Preprocessing Data

Lourdes Sofía Elizondo Guajardo Daniela Diaz Delgado Bernardo Ortega Chávez
Gabriel González Bataller

2021-09-05

Contents

1 National Data Set

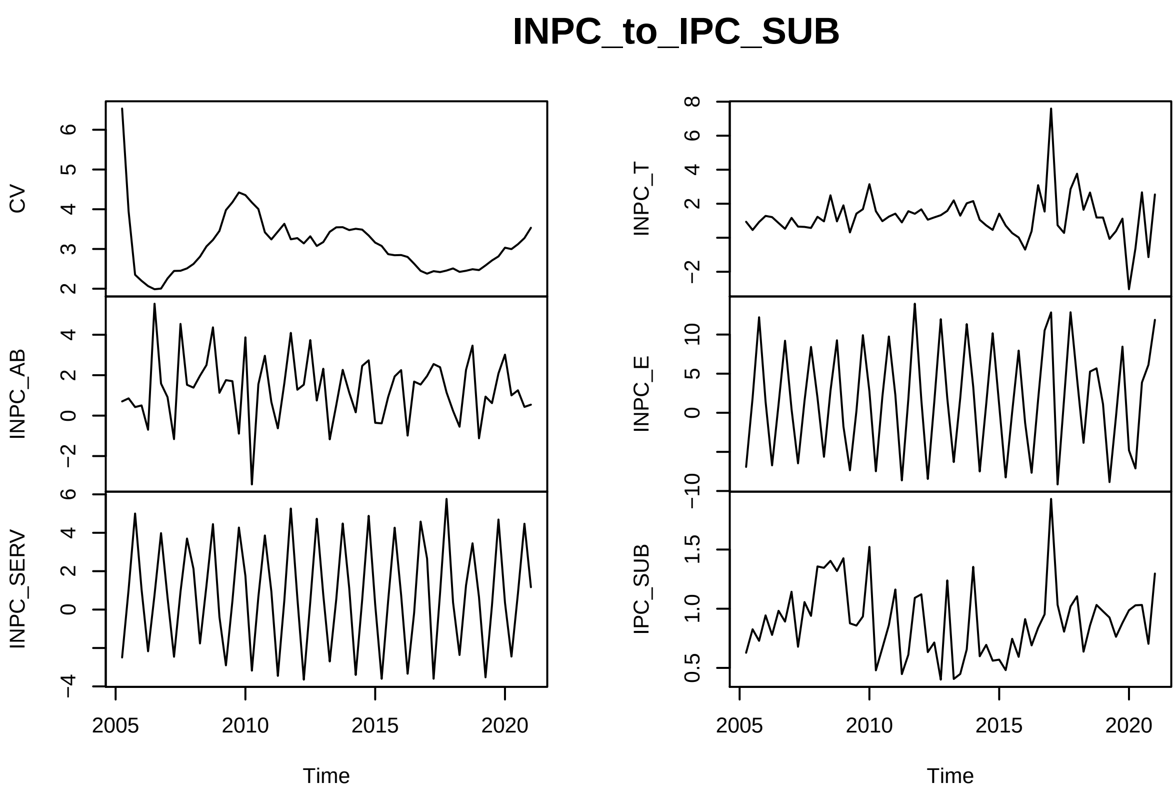
1.1 Loading and visualizing the data

```
BDT2 <- read_excel("BDF NACIONAL.xlsx")
DATA.ts <- ts(BDT2, start = c(2005, 2), frequency = 4)

CV <- DATA.ts[ ,1]
INPC_AB <- DATA.ts[ ,2]
INPC_SERV <- DATA.ts[ ,3]
INPC_T<- DATA.ts[ ,4]
INPC_E <- DATA.ts[ ,5]
IPV <- DATA.ts[ ,6]
IPC_SUB <- DATA.ts[ ,7]
REMESAS <- DATA.ts[ ,8]
INT <- DATA.ts[ ,9]
CONF <- DATA.ts[ ,10]
M1 <- DATA.ts[ ,11]
DEBT <- DATA.ts[ ,12]
EX <- DATA.ts[ ,13]
PIB <- DATA.ts[,14]
DESEMPLEO <- DATA.ts[,15]
IGAE <- DATA.ts[,16]

head(DATA.ts)
```

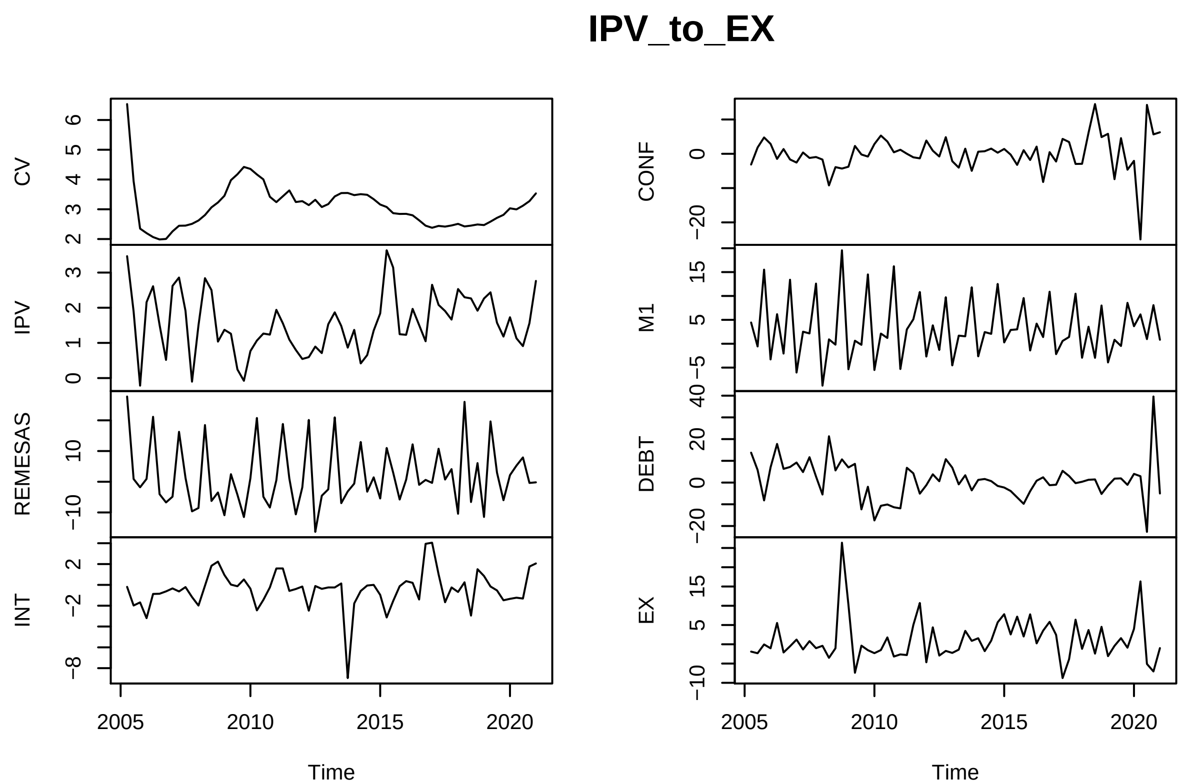
```
##                CV    INPC_AB  INPC_SERV    INPC_T    INPC_E        IPV   IPC_SUB
## 2005 Q2 6.534773  0.7018551 -2.4951237 0.9394907 -6.924964  3.4660615 0.6277530
## 2005 Q3 3.947015  0.8525432  1.0491358 0.4557571  1.906007  1.8943170 0.8263839
## 2005 Q4 2.350651  0.4220444  5.0026781 0.9256332 12.211712 -0.2152642 0.7288090
## 2006 Q1 2.199177  0.5030827  1.0508060 1.2841778  1.429080  2.1572857 0.9439550
## 2006 Q2 2.066763 -0.6987303 -2.1718324 1.2125805 -6.725701  2.6108658 0.7785749
## 2006 Q3 1.986694  5.5345808  0.7797826 0.8662647  1.063017  1.5154350 0.9826554
##             REMESAS        INT      CONF         M1      DEBT          EX
## 2005 Q2 27.7734026 -0.1907032 -3.142433  4.4462848 13.768227 -1.90350962
## 2005 Q3  0.9008468 -1.9823262  1.831598 -0.5757901  5.782404 -2.32225035
## 2005 Q4 -1.8000233 -1.6812865  4.768270 15.5243856 -8.226283 -0.04065695
## 2006 Q1  0.9321390 -3.1970260  2.897367 -3.2951687  7.002603 -1.04508266
## 2006 Q2 21.1573268 -0.8704557 -1.492325  6.1874601 17.777689  5.53210143
## 2006 Q3 -4.0398914 -0.8522727  1.383832 -2.0676441  6.294740 -2.13212457
##                PIB  DESEMPLEO       IGAE
## 2005 Q2  5.616308  -9.525851 -0.1121490
## 2005 Q3 -2.287794   8.693642  1.9744905
## 2005 Q4  3.804774 -17.483354  1.7121746
## 2006 Q1 -1.387925  13.107616  0.7367485
## 2006 Q2  4.200068 -11.208719  1.1990516
## 2006 Q3 -1.800185  27.269522  0.4334371
```

```
# PLOTS
INPC_to_IPC_SUB <- ts(cbind(CV,INPC_AB,INPC_SERV,INPC_T,INPC_E,IPC_SUB),start=c(2005,2),frequency=4)
IPV_to_EX <- ts(cbind(CV, IPV, REMESAS, INT, CONF, M1, DEBT, EX),start=c(2005,2),frequency=4)
```

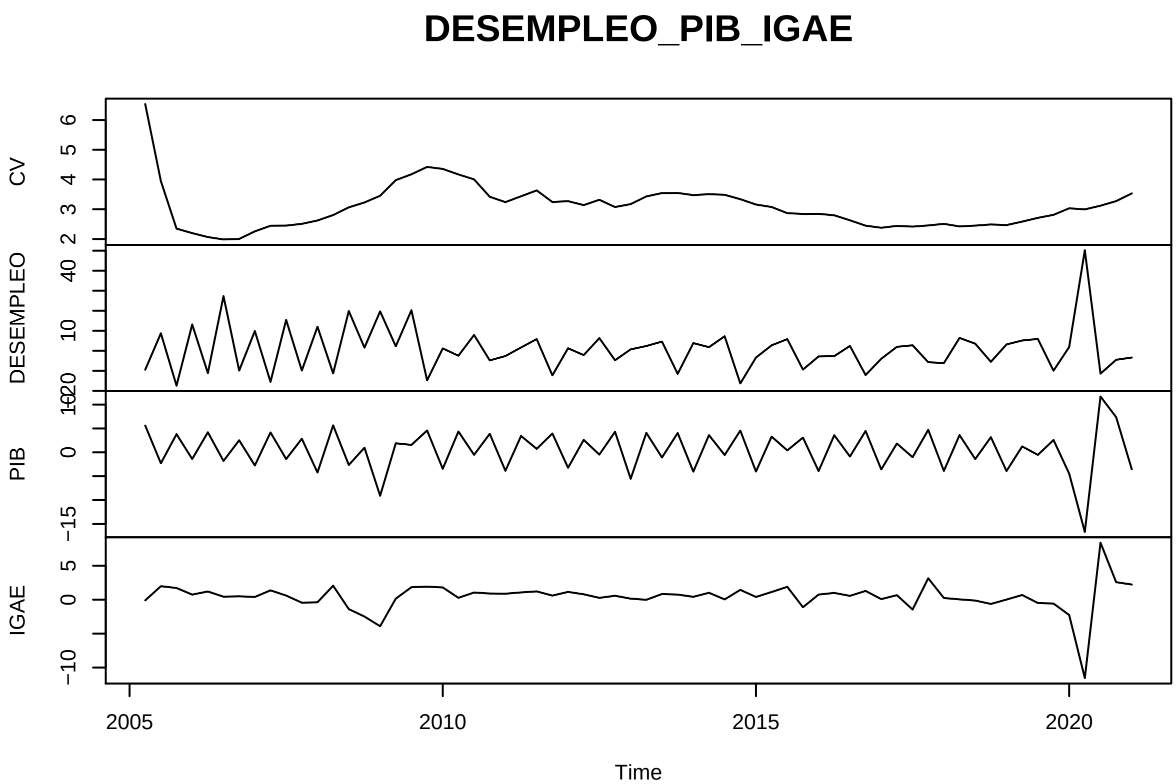
```r
DESEMPLEO_PIB_IGAE <- ts(cbind(CV, DESEMPLEO, PIB, IGAE),start=c(2005,2),frequency=4)
plot(INPC_to_IPC_SUB, cex.lab=0.7)
```




```r
plot(IPV_to_EX, cex.lab=0.7)
```




```r
plot(DESEMPLEO_PIB_IGAE, cex.lab=0.7)
```



1.2 Correlation

```r
# Matrix
round(cor(DATA.ts), 2)
```

```
##               CV INPC_AB INPC_SERV INPC_T INPC_E   IPV IPC_SUB REMESAS   INT
## CV          1.00   -0.11     -0.14  -0.01  -0.13 -0.12   -0.17    0.13 -0.04
## INPC_AB    -0.11    1.00      0.41  -0.04   0.35 -0.19    0.40   -0.41  0.06
## INPC_SERV  -0.14    0.41      1.00   0.13   0.95 -0.40    0.22   -0.67  0.06
## INPC_T     -0.01   -0.04      0.13   1.00   0.35  0.09    0.44   -0.12  0.16
## INPC_E     -0.13    0.35      0.95   0.35   1.00 -0.35    0.27   -0.67  0.09
## IPV        -0.12   -0.19     -0.40   0.09  -0.35  1.00    0.11    0.38  0.10
## IPC_SUB    -0.17    0.40      0.22   0.44   0.27  0.11    1.00   -0.30  0.35
## REMESAS     0.13   -0.41     -0.67  -0.12  -0.67  0.38   -0.30    1.00 -0.02
## INT        -0.04    0.06      0.06   0.16   0.09  0.10    0.35   -0.02  1.00
## CONF       -0.01   -0.10      0.10   0.11   0.16 -0.07   -0.17   -0.04  0.03
## M1          0.01    0.09      0.50  -0.21   0.45 -0.38   -0.31   -0.02  0.00
## DEBT       -0.11   -0.01     -0.12  -0.27  -0.20  0.25   -0.04    0.17  0.16
## EX         -0.08    0.17      0.18  -0.06   0.14 -0.05    0.10   -0.07  0.09
## PIB         0.10   -0.16      0.12  -0.03   0.10 -0.09   -0.45    0.37 -0.05
## DESEMPLEO  -0.03    0.17     -0.34  -0.12  -0.36  0.08    0.17   -0.10 -0.02
## IGAE        0.06   -0.11      0.12   0.23   0.18  0.00   -0.12    0.11 -0.02
##             CONF    M1  DEBT    EX   PIB DESEMPLEO  IGAE
## CV         -0.01  0.01 -0.11 -0.08  0.10     -0.03  0.06
```

```
## INPC_AB   -0.10  0.09 -0.01  0.17 -0.16      0.17 -0.11
## INPC_SERV  0.10  0.50 -0.12  0.18  0.12     -0.34  0.12
## INPC_T     0.11 -0.21 -0.27 -0.06 -0.03     -0.12  0.23
## INPC_E     0.16  0.45 -0.20  0.14  0.10     -0.36  0.18
## IPV       -0.07 -0.38  0.25 -0.05 -0.09      0.08  0.00
## IPC_SUB   -0.17 -0.31 -0.04  0.10 -0.45      0.17 -0.12
## REMESAS   -0.04 -0.02  0.17 -0.07  0.37     -0.10  0.11
## INT        0.03  0.00  0.16  0.09 -0.05     -0.02 -0.02
## CONF       1.00 -0.12 -0.21 -0.45  0.40     -0.35  0.61
## M1        -0.12  1.00  0.06  0.33  0.52     -0.46 -0.01
## DEBT      -0.21  0.06  1.00 -0.03  0.03      0.08 -0.18
## EX        -0.45  0.33 -0.03  1.00 -0.28      0.18 -0.46
## PIB        0.40  0.52  0.03 -0.28  1.00     -0.68  0.67
## DESEMPLEO -0.35 -0.46  0.08  0.18 -0.68      1.00 -0.57
## IGAE       0.61 -0.01 -0.18 -0.46  0.67     -0.57  1.00
```

We filtered the coefficients to only get those greater than 0.5:

```r
# Filtered data
cor_mat <- cor(DATA.ts)
cor_mat[!lower.tri(cor_mat)] <- NA # remove diagonal and redundant values
data.frame( cor_mat) %>%
              rownames_to_column() %>%
              gather(key="variable", value="correlation", -rowname) %>%
              filter(abs(correlation) > 0.5)
```

```
##     rowname  variable correlation
## 1    INPC_E INPC_SERV   0.9504520
## 2   REMESAS INPC_SERV  -0.6702286
## 3   REMESAS    INPC_E  -0.6659018
## 4      IGAE      CONF   0.6126806
## 5       PIB        M1   0.5204366
## 6 DESEMPLEO       PIB  -0.6840906
## 7      IGAE       PIB   0.6724641
## 8      IGAE DESEMPLEO  -0.5749911
```

We will have to choose the variables that minimize the AIC further on.

1.3 Stationarity

With a standard dickey fuller test we checked for stationary and for all cases the test-statistic is smaller than the critical value at a 99% confidence level. This means that all variables are stationary. This was expected given that all variables are variations and given the time series graphs analyzed before.

```r
CV.DF=ur.df(CV, type="trend",lags=0)
summary(CV.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
```

```
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.32841 -0.18122 -0.02495  0.19757  0.74448
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.843651   0.220195   3.831 0.000307 ***
## z.lag.1     -0.322291   0.060169  -5.356 1.41e-06 ***
## tt           0.003095   0.002426   1.276 0.206945
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3413 on 60 degrees of freedom
## Multiple R-squared:  0.3693, Adjusted R-squared:  0.3483
## F-statistic: 17.57 on 2 and 60 DF,  p-value: 9.857e-07
##
##
## Value of test-statistic is: -5.3564 12.1221 17.5696
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
INPC_AB.DF=ur.df(INPC_AB, type = "trend", lags = 0)
summary(INPC_AB.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.3780 -0.9275  0.1065  0.8337  3.6257
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.815148   0.446123   4.069  0.00014 ***
## z.lag.1     -1.182395   0.126841  -9.322 2.85e-13 ***
## tt          -0.006747   0.011011  -0.613  0.54233
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.588 on 60 degrees of freedom
## Multiple R-squared:  0.5917, Adjusted R-squared:  0.5781
## F-statistic: 43.47 on 2 and 60 DF,  p-value: 2.141e-12
##
##
## Value of test-statistic is: -9.3218 28.9792 43.4687
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
INPC_SERV.DF=ur.df(INPC_SERV, type = "trend", lags = 0)
summary(INPC_SERV.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -4.498 -1.119 -0.122  2.274  4.989
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.944304   0.706587   1.336    0.186
## z.lag.1     -0.996823   0.127556  -7.815 1.01e-10 ***
## tt          -0.003449   0.019023  -0.181    0.857
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.746 on 60 degrees of freedom
## Multiple R-squared:  0.5046, Adjusted R-squared:  0.4881
## F-statistic: 30.56 on 2 and 60 DF,  p-value: 7.039e-10
##
##
## Value of test-statistic is: -7.8148 20.3845 30.5627
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
INPC_T.DF=ur.df(INPC_T, type = "trend", lags = 0)
summary(INPC_T.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2712 -0.5992 -0.0441  0.3318  6.3139
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.123528   0.381854   2.942  0.00463 **
## z.lag.1     -0.899208   0.129436  -6.947 3.08e-09 ***
## tt           0.000113   0.009357   0.012  0.99040
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.35 on 60 degrees of freedom
## Multiple R-squared:  0.4459, Adjusted R-squared:  0.4275
## F-statistic: 24.14 on 2 and 60 DF,  p-value: 2.028e-08
##
##
## Value of test-statistic is: -6.9471 16.1038 24.1445
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
INPC_E.DF=ur.df(INPC_E, type = "trend", lags = 0)
summary(INPC_E.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.4250  -4.7732   0.0707   5.0357  12.1612
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.518831   1.723935   0.881    0.382
## z.lag.1     -0.973297   0.129757  -7.501 3.49e-10 ***
## tt           0.008326   0.046609   0.179    0.859
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.726 on 60 degrees of freedom
## Multiple R-squared:  0.4839, Adjusted R-squared:  0.4667
## F-statistic: 28.13 on 2 and 60 DF,  p-value: 2.404e-09
##
##
## Value of test-statistic is: -7.5009 18.7968 28.1331
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
IPV.DF=ur.df(IPV, type = "trend", lags = 0)
summary(IPV.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.64677 -0.54579 -0.05738  0.37890  1.93287
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.751867   0.252184   2.981  0.00414 **
## z.lag.1     -0.649245   0.115106  -5.640 4.85e-07 ***
## tt           0.007599   0.005270   1.442  0.15455
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7572 on 60 degrees of freedom
## Multiple R-squared:  0.3523, Adjusted R-squared:  0.3307
```

```
## F-statistic: 16.32 on 2 and 60 DF,  p-value: 2.196e-06
##
##
## Value of test-statistic is: -5.6404 10.8822 16.3165
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
IPC_SUB.DF=ur.df(IPC_SUB, type = "trend", lags = 0)
summary(IPC_SUB.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.56079 -0.19460 -0.01208  0.16223  1.02503
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.7674344  0.1423595   5.391 1.24e-06 ***
## z.lag.1     -0.8179329  0.1280369  -6.388 2.74e-08 ***
## tt          -0.0008094  0.0021107  -0.383    0.703
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3038 on 60 degrees of freedom
## Multiple R-squared:  0.4049, Adjusted R-squared:  0.385
## F-statistic: 20.41 on 2 and 60 DF,  p-value: 1.731e-07
##
##
## Value of test-statistic is: -6.3883 13.6318 20.4091
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
REMESAS.DF=ur.df(REMESAS, type = "trend", lags = 0)
summary(REMESAS.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.731  -6.546  -2.088   4.768  20.987
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.19551    2.42641   0.081    0.936
## z.lag.1     -1.25566    0.11717 -10.716 1.46e-15 ***
## tt           0.05333    0.06578   0.811    0.421
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.492 on 60 degrees of freedom
## Multiple R-squared:  0.6574, Adjusted R-squared:  0.646
## F-statistic: 57.58 on 2 and 60 DF,  p-value: 1.102e-14
##
##
## Value of test-statistic is: -10.7163 38.4297 57.5757
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
INT.DF=ur.df(INT, type = "trend", lags = 0)
summary(INT.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6971 -0.5969  0.1830  0.5845  4.5021
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.64440    0.45846  -1.406    0.165
## z.lag.1     -0.72530    0.12564  -5.773 2.94e-07 ***
## tt           0.01031    0.01223   0.842    0.403
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.759 on 60 degrees of freedom
## Multiple R-squared:  0.3579, Adjusted R-squared:  0.3365
## F-statistic: 16.72 on 2 and 60 DF,  p-value: 1.69e-06
##
##
## Value of test-statistic is: -5.7726 11.1566 16.7219
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
CONF.DF=ur.df(CONF, type = "trend", lags = 0)
summary(CONF.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.8695  -2.1858   0.0963   2.9814  13.9831
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.64097    1.38775  -0.462    0.646
## z.lag.1     -1.02442    0.12994  -7.884 7.74e-11 ***
## tt           0.02469    0.03773   0.655    0.515
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.433 on 60 degrees of freedom
## Multiple R-squared:  0.5089, Adjusted R-squared:  0.4925
## F-statistic: 31.08 on 2 and 60 DF,  p-value: 5.446e-10
##
##
## Value of test-statistic is: -7.8836 20.7378 31.0828
##
```

```
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
M1.DF=ur.df(M1, type = "trend", lags = 0)
summary(M1.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.508 -3.995 -1.376  3.492 14.631
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.911621   1.417883   3.464 0.000988 ***
## z.lag.1     -1.520906   0.110307 -13.788  < 2e-16 ***
## tt          -0.005347   0.037284  -0.143 0.886438
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.381 on 60 degrees of freedom
## Multiple R-squared:  0.7601, Adjusted R-squared:  0.7521
## F-statistic: 95.05 on 2 and 60 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -13.7879 63.3718 95.0541
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
DEBT.DF=ur.df(DEBT, type = "trend", lags = 0)
summary(DEBT.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
```

```
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.877  -4.220   0.516   3.734  40.187
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.04461    2.38858   1.275    0.207
## z.lag.1     -1.01239    0.12792  -7.914 6.86e-11 ***
## tt          -0.06282    0.06433  -0.977    0.333
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.19 on 60 degrees of freedom
## Multiple R-squared:  0.5108, Adjusted R-squared:  0.4945
## F-statistic: 31.33 on 2 and 60 DF,  p-value: 4.823e-10
##
##
## Value of test-statistic is: -7.9144 20.9092 31.3306
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
EX.DF=ur.df(EX, type = "trend", lags = 0)
summary(EX.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.165  -3.030  -1.143   2.591  25.596
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.741753   1.433021   0.518    0.607
## z.lag.1     -0.909574   0.128570  -7.075 1.87e-09 ***
## tt           0.008888   0.038962   0.228    0.820
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.613 on 60 degrees of freedom
## Multiple R-squared:  0.455,  Adjusted R-squared:  0.4368
## F-statistic: 25.04 on 2 and 60 DF,  p-value: 1.237e-08
##
##
## Value of test-statistic is: -7.0746 16.6965 25.0446
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```
PIB.DF=ur.df(PIB, type="trend",lags=0)
summary(PIB.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.109  -1.579   0.345   1.894  12.545
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.07153    0.99482   1.077    0.286
## z.lag.1     -1.47277    0.11319 -13.012   <2e-16 ***
## tt          -0.01174    0.02692  -0.436    0.664
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.882 on 60 degrees of freedom
## Multiple R-squared:  0.7383, Adjusted R-squared:  0.7296
## F-statistic: 84.65 on 2 and 60 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -13.0117 56.4653 84.6537
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
DESEMPLEO.DF=ur.df(DESEMPLEO, type = "trend", lags = 0)
summary(DESEMPLEO.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.192  -7.071  -0.982   5.104  50.710
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.33016    2.69020   0.866    0.390
## z.lag.1     -1.40847    0.11690 -12.049   <2e-16 ***
## tt          -0.03506    0.07302  -0.480    0.633
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.54 on 60 degrees of freedom
## Multiple R-squared:  0.7077, Adjusted R-squared:  0.698
## F-statistic: 72.63 on 2 and 60 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -12.049 48.425 72.6348
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

```r
IGAE.DF=ur.df(IGAE, type = "trend", lags = 0)
summary(IGAE.DF)
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
## lm(formula = z.diff ~ z.lag.1 + 1 + tt)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.9771  -0.3148   0.2031   0.6921   7.1577
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.770689   0.569008   1.354    0.181
## z.lag.1     -1.087418   0.129427  -8.402 1.01e-11 ***
## tt          -0.009125   0.015301  -0.596    0.553
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.2 on 60 degrees of freedom
## Multiple R-squared:  0.5406, Adjusted R-squared:  0.5253
## F-statistic:  35.3 on 2 and 60 DF,  p-value: 7.34e-11
##
##
## Value of test-statistic is: -8.4018 23.5411 35.3027
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -4.04 -3.45 -3.15
## phi2  6.50  4.88  4.16
## phi3  8.73  6.49  5.47
```

2 Estimating lags

Given that each variable may have a different lagged effect on the default rate (CV) we perform individual ARIMAs of each stationary variable with CV with different *lags*, to determine which amount of lags gives us the minimum AIC. We do not, however, compute an ARIMA with zero lags, because one would not have the necessary inputs to run the ARIMA and forecast. Thus the comparison below is between: AIC with 1 lags, with 2 lags and with 3 lags.

We chose the right amount of lags for each case, minimizing AIC.

```
# INPC_AB
INPC_AB_v<-as.vector(INPC_AB)
CV_v<-as.vector(CV)
INPC_AB_v2<-cbind(INPC_AB_v,CV_v)
colnames(INPC_AB_v2)<-c("INPC_AB","CV")
a<- lag(INPC_AB_v,0)
x<- lag(INPC_AB_v,1)
y<- lag(INPC_AB_v,2)
z<- lag(INPC_AB_v,3)
INPC_AB_lags <- cbind(x,y,z)

fitINPC_AB1 <- auto.arima(INPC_AB_v2[4: 63,2], xreg=INPC_AB_lags[4: 63,1], d=0)
fitINPC_AB2 <- auto.arima(INPC_AB_v2[4: 63,2], xreg=INPC_AB_lags[4: 63,1:2], d=0)
fitINPC_AB3 <- auto.arima(INPC_AB_v2[4: 63,2], xreg=INPC_AB_lags[4: 63,1:3], d=0)
AIC_INPC_AB <- cbind(fitINPC_AB1$aic,fitINPC_AB2$aic,fitINPC_AB3$aic)
colnames(AIC_INPC_AB)<-c("1 lag","2 lags", "3 lags")
```

```r
# INPC_SERV
INPC_SERV_v<-as.vector(INPC_SERV)
CV_v<-as.vector(CV)
INPC_SERV_v2<-cbind(INPC_SERV_v,CV_v)
colnames(INPC_SERV_v2)<-c("INPC_SERV","CV")
a<- lag(INPC_SERV_v,0)
x<- lag(INPC_SERV_v,1)
y<- lag(INPC_SERV_v,2)
z<- lag(INPC_SERV_v,3)
INPC_SERV_lags <- cbind(x,y,z)
fitINPC_SERV1 <- auto.arima(INPC_SERV_v2[4: 63,2], xreg=INPC_SERV_lags[4: 63,1], d=0)
fitINPC_SERV2 <- auto.arima(INPC_SERV_v2[4: 63,2], xreg=INPC_SERV_lags[4: 63,1:2], d=0)
fitINPC_SERV3 <- auto.arima(INPC_SERV_v2[4: 63,2], xreg=INPC_SERV_lags[4: 63,1:3], d=0)
AIC_INPC_SERV <- cbind(fitINPC_SERV1$aic,fitINPC_SERV2$aic,fitINPC_SERV3$aic)
colnames(AIC_INPC_SERV)<-c("1 lag","2 lags", "3 lags")

# INPC_T
INPC_T_v<-as.vector(INPC_T)
CV_v<-as.vector(CV)
INPC_T_v2<-cbind(INPC_T_v,CV_v)
colnames(INPC_T_v2)<-c("INPC_T","CV")
a<- lag(INPC_T_v,0)
x<- lag(INPC_T_v,1)
y<- lag(INPC_T_v,2)
z<- lag(INPC_T_v,3)
INPC_T_lags <- cbind(x,y,z)
fitINPC_T1 <- auto.arima(INPC_T_v2[4: 63,2], xreg=INPC_T_lags[4: 63,1], d=0)
fitINPC_T2 <- auto.arima(INPC_T_v2[4: 63,2], xreg=INPC_T_lags[4: 63,1:2], d=0)
fitINPC_T3 <- auto.arima(INPC_T_v2[4: 63,2], xreg=INPC_T_lags[4: 63,1:3], d=0)
AIC_INPC_T <- cbind(fitINPC_T1$aic,fitINPC_T2$aic,fitINPC_T3$aic)
colnames(AIC_INPC_T)<-c("1 lag","2 lags", "3 lags")

# INPC_E
INPC_E_v<-as.vector(INPC_E)
CV_v<-as.vector(CV)
INPC_E_v2<-cbind(INPC_E_v,CV_v)
colnames(INPC_E_v2)<-c("INPC_E","CV")
a<- lag(INPC_E_v,0)
x<- lag(INPC_E_v,1)
y<- lag(INPC_E_v,2)
z<- lag(INPC_E_v,3)
INPC_E_lags <- cbind(x,y,z)
fitINPC_E1 <- auto.arima(INPC_E_v2[4: 63,2], xreg=INPC_E_lags[4: 63,1], d=0)
fitINPC_E2 <- auto.arima(INPC_E_v2[4: 63,2], xreg=INPC_E_lags[4: 63,1:2], d=0)
fitINPC_E3 <- auto.arima(INPC_E_v2[4: 63,2], xreg=INPC_E_lags[4: 63,1:3], d=0)
AIC_INPC_E <- cbind(fitINPC_E1$aic,fitINPC_E2$aic,fitINPC_E3$aic)
colnames(AIC_INPC_E)<-c("1 lag","2 lags", "3 lags")

# IPV
IPV_v<-as.vector(IPV)
CV_v<-as.vector(CV)
IPV_v2<-cbind(IPV_v,CV_v)
colnames(IPV_v2)<-c("IPV","CV")
```

```r
a<- lag(IPV_v,0)
x<- lag(IPV_v,1)
y<- lag(IPV_v,2)
z<- lag(IPV_v,3)
IPV_lags <- cbind(x,y,z)
fitIPV1 <- auto.arima(IPV_v2[4: 63,2], xreg=IPV_lags[4: 63,1], d=0)
fitIPV2 <- auto.arima(IPV_v2[4: 63,2], xreg=IPV_lags[4: 63,1:2], d=0)
fitIPV3 <- auto.arima(IPV_v2[4: 63,2], xreg=IPV_lags[4: 63,1:3], d=0)
AIC_IPV <- cbind(fitIPV1$aic,fitIPV2$aic,fitIPV3$aic)
colnames(AIC_IPV)<-c("1 lag","2 lags", "3 lags")

# IPC_SUB
IPC_SUB_v<-as.vector(IPC_SUB)
CV_v<-as.vector(CV)
IPC_SUB_v2<-cbind(IPC_SUB_v,CV_v)
colnames(IPC_SUB_v2)<-c("IPC_SUB","CV")
a<- lag(IPC_SUB_v,0)
x<- lag(IPC_SUB_v,1)
y<- lag(IPC_SUB_v,2)
z<- lag(IPC_SUB_v,3)
IPC_SUB_lags <- cbind(x,y,z)
fitIPC_SUB1 <- auto.arima(IPC_SUB_v2[4: 63,2], xreg=IPC_SUB_lags[4: 63,1], d=0)
fitIPC_SUB2 <- auto.arima(IPC_SUB_v2[4: 63,2], xreg=IPC_SUB_lags[4: 63,1:2], d=0)
fitIPC_SUB3 <- auto.arima(IPC_SUB_v2[4: 63,2], xreg=IPC_SUB_lags[4: 63,1:3], d=0)
AIC_IPC_SUB <- cbind(fitIPC_SUB1$aic,fitIPC_SUB2$aic,fitIPC_SUB3$aic)
colnames(AIC_IPC_SUB)<-c("1 lag","2 lags", "3 lags")

# REMESAS
REMESAS_v<-as.vector(REMESAS)
CV_v<-as.vector(CV)
REMESAS_v2<-cbind(REMESAS_v,CV_v)
colnames(REMESAS_v2)<-c("REMESAS","CV")
a<- lag(REMESAS_v,0)
x<- lag(REMESAS_v,1)
y<- lag(REMESAS_v,2)
z<- lag(REMESAS_v,3)
REMESAS_lags <- cbind(x,y,z)
fitREMESAS1 <- auto.arima(REMESAS_v2[4: 63,2], xreg=REMESAS_lags[4: 63,1], d=0)
fitREMESAS2 <- auto.arima(REMESAS_v2[4: 63,2], xreg=REMESAS_lags[4: 63,1:2], d=0)
fitREMESAS3 <- auto.arima(REMESAS_v2[4: 63,2], xreg=REMESAS_lags[4: 63,1:3], d=0)
AIC_REMESAS <- cbind(fitREMESAS1$aic,fitREMESAS2$aic,fitREMESAS3$aic)
colnames(AIC_REMESAS)<-c("1 lag","2 lags", "3 lags")

# INT
INT_v<-as.vector(INT)
CV_v<-as.vector(CV)
INT_v2<-cbind(INT_v,CV_v)
colnames(INT_v2)<-c("INT","CV")
a<- lag(INT_v,0)
x<- lag(INT_v,1)
y<- lag(INT_v,2)
z<- lag(INT_v,3)
INT_lags <- cbind(x,y,z)
```

```r
fitINT1 <- auto.arima(INT_v2[4: 63,2], xreg=INT_lags[4: 63,1], d=0)
fitINT2 <- auto.arima(INT_v2[4: 63,2], xreg=INT_lags[4: 63,1:2], d=0)
fitINT3 <- auto.arima(INT_v2[4: 63,2], xreg=INT_lags[4: 63,1:3], d=0)
AIC_INT <- cbind(fitINT1$aic,fitINT2$aic,fitINT3$aic)
colnames(AIC_INT)<-c("1 lag","2 lags", "3 lags")

# CONF
CONF_v<-as.vector(CONF)
CV_v<-as.vector(CV)
CONF_v2<-cbind(CONF_v,CV_v)
colnames(CONF_v2)<-c("CONF","CV")
a<- lag(CONF_v,0)
x<- lag(CONF_v,1)
y<- lag(CONF_v,2)
z<- lag(CONF_v,3)
CONF_lags <- cbind(x,y,z)
fitCONF1 <- auto.arima(CONF_v2[4: 63,2], xreg=CONF_lags[4: 63,1], d=0)
fitCONF2 <- auto.arima(CONF_v2[4: 63,2], xreg=CONF_lags[4: 63,1:2], d=0)
fitCONF3 <- auto.arima(CONF_v2[4: 63,2], xreg=CONF_lags[4: 63,1:3], d=0)
AIC_CONF <- cbind(fitCONF1$aic,fitCONF2$aic,fitCONF3$aic)
colnames(AIC_CONF)<-c("1 lag","2 lags", "3 lags")

# M1
M1_v<-as.vector(M1)
CV_v<-as.vector(CV)
M1_v2<-cbind(M1_v,CV_v)
colnames(M1_v2)<-c("M1","CV")
a<- lag(M1_v,0)
x<- lag(M1_v,1)
y<- lag(M1_v,2)
z<- lag(M1_v,3)
M1_lags <- cbind(x,y,z)
fitM11 <- auto.arima(M1_v2[4: 63,2], xreg=M1_lags[4: 63,1], d=0)
fitM12 <- auto.arima(M1_v2[4: 63,2], xreg=M1_lags[4: 63,1:2], d=0)
fitM13 <- auto.arima(M1_v2[4: 63,2], xreg=M1_lags[4: 63,1:3], d=0)
AIC_M1 <- cbind(fitM11$aic,fitM12$aic,fitM13$aic)
colnames(AIC_M1)<-c("1 lag","2 lags", "3 lags")

# DEBT
DEBT_v<-as.vector(DEBT)
CV_v<-as.vector(CV)
DEBT_v2<-cbind(DEBT_v,CV_v)
colnames(DEBT_v2)<-c("DEBT","CV")
a<- lag(DEBT_v,0)
x<- lag(DEBT_v,1)
y<- lag(DEBT_v,2)
z<- lag(DEBT_v,3)
DEBT_lags <- cbind(x,y,z)
fitDEBT1 <- auto.arima(DEBT_v2[4: 63,2], xreg=DEBT_lags[4: 63,1], d=0)
fitDEBT2 <- auto.arima(DEBT_v2[4: 63,2], xreg=DEBT_lags[4: 63,1:2], d=0)
fitDEBT3 <- auto.arima(DEBT_v2[4: 63,2], xreg=DEBT_lags[4: 63,1:3], d=0)
AIC_DEBT <- cbind(fitDEBT1$aic,fitDEBT2$aic,fitDEBT3$aic)
colnames(AIC_DEBT)<-c("1 lag","2 lags", "3 lags")
```

```r
# EX
EX_v<-as.vector(EX)
CV_v<-as.vector(CV)
EX_v2<-cbind(EX_v,CV_v)
colnames(EX_v2)<-c("EX","CV")
a<- lag(EX_v,0)
x<- lag(EX_v,1)
y<- lag(EX_v,2)
z<- lag(EX_v,3)
EX_lags <- cbind(x,y,z)
fitEX1 <- auto.arima(EX_v2[4: 63,2], xreg=EX_lags[4: 63,1], d=0)
fitEX2 <- auto.arima(EX_v2[4: 63,2], xreg=EX_lags[4: 63,1:2], d=0)
fitEX3 <- auto.arima(EX_v2[4: 63,2], xreg=EX_lags[4: 63,1:3], d=0)
AIC_EX <- cbind(fitEX1$aic,fitEX2$aic,fitEX3$aic)
colnames(AIC_EX)<-c("1 lag","2 lags", "3 lags")

# PIB
PIB_v<-as.vector(PIB)
CV_v<-as.vector(CV)
PIB_v2<-cbind(PIB_v,CV_v)
colnames(PIB_v2)<-c("PIB","CV")
a<- lag(PIB_v,0)
x<- lag(PIB_v,1)
y<- lag(PIB_v,2)
z<- lag(PIB_v,3)
PIB_lags <- cbind(x,y,z)
fitPIB1 <- auto.arima(PIB_v2[4: 63,2], xreg=PIB_lags[4: 63,1], d=0)
fitPIB2 <- auto.arima(PIB_v2[4: 63,2], xreg=PIB_lags[4: 63,1:2], d=0)
fitPIB3 <- auto.arima(PIB_v2[4: 63,2], xreg=PIB_lags[4: 63,1:3], d=0)
AIC_PIB <- cbind(fitPIB1$aic,fitPIB2$aic,fitPIB3$aic)
colnames(AIC_PIB)<-c("1 lag","2 lags", "3 lags")

# DESEMPLEO
DESEMPLEO_v<-as.vector(DESEMPLEO)
CV_v<-as.vector(CV)
DESEMPLEO_v2<-cbind(DESEMPLEO_v,CV_v)
colnames(DESEMPLEO_v2)<-c("DESEMPLEO","CV")
a<- lag(DESEMPLEO_v,0)
x<- lag(DESEMPLEO_v,1)
y<- lag(DESEMPLEO_v,2)
z<- lag(DESEMPLEO_v,3)
DESEMPLEO_lags <- cbind(x,y,z)
fitDESEMPLEO1 <- auto.arima(DESEMPLEO_v2[4: 63,2], xreg=DESEMPLEO_lags[4: 63,1], d=0)
fitDESEMPLEO2 <- auto.arima(DESEMPLEO_v2[4: 63,2], xreg=DESEMPLEO_lags[4: 63,1:2], d=0)
fitDESEMPLEO3 <- auto.arima(DESEMPLEO_v2[4: 63,2], xreg=DESEMPLEO_lags[4: 63,1:3], d=0)
AIC_DESEMPLEO <- cbind(fitDESEMPLEO1$aic,fitDESEMPLEO2$aic,fitDESEMPLEO3$aic)
colnames(AIC_DESEMPLEO)<-c("1 lag","2 lags", "3 lags")

# IGAE
IGAE_v<-as.vector(IGAE)
CV_v<-as.vector(CV)
IGAE_v2<-cbind(IGAE_v,CV_v)
colnames(IGAE_v2)<-c("IGAE","CV")
```

```
a<- lag(IGAE_v,0)
x<- lag(IGAE_v,1)
y<- lag(IGAE_v,2)
z<- lag(IGAE_v,3)
IGAE_lags <- cbind(x,y,z)
fitIGAE1 <- auto.arima(IGAE_v2[4: 63,2], xreg=IGAE_lags[4: 63,1], d=0)
fitIGAE2 <- auto.arima(IGAE_v2[4: 63,2], xreg=IGAE_lags[4: 63,1:2], d=0)
fitIGAE3 <- auto.arima(IGAE_v2[4: 63,2], xreg=IGAE_lags[4: 63,1:3], d=0)
AIC_IGAE <- cbind(fitIGAE1$aic,fitIGAE2$aic,fitIGAE3$aic)
colnames(AIC_IGAE)<-c("1 lag","2 lags", "3 lags")
```

```
AICs<-rbind(AIC_INPC_AB, AIC_INPC_SERV, AIC_INPC_T, AIC_INPC_E, AIC_IPV, AIC_IPC_SUB, AIC_REMESAS, AIC_I
rownames(AICs)<-c("INPC_AB", "INPC_SERV", "INPC_T", "INPC_E", "IPV", "IPC_SUB", "REMESAS", "INT", "CONF
AICs
```

```
##                1 lag    2 lags    3 lags
## INPC_AB   -44.34416 -42.35572 -44.86912
## INPC_SERV -40.94426 -41.47433 -40.05878
## INPC_T    -38.46759 -36.51221 -34.56707
## INPC_E    -40.69909 -40.30348 -38.65377
## IPV       -38.30869 -37.48287 -35.61254
## IPC_SUB   -38.32896 -40.77693 -41.65838
## REMESAS   -42.09516 -40.11344 -44.56620
## INT       -38.31281 -37.70656 -36.44092
## CONF      -38.26051 -36.45590 -37.13657
## M1        -39.03681 -39.52251 -41.54306
## DEBT      -38.28492 -38.46186 -36.58643
## EX        -38.38692 -36.72289 -34.77319
## PIB       -38.43160 -38.84621 -38.07924
## DESEMPLEO -38.46382 -36.65742 -34.66384
## IGAE      -38.38967 -36.41858 -36.61604
```

Analyzing the results above, we created a new excel file (called BDF2) where for each variable we included:

1. The variable with 1 lag
2. The subsequent lagged variables until the minimum AIC is reached

For example, for INPC_AB, the number of lags that minimizes AIC is three. In this case, we include INPC_AB (1 lag), INPC_AB_2 (2 lags) and INPC_AB_3 (3 lags). Like so:

```
BDF2 <- read_excel("BDF NACIONAL LAGS.xlsx")
BDF2 <- ts(BDF2, start = c(2006,1),frequency = 4)
head(BDF2)
```

```
##               CV    INPC_AB  INPC_AB_2  INPC_AB_3  INPC_SERV INPC_SERV_2
## 2006 Q1 2.199177  0.4220444  0.8525432  0.7018551  5.0026781   1.0491358
## 2006 Q2 2.066763  0.5030827  0.4220444  0.8525432  1.0508060   5.0026781
## 2006 Q3 1.986694 -0.6987303  0.5030827  0.4220444 -2.1718324   1.0508060
## 2006 Q4 2.003475  5.5345808 -0.6987303  0.5030827  0.7797826  -2.1718324
## 2007 Q1 2.257972  1.5850926  5.5345808 -0.6987303  3.9809778   0.7797826
## 2007 Q2 2.447487  0.9136926  1.5850926  5.5345808  0.6306750   3.9809778
```

```
##            INPC_T     INPC_E        IPV   IPC_SUB IPC_SUB_2 IPC_SUB_3   REMESAS
## 2006 Q1 0.9256332 12.2117117 -0.2152642 0.7288090 0.8263839 0.6277530 -1.800023
## 2006 Q2 1.2841778  1.4290805  2.1572857 0.9439550 0.7288090 0.8263839  0.932139
## 2006 Q3 1.2125805 -6.7257009  2.6108658 0.7785749 0.9439550 0.7288090 21.157327
## 2006 Q4 0.8662647  1.0630173  1.5154350 0.9826554 0.7785749 0.9439550 -4.039891
## 2007 Q1 0.5247664  9.2075874  0.5160339 0.8913585 0.9826554 0.7785749 -6.732341
## 2007 Q2 1.1675579  0.4696561  2.6219289 1.1435721 0.8913585 0.9826554 -4.853932
##          REMESAS_2  REMESAS_3        INT      CONF        M1       M1_2
## 2006 Q1  0.9008468 27.7734026 -1.6812865  4.768270 15.524386 -0.5757901
## 2006 Q2 -1.8000233  0.9008468 -3.1970260  2.897367 -3.295169 15.5243856
## 2006 Q3  0.9321390 -1.8000233 -0.8704557 -1.492325  6.187460 -3.2951687
## 2006 Q4 21.1573268  0.9321390 -0.8522727  1.383832 -2.067644  6.1874601
## 2007 Q1 -4.0398914 21.1573268 -0.6251628 -1.650998 13.407490 -2.0676441
## 2007 Q2 -6.7323410 -4.0398914 -0.3407602 -2.555097 -6.039887 13.4074901
##               M1_3      DEBT    DEBT_2       PIB     PIB_2  DESEMPLEO      IGAE
## 2006 Q1  4.4462848 -8.226283  5.782404  3.804774 -2.287794 -17.483354 1.7121746
## 2006 Q2 -0.5757901  7.002603 -8.226283 -1.387925  3.804774  13.107616 0.7367485
## 2006 Q3 15.5243856 17.777689  7.002603  4.200068 -1.387925 -11.208719 1.1990516
## 2006 Q4 -3.2951687  6.294740 17.777689 -1.800185  4.200068  27.269522 0.4334371
## 2007 Q1  6.1874601  7.173895  6.294740  2.528630 -1.800185  -9.953840 0.4903945
## 2007 Q2 -2.0676441  9.223489  7.173895 -2.750982  2.528630   9.819431 0.3878567
##                  EX
## 2006 Q1 -0.04065695
## 2006 Q2 -1.04508266
## 2006 Q3  5.53210143
## 2006 Q4 -2.13212457
## 2007 Q1 -0.51285299
## 2007 Q2  1.21731087
```

3 Models & hypothesis

Now, with all variables stationary and with the right amount of lags, we test *different combinations of variables*, to remove those that do not contribute to the minimization of AIC. (Note that the significance of the coefficients is not very relevant for the forecast model and for the scope of this investigation)

We did this in accordance to the next hypothesis and in line with what literature in similar studies has done:

3.1 Hypothesis

3.1.1 Borrower's ability to pay

(explain)

3.1.2 Willingness of consumers to pay their mortgage

(explain)

3.1.3 Confidence and cost indicators for home purchase

(explain)

```
training_set<-ts(BDF2[1:59,],start = c(2006,1),frequency=4)
```

3.2 Testing models

H1 23:24 PIB 26: IGAE

H2 2:7 INPC_AB a T 8: INPC_E 10:12 IPC_SUB

CONSTANT 13:15 REMESAS 9: IPV 25: DESEMPLEO

VARIABLE 18:20 M1 17: CONF 27: EX 21:22 DEBT 16: INT

Model 1: 25,13:15,9, 23:24, 2:7 + variable Model 2: 25,13:15,9, 23:24, 8,10:12 + variable Model 3: 25,13:15,9, 26, 2:7 + variable Model 4: 25,13:15,9, 26, 8,10:12 + variable

```
# Modelo 1
modelo_1.1 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 17, 27, 21:22, 16)])
modelo_1.2 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 17, 27, 21:22)])
modelo_1.3 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 17, 27, 16)])
modelo_1.4 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 17, 27)])
modelo_1.5 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 17, 21:22, 16)])
modelo_1.6 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 17, 21:22)])
modelo_1.7 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 17, 16)])
modelo_1.8 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 17)])
modelo_1.9 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 27, 21:22, 16)])
modelo_1.10 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 27, 21:22)])
modelo_1.11 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 27, 16)])
modelo_1.12 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 27)])
modelo_1.13 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 21:22, 16)])
modelo_1.14 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 21:22)])
modelo_1.15 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20, 16)])
modelo_1.16 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 18:20)])
modelo_1.17 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 17, 27, 21:22, 16)])
modelo_1.18 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 17, 27, 21:22)])
modelo_1.19 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 17, 27, 16)])
```

```r
modelo_1.20 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 17, 27)])
modelo_1.21 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 17, 21:22, 16)])
modelo_1.22 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 17, 21:22)])
modelo_1.23 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 17, 16)])
modelo_1.24 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 17)])
modelo_1.25 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 27, 21:22, 16)])
modelo_1.26 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 27, 21:22)])
modelo_1.27 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 27, 16)])
modelo_1.28 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 27)])
modelo_1.29 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 21:22, 16)])
modelo_1.30 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 21:22)])
modelo_1.31 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7, 16)])
modelo_1.32 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  2:7)])
```

```r
# Modelo 2
modelo_2.1 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17, 27, 21:22, 16)])
modelo_2.2 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17, 27, 21:22)])
modelo_2.3 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17, 27, 16)])
modelo_2.4 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17, 27)])
modelo_2.5 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17, 21:22, 16)])
modelo_2.6 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17, 21:22)])
modelo_2.7 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17, 16)])
modelo_2.8 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17)])
modelo_2.9 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 27, 21:22, 16)])
modelo_2.10 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 27, 21:22)])
modelo_2.11 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 27, 16)])
modelo_2.12 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 27)])
modelo_2.13 <- auto.arima(training_set[,"CV"],
```

```r
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 21:22, 16)])
modelo_2.14 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 21:22)])
modelo_2.15 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 16)])
modelo_2.16 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 18:20)])
modelo_2.17 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 17, 27, 21:22, 16)])
modelo_2.18 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 17, 27, 21:22)])
modelo_2.19 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 17, 27, 16)])
modelo_2.20 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 17, 27)])
modelo_2.21 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 17, 21:22, 16)])
modelo_2.22 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 17, 21:22)])
modelo_2.23 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 17, 16)])
modelo_2.24 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 17)])
modelo_2.25 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 27, 21:22, 16)])
modelo_2.26 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 27, 21:22)])
modelo_2.27 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 27, 16)])
modelo_2.28 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 27)])
modelo_2.29 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 21:22, 16)])
modelo_2.30 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 21:22)])
modelo_2.31 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12, 16)])
modelo_2.32 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 23:24,  8, 10:12)])
```

```r
# Modelo 3
modelo_3.1 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 17, 27, 21:22, 16)])
modelo_3.2 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 17, 27, 21:22)])
modelo_3.3 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 17, 27, 16)])
modelo_3.4 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 17, 27)])
modelo_3.5 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 17, 21:22, 16)])
modelo_3.6 <- auto.arima(training_set[,"CV"],
                             xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 17, 21:22)])
```

```r
modelo_3.7 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 17, 16)])
modelo_3.8 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 17)])
modelo_3.9 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 27, 21:22, 16)])
modelo_3.10 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 27, 21:22)])
modelo_3.11 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 27, 16)])
modelo_3.12 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 27)])
modelo_3.13 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 21:22, 16)])
modelo_3.14 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 21:22)])
modelo_3.15 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20, 16)])
modelo_3.16 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 18:20)])
modelo_3.17 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 17, 27, 21:22, 16)])
modelo_3.18 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 17, 27, 21:22)])
modelo_3.19 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 17, 27, 16)])
modelo_3.20 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 17, 27)])
modelo_3.21 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 17, 21:22, 16)])
modelo_3.22 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 17, 21:22)])
modelo_3.23 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 17, 16)])
modelo_3.24 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 17)])
modelo_3.25 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 27, 21:22, 16)])
modelo_3.26 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 27, 21:22)])
modelo_3.27 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 27, 16)])
modelo_3.28 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 27)])
modelo_3.29 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 21:22, 16)])
modelo_3.30 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 21:22)])
modelo_3.31 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7, 16)])
modelo_3.32 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  2:7)])
```

```r
# Modelo 4
modelo_4.1 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 17, 27, 21:22, 16)])
modelo_4.2 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 17, 27, 21:22)])
modelo_4.3 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 17, 27, 16)])
modelo_4.4 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 17, 27)])
modelo_4.5 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 17, 21:22, 16)])
modelo_4.6 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 17, 21:22)])
modelo_4.7 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 17, 16)])
modelo_4.8 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 17)])
modelo_4.9 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 27, 21:22, 16)])
modelo_4.10 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 27, 21:22)])
modelo_4.11 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 27, 16)])
modelo_4.12 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 27)])
modelo_4.13 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 21:22, 16)])
modelo_4.14 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 21:22)])
modelo_4.15 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20, 16)])
modelo_4.16 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 18:20)])
modelo_4.17 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 17, 27, 21:22, 16)])
modelo_4.18 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 17, 27, 21:22)])
modelo_4.19 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 17, 27, 16)])
modelo_4.20 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 17, 27)])
modelo_4.21 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 17, 21:22, 16)])
modelo_4.22 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 17, 21:22)])
modelo_4.23 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 17, 16)])
modelo_4.24 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 17)])
modelo_4.25 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 27, 21:22, 16)])
modelo_4.26 <- auto.arima(training_set[,"CV"],
                         xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 27, 21:22)])
```

```r
modelo_4.27 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 27, 16)])
modelo_4.28 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 27)])
modelo_4.29 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 21:22, 16)])
modelo_4.30 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 21:22)])
modelo_4.31 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 26,  8, 10:12, 16)])
modelo_4.32 <- auto.arima(training_set[,"CV"],
                          xreg=training_set[,c(25,13:15,9, 26,  8, 10:12)])
```

```r
#AIC comparison
AICs_modelo1<-rbind(modelo_1.1$aic,modelo_1.2$aic,modelo_1.3$aic,modelo_1.4$aic,modelo_1.5$aic,modelo_1

AICs_modelo2<-rbind(modelo_2.1$aic,modelo_2.2$aic,modelo_2.3$aic,modelo_2.4$aic,modelo_2.5$aic,modelo_2

AICs_modelo3<-rbind(modelo_3.1$aic,modelo_3.2$aic,modelo_3.3$aic,modelo_3.4$aic,modelo_3.5$aic,modelo_3

AICs_modelo4<-rbind(modelo_4.1$aic,modelo_4.2$aic,modelo_4.3$aic,modelo_4.4$aic,modelo_4.5$aic,modelo_4

AICs_modelos<-cbind(AICs_modelo1,AICs_modelo2,AICs_modelo3,AICs_modelo4)
colnames(AICs_modelos)<-c("Modelo 1","Modelo 2", "Modelo 3", "Modelo 4")
rownames(AICs_modelos)<-c(1:32)

AICs_modelos
```

```
##     Modelo 1  Modelo 2  Modelo 3  Modelo 4
## 1  -51.96869 -52.24401 -53.48009 -50.32843
## 2  -53.44749 -53.98491 -54.67169 -51.09328
## 3  -51.14264 -55.47369 -51.47582 -53.21826
## 4  -53.11779 -62.52516 -52.15267 -54.05153
## 5  -53.96401 -54.18569 -55.44268 -52.02117
## 6  -55.43987 -55.97857 -56.62793 -53.08792
## 7  -53.05939 -57.45682 -53.46903 -54.31888
## 8  -55.02509 -64.52464 -54.14089 -55.80039
## 9  -53.52958 -49.57884 -55.22575 -50.50375
## 10 -54.78758 -51.55611 -56.34255 -51.43826
## 11 -51.44055 -51.01616 -53.27547 -53.13651
## 12 -52.33953 -53.00788 -53.79828 -54.16871
## 13 -55.52177 -51.26437 -57.18496 -51.93698
## 14 -56.77317 -53.26157 -58.29190 -53.32625
## 15 -53.41645 -52.39010 -55.27188 -53.63091
## 16 -54.33953 -54.17703 -55.77074 -55.41949
## 17 -50.49885 -49.47908 -52.71045 -52.11599
## 18 -52.16573 -51.46601 -54.38267 -53.86981
## 19 -51.12127 -53.75376 -49.82791 -53.02352
## 20 -53.10632 -55.59215 -52.91542 -54.96211
## 21 -51.83870 -50.97404 -53.03099 -52.86733
## 22 -53.26559 -52.97155 -54.59639 -54.84813
## 23 -52.92567 -52.30242 -51.04859 -54.76216
## 24 -54.91320 -54.13034 -54.67138 -56.75296
```

```
## 25 -52.43449 -50.94898 -54.66645 -48.49475
## 26 -54.14197 -52.91952 -56.36576 -50.32948
## 27 -50.50777 -51.63444 -51.74564 -51.91421
## 28 -52.86948 -53.36676 -52.38744 -53.91220
## 29 -53.74425 -52.63237 -54.84771 -50.27650
## 30 -55.19472 -54.63172 -56.45737 -51.84772
## 31 -52.06149 -53.63353 -52.89826 -53.85234
## 32 -54.84836 -55.32332 -53.68042 -55.85044
```

3.3 Best model

```
min(AICs_modelos)
```

```
## [1] -64.52464
```

The model with the lowest AIC is **model_2.8** with AIC = -64.52464. This model includes:

1. Borrower's ability to pay: (A) PIB instead of IGAE and (B) Unemployment and Remittances.
2. Willingness of consumers to pay their mortgage: (A) INPC_E and IPC_SUB instead of the disaggregated INPC and (B) M1.
3. Confidence and cost indicators for home purchase: IPV and consumer's confidence (CONF).

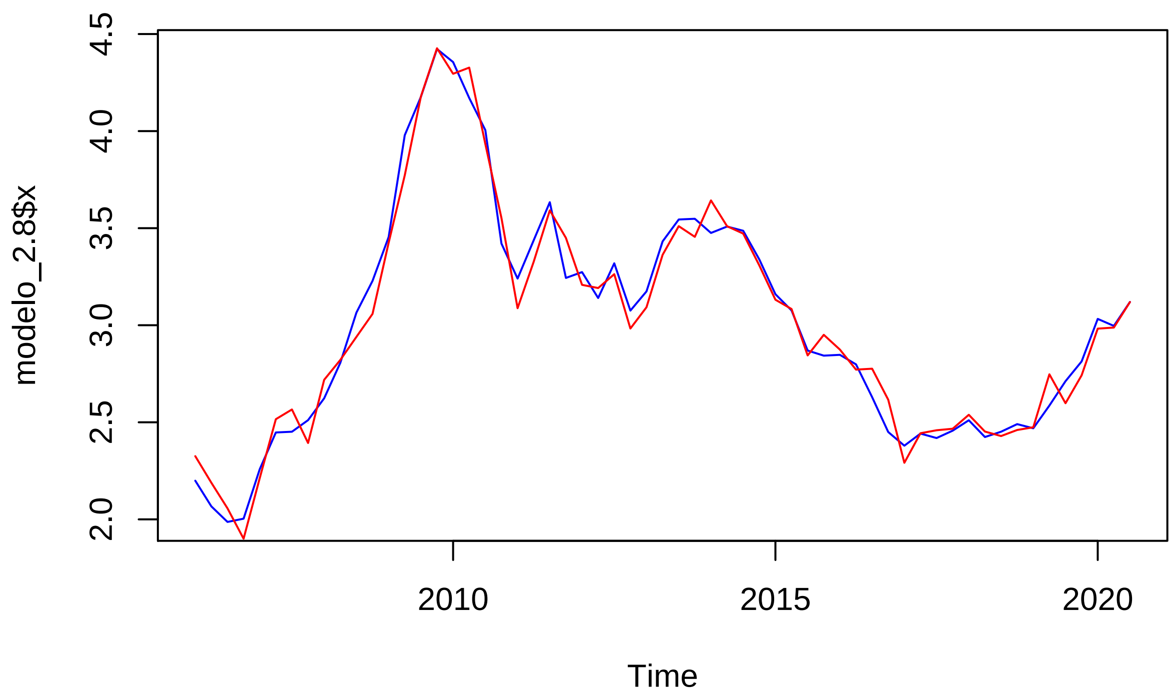
At the end, the best model includes: IGAE, DESEMPLEO, REMESAS, INPC_E, IPC_SUB, IPV and CONF. **A total of 8 regressors.** M1 ended up in our final model; **that alone is a relevant finding**.

The selected model is an ARIMA(2,0,3):

```
summary(modelo_2.8)
```

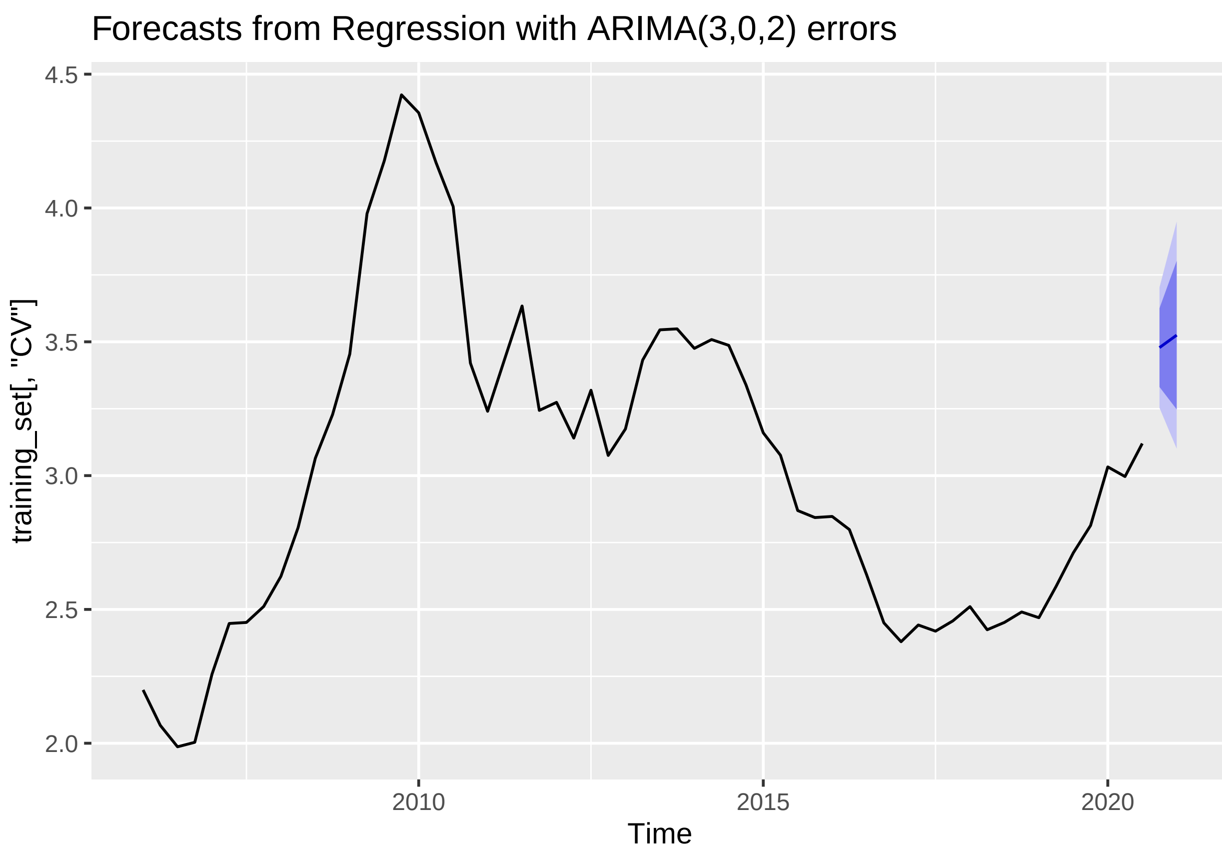
```
## Series: training_set[, "CV"]
## Regression with ARIMA(3,0,2) errors
##
## Coefficients:
##          ar1     ar2      ar3     ma1     ma2  intercept  DESEMPLEO  REMESAS
##       0.7214  0.6970  -0.5555  0.8796  0.4913     2.7034    -0.0014  -0.0003
## s.e.  0.2003  0.1619   0.1593  0.3230  0.2418     0.2424     0.0017   0.0019
##       REMESAS_2  REMESAS_3      IPV     PIB   PIB_2  INPC_E  IPC_SUB  IPC_SUB_2
##         -0.0023    -0.0094  -0.0913  0.0238  0.0066  0.0086   0.1134     0.1635
## s.e.     0.0017     0.0016   0.0222  0.0077  0.0082  0.0025   0.0723     0.0554
##       IPC_SUB_3       M1    M1_2    M1_3     CONF
##          0.2120  -0.0057  0.0042  0.0068  -0.0097
## s.e.     0.0573   0.0040  0.0044  0.0052   0.0027
##
## sigma^2 estimated as 0.01315:  log likelihood=54.26
## AIC=-64.52   AICc=-36.41   BIC=-18.82
##
## Training set error measures:
##                      ME       RMSE        MAE         MPE     MAPE      MASE
## Training set 0.00301571 0.09204153 0.07288061 -0.04339445 2.518327 0.1864579
##                     ACF1
## Training set -0.02899452
```

```r
plot(modelo_2.8$x,col="blue")
lines(fitted(modelo_2.8),col="red")
```



Forecast for last 2 quarters:

```r
test_set1<-as.matrix(BDF2[60:61,c(25,13:15,9, 23:24,  8, 10:12, 18:20, 17)])
test_set<-t(test_set1)

library("forecast")
forecast_cv<-forecast(modelo_2.8,xreg=test_set1)
autoplot(forecast_cv)
```




```
forecast_cv
```

```
##         Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2020 Q4       3.478257 3.331278 3.625235 3.253472 3.703041
## 2021 Q1       3.524858 3.247420 3.802296 3.100553 3.949163
```

```
forecast_2<-c("3.478257","3.524858")
comparison<-as.data.frame(cbind(tail(DATA.ts[63:64,"CV"]),forecast_2))
colnames(comparison)<-c("Actual CV","Forecasted CV")
rownames(comparison)<-c("2020 Q4","2021 Q1")
comparison
```

```
##                Actual CV Forecasted CV
## 2020 Q4 3.27408756441129      3.478257
## 2021 Q1 3.53406391042347      3.524858
```

3.4 Data Analysis

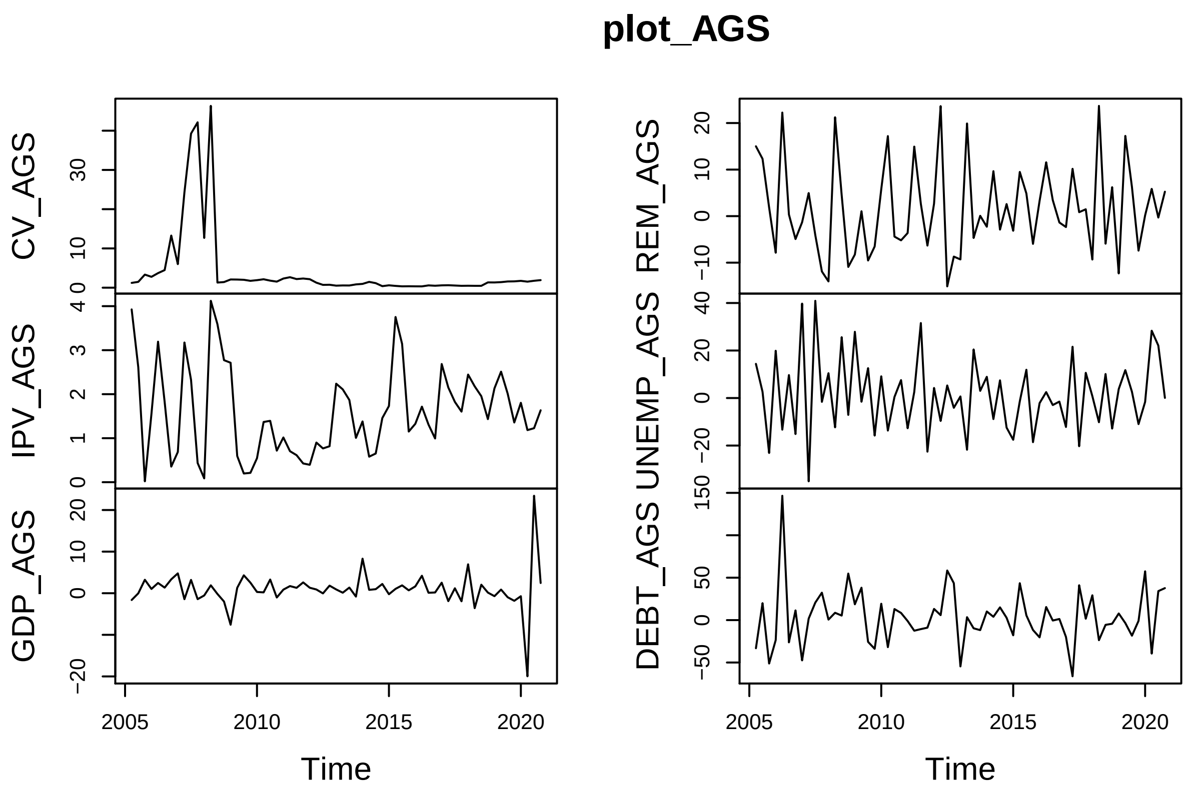
```
# (por poner, HTML format)
```

4 State Data Set

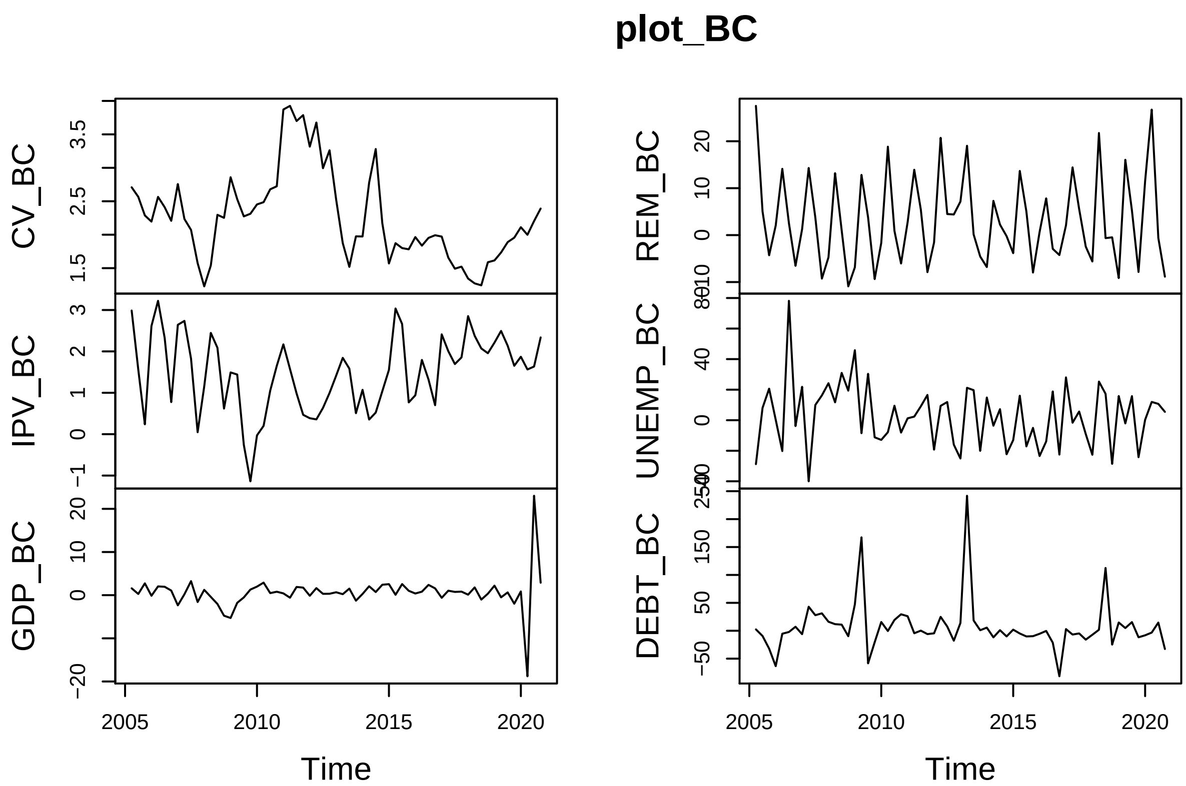
Now we intend to follow a very similar methodology for the state level data.

4.1 Loading and visualizing the data: 32 states

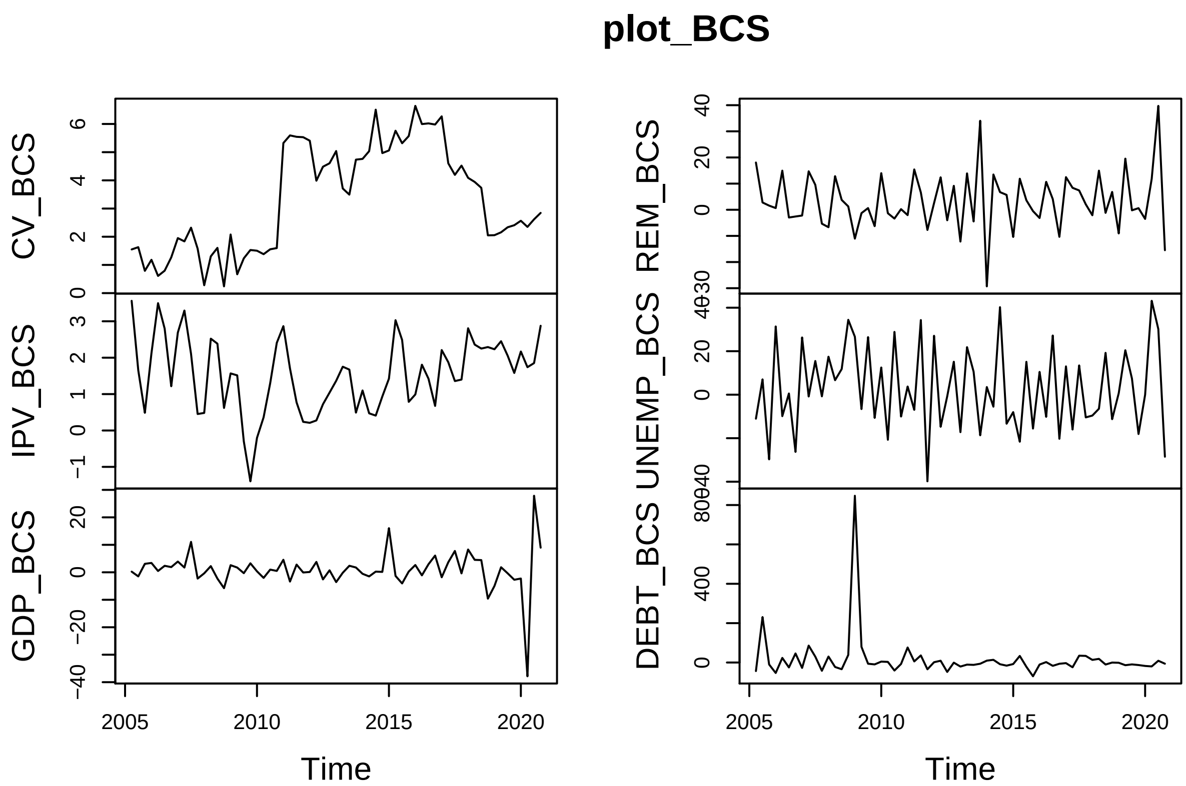
```
AGS<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 1)), start=c(2005,2), end=c(2020,4),  frequency=4)
BC<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 2)), start=c(2005,2), end=c(2020,4),   frequency=4)
BCS<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 3)), start=c(2005,2), end=c(2020,4),   frequency=4)
CAMP<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 4)), start=c(2005,2), end=c(2020,4),   frequency=4)
CDMX<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 5)), start=c(2005,2), end=c(2020,4),   frequency=4)
CHIH<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 6)), start=c(2005,2), end=c(2020,4),   frequency=4)
CHIS<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 7)), start=c(2005,2), end=c(2020,4),   frequency=4)
COAH<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 8)), start=c(2005,2), end=c(2020,4),   frequency=4)
COL<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 9)), start=c(2005,2), end=c(2020,4),   frequency=4)
DGO<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 10)), start=c(2005,2), end=c(2020,4),   frequency=4)
GRO<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 11)), start=c(2005,2), end=c(2020,4),   frequency=4)
GTO<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 12)), start=c(2005,2), end=c(2020,4),   frequency=4)
HGO<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 13)), start=c(2005,2), end=c(2020,4),   frequency=4)
JAL<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 14)), start=c(2005,2), end=c(2020,4),   frequency=4)
MEX<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 15)), start=c(2005,2), end=c(2020,4),   frequency=4)
MICH<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 16)), start=c(2005,2), end=c(2020,4),   frequency=4)
MOR<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 17)), start=c(2005,2), end=c(2020,4),   frequency=4)
NAY<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 18)), start=c(2005,2), end=c(2020,4),   frequency=4)
NL<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 19)), start=c(2005,2),  end=c(2020,4),  frequency=4)
OAXACA<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 20)), start=c(2005,2), end=c(2020,4),   frequency=4)
PUE<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 21)), start=c(2005,2), end=c(2020,4),   frequency=4)
Q_ROO<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 22)), start=c(2005,2), end=c(2020,4),   frequency=4)
QRO<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 23)), start=c(2005,2), end=c(2020,4),   frequency=4)
SIN<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 24)), start=c(2005,2), end=c(2020,4),   frequency=4)
SLP<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 25)), start=c(2005,2), end=c(2020,4),   frequency=4)
SON<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 26)), start=c(2005,2), end=c(2020,4),   frequency=4)
TAB<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 27)), start=c(2005,2), end=c(2020,4),   frequency=4)
TAMPS<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 28)), start=c(2005,2), end=c(2020,4),    frequency=4)
TLAX<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 29)), start=c(2005,2), end=c(2020,4),   frequency=4)
VER<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 30)), start=c(2005,2), end=c(2020,4),   frequency=4)
YUC<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 31)), start=c(2005,2), end=c(2020,4),   frequency=4)
ZAC<-ts(data=(read_excel("ESTADOS.xlsx", sheet = 32)), start=c(2005,2), end=c(2020,4),   frequency=4)

CV_AGS <- AGS[ ,5]
IPV_AGS <- AGS[ ,4]
GDP_AGS <- AGS[,6]
REM_AGS <- AGS[,10]
UNEMP_AGS <- AGS[,11]
DEBT_AGS <- AGS[,12]
plot_AGS<-ts(cbind(CV_AGS,IPV_AGS,GDP_AGS,REM_AGS,UNEMP_AGS,DEBT_AGS),start=c(2005,2),frequency=4)
plot(plot_AGS)
```



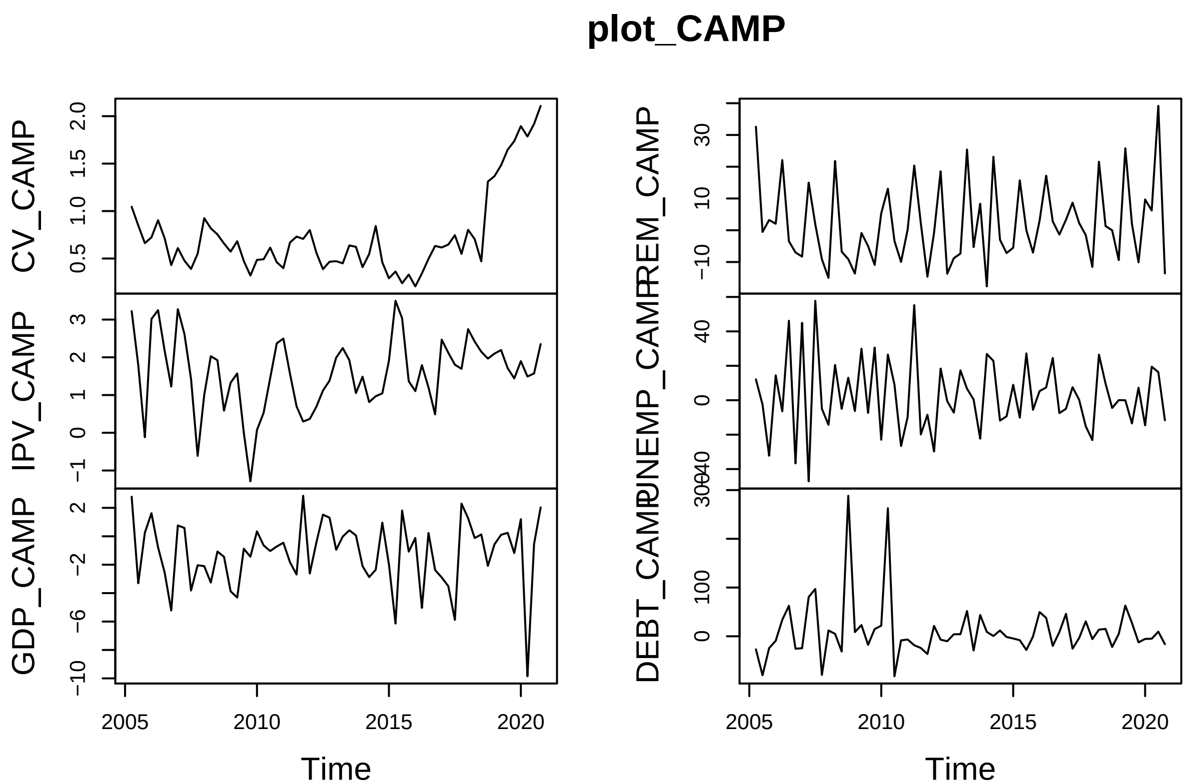

```
CV_BC <- BC[ ,5]
IPV_BC <- BC[ ,4]
GDP_BC <- BC[,6]
REM_BC <- BC[,10]
UNEMP_BC <- BC[,11]
DEBT_BC <- BC[,12]
plot_BC<-ts(cbind(CV_BC,IPV_BC,GDP_BC,REM_BC,UNEMP_BC,DEBT_BC),start=c(2005,2),frequency=4)
plot(plot_BC)
```



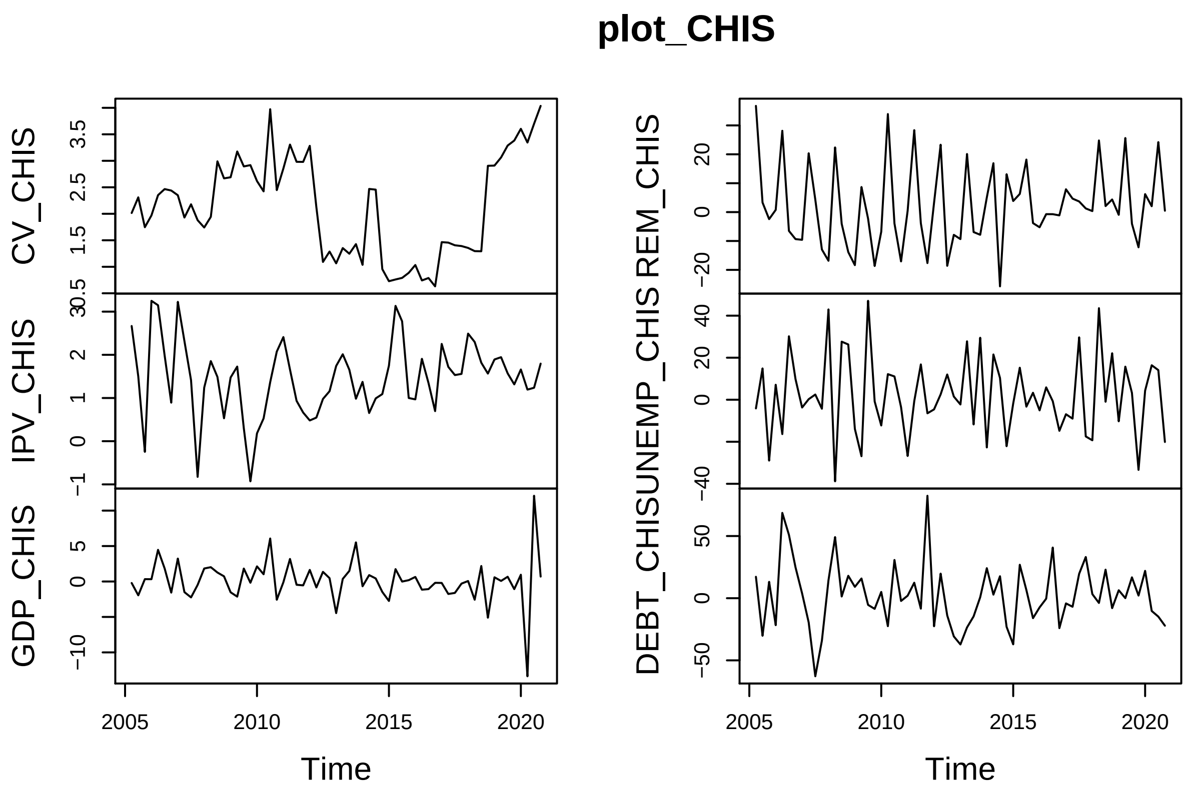

```
CV_BCS <- BCS[ ,5]
IPV_BCS <- BCS[ ,4]
GDP_BCS <- BCS[,6]
REM_BCS <- BCS[,10]
UNEMP_BCS <- BCS[,11]
DEBT_BCS <- BCS[,12]
plot_BCS<-ts(cbind(CV_BCS,IPV_BCS,GDP_BCS,REM_BCS,UNEMP_BCS,DEBT_BCS),start=c(2005,2),frequency=4)
plot(plot_BCS)
```



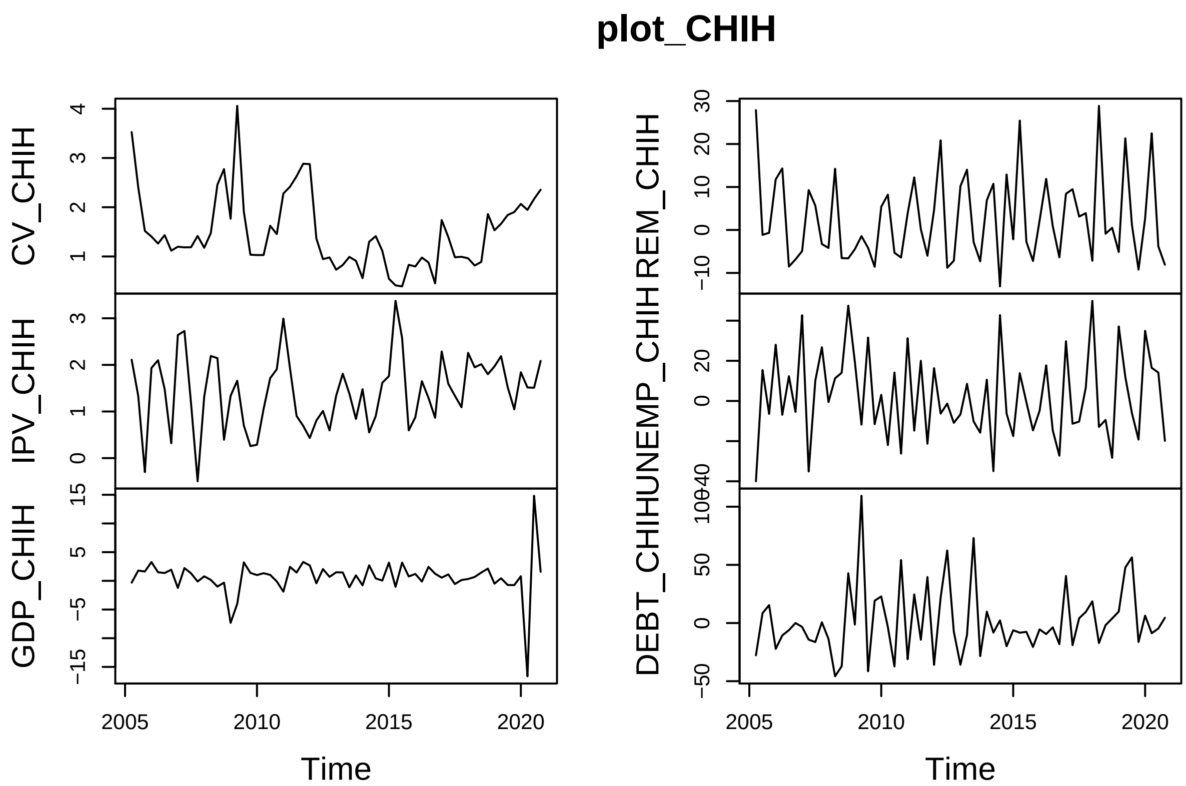

```
CV_CAMP <- CAMP[ ,5]
IPV_CAMP <- CAMP[ ,4]
GDP_CAMP <- CAMP[,6]
REM_CAMP <- CAMP[,10]
UNEMP_CAMP <- CAMP[,11]
DEBT_CAMP <- CAMP[,12]
plot_CAMP<-ts(cbind(CV_CAMP,IPV_CAMP,GDP_CAMP,REM_CAMP,UNEMP_CAMP,DEBT_CAMP),start=c(2005,2),frequency=4
plot(plot_CAMP)
```



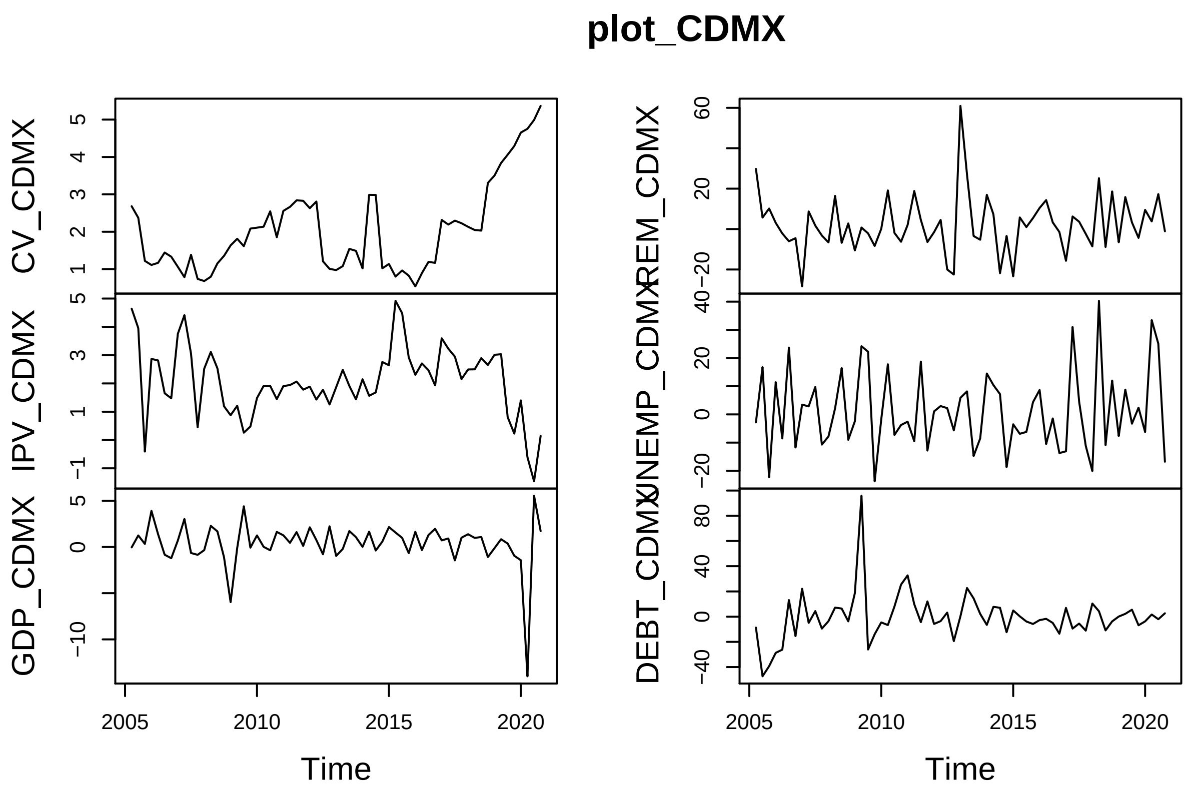

```
CV_CHIS <- CHIS[ ,5]
IPV_CHIS <- CHIS[ ,4]
GDP_CHIS <- CHIS[,6]
REM_CHIS <- CHIS[,10]
UNEMP_CHIS <- CHIS[,11]
DEBT_CHIS <- CHIS[,12]
plot_CHIS<-ts(cbind(CV_CHIS,IPV_CHIS,GDP_CHIS,REM_CHIS,UNEMP_CHIS,DEBT_CHIS),start=c(2005,2),frequency=4
plot(plot_CHIS)
```



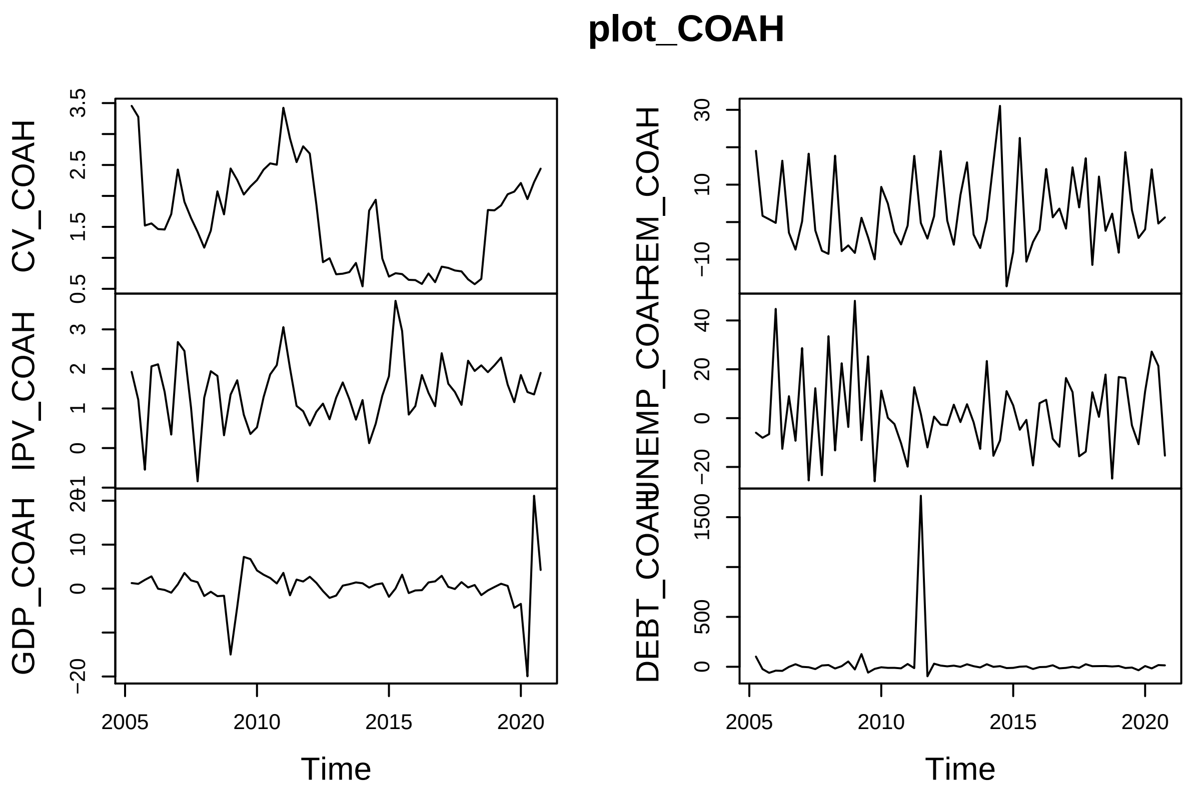

```
CV_CHIH <- CHIH[ ,5]
IPV_CHIH <- CHIH[ ,4]
GDP_CHIH <- CHIH[,6]
REM_CHIH <- CHIH[,10]
UNEMP_CHIH <- CHIH[,11]
DEBT_CHIH <- CHIH[,12]
plot_CHIH<-ts(cbind(CV_CHIH,IPV_CHIH,GDP_CHIH,REM_CHIH,UNEMP_CHIH,DEBT_CHIH),start=c(2005,2),frequency=4
plot(plot_CHIH)
```



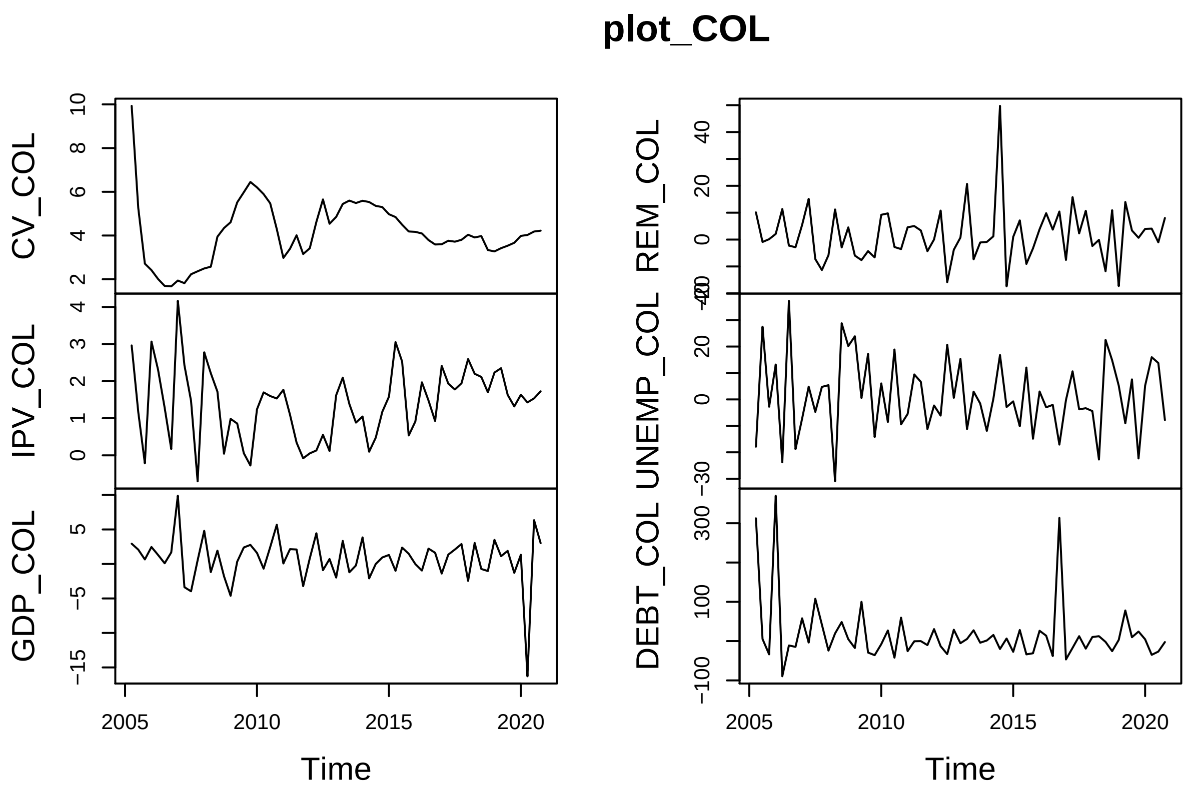

```
CV_CDMX <- CDMX[ ,5]
IPV_CDMX <- CDMX[ ,4]
GDP_CDMX <- CDMX[,6]
REM_CDMX <- CDMX[,10]
UNEMP_CDMX <- CDMX[,11]
DEBT_CDMX <- CDMX[,12]
plot_CDMX<-ts(cbind(CV_CDMX,IPV_CDMX,GDP_CDMX,REM_CDMX,UNEMP_CDMX,DEBT_CDMX),start=c(2005,2),frequency=4
plot(plot_CDMX)
```



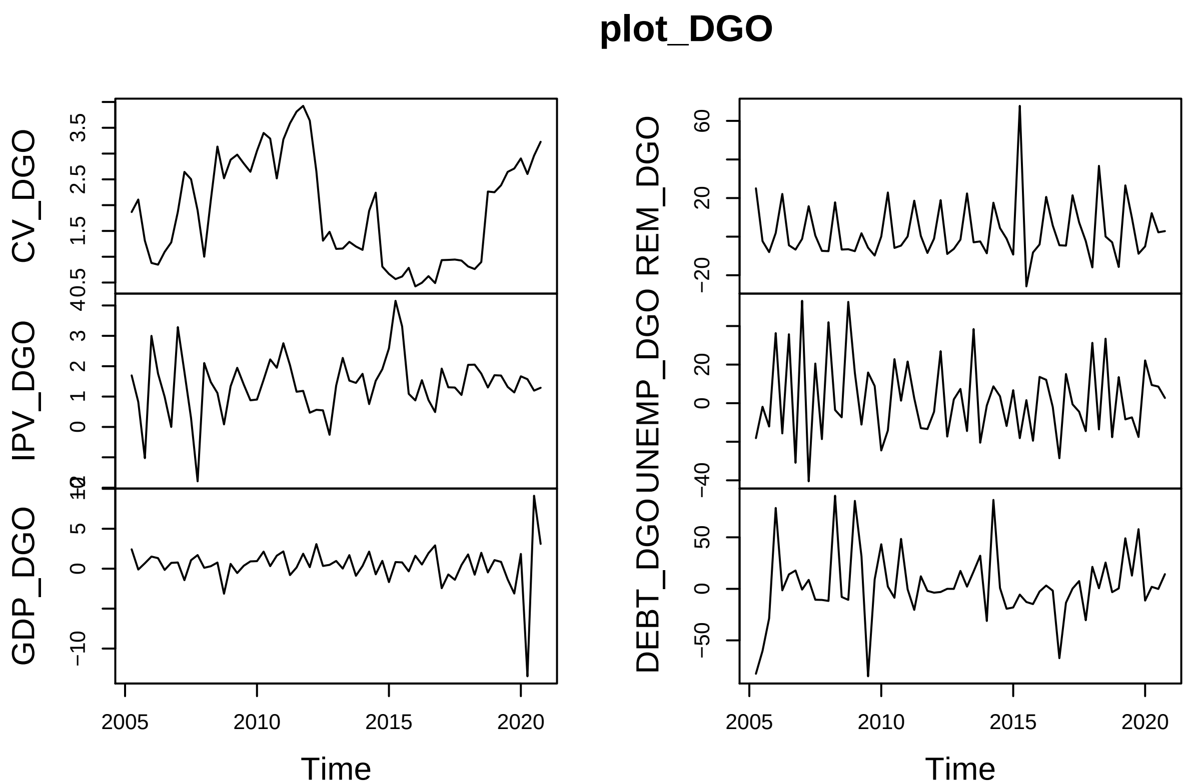

```
CV_COAH <- COAH[ ,5]
IPV_COAH <- COAH[ ,4]
GDP_COAH <- COAH[,6]
REM_COAH <- COAH[,10]
UNEMP_COAH <- COAH[,11]
DEBT_COAH <- COAH[,12]
plot_COAH<-ts(cbind(CV_COAH,IPV_COAH,GDP_COAH,REM_COAH,UNEMP_COAH,DEBT_COAH),start=c(2005,2),frequency=4
plot(plot_COAH)
```



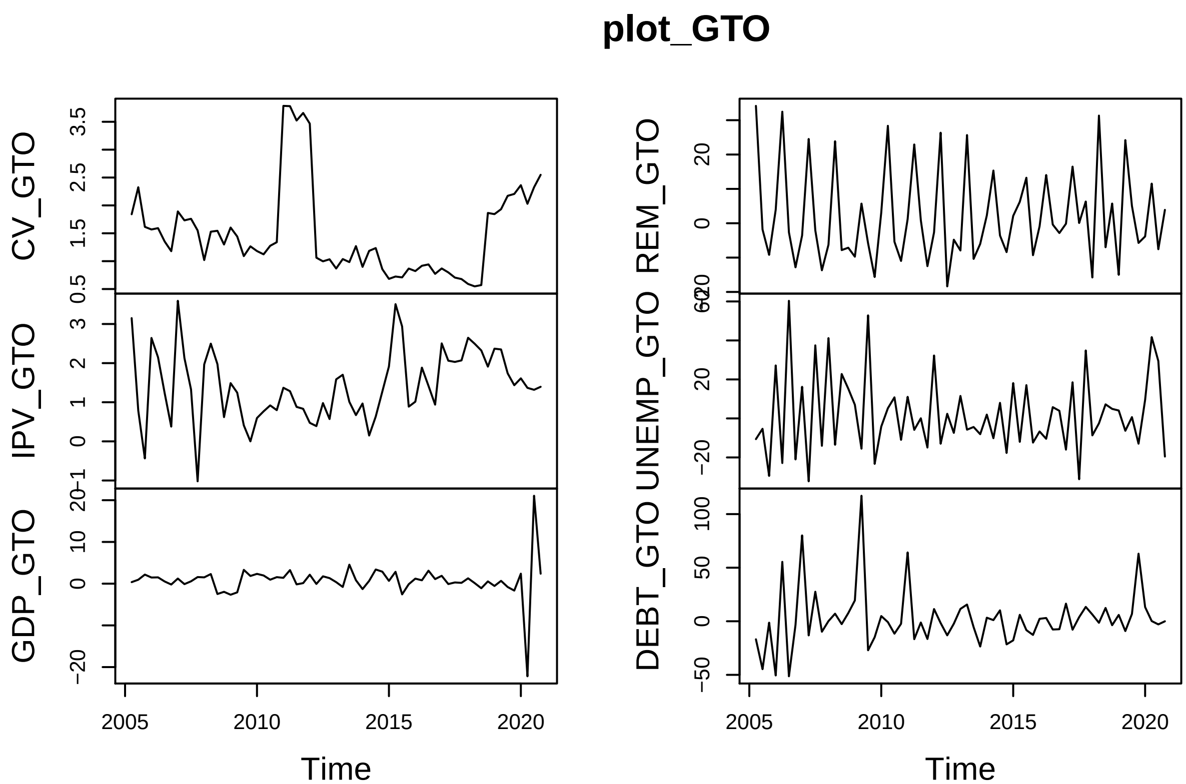

```
CV_COL <- COL[ ,5]
IPV_COL <- COL[ ,4]
GDP_COL <- COL[,6]
REM_COL <- COL[,10]
UNEMP_COL <- COL[,11]
DEBT_COL <- COL[,12]
plot_COL<-ts(cbind(CV_COL,IPV_COL,GDP_COL,REM_COL,UNEMP_COL,DEBT_COL),start=c(2005,2),frequency=4)
plot(plot_COL)
```



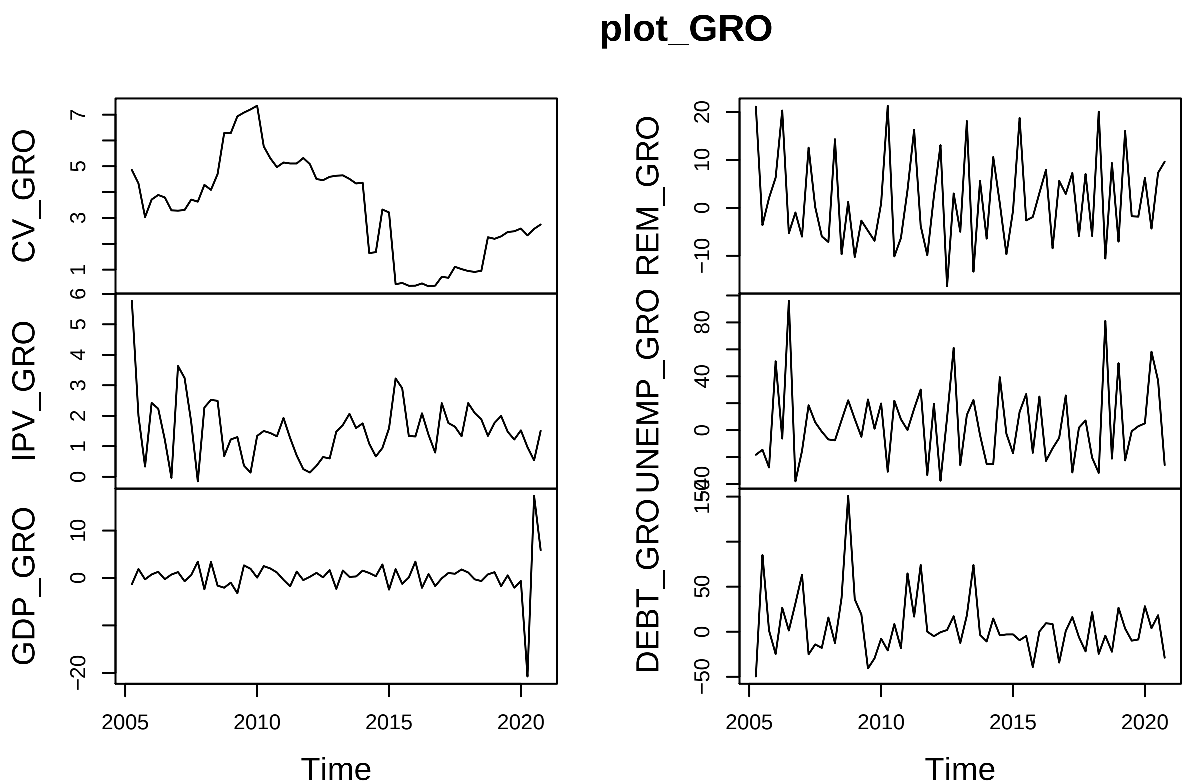

```
CV_DGO <- DGO[ ,5]
IPV_DGO <- DGO[ ,4]
GDP_DGO <- DGO[,6]
REM_DGO <- DGO[,10]
UNEMP_DGO <- DGO[,11]
DEBT_DGO <- DGO[,12]
plot_DGO<-ts(cbind(CV_DGO,IPV_DGO,GDP_DGO,REM_DGO,UNEMP_DGO,DEBT_DGO),start=c(2005,2),frequency=4)
plot(plot_DGO)
```



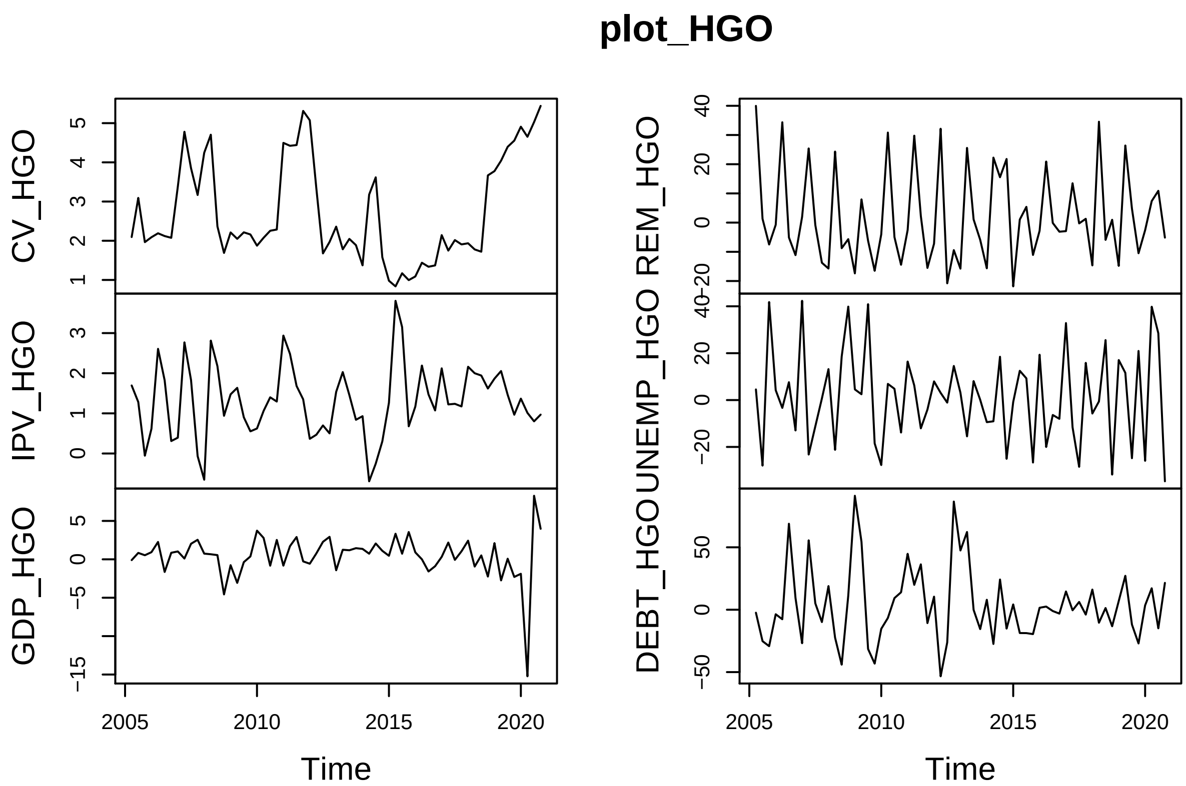

```
CV_GTO <- GTO[ ,5]
IPV_GTO <- GTO[ ,4]
GDP_GTO <- GTO[,6]
REM_GTO <- GTO[,10]
UNEMP_GTO <- GTO[,11]
DEBT_GTO <- GTO[,12]
plot_GTO<-ts(cbind(CV_GTO,IPV_GTO,GDP_GTO,REM_GTO,UNEMP_GTO,DEBT_GTO),start=c(2005,2),frequency=4)
plot(plot_GTO)
```



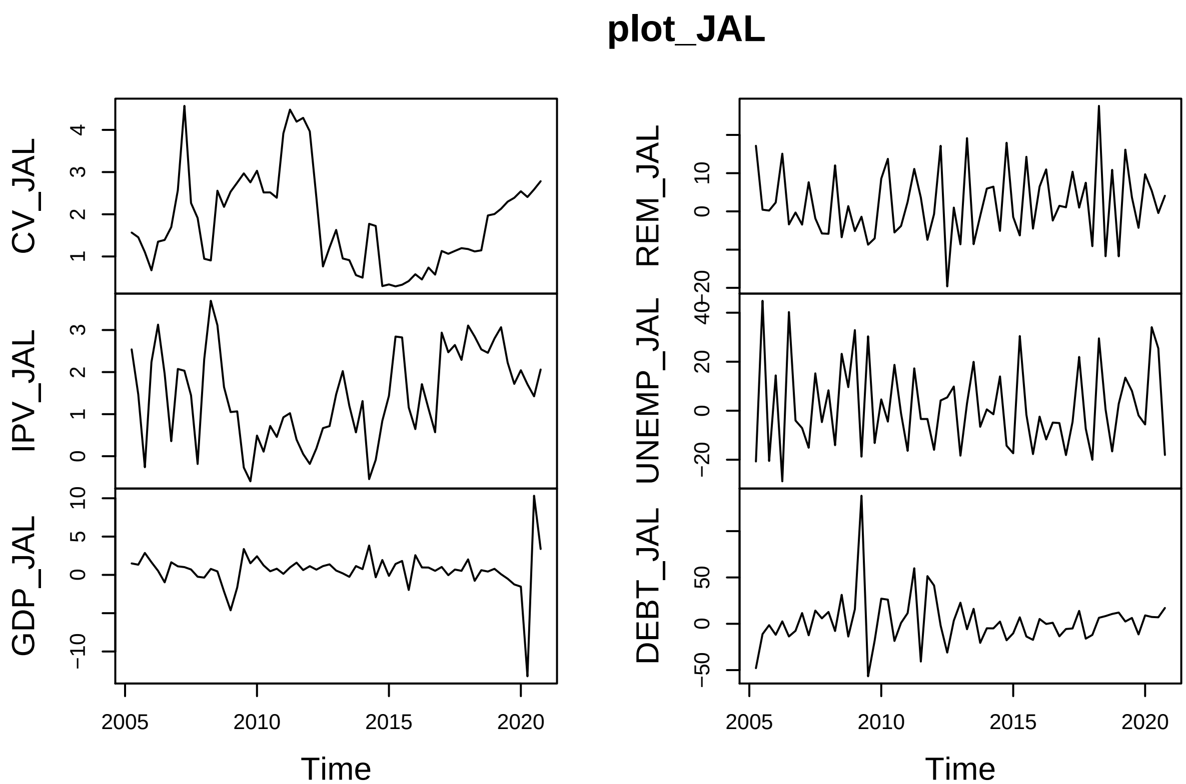

```
CV_GRO <- GRO[ ,5]
IPV_GRO <- GRO[ ,4]
GDP_GRO <- GRO[,6]
REM_GRO <- GRO[,10]
UNEMP_GRO <- GRO[,11]
DEBT_GRO <- GRO[,12]
plot_GRO<-ts(cbind(CV_GRO,IPV_GRO,GDP_GRO,REM_GRO,UNEMP_GRO,DEBT_GRO),start=c(2005,2),frequency=4)
plot(plot_GRO)
```



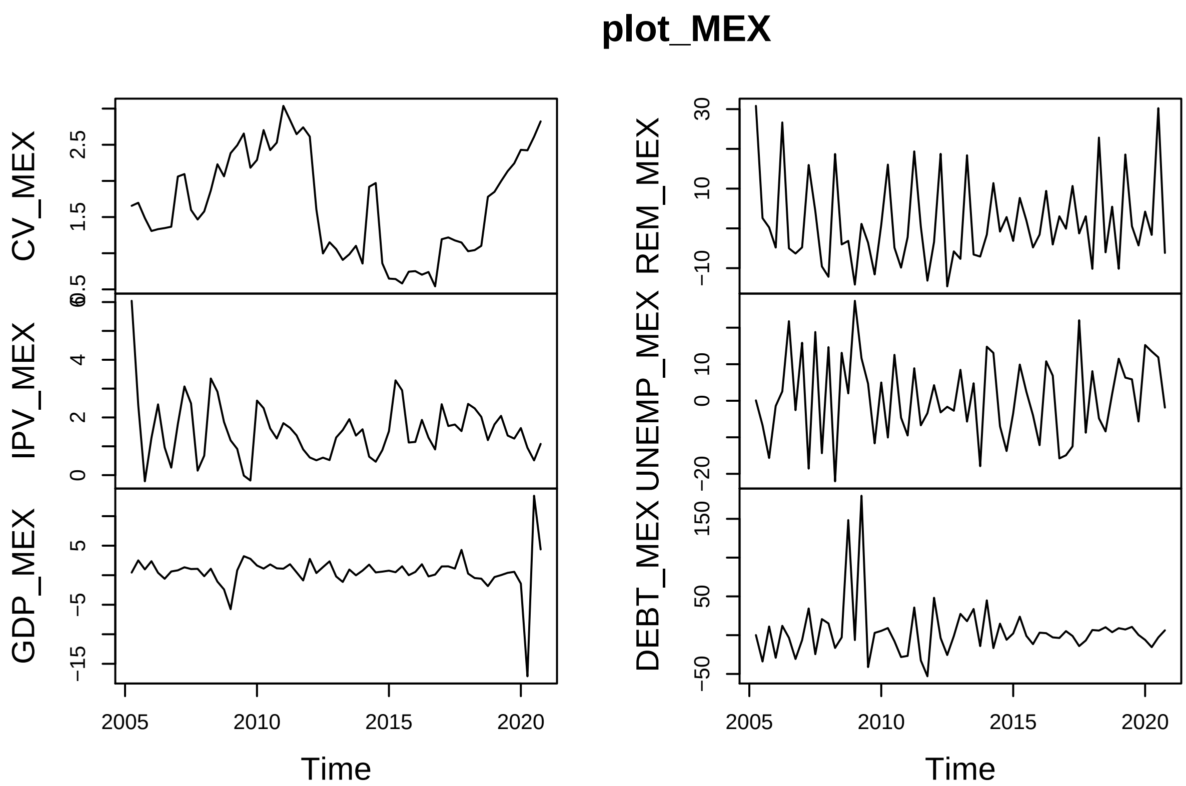

```
CV_HGO <- HGO[ ,5]
IPV_HGO <- HGO[ ,4]
GDP_HGO <- HGO[,6]
REM_HGO <- HGO[,10]
UNEMP_HGO <- HGO[,11]
DEBT_HGO <- HGO[,12]
plot_HGO<-ts(cbind(CV_HGO,IPV_HGO,GDP_HGO,REM_HGO,UNEMP_HGO,DEBT_HGO),start=c(2005,2),frequency=4)
plot(plot_HGO)
```



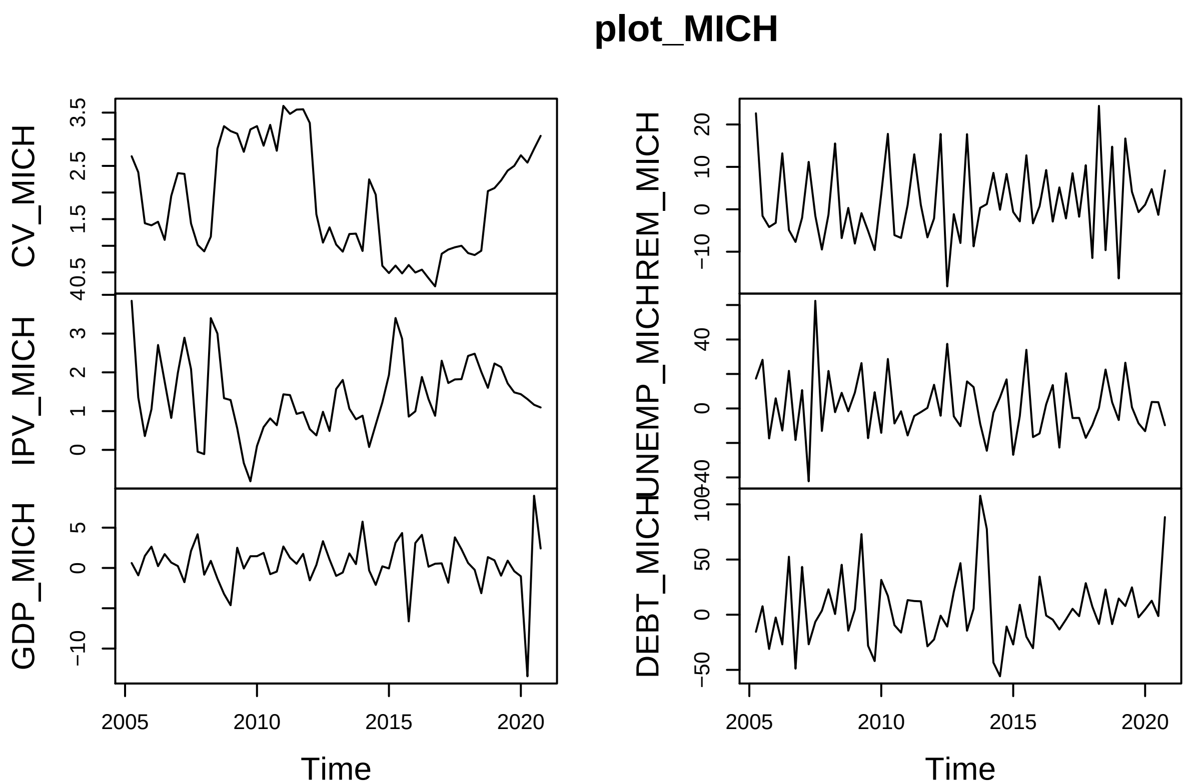

```
CV_JAL <- JAL[ ,5]
IPV_JAL <- JAL[ ,4]
GDP_JAL <- JAL[,6]
REM_JAL <- JAL[,10]
UNEMP_JAL <- JAL[,11]
DEBT_JAL <- JAL[,12]
plot_JAL<-ts(cbind(CV_JAL,IPV_JAL,GDP_JAL,REM_JAL,UNEMP_JAL,DEBT_JAL),start=c(2005,2),frequency=4)
plot(plot_JAL)
```



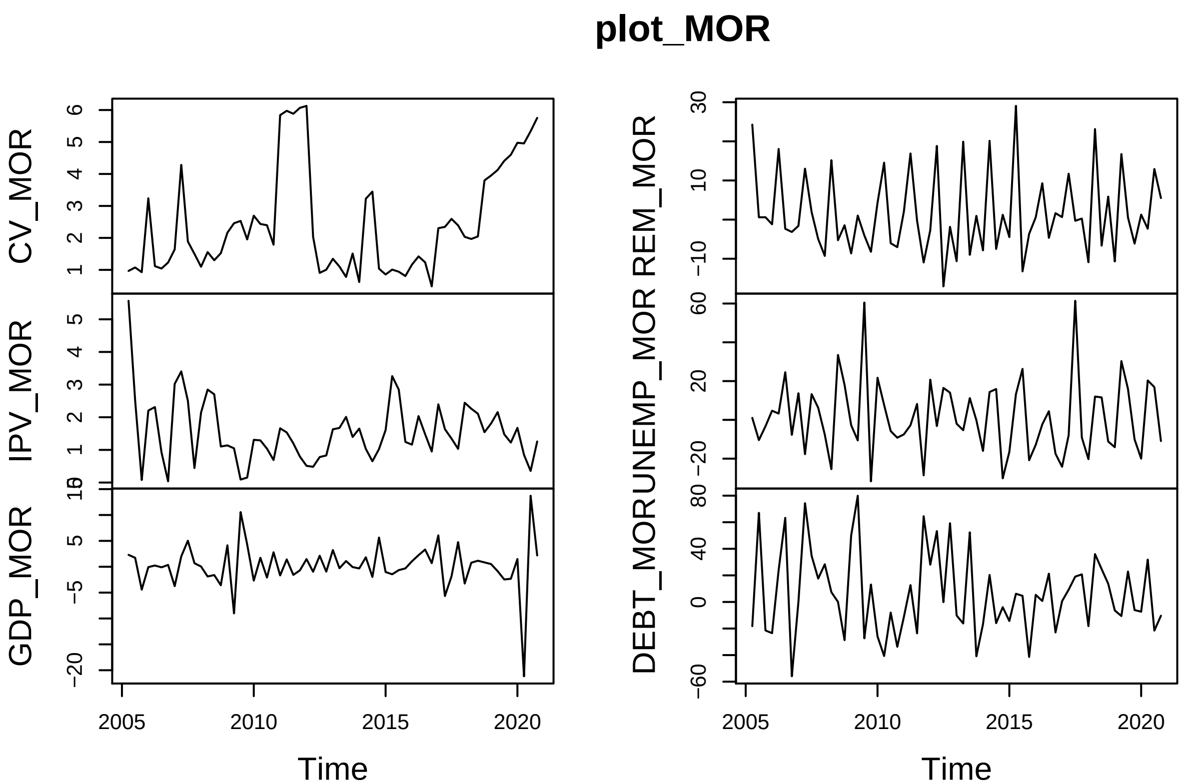

```
CV_MEX <- MEX[ ,5]
IPV_MEX <- MEX[ ,4]
GDP_MEX <- MEX[,6]
REM_MEX <- MEX[,10]
UNEMP_MEX <- MEX[,11]
DEBT_MEX <- MEX[,12]
plot_MEX<-ts(cbind(CV_MEX,IPV_MEX,GDP_MEX,REM_MEX,UNEMP_MEX,DEBT_MEX),start=c(2005,2),frequency=4)
plot(plot_MEX)
```



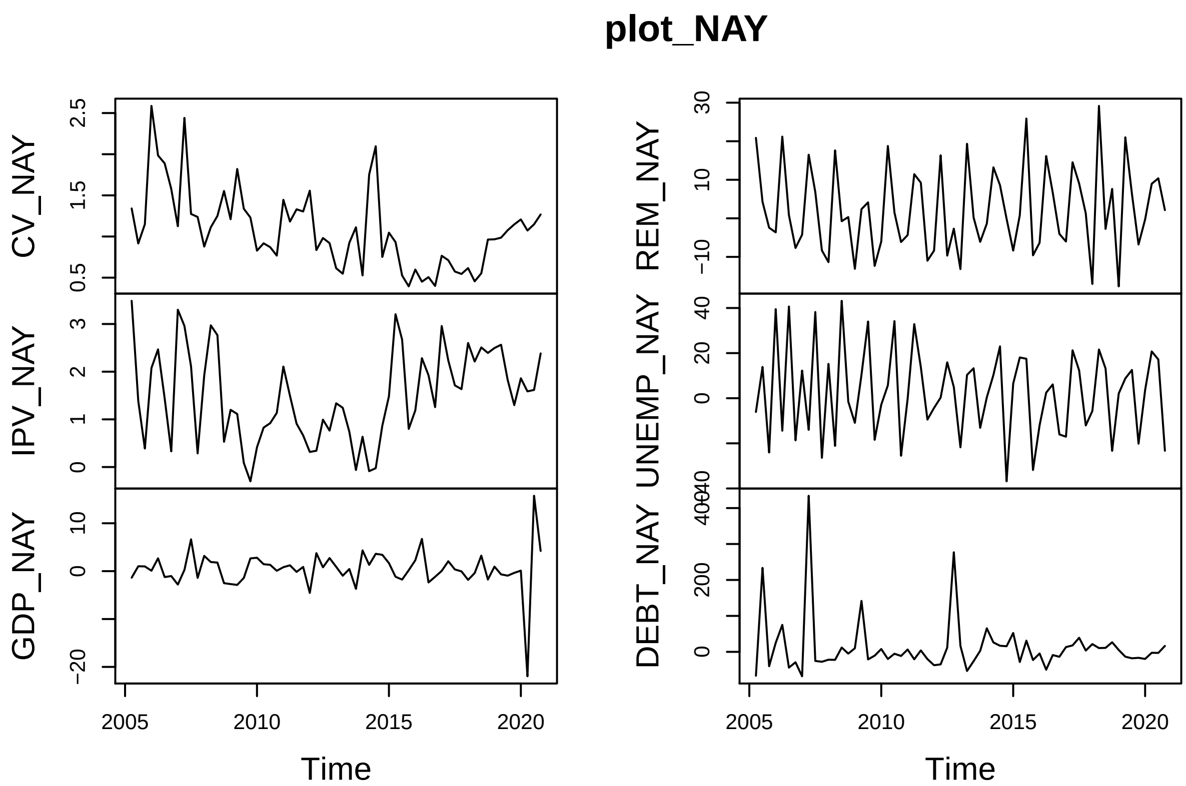

```r
CV_MICH <- MICH[ ,5]
IPV_MICH <- MICH[ ,4]
GDP_MICH <- MICH[,6]
REM_MICH <- MICH[,10]
UNEMP_MICH <- MICH[,11]
DEBT_MICH <- MICH[,12]
plot_MICH<-ts(cbind(CV_MICH,IPV_MICH,GDP_MICH,REM_MICH,UNEMP_MICH,DEBT_MICH),start=c(2005,2),frequency=4
plot(plot_MICH)
```



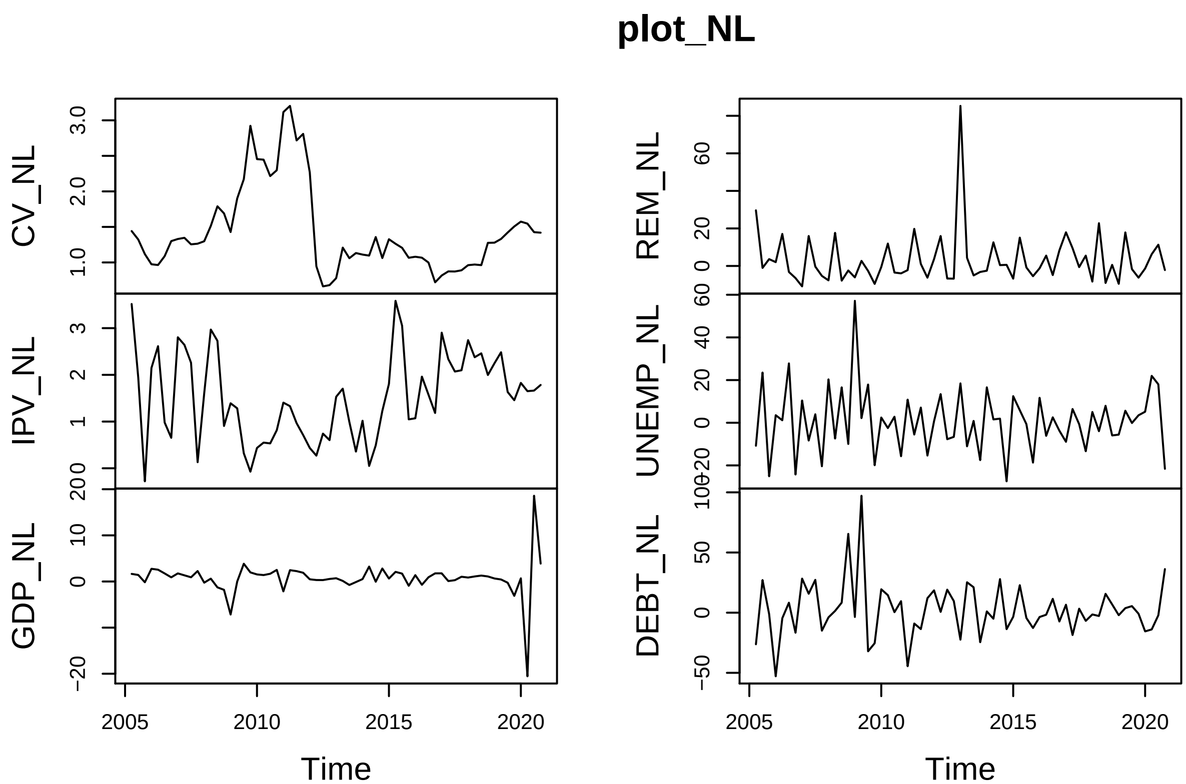

```
CV_MOR <- MOR[ ,5]
IPV_MOR <- MOR[ ,4]
GDP_MOR <- MOR[,6]
REM_MOR <- MOR[,10]
UNEMP_MOR <- MOR[,11]
DEBT_MOR <- MOR[,12]
plot_MOR<-ts(cbind(CV_MOR,IPV_MOR,GDP_MOR,REM_MOR,UNEMP_MOR,DEBT_MOR),start=c(2005,2),frequency=4)
plot(plot_MOR)
```



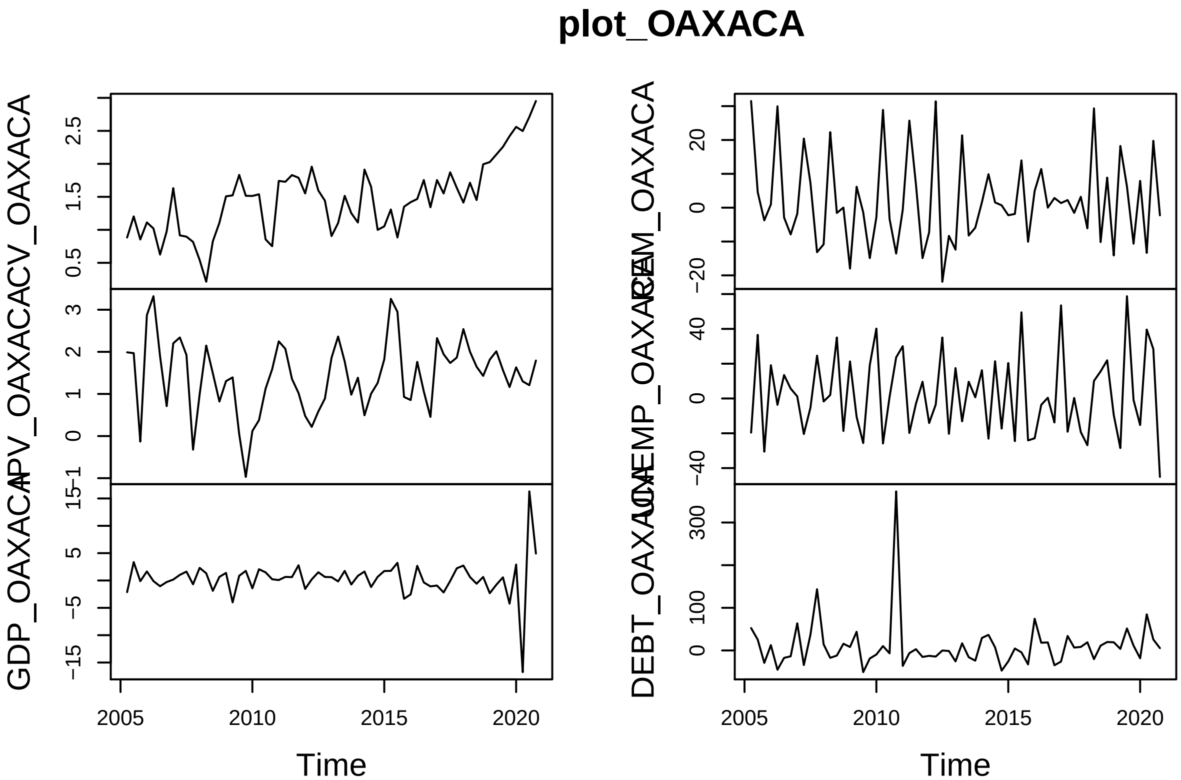

```
CV_NAY <- NAY[ ,5]
IPV_NAY <- NAY[ ,4]
GDP_NAY <- NAY[,6]
REM_NAY <- NAY[,10]
UNEMP_NAY <- NAY[,11]
DEBT_NAY <- NAY[,12]
plot_NAY<-ts(cbind(CV_NAY,IPV_NAY,GDP_NAY,REM_NAY,UNEMP_NAY,DEBT_NAY),start=c(2005,2),frequency=4)
plot(plot_NAY)
```



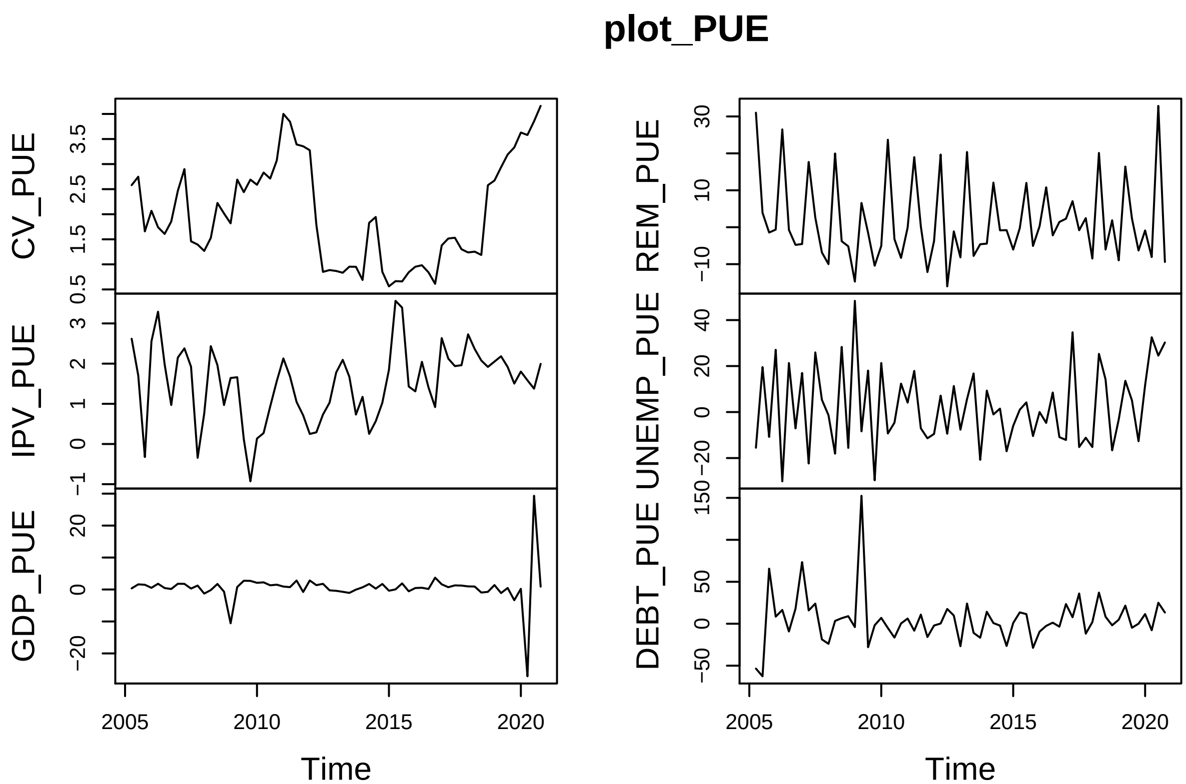

```
CV_NL <- NL[ ,5]
IPV_NL <- NL[ ,4]
GDP_NL <- NL[,6]
REM_NL <- NL[,10]
UNEMP_NL <- NL[,11]
DEBT_NL <- NL[,12]
plot_NL<-ts(cbind(CV_NL,IPV_NL,GDP_NL,REM_NL,UNEMP_NL,DEBT_NL),start=c(2005,2),frequency=4)
plot(plot_NL)
```



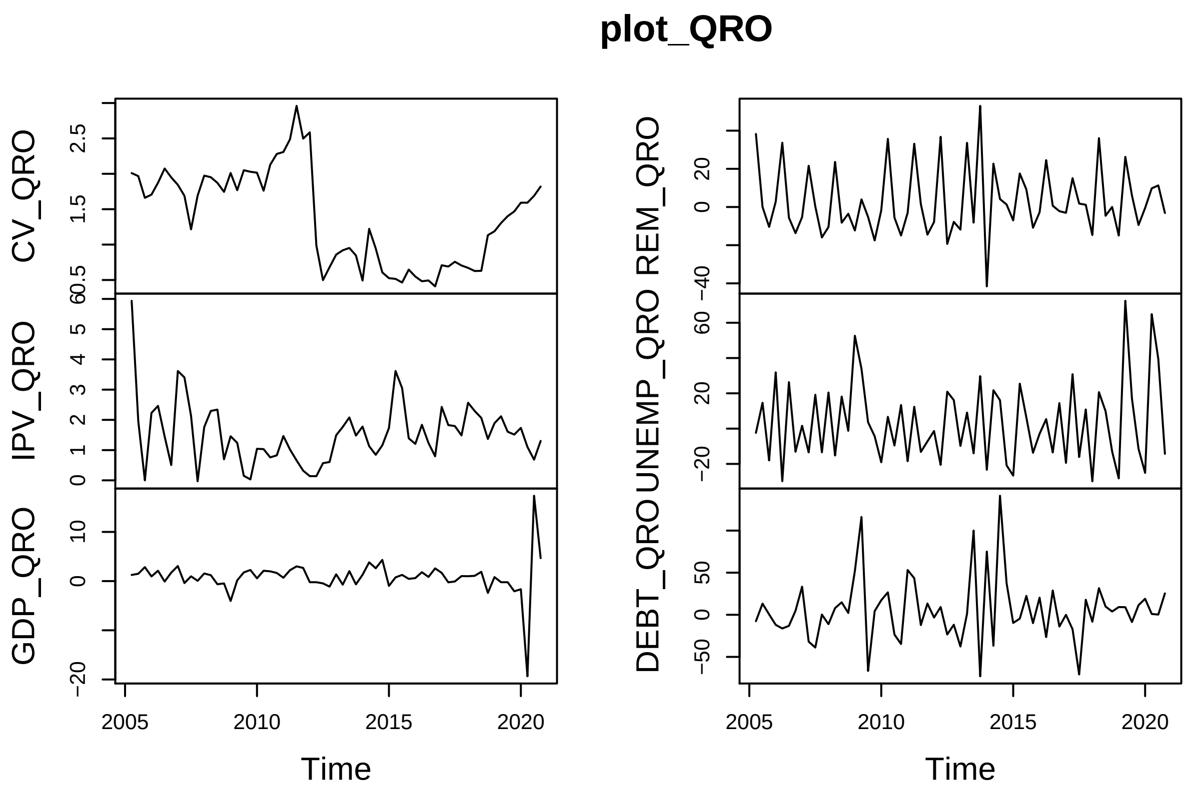

```
CV_OAXACA <- OAXACA[ ,5]
IPV_OAXACA <- OAXACA[ ,4]
GDP_OAXACA <- OAXACA[,6]
REM_OAXACA <- OAXACA[,10]
UNEMP_OAXACA <- OAXACA[,11]
DEBT_OAXACA <- OAXACA[,12]
plot_OAXACA<-ts(cbind(CV_OAXACA,IPV_OAXACA,GDP_OAXACA,REM_OAXACA,UNEMP_OAXACA,DEBT_OAXACA),start=c(2005
plot(plot_OAXACA)
```



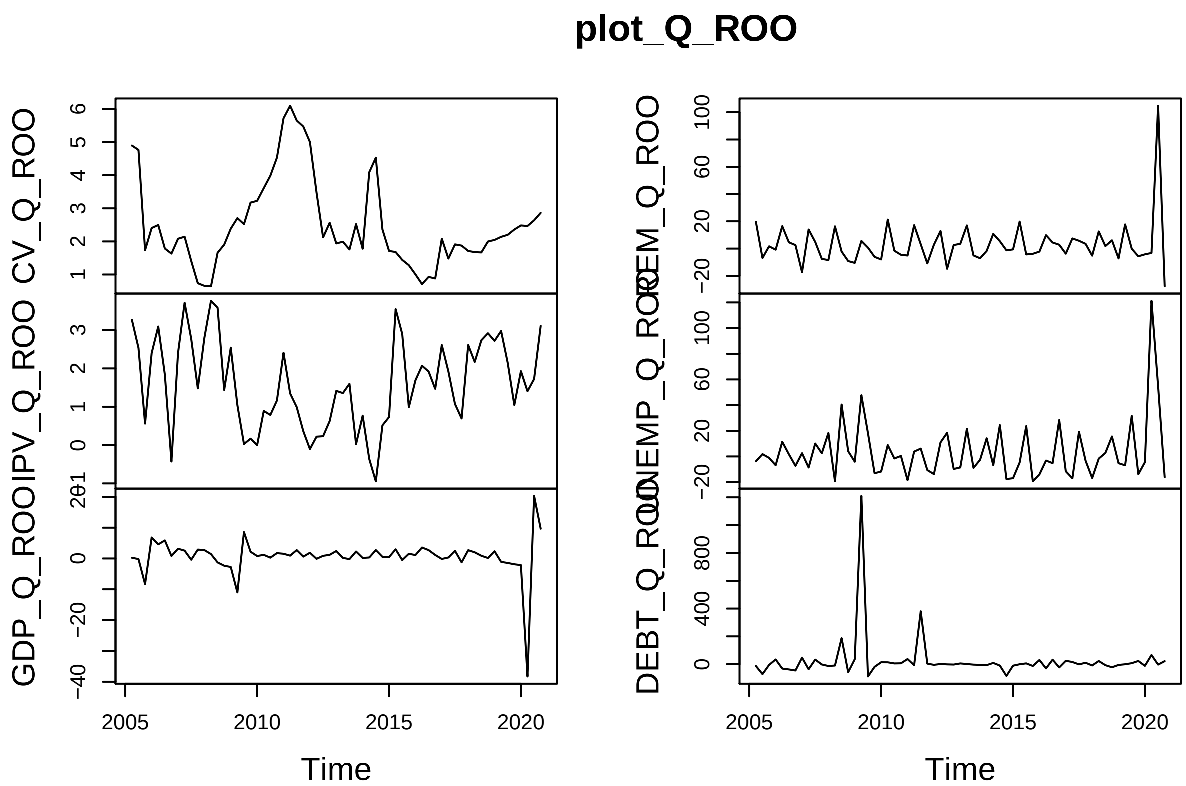

```
CV_PUE <- PUE[ ,5]
IPV_PUE <- PUE[ ,4]
GDP_PUE <- PUE[,6]
REM_PUE <- PUE[,10]
UNEMP_PUE <- PUE[,11]
DEBT_PUE <- PUE[,12]
plot_PUE<-ts(cbind(CV_PUE,IPV_PUE,GDP_PUE,REM_PUE,UNEMP_PUE,DEBT_PUE),start=c(2005,2),frequency=4)
plot(plot_PUE)
```



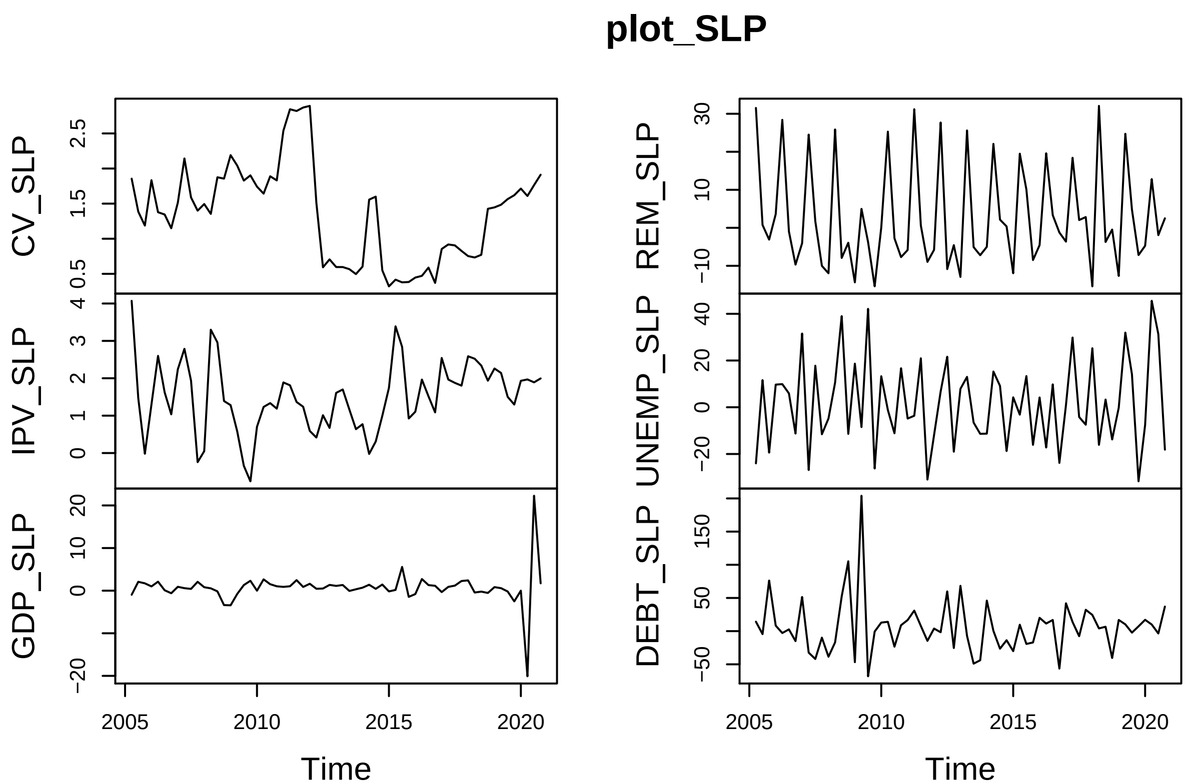

```
CV_QRO <- QRO[ ,5]
IPV_QRO <- QRO[ ,4]
GDP_QRO <- QRO[,6]
REM_QRO <- QRO[,10]
UNEMP_QRO <- QRO[,11]
DEBT_QRO <- QRO[,12]
plot_QRO<-ts(cbind(CV_QRO,IPV_QRO,GDP_QRO,REM_QRO,UNEMP_QRO,DEBT_QRO),start=c(2005,2),frequency=4)
plot(plot_QRO)
```



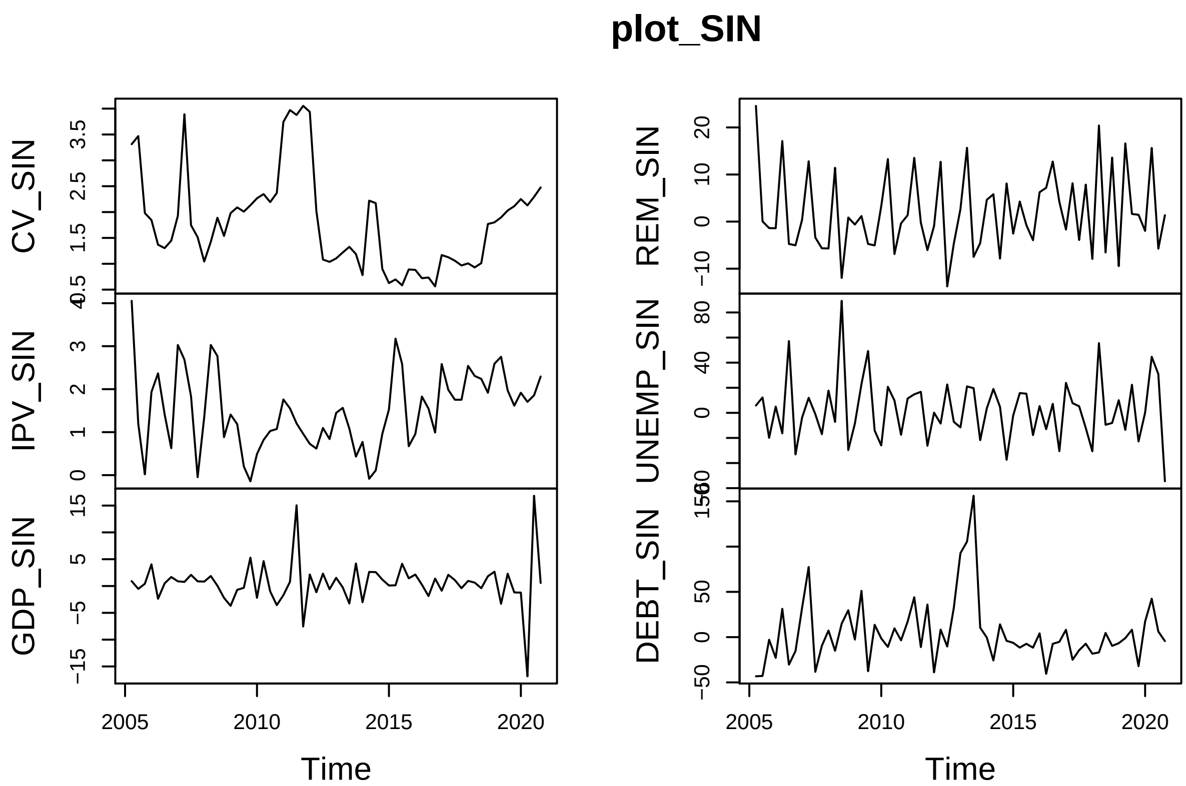

```
CV_Q_ROO <- Q_ROO[ ,5]
IPV_Q_ROO <- Q_ROO[ ,4]
GDP_Q_ROO <- Q_ROO[,6]
REM_Q_ROO <- Q_ROO[,10]
UNEMP_Q_ROO <- Q_ROO[,11]
DEBT_Q_ROO <- Q_ROO[,12]
plot_Q_ROO<-ts(cbind(CV_Q_ROO,IPV_Q_ROO,GDP_Q_ROO,REM_Q_ROO,UNEMP_Q_ROO,DEBT_Q_ROO),start=c(2005,2),freq
plot(plot_Q_ROO)
```



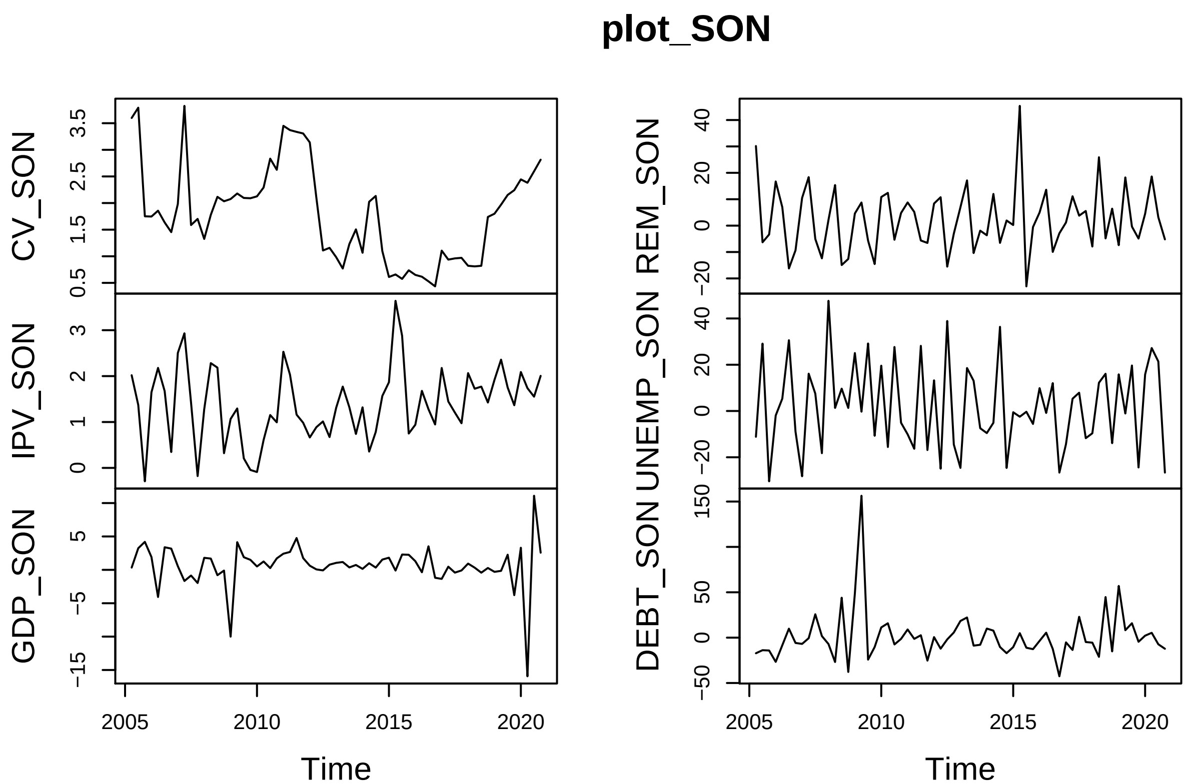

```
CV_SLP <- SLP[ ,5]
IPV_SLP <- SLP[ ,4]
GDP_SLP <- SLP[,6]
REM_SLP <- SLP[,10]
UNEMP_SLP <- SLP[,11]
DEBT_SLP <- SLP[,12]
plot_SLP<-ts(cbind(CV_SLP,IPV_SLP,GDP_SLP,REM_SLP,UNEMP_SLP,DEBT_SLP),start=c(2005,2),frequency=4)
plot(plot_SLP)
```



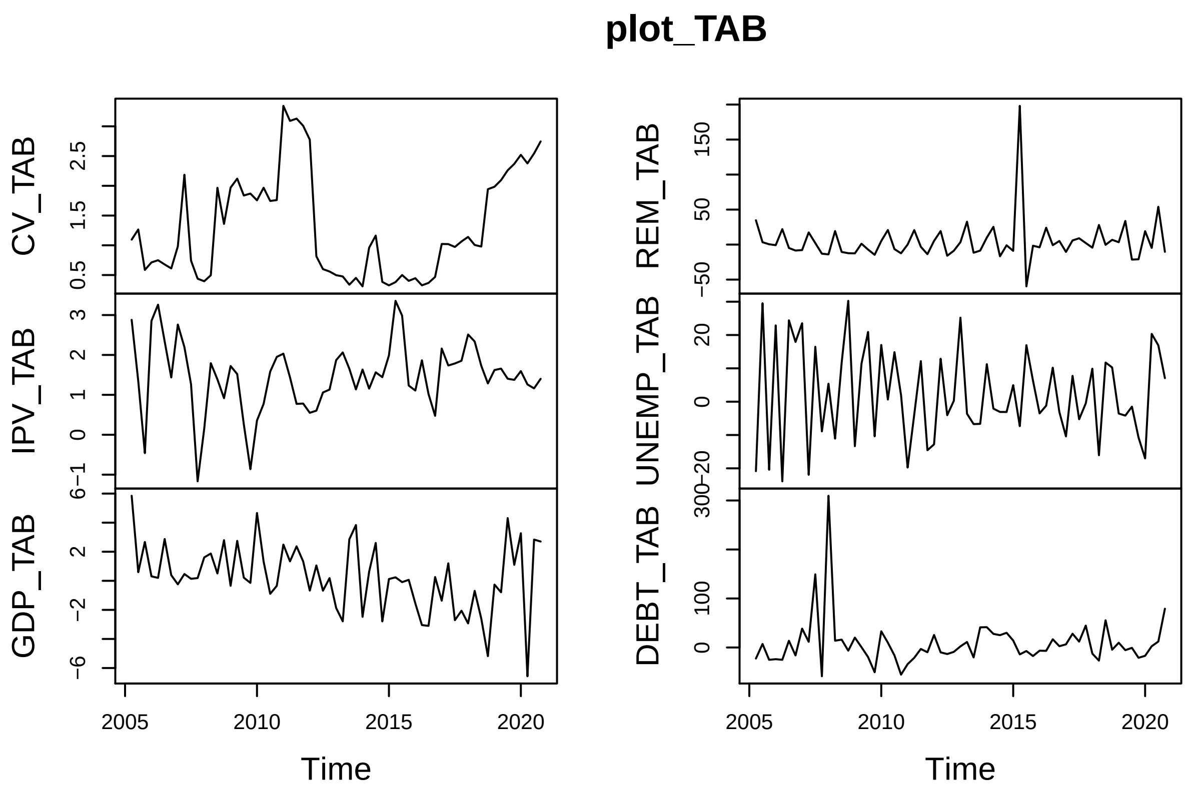

```
CV_SIN <- SIN[ ,5]
IPV_SIN <- SIN[ ,4]
GDP_SIN <- SIN[,6]
REM_SIN <- SIN[,10]
UNEMP_SIN <- SIN[,11]
DEBT_SIN <- SIN[,12]
plot_SIN<-ts(cbind(CV_SIN,IPV_SIN,GDP_SIN,REM_SIN,UNEMP_SIN,DEBT_SIN),start=c(2005,2),frequency=4)
plot(plot_SIN)
```



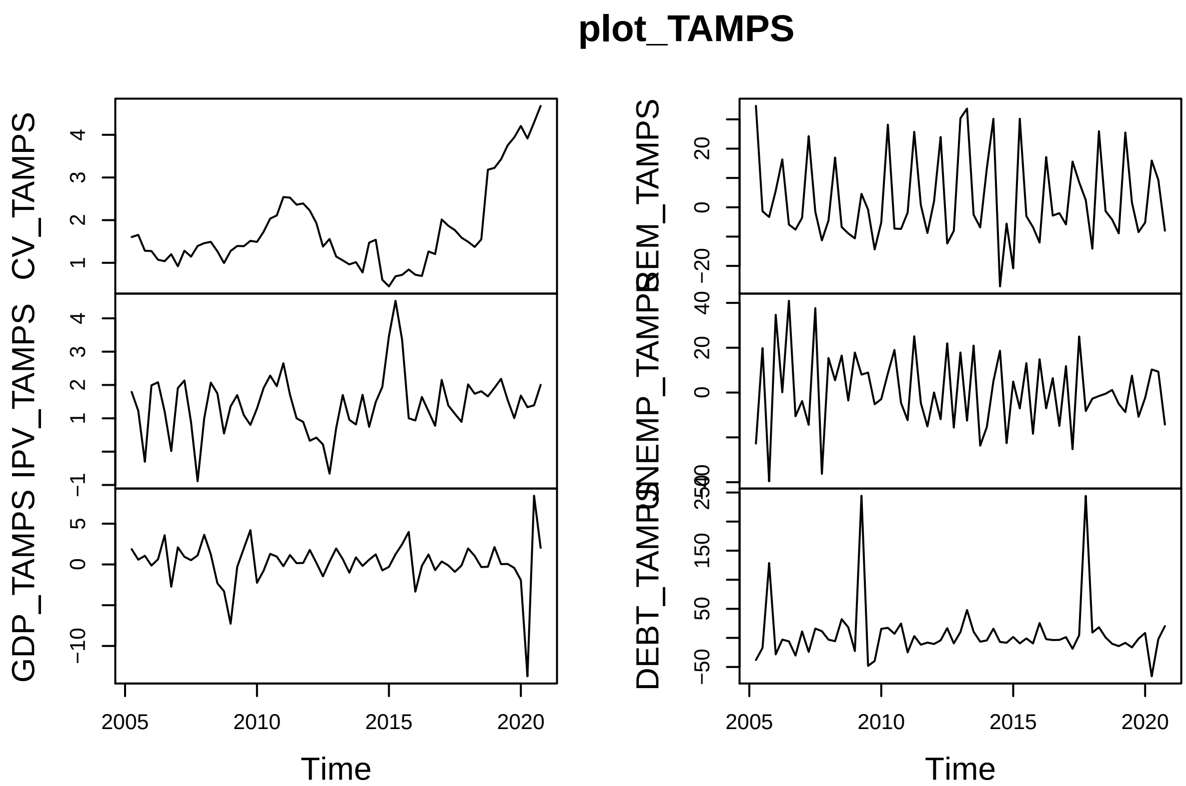

```r
CV_SON <- SON[ ,5]
IPV_SON <- SON[ ,4]
GDP_SON <- SON[,6]
REM_SON <- SON[,10]
UNEMP_SON <- SON[,11]
DEBT_SON <- SON[,12]
plot_SON<-ts(cbind(CV_SON,IPV_SON,GDP_SON,REM_SON,UNEMP_SON,DEBT_SON),start=c(2005,2),frequency=4)
plot(plot_SON)
```



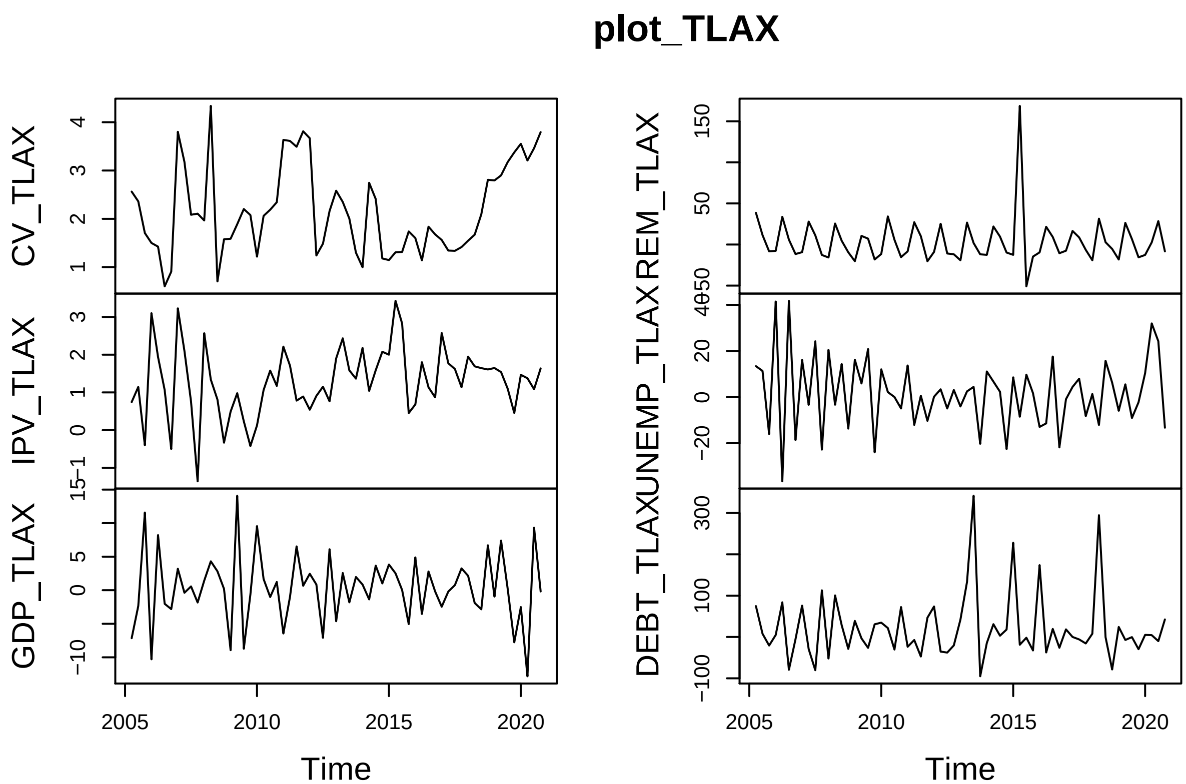

```r
CV_TAB <- TAB[ ,5]
IPV_TAB <- TAB[ ,4]
GDP_TAB <- TAB[,6]
REM_TAB <- TAB[,10]
UNEMP_TAB <- TAB[,11]
DEBT_TAB <- TAB[,12]
plot_TAB<-ts(cbind(CV_TAB,IPV_TAB,GDP_TAB,REM_TAB,UNEMP_TAB,DEBT_TAB),start=c(2005,2),frequency=4)
plot(plot_TAB)
```



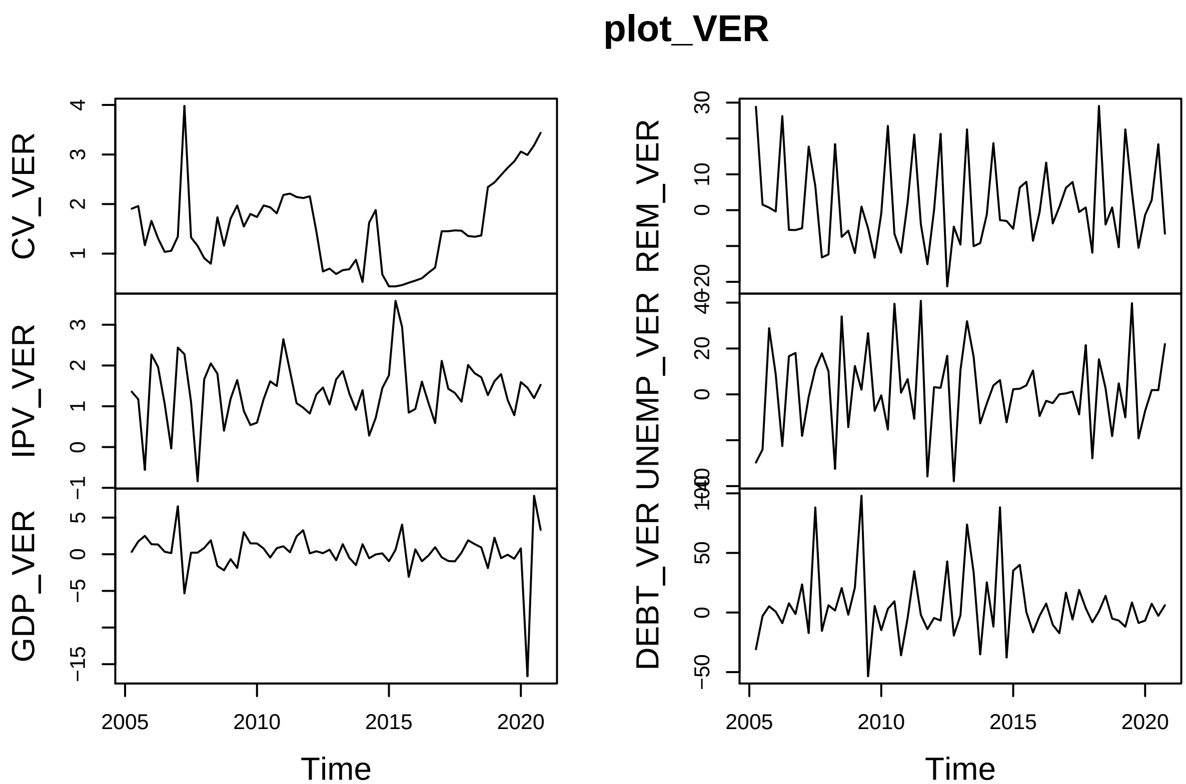

```
CV_TAMPS <- TAMPS[ ,5]
IPV_TAMPS <- TAMPS[ ,4]
GDP_TAMPS <- TAMPS[,6]
REM_TAMPS <- TAMPS[,10]
UNEMP_TAMPS <- TAMPS[,11]
DEBT_TAMPS <- TAMPS[,12]
plot_TAMPS<-ts(cbind(CV_TAMPS,IPV_TAMPS,GDP_TAMPS,REM_TAMPS,UNEMP_TAMPS,DEBT_TAMPS),start=c(2005,2),fre
plot(plot_TAMPS)
```



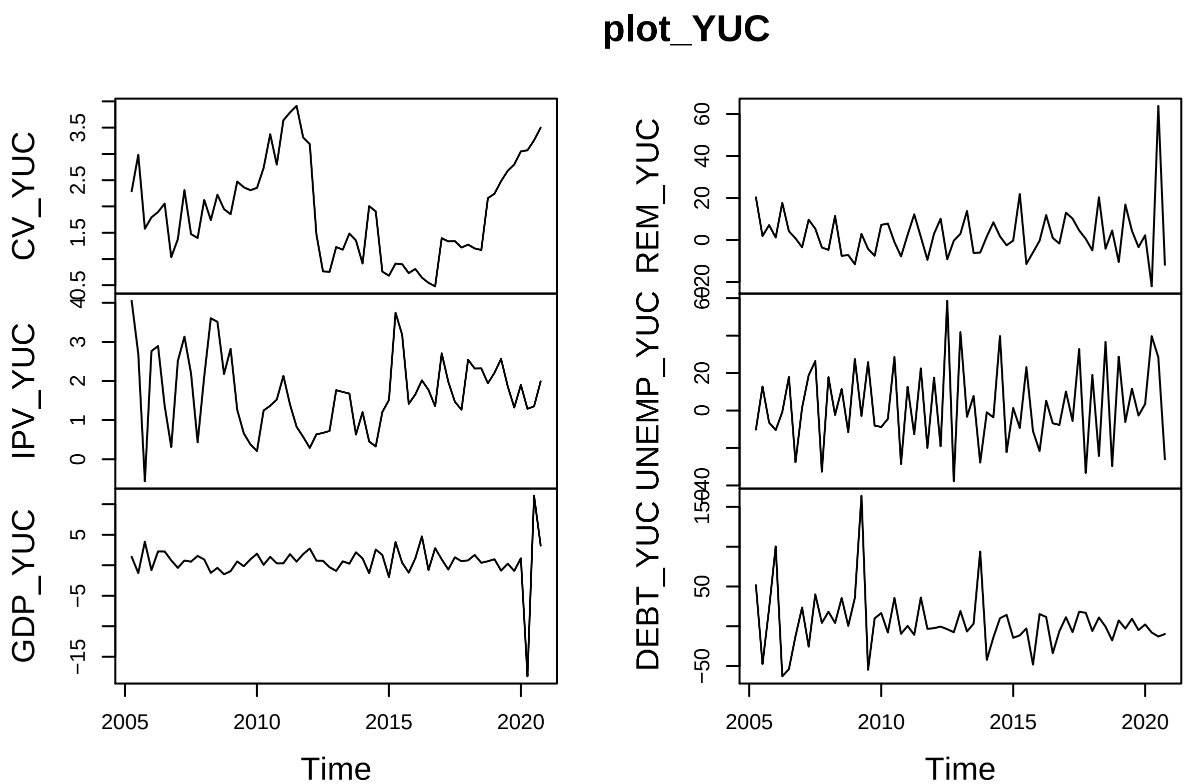

```
CV_TLAX <- TLAX[ ,5]
IPV_TLAX <- TLAX[ ,4]
GDP_TLAX <- TLAX[,6]
REM_TLAX <- TLAX[,10]
UNEMP_TLAX <- TLAX[,11]
DEBT_TLAX <- TLAX[,12]
plot_TLAX<-ts(cbind(CV_TLAX,IPV_TLAX,GDP_TLAX,REM_TLAX,UNEMP_TLAX,DEBT_TLAX),start=c(2005,2),frequency=4
plot(plot_TLAX)
```



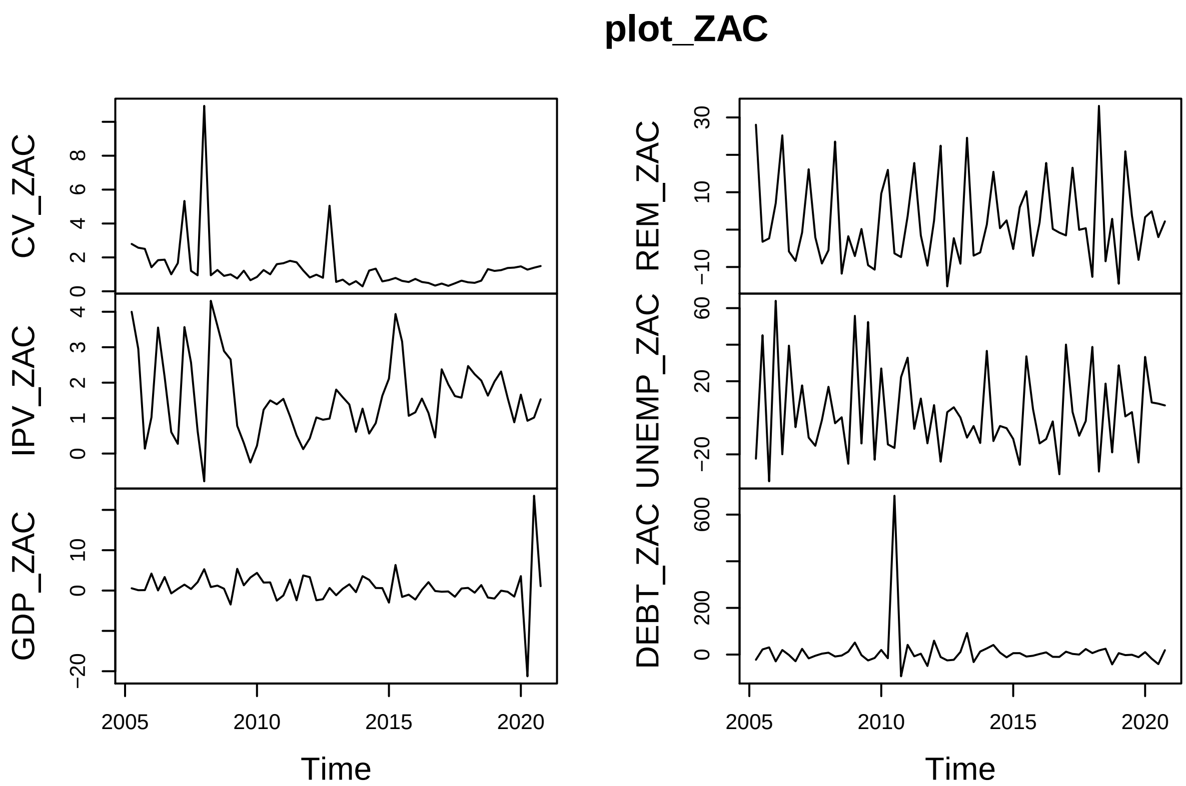

```
CV_VER <- VER[ ,5]
IPV_VER <- VER[ ,4]
GDP_VER <- VER[,6]
REM_VER <- VER[,10]
UNEMP_VER <- VER[,11]
DEBT_VER <- VER[,12]
plot_VER<-ts(cbind(CV_VER,IPV_VER,GDP_VER,REM_VER,UNEMP_VER,DEBT_VER),start=c(2005,2),frequency=4)
plot(plot_VER)
```




```r
CV_YUC <- YUC[ ,5]
IPV_YUC <- YUC[ ,4]
GDP_YUC <- YUC[,6]
REM_YUC <- YUC[,10]
UNEMP_YUC <- YUC[,11]
DEBT_YUC <- YUC[,12]
plot_YUC<-ts(cbind(CV_YUC,IPV_YUC,GDP_YUC,REM_YUC,UNEMP_YUC,DEBT_YUC),start=c(2005,2),frequency=4)
plot(plot_YUC)
```




```
CV_ZAC <- ZAC[ ,5]
IPV_ZAC <- ZAC[ ,4]
GDP_ZAC <- ZAC[,6]
REM_ZAC <- ZAC[,10]
UNEMP_ZAC <- ZAC[,11]
DEBT_ZAC <- ZAC[,12]
plot_ZAC<-ts(cbind(CV_ZAC,IPV_ZAC,GDP_ZAC,REM_ZAC,UNEMP_ZAC,DEBT_ZAC),start=c(2005,2),frequency=4)
plot(plot_ZAC)
```



4.2 Correlation

```r
round(cor(plot_AGS),2)
```

```
##           CV_AGS IPV_AGS GDP_AGS REM_AGS UNEMP_AGS DEBT_AGS
## CV_AGS      1.00    0.14    0.01   -0.01      0.00     0.11
## IPV_AGS     0.14    1.00   -0.10    0.37      0.04     0.18
## GDP_AGS     0.01   -0.10    1.00   -0.08     -0.02     0.18
## REM_AGS    -0.01    0.37   -0.08    1.00     -0.16     0.00
## UNEMP_AGS   0.00    0.04   -0.02   -0.16      1.00    -0.14
## DEBT_AGS    0.11    0.18    0.18    0.00     -0.14     1.00
```

```r
round(cor(plot_BC),2)
```

```
##          CV_BC IPV_BC GDP_BC REM_BC UNEMP_BC DEBT_BC
## CV_BC     1.00  -0.28   0.01   0.10     0.00   -0.04
## IPV_BC   -0.28   1.00  -0.01   0.35     0.01    0.12
## GDP_BC    0.01  -0.01   1.00  -0.16    -0.08   -0.06
## REM_BC    0.10   0.35  -0.16   1.00    -0.08    0.24
## UNEMP_BC  0.00   0.01  -0.08  -0.08     1.00    0.14
## DEBT_BC  -0.04   0.12  -0.06   0.24     0.14    1.00
```

```r
round(cor(plot_BCS),2)
```

```
##           CV_BCS IPV_BCS GDP_BCS REM_BCS UNEMP_BCS DEBT_BCS
## CV_BCS      1.00   -0.12    0.08    0.03     -0.08    -0.12
## IPV_BCS    -0.12    1.00    0.05    0.17     -0.01     0.03
## GDP_BCS     0.08    0.05    1.00    0.08     -0.11     0.05
## REM_BCS     0.03    0.17    0.08    1.00      0.15    -0.13
## UNEMP_BCS  -0.08   -0.01   -0.11    0.15      1.00     0.15
## DEBT_BCS   -0.12    0.03    0.05   -0.13      0.15     1.00
```

```r
round(cor(plot_CAMP),2)
```

```
##            CV_CAMP IPV_CAMP GDP_CAMP REM_CAMP UNEMP_CAMP DEBT_CAMP
## CV_CAMP       1.00     0.15     0.02     0.17      -0.09     -0.09
## IPV_CAMP      0.15     1.00     0.09     0.27       0.10     -0.09
## GDP_CAMP      0.02     0.09     1.00    -0.05      -0.14     -0.02
## REM_CAMP      0.17     0.27    -0.05     1.00       0.18      0.14
## UNEMP_CAMP   -0.09     0.10    -0.14     0.18       1.00      0.15
## DEBT_CAMP    -0.09    -0.09    -0.02     0.14       0.15      1.00
```

```r
round(cor(plot_CHIS),2)
```

```
##            CV_CHIS IPV_CHIS GDP_CHIS REM_CHIS UNEMP_CHIS DEBT_CHIS
## CV_CHIS       1.00    -0.15     0.14    -0.01       0.04      0.14
## IPV_CHIS     -0.15     1.00     0.02     0.35      -0.07      0.16
## GDP_CHIS      0.14     0.02     1.00     0.11       0.05      0.22
## REM_CHIS     -0.01     0.35     0.11     1.00      -0.01      0.01
## UNEMP_CHIS    0.04    -0.07     0.05    -0.01       1.00     -0.09
## DEBT_CHIS     0.14     0.16     0.22     0.01      -0.09      1.00
```

```r
round(cor(plot_CHIH),2)
```

```
##            CV_CHIH IPV_CHIH GDP_CHIH REM_CHIH UNEMP_CHIH DEBT_CHIH
## CV_CHIH       1.00    -0.01    -0.09    -0.02       0.09      0.21
## IPV_CHIH     -0.01     1.00    -0.09     0.36       0.10     -0.08
## GDP_CHIH     -0.09    -0.09     1.00    -0.20      -0.14     -0.11
## REM_CHIH     -0.02     0.36    -0.20     1.00      -0.14     -0.16
## UNEMP_CHIH    0.09     0.10    -0.14    -0.14       1.00     -0.05
## DEBT_CHIH     0.21    -0.08    -0.11    -0.16      -0.05      1.00
```

```r
round(cor(plot_CDMX),2)
```

```
##            CV_CDMX IPV_CDMX GDP_CDMX REM_CDMX UNEMP_CDMX DEBT_CDMX
## CV_CDMX       1.00    -0.42    -0.20     0.07       0.15      0.07
## IPV_CDMX     -0.42     1.00     0.22     0.06      -0.04     -0.04
## GDP_CDMX     -0.20     0.22     1.00    -0.02      -0.11     -0.14
## REM_CDMX      0.07     0.06    -0.02     1.00       0.30     -0.07
## UNEMP_CDMX    0.15    -0.04    -0.11     0.30       1.00      0.15
## DEBT_CDMX     0.07    -0.04    -0.14    -0.07       0.15      1.00
```

```r
round(cor(plot_COAH),2)
```

```
##            CV_COAH IPV_COAH GDP_COAH REM_COAH UNEMP_COAH DEBT_COAH
## CV_COAH       1.00     0.01     0.07     0.05      -0.04      0.15
## IPV_COAH      0.01     1.00    -0.02     0.17       0.19     -0.04
## GDP_COAH      0.07    -0.02     1.00    -0.04      -0.20      0.04
## REM_COAH      0.05     0.17    -0.04     1.00      -0.17     -0.02
## UNEMP_COAH   -0.04     0.19    -0.20    -0.17       1.00     -0.01
## DEBT_COAH     0.15    -0.04     0.04    -0.02      -0.01      1.00
```

```r
round(cor(plot_COL),2)
```

```
##           CV_COL IPV_COL GDP_COL REM_COL UNEMP_COL DEBT_COL
## CV_COL      1.00   -0.13    0.04    0.04      0.02     0.13
## IPV_COL    -0.13    1.00    0.18    0.15     -0.10     0.23
## GDP_COL     0.04    0.18    1.00    0.01     -0.18     0.12
## REM_COL     0.04    0.15    0.01    1.00     -0.15     0.07
## UNEMP_COL   0.02   -0.10   -0.18   -0.15      1.00    -0.16
## DEBT_COL    0.13    0.23    0.12    0.07     -0.16     1.00
```

```r
round(cor(plot_DGO),2)
```

```
##           CV_DGO IPV_DGO GDP_DGO REM_DGO UNEMP_DGO DEBT_DGO
## CV_DGO      1.00   -0.13    0.01   -0.06     -0.02     0.11
## IPV_DGO    -0.13    1.00   -0.05    0.30      0.21     0.10
## GDP_DGO     0.01   -0.05    1.00   -0.03      0.03    -0.02
## REM_DGO    -0.06    0.30   -0.03    1.00     -0.24     0.07
## UNEMP_DGO  -0.02    0.21    0.03   -0.24      1.00     0.10
## DEBT_DGO    0.11    0.10   -0.02    0.07      0.10     1.00
```

```r
round(cor(plot_GTO),2)
```

```
##           CV_GTO IPV_GTO GDP_GTO REM_GTO UNEMP_GTO DEBT_GTO
## CV_GTO      1.00   -0.14    0.04    0.02      0.03     0.12
## IPV_GTO    -0.14    1.00   -0.07    0.36      0.10     0.19
## GDP_GTO     0.04   -0.07    1.00   -0.14     -0.08    -0.10
## REM_GTO     0.02    0.36   -0.14    1.00     -0.13     0.06
## UNEMP_GTO   0.03    0.10   -0.08   -0.13      1.00    -0.14
## DEBT_GTO    0.12    0.19   -0.10    0.06     -0.14     1.00
```

```r
round(cor(plot_GRO),2)
```

```
##           CV_GRO IPV_GRO GDP_GRO REM_GRO UNEMP_GRO DEBT_GRO
## CV_GRO      1.00   -0.20    0.00   -0.15      0.03     0.22
## IPV_GRO    -0.20    1.00   -0.09    0.31     -0.05    -0.06
## GDP_GRO     0.00   -0.09    1.00    0.13     -0.09    -0.05
## REM_GRO    -0.15    0.31    0.13    1.00     -0.26    -0.22
## UNEMP_GRO   0.03   -0.05   -0.09   -0.26      1.00     0.10
## DEBT_GRO    0.22   -0.06   -0.05   -0.22      0.10     1.00
```

```
round(cor(plot_HGO),2)
```

```
##           CV_HGO IPV_HGO GDP_HGO REM_HGO UNEMP_HGO DEBT_HGO
## CV_HGO      1.00   -0.04   -0.11    0.11     -0.04     0.07
## IPV_HGO    -0.04    1.00   -0.05    0.25     -0.07     0.19
## GDP_HGO    -0.11   -0.05    1.00   -0.02     -0.24    -0.15
## REM_HGO     0.11    0.25   -0.02    1.00     -0.10    -0.06
## UNEMP_HGO  -0.04   -0.07   -0.24   -0.10      1.00     0.00
## DEBT_HGO    0.07    0.19   -0.15   -0.06      0.00     1.00
```

```
round(cor(plot_JAL),2)
```

```
##           CV_JAL IPV_JAL GDP_JAL REM_JAL UNEMP_JAL DEBT_JAL
## CV_JAL      1.00   -0.29   -0.02   -0.05      0.03     0.33
## IPV_JAL    -0.29    1.00   -0.13    0.23      0.03    -0.02
## GDP_JAL    -0.02   -0.13    1.00   -0.01     -0.18    -0.14
## REM_JAL    -0.05    0.23   -0.01    1.00     -0.14    -0.04
## UNEMP_JAL   0.03    0.03   -0.18   -0.14      1.00    -0.04
## DEBT_JAL    0.33   -0.02   -0.14   -0.04     -0.04     1.00
```

```
round(cor(plot_MEX),2)
```

```
##           CV_MEX IPV_MEX GDP_MEX REM_MEX UNEMP_MEX DEBT_MEX
## CV_MEX      1.00   -0.11    0.02    0.02      0.15     0.03
## IPV_MEX    -0.11    1.00   -0.05    0.45      0.03     0.04
## GDP_MEX     0.02   -0.05    1.00    0.22     -0.17    -0.03
## REM_MEX     0.02    0.45    0.22    1.00     -0.11     0.08
## UNEMP_MEX   0.15    0.03   -0.17   -0.11      1.00     0.12
## DEBT_MEX    0.03    0.04   -0.03    0.08      0.12     1.00
```

```
round(cor(plot_MICH),2)
```

```
##            CV_MICH IPV_MICH GDP_MICH REM_MICH UNEMP_MICH DEBT_MICH
## CV_MICH       1.00    -0.24    -0.14    -0.03      -0.01      0.00
## IPV_MICH     -0.24     1.00    -0.01     0.32       0.10      0.04
## GDP_MICH     -0.14    -0.01     1.00    -0.01       0.03      0.16
## REM_MICH     -0.03     0.32    -0.01     1.00       0.00     -0.03
## UNEMP_MICH   -0.01     0.10     0.03     0.00       1.00      0.03
## DEBT_MICH     0.00     0.04     0.16    -0.03       0.03      1.00
```

```
round(cor(plot_MOR),2)
```

```
##           CV_MOR IPV_MOR GDP_MOR REM_MOR UNEMP_MOR DEBT_MOR
## CV_MOR      1.00   -0.18   -0.06    0.08      0.03     0.06
## IPV_MOR    -0.18    1.00   -0.03    0.35     -0.04     0.10
## GDP_MOR    -0.06   -0.03    1.00    0.05     -0.04    -0.13
## REM_MOR     0.08    0.35    0.05    1.00      0.04     0.03
## UNEMP_MOR   0.03   -0.04   -0.04    0.04      1.00     0.00
## DEBT_MOR    0.06    0.10   -0.13    0.03      0.00     1.00
```

```r
round(cor(plot_NAY),2)
```

```
##           CV_NAY IPV_NAY GDP_NAY REM_NAY UNEMP_NAY DEBT_NAY
## CV_NAY      1.00   -0.08   -0.06    0.04      0.19     0.27
## IPV_NAY    -0.08    1.00    0.00    0.28      0.14     0.08
## GDP_NAY    -0.06    0.00    1.00    0.05      0.00     0.05
## REM_NAY     0.04    0.28    0.05    1.00      0.29     0.09
## UNEMP_NAY   0.19    0.14    0.00    0.29      1.00    -0.04
## DEBT_NAY    0.27    0.08    0.05    0.09     -0.04     1.00
```

```r
round(cor(plot_NL),2)
```

```
##          CV_NL IPV_NL GDP_NL REM_NL UNEMP_NL DEBT_NL
## CV_NL     1.00  -0.30   0.01  -0.14    -0.02   -0.02
## IPV_NL   -0.30   1.00   0.00   0.22     0.16    0.03
## GDP_NL    0.01   0.00   1.00   0.09    -0.20    0.01
## REM_NL   -0.14   0.22   0.09   1.00     0.06   -0.16
## UNEMP_NL -0.02   0.16  -0.20   0.06     1.00   -0.02
## DEBT_NL  -0.02   0.03   0.01  -0.16    -0.02    1.00
```

```r
round(cor(plot_OAXACA),2)
```

```
##              CV_OAXACA IPV_OAXACA GDP_OAXACA REM_OAXACA UNEMP_OAXACA
## CV_OAXACA         1.00      -0.09       0.03       0.01        -0.04
## IPV_OAXACA       -0.09       1.00       0.08       0.32         0.05
## GDP_OAXACA        0.03       0.08       1.00       0.22        -0.03
## REM_OAXACA        0.01       0.32       0.22       1.00        -0.23
## UNEMP_OAXACA     -0.04       0.05      -0.03      -0.23         1.00
## DEBT_OAXACA      -0.06      -0.02      -0.12      -0.16         0.16
##              DEBT_OAXACA
## CV_OAXACA          -0.06
## IPV_OAXACA         -0.02
## GDP_OAXACA         -0.12
## REM_OAXACA         -0.16
## UNEMP_OAXACA        0.16
## DEBT_OAXACA         1.00
```

```r
round(cor(plot_PUE),2)
```

```
##           CV_PUE IPV_PUE GDP_PUE REM_PUE UNEMP_PUE DEBT_PUE
## CV_PUE      1.00   -0.10    0.06    0.11      0.17     0.05
## IPV_PUE    -0.10    1.00   -0.08    0.29      0.12     0.12
## GDP_PUE     0.06   -0.08    1.00    0.38     -0.14     0.12
## REM_PUE     0.11    0.29    0.38    1.00     -0.07     0.09
## UNEMP_PUE   0.17    0.12   -0.14   -0.07      1.00     0.02
## DEBT_PUE    0.05    0.12    0.12    0.09      0.02     1.00
```

```r
round(cor(plot_QRO),2)
```

```
##           CV_QRO IPV_QRO GDP_QRO REM_QRO UNEMP_QRO DEBT_QRO
## CV_QRO      1.00   -0.18    0.05   -0.04      0.02     0.01
## IPV_QRO    -0.18    1.00   -0.04    0.32      0.02     0.00
## GDP_QRO     0.05   -0.04    1.00    0.00     -0.21     0.04
## REM_QRO    -0.04    0.32    0.00    1.00      0.26    -0.15
## UNEMP_QRO   0.02    0.02   -0.21    0.26      1.00    -0.01
## DEBT_QRO    0.01    0.00    0.04   -0.15     -0.01     1.00
```

```r
round(cor(plot_Q_ROO),2)
```

```
##             CV_Q_ROO IPV_Q_ROO GDP_Q_ROO REM_Q_ROO UNEMP_Q_ROO DEBT_Q_ROO
## CV_Q_ROO        1.00     -0.31      0.01      0.04       -0.05       0.10
## IPV_Q_ROO      -0.31      1.00      0.07      0.08       -0.01      -0.02
## GDP_Q_ROO       0.01      0.07      1.00      0.32       -0.50      -0.27
## REM_Q_ROO       0.04      0.08      0.32      1.00        0.26       0.00
## UNEMP_Q_ROO    -0.05     -0.01     -0.50      0.26        1.00       0.30
## DEBT_Q_ROO      0.10     -0.02     -0.27      0.00        0.30       1.00
```

```r
round(cor(plot_SLP),2)
```

```
##           CV_SLP IPV_SLP GDP_SLP REM_SLP UNEMP_SLP DEBT_SLP
## CV_SLP      1.00   -0.10   -0.01    0.04      0.03     0.11
## IPV_SLP    -0.10    1.00   -0.02    0.41      0.14     0.06
## GDP_SLP    -0.01   -0.02    1.00   -0.04     -0.02    -0.07
## REM_SLP     0.04    0.41   -0.04    1.00      0.05     0.00
## UNEMP_SLP   0.03    0.14   -0.02    0.05      1.00     0.01
## DEBT_SLP    0.11    0.06   -0.07    0.00      0.01     1.00
```

```r
round(cor(plot_SIN),2)
```

```
##           CV_SIN IPV_SIN GDP_SIN REM_SIN UNEMP_SIN DEBT_SIN
## CV_SIN      1.00   -0.03    0.04    0.08      0.06     0.04
## IPV_SIN    -0.03    1.00   -0.03    0.32      0.19    -0.02
## GDP_SIN     0.04   -0.03    1.00   -0.21      0.05    -0.23
## REM_SIN     0.08    0.32   -0.21    1.00     -0.01     0.03
## UNEMP_SIN   0.06    0.19    0.05   -0.01      1.00     0.07
## DEBT_SIN    0.04   -0.02   -0.23    0.03      0.07     1.00
```

```r
round(cor(plot_SON),2)
```

```
##           CV_SON IPV_SON GDP_SON REM_SON UNEMP_SON DEBT_SON
## CV_SON      1.00    0.01    0.06    0.08      0.10     0.02
## IPV_SON     0.01    1.00   -0.07    0.37      0.03     0.09
## GDP_SON     0.06   -0.07    1.00   -0.18     -0.06    -0.08
## REM_SON     0.08    0.37   -0.18    1.00     -0.10     0.04
## UNEMP_SON   0.10    0.03   -0.06   -0.10      1.00     0.16
## DEBT_SON    0.02    0.09   -0.08    0.04      0.16     1.00
```

```r
round(cor(plot_TAB),2)
```

```
##           CV_TAB IPV_TAB GDP_TAB REM_TAB UNEMP_TAB DEBT_TAB
## CV_TAB      1.00   -0.13    0.21   -0.06     -0.12    -0.15
## IPV_TAB    -0.13    1.00   -0.10    0.29      0.00    -0.09
## GDP_TAB     0.21   -0.10    1.00    0.00     -0.15     0.02
## REM_TAB    -0.06    0.29    0.00    1.00     -0.19    -0.08
## UNEMP_TAB  -0.12    0.00   -0.15   -0.19      1.00     0.19
## DEBT_TAB   -0.15   -0.09    0.02   -0.08      0.19     1.00
```

```r
round(cor(plot_TAMPS),2)
```

```
##             CV_TAMPS IPV_TAMPS GDP_TAMPS REM_TAMPS UNEMP_TAMPS DEBT_TAMPS
## CV_TAMPS        1.00      0.01     -0.05      0.00       -0.04      -0.12
## IPV_TAMPS       0.01      1.00      0.07      0.21        0.17      -0.10
## GDP_TAMPS      -0.05      0.07      1.00      0.02       -0.07       0.05
## REM_TAMPS       0.00      0.21      0.02      1.00       -0.07      -0.01
## UNEMP_TAMPS    -0.04      0.17     -0.07     -0.07        1.00      -0.07
## DEBT_TAMPS     -0.12     -0.10      0.05     -0.01       -0.07       1.00
```

```r
round(cor(plot_TLAX),2)
```

```
##            CV_TLAX IPV_TLAX GDP_TLAX REM_TLAX UNEMP_TLAX DEBT_TLAX
## CV_TLAX       1.00     0.04    -0.03     0.00       0.03      0.04
## IPV_TLAX      0.04     1.00    -0.05     0.26       0.25      0.05
## GDP_TLAX     -0.03    -0.05     1.00     0.14      -0.25      0.03
## REM_TLAX      0.00     0.26     0.14     1.00      -0.03      0.02
## UNEMP_TLAX    0.03     0.25    -0.25    -0.03       1.00     -0.16
## DEBT_TLAX     0.04     0.05     0.03     0.02      -0.16      1.00
```

```r
round(cor(plot_VER),2)
```

```
##           CV_VER IPV_VER GDP_VER REM_VER UNEMP_VER DEBT_VER
## CV_VER      1.00    0.00   -0.13    0.15      0.02    -0.10
## IPV_VER     0.00    1.00    0.05    0.26     -0.12     0.20
## GDP_VER    -0.13    0.05    1.00    0.01     -0.03    -0.05
## REM_VER     0.15    0.26    0.01    1.00     -0.16     0.03
## UNEMP_VER   0.02   -0.12   -0.03   -0.16      1.00     0.25
## DEBT_VER   -0.10    0.20   -0.05    0.03      0.25     1.00
```

```r
round(cor(plot_YUC),2)
```

```
##           CV_YUC IPV_YUC GDP_YUC REM_YUC UNEMP_YUC DEBT_YUC
## CV_YUC      1.00   -0.06   -0.04    0.06      0.14     0.05
## IPV_YUC    -0.06    1.00   -0.07    0.15      0.13     0.03
## GDP_YUC    -0.04   -0.07    1.00    0.56     -0.14     0.03
## REM_YUC     0.06    0.15    0.56    1.00     -0.06    -0.07
## UNEMP_YUC   0.14    0.13   -0.14   -0.06      1.00    -0.04
## DEBT_YUC    0.05    0.03    0.03   -0.07     -0.04     1.00
```

```r
round(cor(plot_ZAC),2)
```

```
##            CV_ZAC IPV_ZAC GDP_ZAC REM_ZAC UNEMP_ZAC DEBT_ZAC
## CV_ZAC       1.00   -0.13    0.12    0.01      0.07    -0.05
## IPV_ZAC     -0.13    1.00   -0.07    0.34      0.01     0.01
## GDP_ZAC      0.12   -0.07    1.00   -0.02      0.02     0.02
## REM_ZAC      0.01    0.34   -0.02    1.00     -0.29    -0.03
## UNEMP_ZAC    0.07    0.01    0.02   -0.29      1.00    -0.08
## DEBT_ZAC    -0.05    0.01    0.02   -0.03     -0.08     1.00
```

4.3 Data Analysis

4.3.1 Delincquency rates by State

```r
library("ellipsis")
```

```
## Warning: package 'ellipsis' was built under R version 4.0.2
```

```r
library("cli")
```

```
## Warning: package 'cli' was built under R version 4.0.2
```

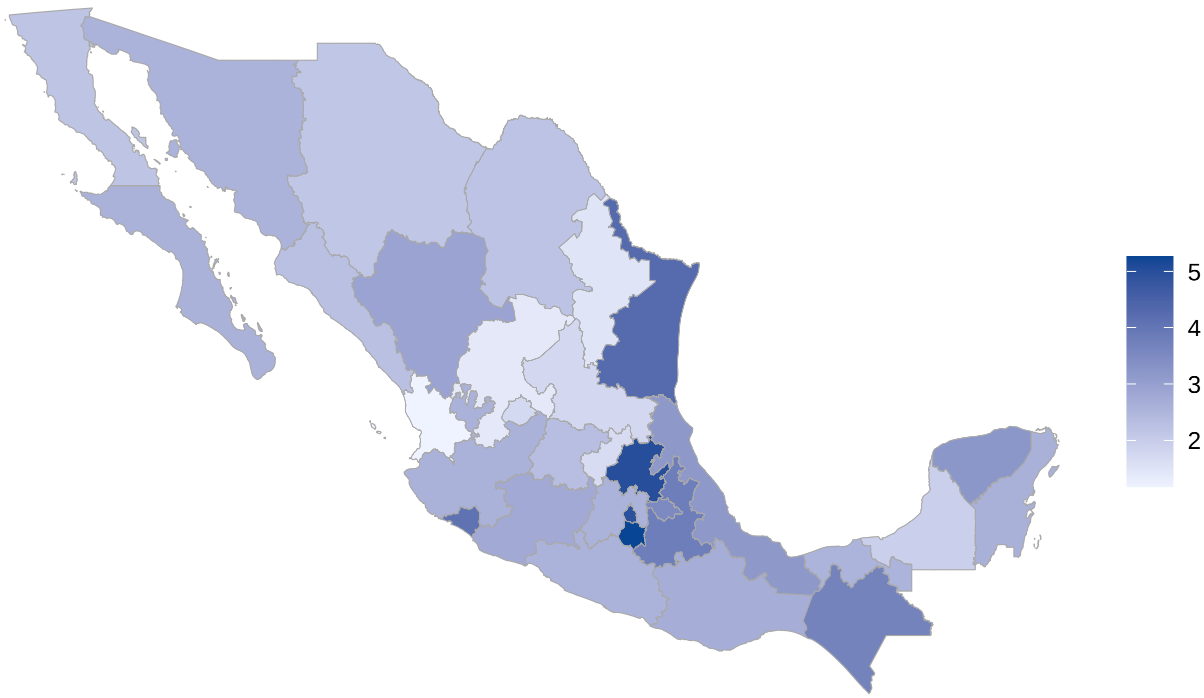
```r
library("devtools")
```

```
## Warning: package 'devtools' was built under R version 4.0.2
```

```
## Error in get(genname, envir = envir) : object 'testthat_print' not found
```

```r
library("mxmaps")

CV_STATES<-cbind(CV_AGS,CV_BC,CV_BCS,CV_CAMP,CV_CDMX,CV_CHIH,CV_CHIS,CV_COAH,CV_COL,CV_DGO,CV_GRO,CV_GT

KEY<-c(01,  02, 03, 04, 09, 08, 07, 05, 06, 10, 12, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 23, 22, 25,

CV_2020<-CV_STATES[60:63,]

CV_2020_MEAN<-c(mean(CV_2020[,1]),mean(CV_2020[,2]),mean(CV_2020[,3]),mean(CV_2020[,4]),mean(CV_2020[,5

MAPCV2020<-as.data.frame(t(rbind(CV_2020_MEAN,KEY)))
colnames(MAPCV2020)<-c("value","region")

MAP <- mxstate_choropleth(MAPCV2020, num_colors=1, title="Average delincquency rate by state (2020)")
MAP
```

Average delincquency rate by state (2020)



4.3.2 GDP by State

4.3.3

5 Stationarity

(development of state level models is still pending)