	2.228504	2.073281
1981/03	996616	5.101777	2.643333
1981/04	1002834	3.599360	2.946432
1982/01	993543	2.493895	3.576928
1982/02	985253	4.101902	3.758169
1982/03	985001	2.797385	3.851869

6 rows

Se observa que los datos son trimestrales

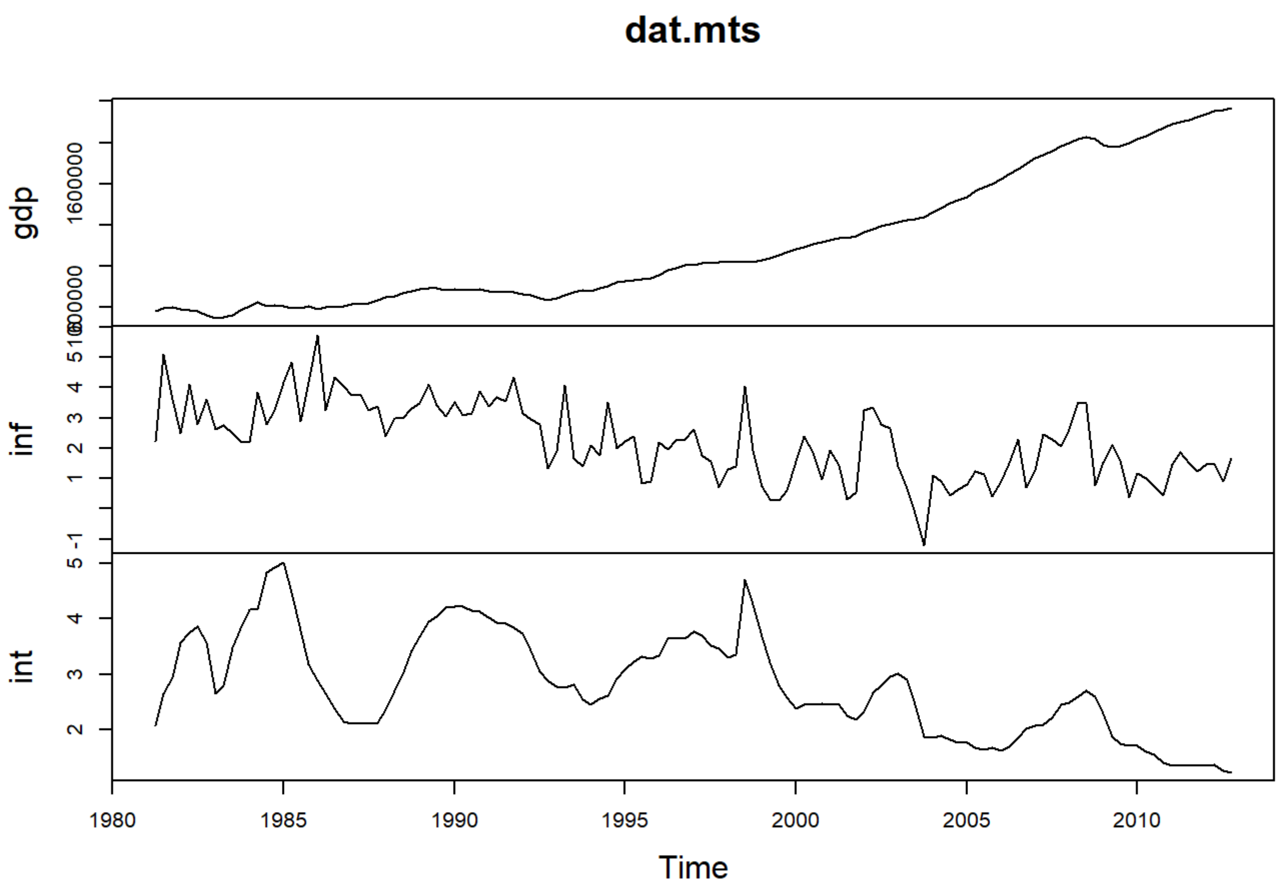
Análisis exploratorio de los datos (EDA)

```
library(tseries)
```

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

```
# Convertir las series a objeto ts
gdp <- ts(data$gdp, start = c(1981, 2), freq = 4) # Se establece el inicio en el año 1981, se
gundo trimestre, con una frecuencia de 4 (trimestral).
inf <- ts(data$inf, start = c(1981, 2), freq = 4)
int <- ts(data$int, start = c(1981, 2), freq = 4)
```

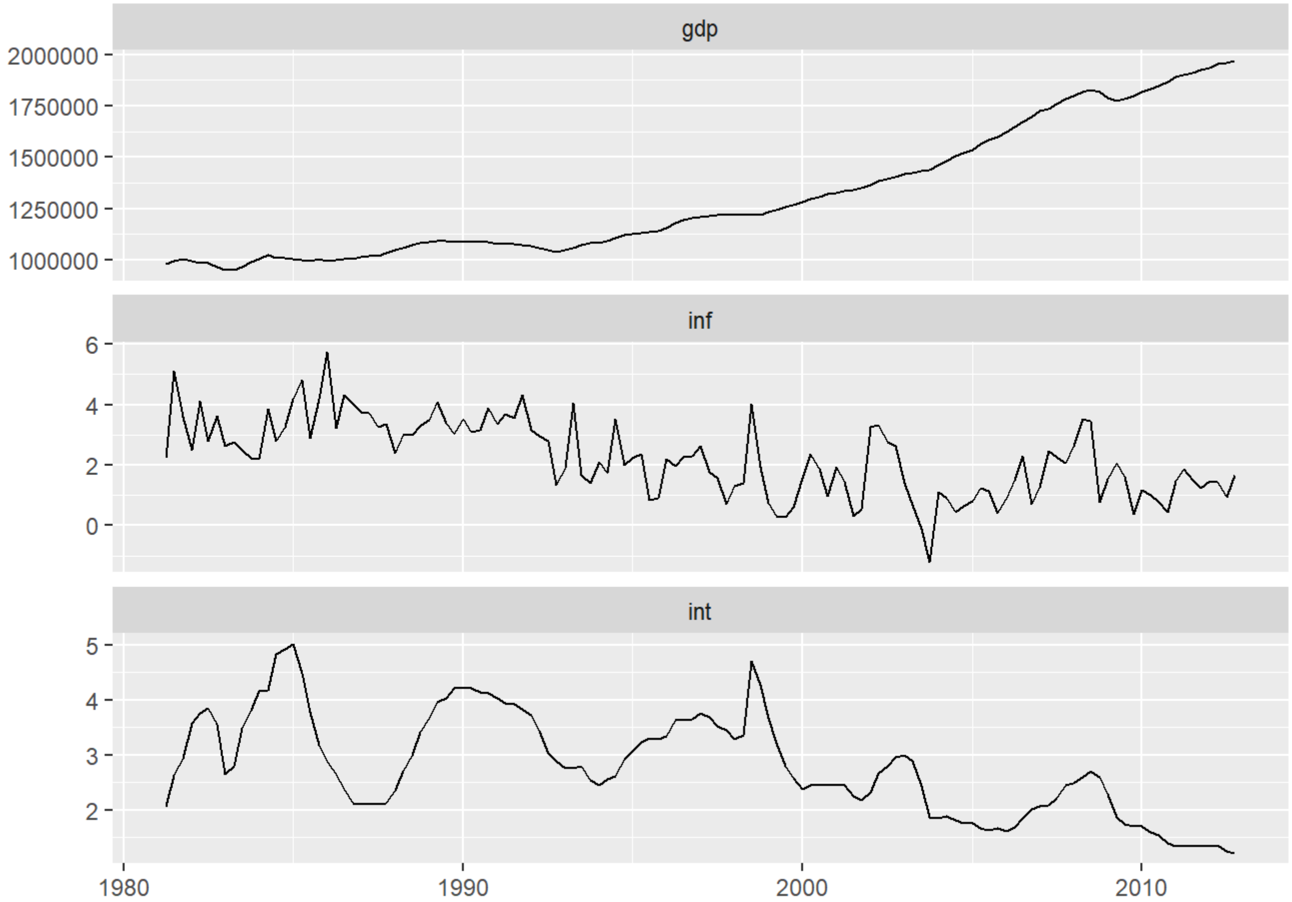
```
dat.mts=cbind(gdp, inf, int)  # Combinar series en matriz dat.mts

plot(dat.mts)  # Graficar las series temporales
```




```
# Con autoplot:
library(ggplot2)  # Cargar librería ggplot2
library(ggfortify)  # Cargar librería ggfortify
```

```
## Warning: package 'ggfortify' was built under R version 4.3.3
```

```
autoplot(dat.mts)  # Generar gráfico con ggplot2
```



Dividir la serie en conjunto de entrenamiento y prueba

```
library(dplyr)
```

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

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

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

```
n_obs=10 # nro de obs
end=dim(dat.mts)[1] # indice final del ultimo dato del dataset
X_train = dat.mts [1:(end-n_obs),]  # Crear conjunto de entrenamiento
X_test = dat.mts [(end-n_obs+1):end,]  # Crear conjunto de prueba
dim(X_test)  # Mostrar dimensiones de X_test
```

```
## [1] 10  3
```

Se obtiene 10 datos reservados y 3 columnas

Prueba de estacionariedad

- H0: Serie = Estacionria
- H1: Serie != Estacionaria

```
apply(X_train, 2, adf.test)  # Realizar prueba ADF por columnas en X_train
```

```
## $gdp
##
##  Augmented Dickey-Fuller Test
##
## data:  newX[, i]
## Dickey-Fuller = -0.92653, Lag order = 4, p-value = 0.9462
## alternative hypothesis: stationary
##
##
## $inf
##
##  Augmented Dickey-Fuller Test
##
## data:  newX[, i]
## Dickey-Fuller = -3.701, Lag order = 4, p-value = 0.02723
## alternative hypothesis: stationary
##
##
## $int
##
##  Augmented Dickey-Fuller Test
##
## data:  newX[, i]
## Dickey-Fuller = -3.4262, Lag order = 4, p-value = 0.05361
## alternative hypothesis: stationary
```

- Se observa que GDP e INT no son estacionarias.
- INF si es estacionaria

Estudiar un modelo para ver la relación entre la tasa de inflación y la tasa de interés INF e INT

```
# para así que obviar el hecho
# de que GDP no sea estacionaria porque no la vamos a utilizar
X_train_new <- X_train[,2:3]  # Seleccionar columnas relevantes para el modelo
```

VAR modeling

```
# Identificación del orden del modelo
library(vars)
```

```
## Warning: package 'vars' was built under R version 4.3.3
```

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

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## 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
```

```
VARselect(X_train_new, type = "none", lag.max = 12)  # Seleccionar orden del modelo VAR
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      3      2      3
##
## $criteria
##                  1           2           3          4           5           6
## AIC(n) -2.88980087 -3.09900126 -3.14959503 -3.0958107 -3.04221385 -3.00203620
## HQ(n)  -2.84883184 -3.01706320 -3.02668793 -2.9319345 -2.83736869 -2.75622201
## SC(n)  -2.78869762 -2.89679476 -2.84628527 -2.6913977 -2.53669759 -2.39541669
## FPE(n)  0.05558779  0.04509754  0.04288017  0.0452651  0.04778441  0.04978557
##                  7           8           9          10          11          12
## AIC(n) -3.04317179 -3.06059055 -3.00932952 -2.95051387 -2.89561743 -2.84588843
## HQ(n)  -2.75638857 -2.73283830 -2.64060823 -2.54082355 -2.44495808 -2.35426004
## SC(n)  -2.33544903 -2.25176454 -2.09940025 -1.93948136 -1.78348166 -1.63264941
## FPE(n)  0.04783598  0.04708475  0.04966319  0.05280794  0.05596551  0.05904546
```

Según el criterio AIC es que se debe considerar hasta 3 retrasos (orden 3)

Creando el modelo VAR

```
var.a <- vars::VAR(X_train_new,

                   lag.max = 10, # max retrasos

                   ic = "AIC", # criterio

                   type = "const")  # Crear modelo VAR

summary(var.a)  # Resumen del modelo
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: inf, int
## Deterministic variables: const
## Sample size: 114
## Log Likelihood: -140.413
## Roots of the characteristic polynomial:
## 0.8438 0.8438 0.5384 0.5179 0.5179 0.0106
## Call:
## vars::VAR(y = X_train_new, type = "const", lag.max = 10, ic = "AIC")
##
##
## Estimation results for equation inf:
## ====================================
## inf = inf.l1 + int.l1 + inf.l2 + int.l2 + inf.l3 + int.l3 + const
##
##         Estimate Std. Error t value Pr(>|t|)
## inf.l1  0.498461   0.097259   5.125 1.33e-06 ***
## int.l1  0.569913   0.349463   1.631   0.1059
## inf.l2  0.008638   0.105312   0.082   0.9348
## int.l2 -0.911124   0.560890  -1.624   0.1072
## inf.l3  0.243582   0.093113   2.616   0.0102 *
## int.l3  0.487111   0.334374   1.457   0.1481
## const   0.109249   0.321186   0.340   0.7344
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.8855 on 107 degrees of freedom
## Multiple R-Squared: 0.527,   Adjusted R-squared: 0.5005
## F-statistic: 19.87 on 6 and 107 DF,  p-value: 1.745e-15
##
##
## Estimation results for equation int:
## ====================================
## int = inf.l1 + int.l1 + inf.l2 + int.l2 + inf.l3 + int.l3 + const
##
##        Estimate Std. Error t value Pr(>|t|)
## inf.l1  0.03690    0.02825   1.306   0.1943
## int.l1  1.33982    0.10151  13.200   <2e-16 ***
## inf.l2 -0.01411    0.03059  -0.461   0.6456
## int.l2 -0.38785    0.16292  -2.381   0.0191 *
## inf.l3 -0.01963    0.02705  -0.726   0.4695
## int.l3 -0.03479    0.09712  -0.358   0.7209
## const   0.23088    0.09329   2.475   0.0149 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.2572 on 107 degrees of freedom
## Multiple R-Squared: 0.9151,  Adjusted R-squared: 0.9103
## F-statistic: 192.2 on 6 and 107 DF,  p-value: < 2.2e-16
##
##
```

```
##
## Covariance matrix of residuals:
##         inf     int
## inf 0.78404 0.07849
## int 0.07849 0.06615
##
## Correlation matrix of residuals:
##        inf    int
## inf 1.0000 0.3447
## int 0.3447 1.0000
```

Diagnosis del modelo (Portmanteau test para objetos var)

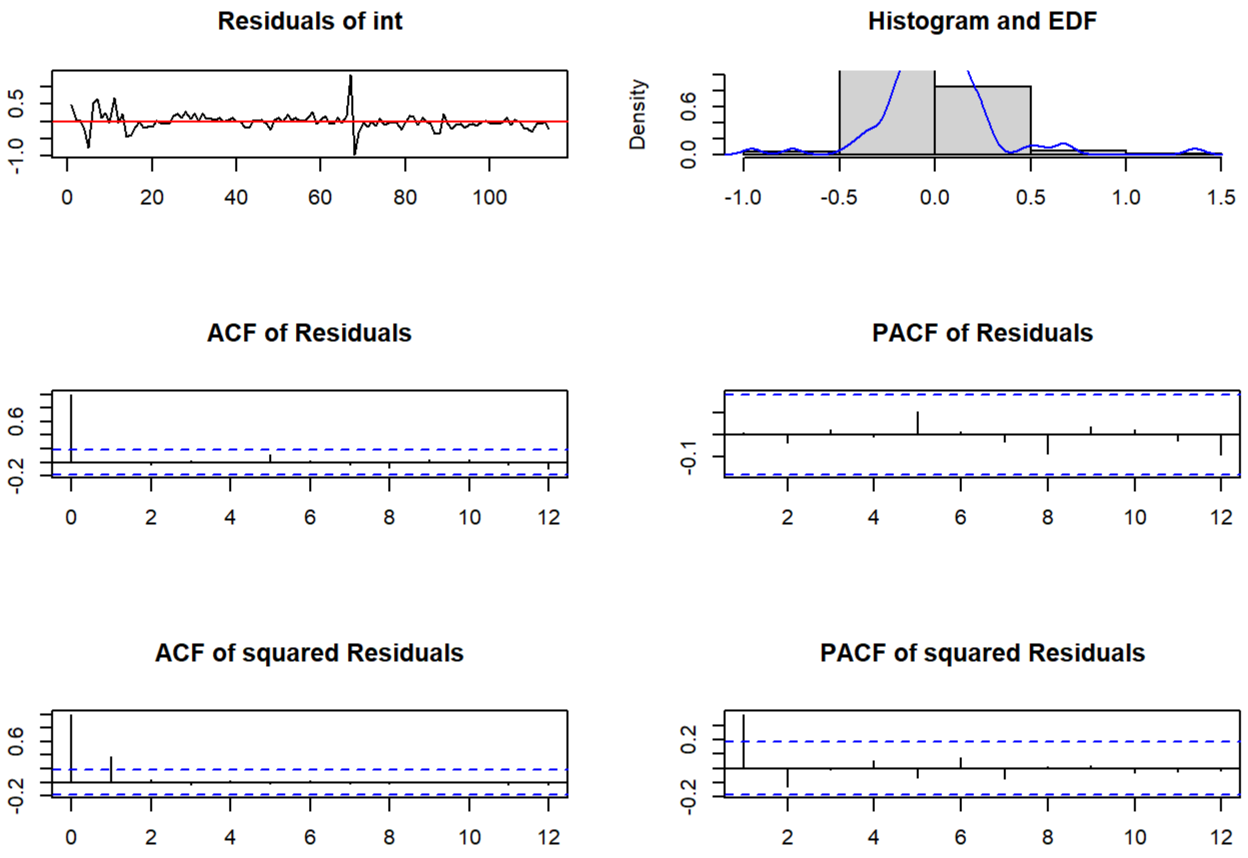
```
bv.serial= serial.test(var.a)  # Realizar test Portmanteau
bv.serial
```

```
##
##  Portmanteau Test (asymptotic)
##
## data:  Residuals of VAR object var.a
## Chi-squared = 39.307, df = 52, p-value = 0.9026
```

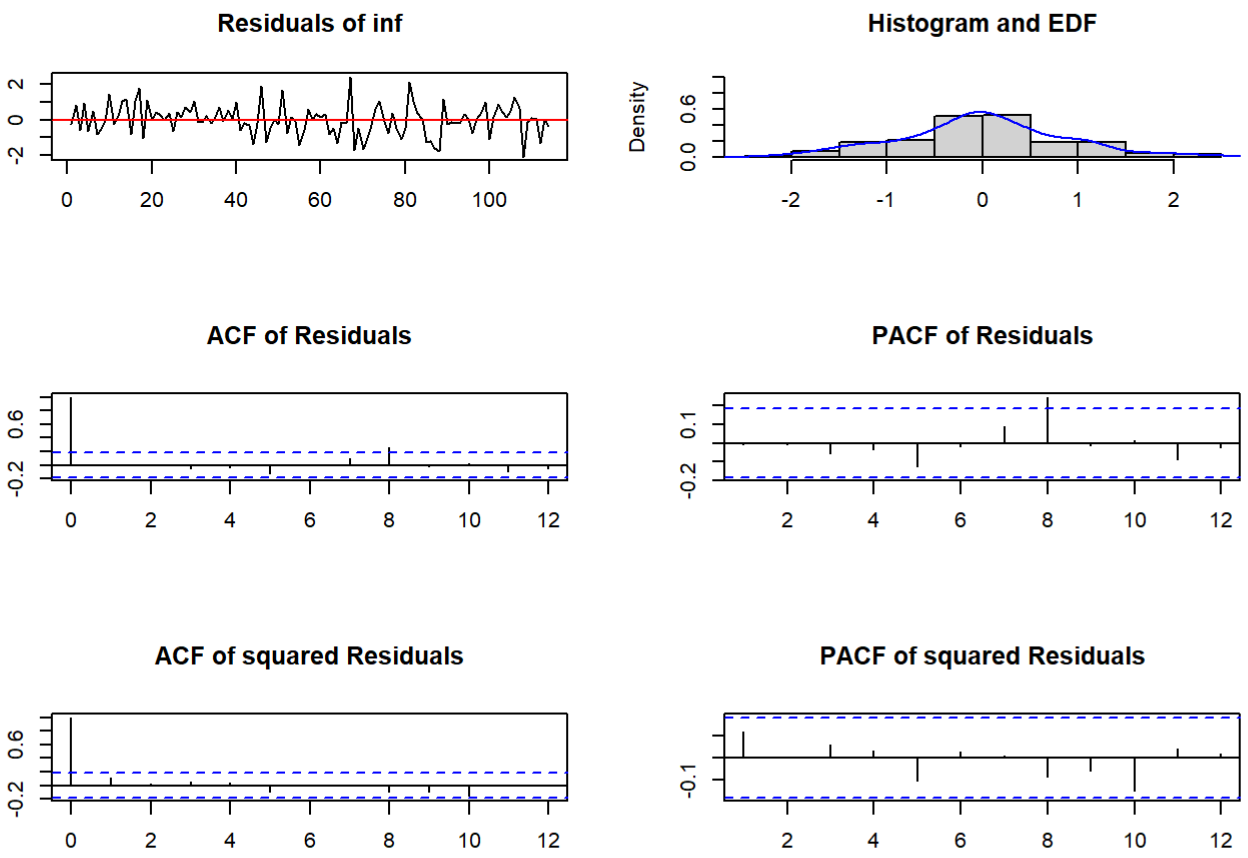
P-value es > 0.05, por lo que no es necesario realizar la diferenciación

Grafico de la diagnosis

```
plot(bv.serial, names = "int")  # Graficar resultados para variable int
```

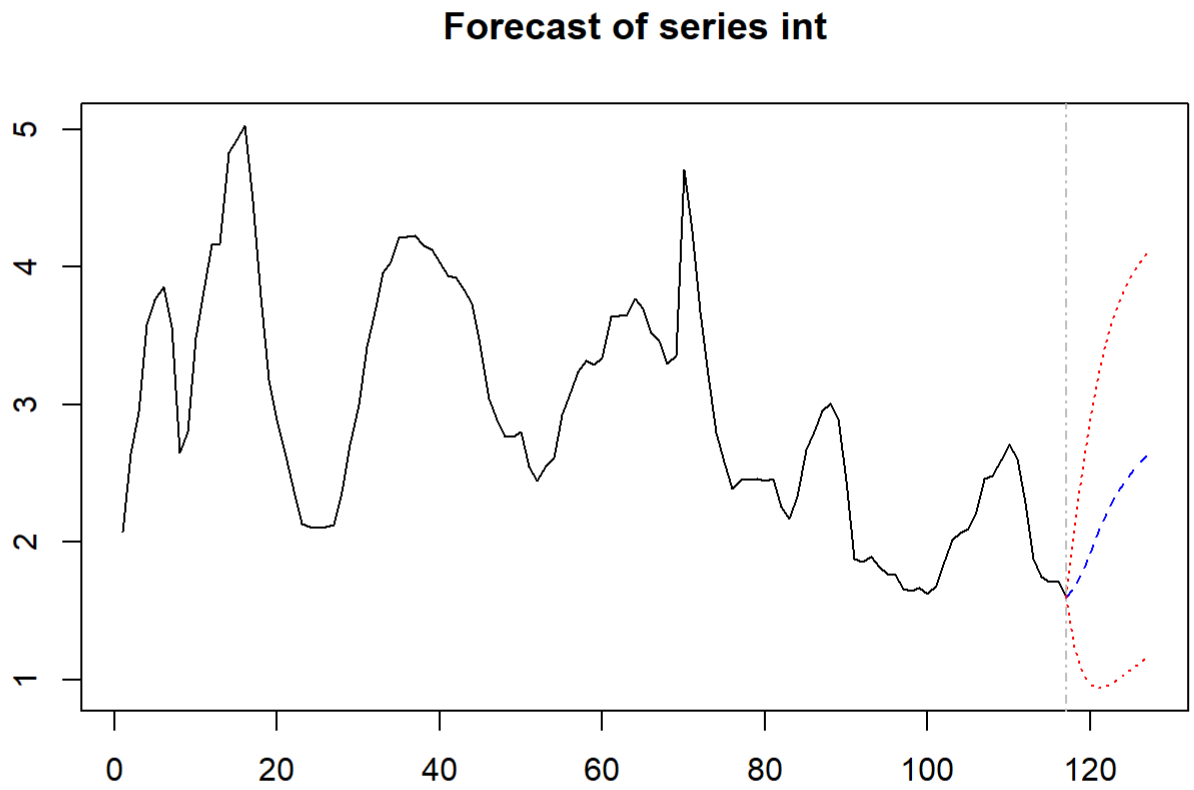



```
plot(bv.serial, names = "inf")  # Graficar resultados para variable inf
```

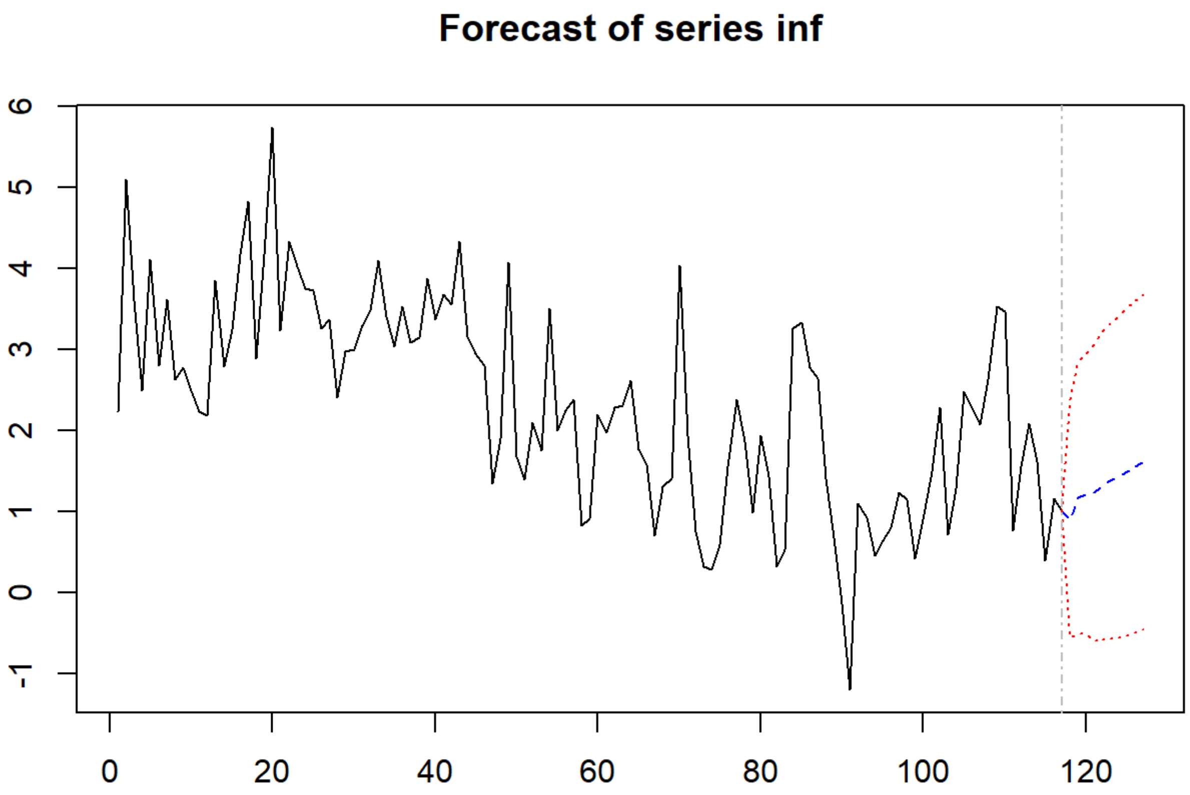


Forecasting

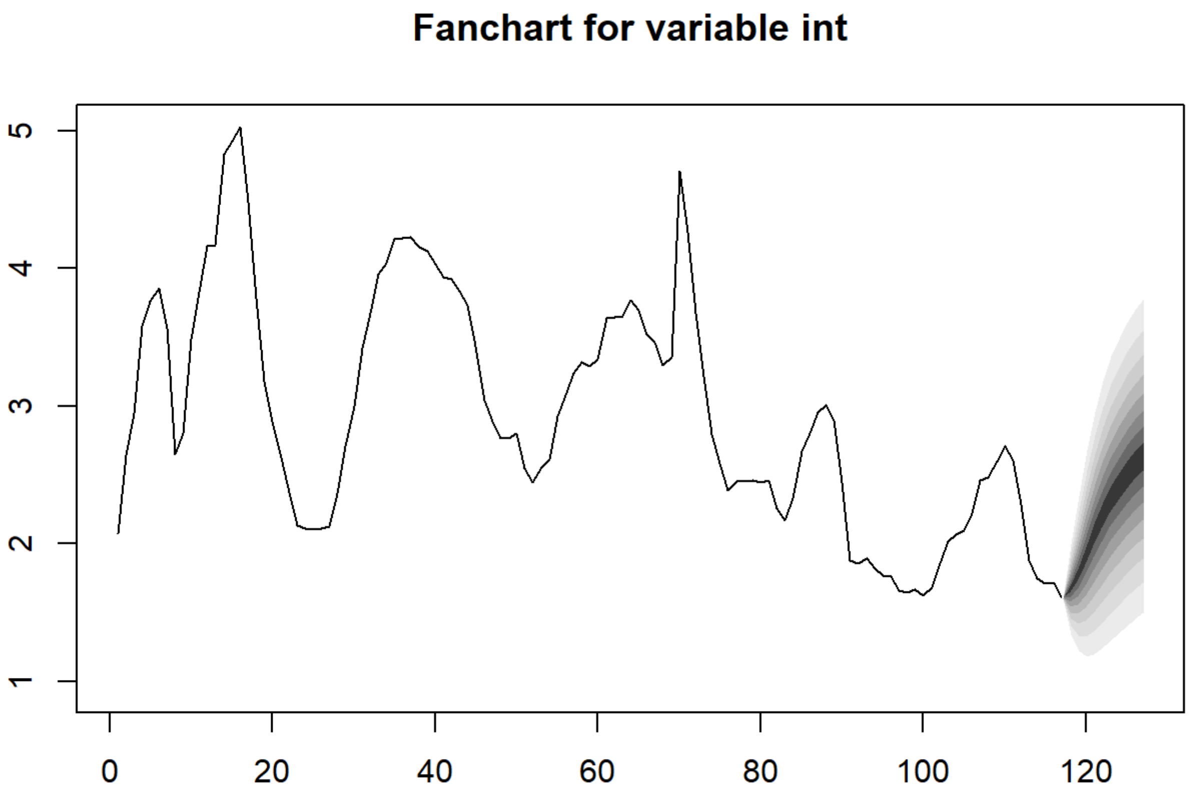
```r
predictions <- predict(var.a, n.ahead = 10, ci = 0.90)  # Realizar pronósticos a 10 periodos con intervalo de confianza del 90%
plot(predictions, names = "int")  # Graficar pronósticos para variable int
```



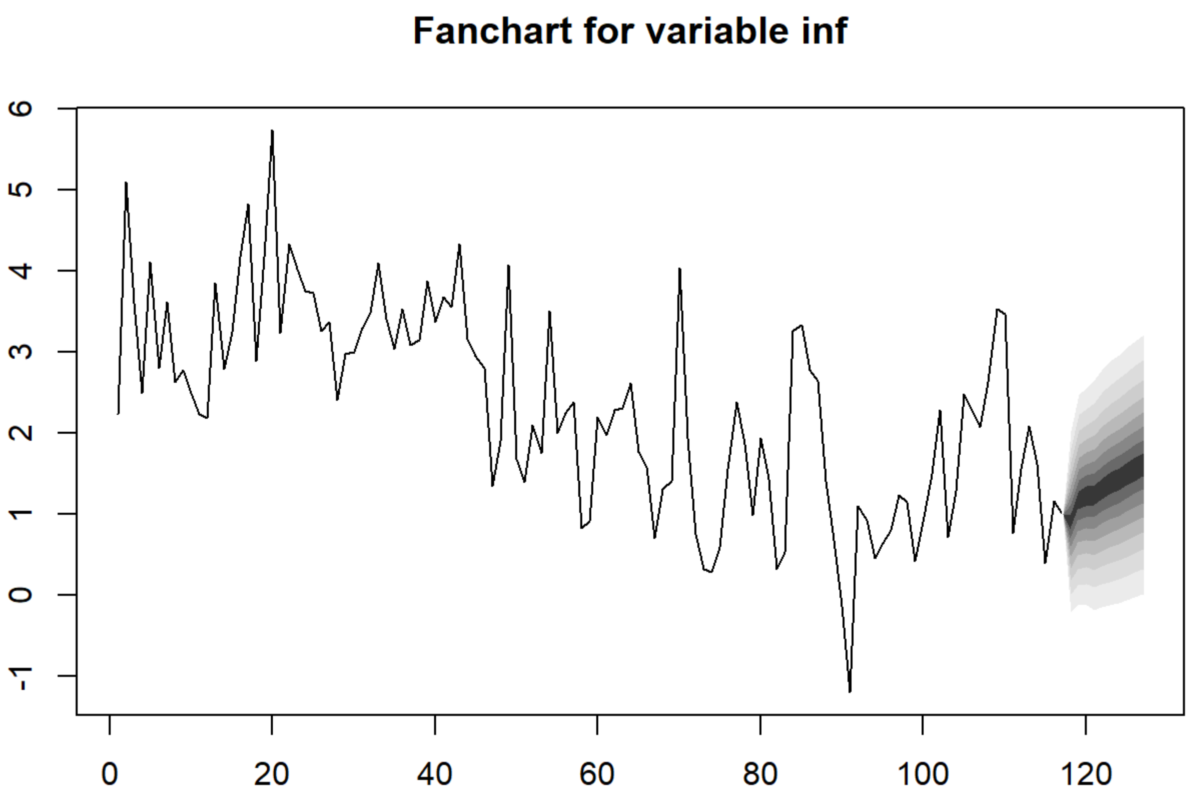

```r
predictions <- predict(var.a, n.ahead = 10, ci = 0.90)  # Realizar pronósticos a 10 periodos en el futuro, con intervalo de confianza del 90%
plot(predictions, names = "inf")  # Graficar pronósticos para variable inf
```




```
# Otro gráfico
fanchart(predictions, names = "int")  # Gráfico adicional para variable int
```



```
fanchart(predictions, names = "inf")  # Gráfico adicional para variable inf
```



Evaluación del modelo

```
pred=predictions$fcst
rmse=sqrt(mean((X_test[,2]-pred$inf)^2))
cat('RMSE inf: ', rmse)
```

```
## RMSE inf:  1.410843
```

```
rmse=sqrt(mean((X_test[,3]-pred$int)^2))
cat('RMSE int: ', rmse)
```

```
## RMSE int:  1.202069
```

Se observa que los resultdos para RMSE son aceptables

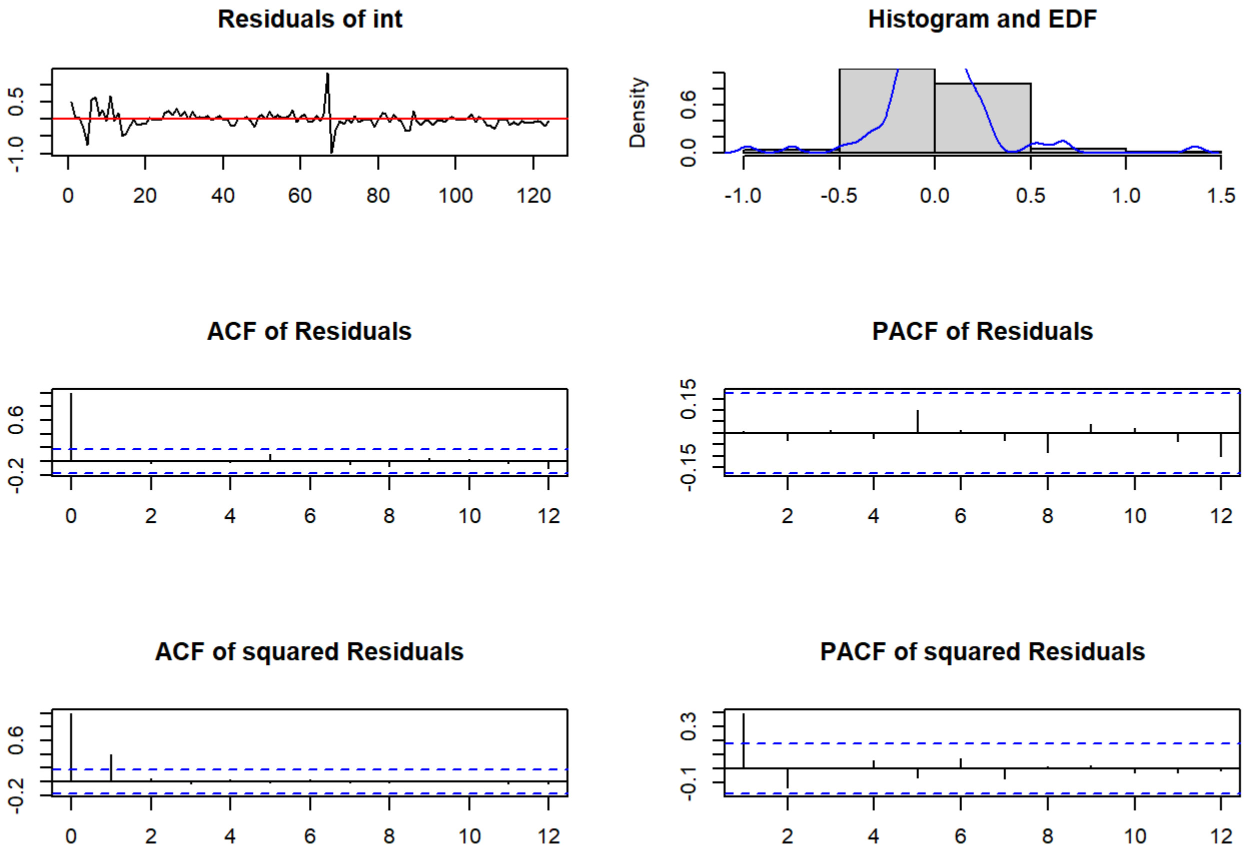
Entrenando con todos los datos

```
# Creando el modelo VAR con todas las variables relevantes
var.a <- vars::VAR(dat.mts[,2:3],

                   lag.max = 10,

                   ic = "AIC",

                   type = "const")  # Crear modelo VAR con todas las variables relevantes

summary(var.a)  # Resumen del nuevo modelo
```

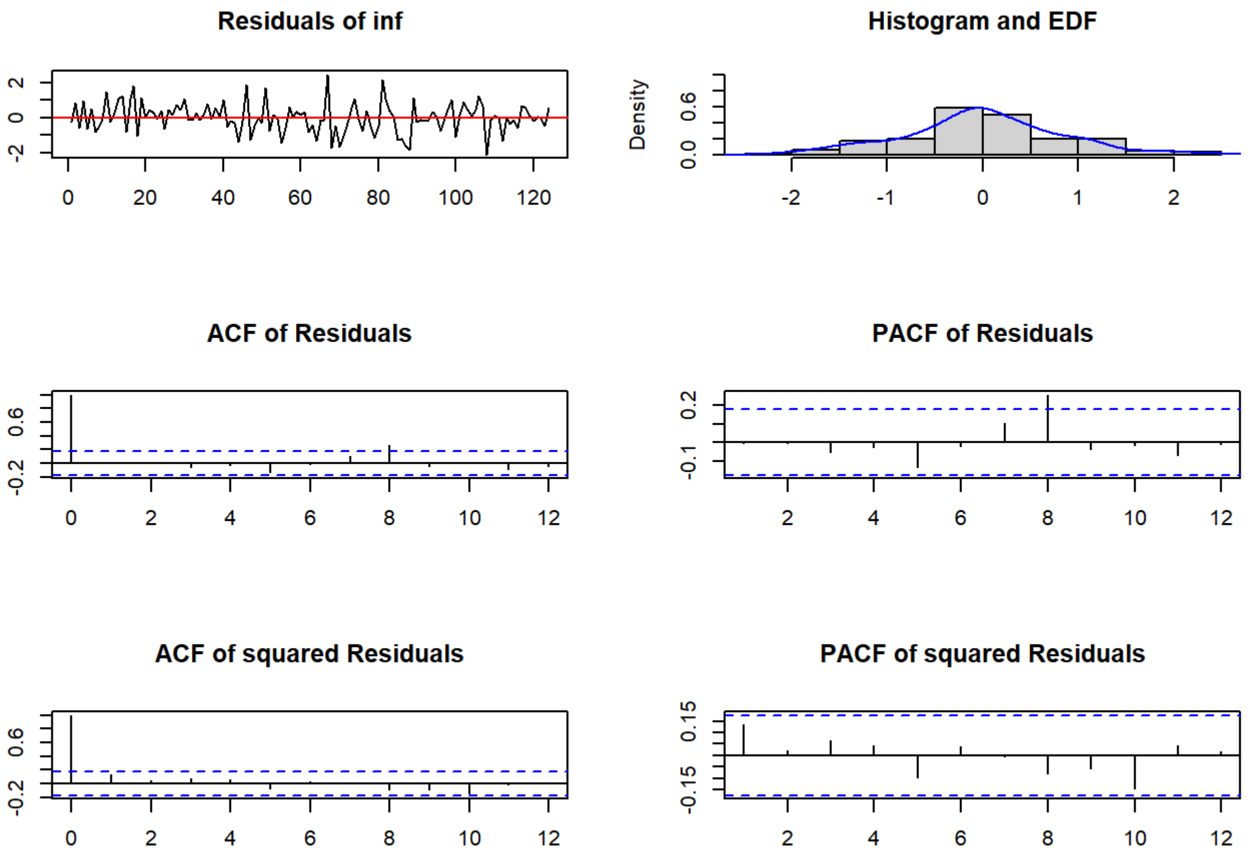
```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: inf, int
## Deterministic variables: const
## Sample size: 124
## Log Likelihood: -145.197
## Roots of the characteristic polynomial:
## 0.8681 0.8681 0.5206 0.5206 0.4745 0.03618
## Call:
## vars::VAR(y = dat.mts[, 2:3], type = "const", lag.max = 10, ic = "AIC")
##
##
## Estimation results for equation inf:
## ====================================
## inf = inf.l1 + int.l1 + inf.l2 + int.l2 + inf.l3 + int.l3 + const
##
##         Estimate Std. Error t value Pr(>|t|)
## inf.l1  0.501210   0.093216   5.377 3.93e-07 ***
## int.l1  0.542232   0.333551   1.626  0.10672
## inf.l2  0.001997   0.101176   0.020  0.98428
## int.l2 -0.877584   0.539561  -1.626  0.10654
## inf.l3  0.244784   0.088920   2.753  0.00685 **
## int.l3  0.469499   0.320319   1.466  0.14541
## const   0.154976   0.263580   0.588  0.55769
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.8556 on 117 degrees of freedom
## Multiple R-Squared: 0.5452,  Adjusted R-squared: 0.5218
## F-statistic: 23.37 on 6 and 117 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation int:
## ====================================
## int = inf.l1 + int.l1 + inf.l2 + int.l2 + inf.l3 + int.l3 + const
##
##        Estimate Std. Error t value Pr(>|t|)
## inf.l1  0.03598    0.02709   1.328   0.1868
## int.l1  1.35979    0.09695  14.026   <2e-16 ***
## inf.l2 -0.01444    0.02941  -0.491   0.6244
## int.l2 -0.40311    0.15683  -2.570   0.0114 *
## inf.l3 -0.01759    0.02585  -0.681   0.4974
## int.l3 -0.01943    0.09311  -0.209   0.8351
## const   0.16099    0.07661   2.101   0.0378 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.2487 on 117 degrees of freedom
## Multiple R-Squared: 0.9322,  Adjusted R-squared: 0.9288
## F-statistic: 268.3 on 6 and 117 DF,  p-value: < 2.2e-16
##
##
```

```
##
## Covariance matrix of residuals:
##          inf     int
## inf 0.73214 0.07237
## int 0.07237 0.06185
##
## Correlation matrix of residuals:
##         inf    int
## inf 1.0000 0.3401
## int 0.3401 1.0000
```

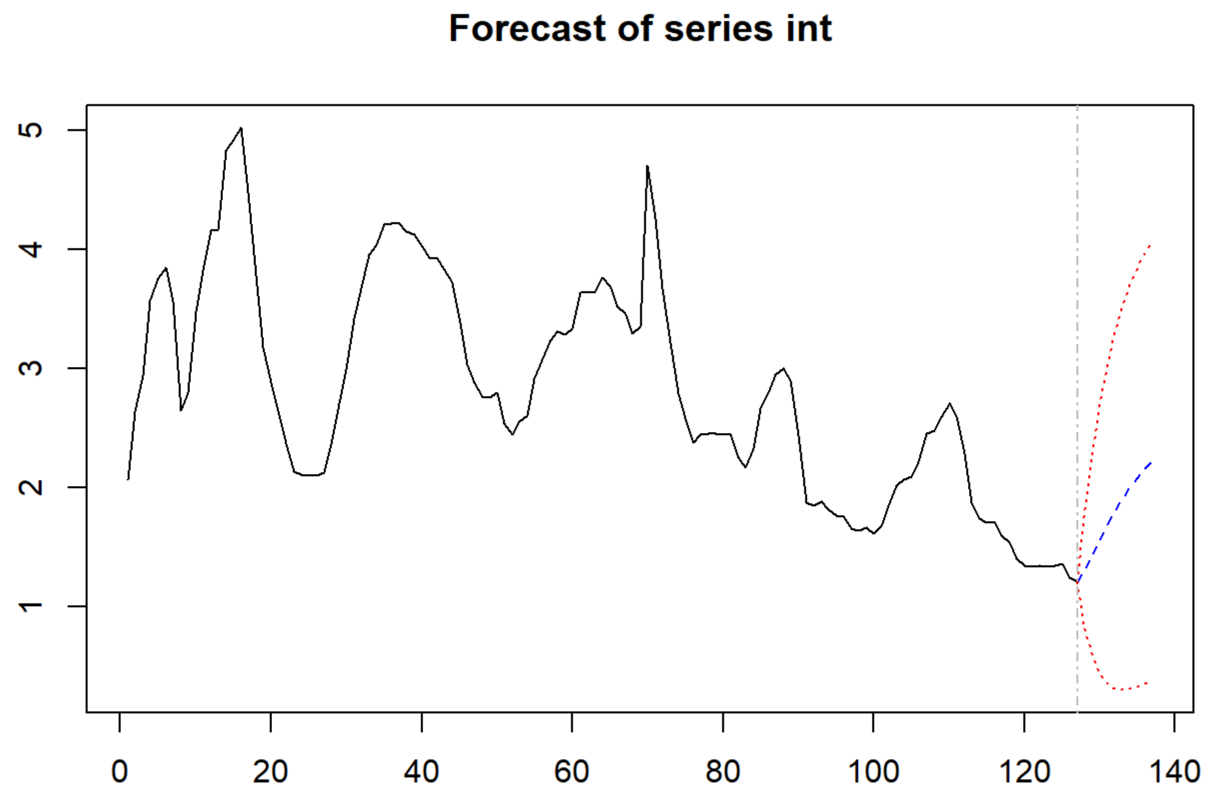
```
## Diagnosis del modelo (Portmanteau test para objetos var)
bv.serial= serial.test(var.a)  # Realizar test Portmanteau nuevamente

plot(bv.serial, names = "int")  # Graficar resultados para variable int
```



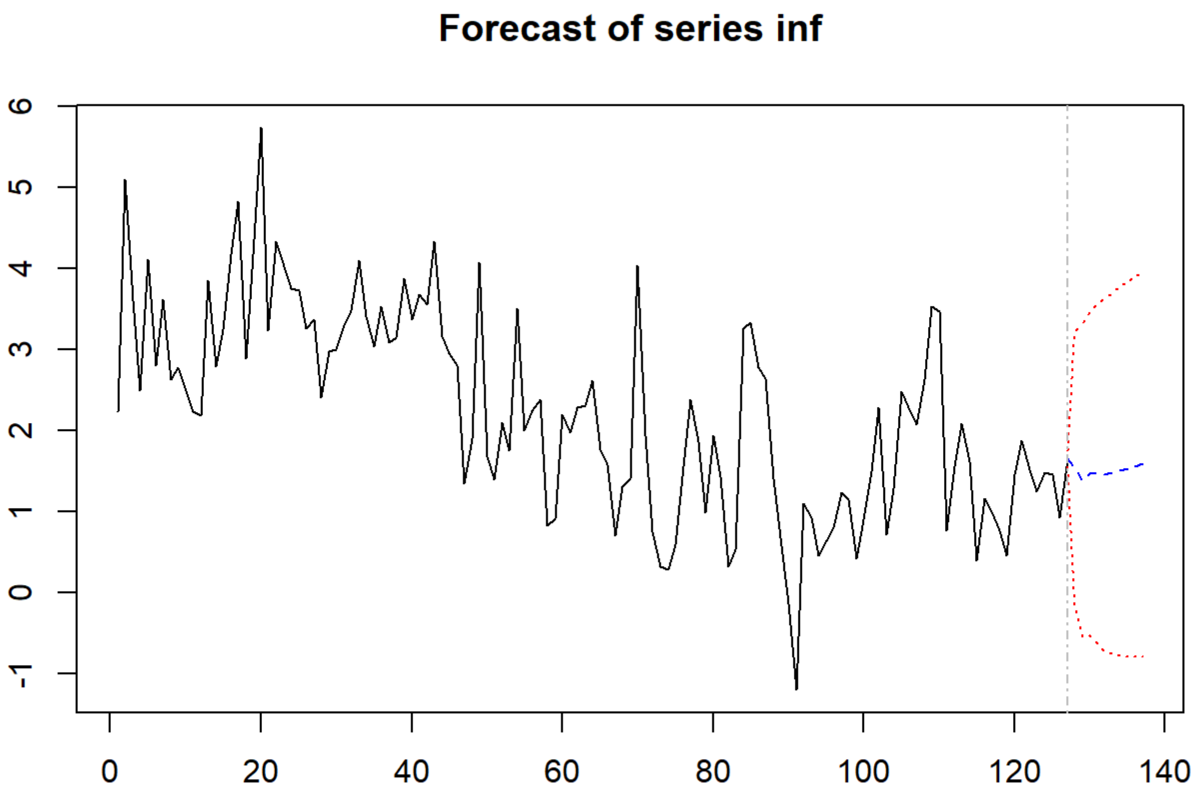

```
plot(bv.serial, names = "inf")  # Graficar resultados para variable inf
```



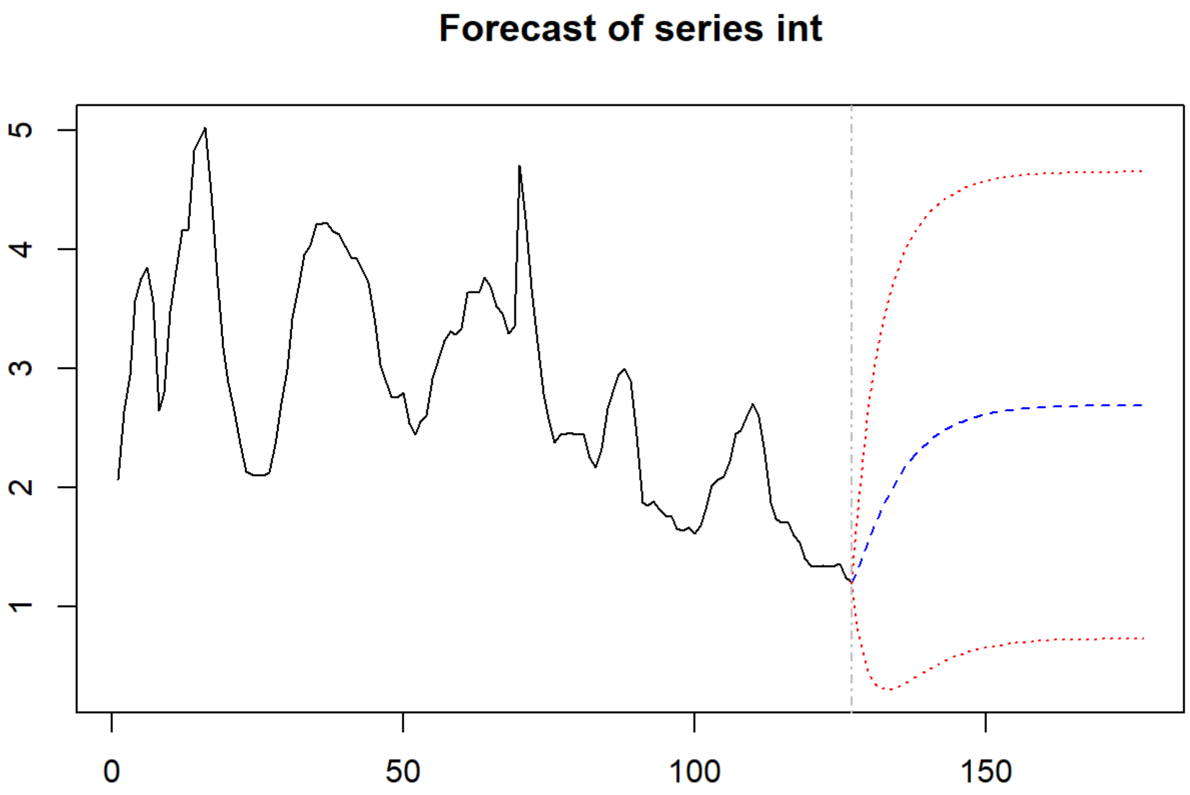

```
## Forecasting usando el modelo VAR (Hallando los pronósticos)
predictions <- predict(var.a, n.ahead = 10, ci = 0.95)  # Realizar pronósticos a futuro con intervalo de confianza del 95%
plot(predictions, names = "int")  # Graficar pronósticos para variable int
```



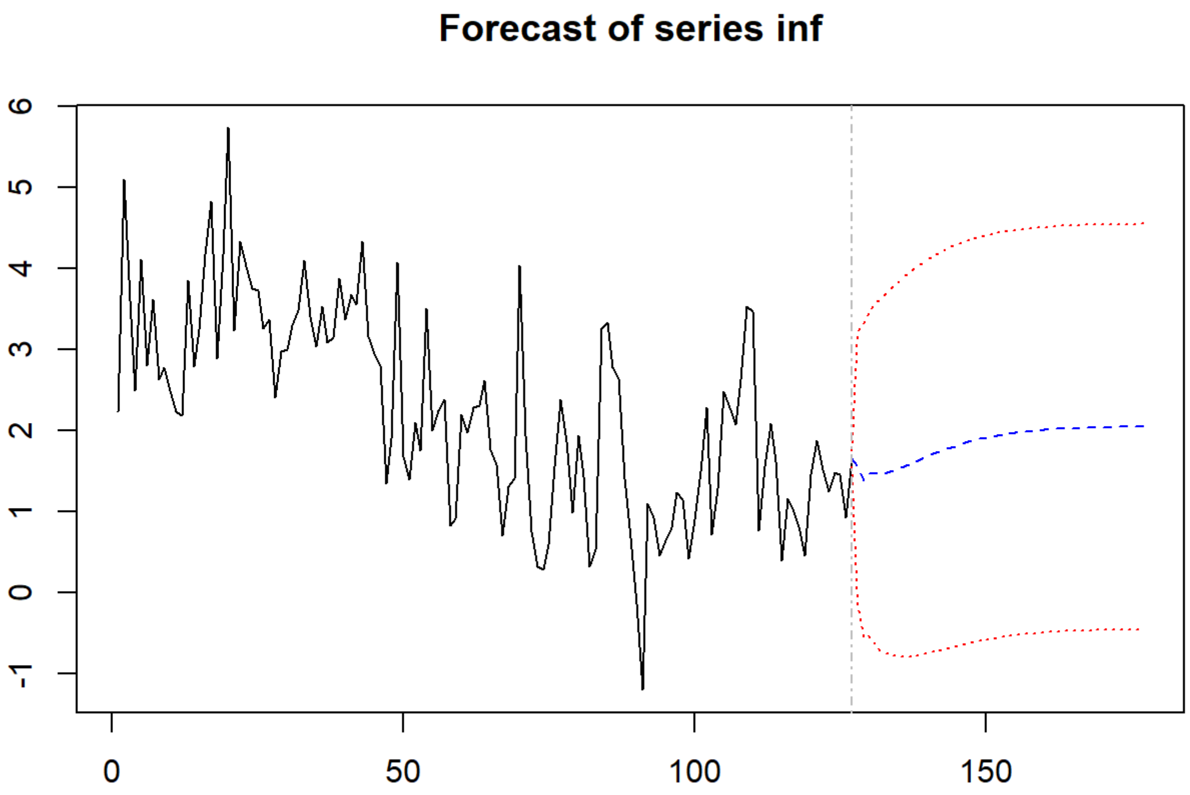

```
## Forecasting usando el modelo VAR (Hallando los pronósticos)
predictions <- predict(var.a, n.ahead = 10, ci = 0.95)   # Realizar pronósticos a futuro con
intervalo de confianza del 95%
plot(predictions, names = "inf")   # Graficar pronósticos para variable inf
```




```
## Más adelante en el futuro "n.ahead=50"
predictions <- predict(var.a, n.ahead = 50, ci = 0.95)    # Realizar pronósticos a futuro con intervalo de confianza del 95%
plot(predictions, names = "int")    # Graficar pronósticos para variable int
```

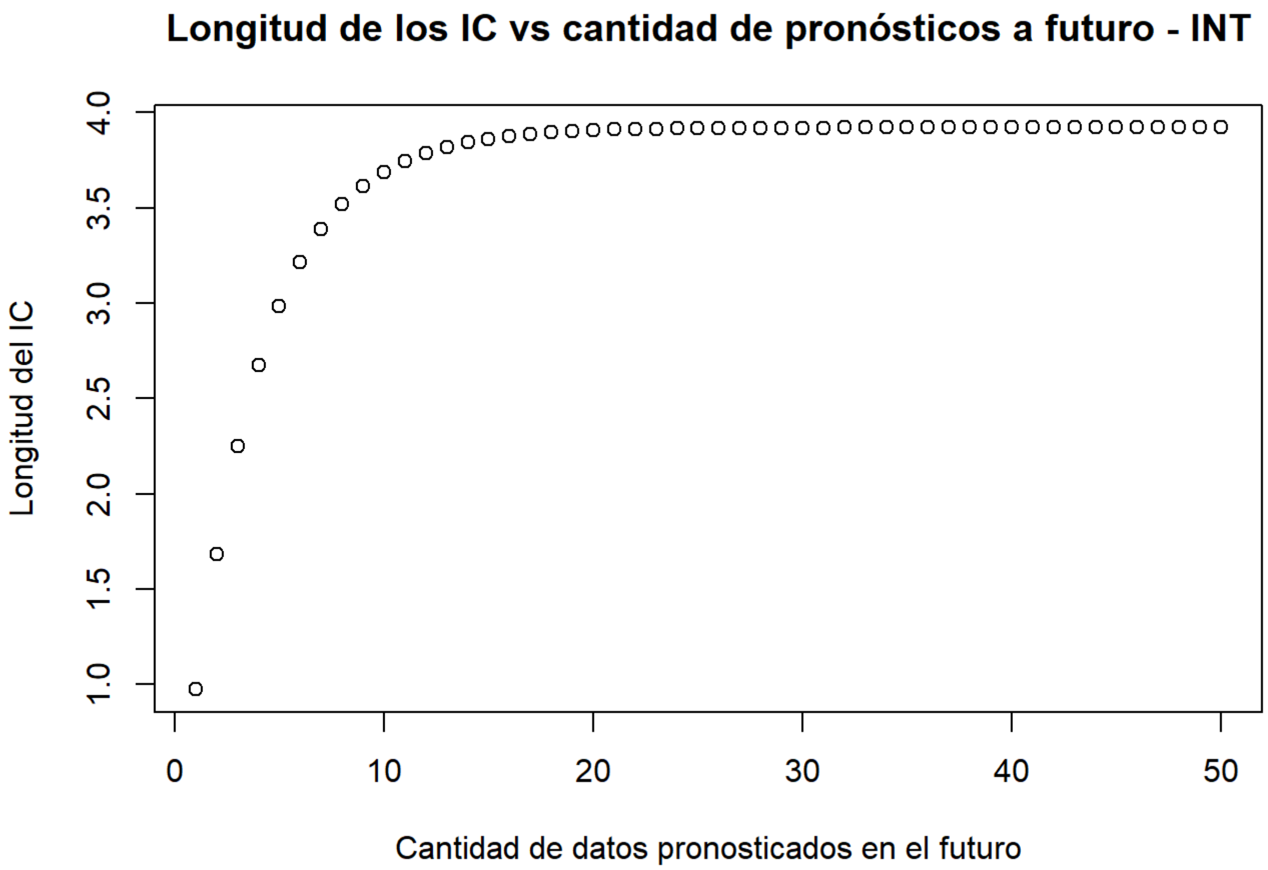



```
#"n.ahead=50"
predictions <- predict(var.a, n.ahead = 50, ci = 0.95)   # Realizar pronósticos a futuro con intervalo de confianza del 95%
plot(predictions, names = "inf")   # Graficar pronósticos para variable inf
```



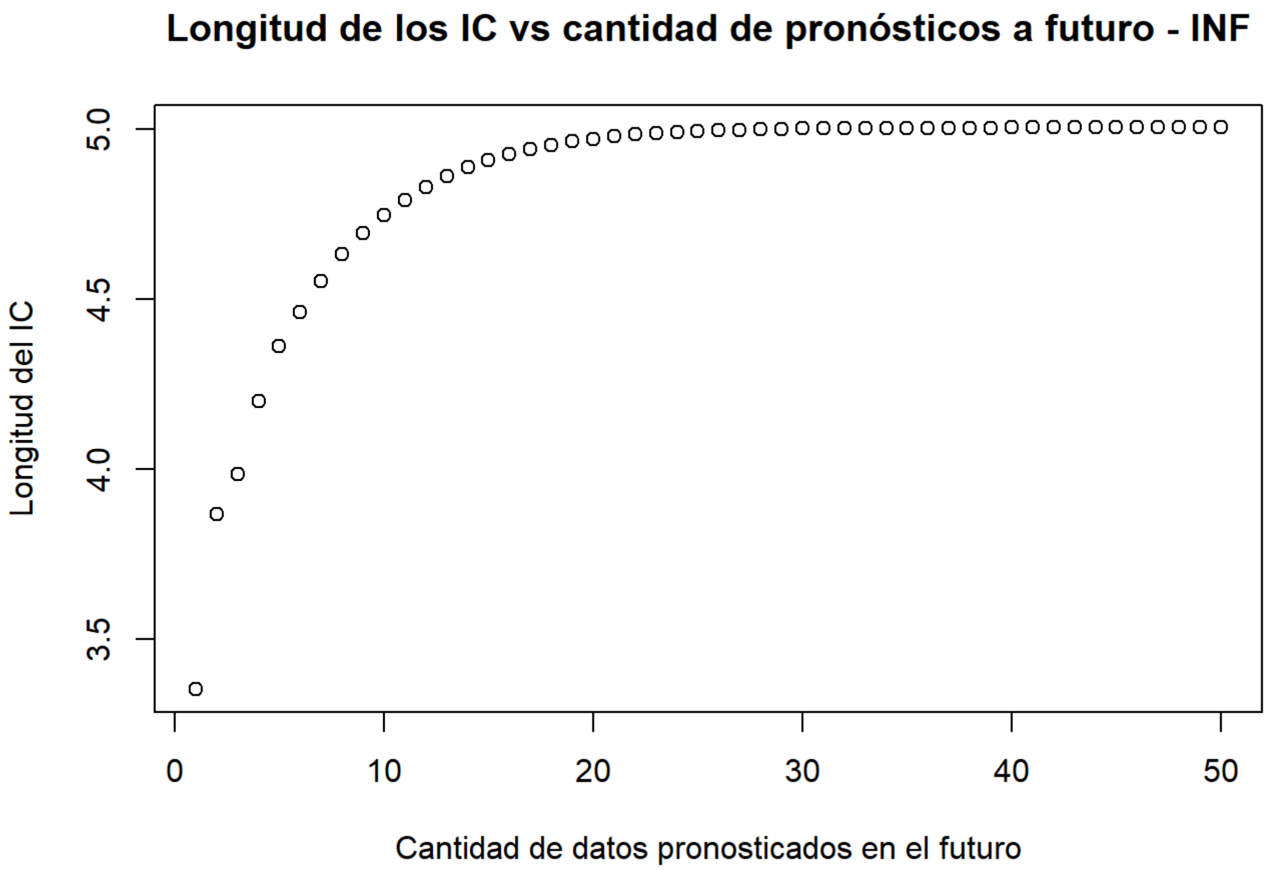
Tamaño de los intervalos de confianza

```
diff_IC_int=predictions$fcst$int[,3]-predictions$fcst$int[,2]   # Calcular longitud de IC par
a variable int
plot(diff_IC_int, main="Longitud de los IC vs cantidad de pronósticos a futuro - INT", xlab
='Cantidad de datos pronosticados en el futuro', ylab='Longitud del IC')   # Graficar longitu
d de IC vs cantidad de pronósticos futuros para variable int
```



Cantidad de datos a futuro a predecir 8 a 10.

```
diff_IC_inf=predictions$fcst$inf[,3]-predictions$fcst$inf[,2]   # Calcular longitud de IC par
a variable inf
plot(diff_IC_inf, main="Longitud de los IC vs cantidad de pronósticos a futuro - INF", xlab
='Cantidad de datos pronosticados en el futuro', ylab='Longitud del IC')   # Graficar longitu
d de IC vs cantidad de pronósticos futuros para variable inf
```



Cantidad de posibles valores a futuro a predecir 10

- A mayor cantidad de datos al futuro a predecir mayor error, por lo tanto se perderá precisión y poder predictivo