Final_VAR

Contents

Methodology

SVAR

We develop a SVAR model in order to assess the impact of short-run shocks from Trump's posts on AHV, and to evaluate whether market volatility can, in turn, influence Trump's posting behavior. In this framework, we systematically pair AHV with one explanatory variable at a time (our X-regressor). The SVAR approach offers the advantage of accounting for structural endogeneity. Our main assumption is that the volatility does not contemporaneously affect Trump's posting activity - neither quantitatively nor qualitatively, while Trump's posts do affect markets instantly. In essence, we impose a short-run restriction on the shock of volatility for all the social media variables.

Based on the information criteria, we found similar results across all specifications, with a recommended lag length of around 70. However, inducing more than 6 lags (corresponding to a full trading day) introduces strong seasonality. Moreover, the higher the number of lags, the greater the persistence of a shock, up to unrealistic levels such as 150 days for the number of Tweets, which seems implausible. Therefore, we chose to fix the number of lags at a maximum of 6. Finally, given the presence of heteroscedasticity and serial correlation in the residuals, we use the Newey-West estimator to compute robust standard errors.

Results

Full Timeframe

As in the ARMA-X framework, we initially estimate a model for each of our five main variables. For all estimations using the SPY ETF market, the significant coefficients of social media variables were consistently negative, while the positive coefficients were large but not significant. (*desc1figVAR*) For the $Tariff$, $Trade$ and $China$ mention variables, the first, second and sometimes forth lag were positive and relatively large (especially in the case of Tariff), while the remaining ones were not. In contrast, for $TweetCount$ and $TweetDummy$, we observed fewer and smaller positive coefficients. At the same time, we found that the contemporaneous effects of the shocks were all positive and relatively strong. This leads to two types of scenarios : either the IRF's experienced a positive shock and remain elevated ($Tariff$, $Trade$ and $China$), or a highly positive shock occurs, but the cumulative effect turns negative after few hours ($TweetDummy$ and $Count$). *IRFtarifVAR IRFdummyVAR*

Finally, except for $Tariff$, all Granger causality tests indicate that Trump's posts have an impact on volatility. However, due to serial correlation in the residuals, these results should be interpreted with caution. Overall, this model suggest that Trump's posts tend to have a positive instantaneous effect on volatility, but with very low persistence. When analyzing the European and Chinese ETF's, we observe similar patterns though with lower magnitude, except for the impact of $Tariff$ and $China$ on the ASHR ETF (Chinese Market), where the cumulative effects show no positive impact. Additionally, the VGK ETF (European Market) appears to react more strongly than ASHR to Trump's posts, especially those mentioning $Trade$ and $Tariff$.

Regarding the impact of AHV on Trump's posts, we find some evidence of a negative effect. For all variables, we observe one or two significantly negative coefficients per variables, typically on the first and fourth lag, alongside many insignificant ones. However, only $TweetCount$ and China pass the Granger test in the US ETF Surprisingly, a large number of Granger tests in the Chinese and European ETFs indicate strong Granger causality, which may point to a limitation of the test itself, as such results appear unrealistic.

Split Sample

For the split framework, the results were similar and striking $IRFchinasecondtermVAR$. While we observed relatively small shock effects and almost entirely negative coefficients during the first term (*descFirsttermfig-*

VAR), leading the cumulative IRF's to indicate a negative impact of posts, the shock effects in the second term were substantially larger, ranging from 5 times (for $TweetCount$ and $TweetDummy$) to as much as 25 times greater (for $Tariff$). the only exception was $Trade$ in the second term, which showed the only negative impact from a shock. Once again, we found positive lagged coefficients in the second term, mostly on the first, second and forth lags. However, none of these coefficients were statistically significant, but the cumulative IRF's clearly show a high positive impact on everything except for $TweetDummy$ and $TweetCount$, whose coefficients and cumulative IRF's displayed similar patterns to those observed in the first term. (*descSecondtermfigVAR*) Moreover, the Granger tests generally failed in both terms, with the sole exception being $China$ in the second term. Regarding the ASHR ETF, we found results similar to those for the US ETF. Surprisingly, in the case of the European market, we observed a positive impact of Trump's posts on AHV during the first term. Nevertheless, the results still indicate a stronger impact of posts during the second term.

Dummy (Overall)

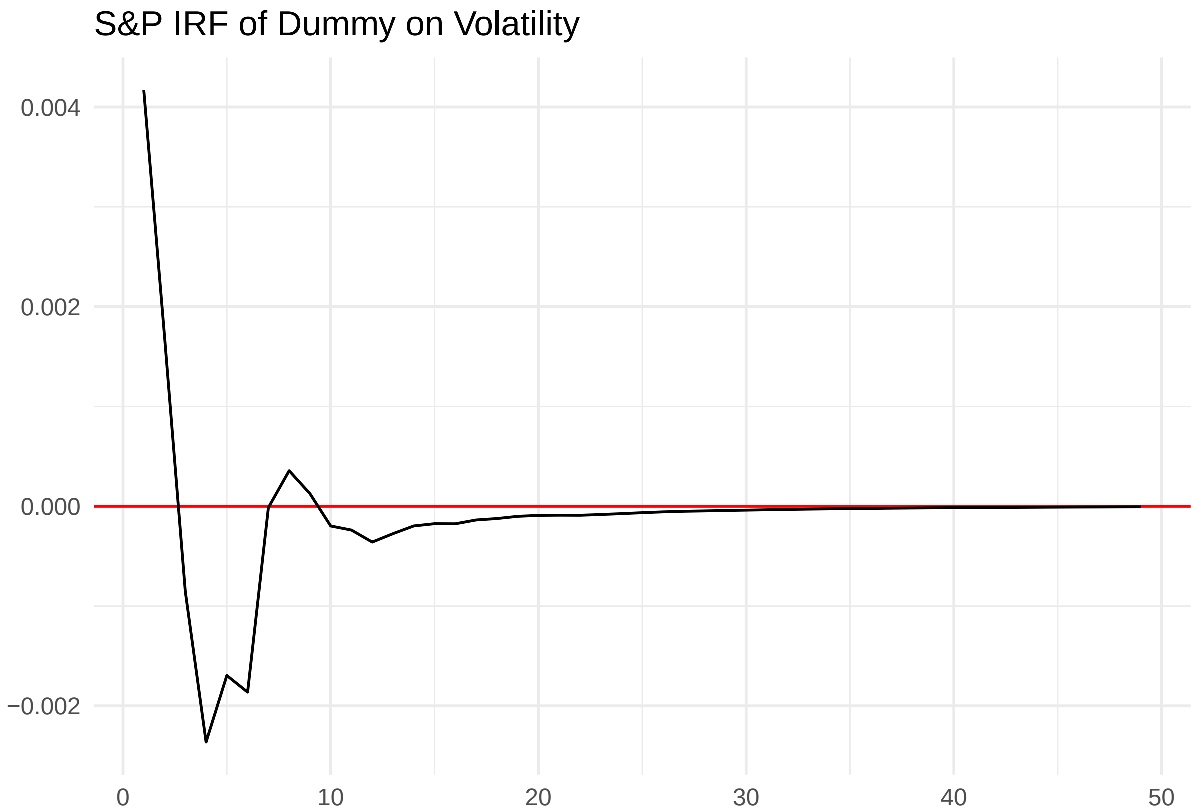
```r
y = cbind(Vdata$dummy, Vdata$SPY_vol)
colnames(y)[1:2] <- c("dummy", "vol")
est.VAR <- VAR(y,p=6)

#extract results
mod_vol = est.VAR$varresult$vol
f = formula(mod_vol)
d = model.frame(mod_vol)
lm_clean = lm(f, data= d)

#apply Newey-West
nw_vcov = NeweyWest(lm_clean, lag=6)
nw_se = sqrt(diag(nw_vcov))

#t-stats
coef = coef(lm_clean)
t_stat = coef/nw_se

#recalculate p-values
robust = 2*(1-pt(abs(t_stat), df = df.residual(lm_clean)))
```

IRF 1

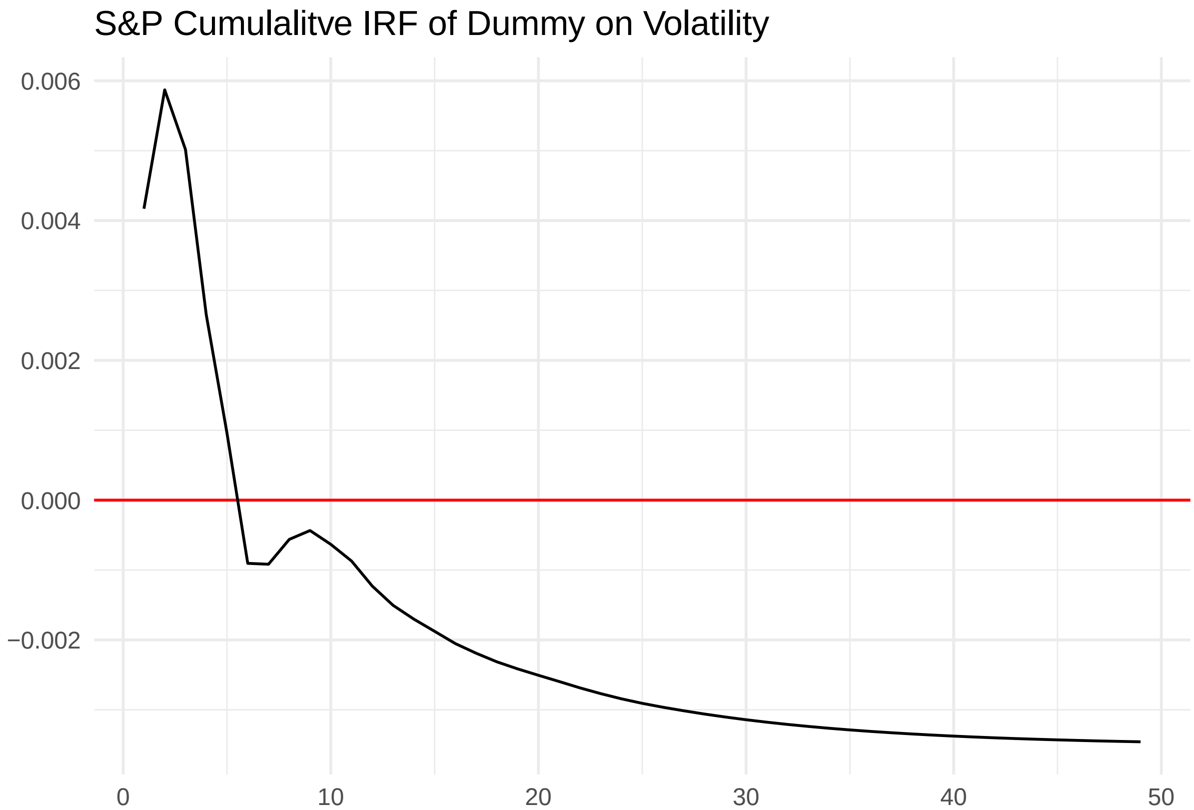
```r
#Recreate a Robust Omega Matrix
U = residuals(est.VAR)
T = nrow(U)
L = 6 #number of lag
Omega = matrix(0, ncol(U), ncol(U))
for(l in 0:L) {
  weight = 1 - l/(L+1)
  Gamma_l_ = t(U[(l+1):T, , drop=FALSE]) %*% U[1:(T-l), , drop=FALSE] /T
  if (l == 0){
    Omega = Omega + Gamma_l_
  } else {
    Omega = Omega + weight*(Gamma_l_ + t(Gamma_l_))
```

```r
  }
}


#make the B matrix
loss <- function(param){
  #Define the restriction
  B <- matrix(c(param[1], param[2], 0, param[3]), ncol = 2)

  #Make BB' approximatively equal to omega
  X <- Omega - B %*% t(B)

  #loss function
  loss <- sum(X^2)
  return(loss)
}

res.opt <- optim(c(1, 0, 1), loss, method = "BFGS")
B.hat <- matrix(c(res.opt$par[1], res.opt$par[2], 0, res.opt$par[3]), ncol = 2)
```

```r
nb.sim = 7*7
#get back the coefficient of est.VAR
phi <- Acoef(est.VAR)
PHI = make.PHI(phi)

#take the constant
constant <- sapply(est.VAR$varresult, function(eq) coef(eq)["const"])
c=as.matrix(constant)

#Simulate the IRF
p <- length(phi)
n <- dim(phi[[1]])[1]


Y <- simul.VAR(c=c, Phi = phi, B = B.hat, nb.sim ,y0.star=rep(0, n*p),
                  indic.IRF = 1, u.shock = c(1,0))



Yd = data.frame(
  period = 1:nrow(Y),
  response = Y[,2])

ggplot(Yd,aes(x=period, y=response)) +
  geom_hline(yintercept = 0, color="red") +
  geom_line() +
  theme_light() +
  ggtitle("S&P IRF of Dummy on Volatility")+
  ylab("")+
  xlab("") +
  theme_minimal()
```




```
ggplot(Yd,aes(x=period, y=cumsum(response))) +
  geom_hline(yintercept = 0, color="red") +
  geom_line() +
  theme_light() +
  ggtitle("S&P Cumulalitve IRF of Dummy on Volatility") +
  ylab("")+
  xlab("") +
  theme_minimal()
```



Tariff (Overall)

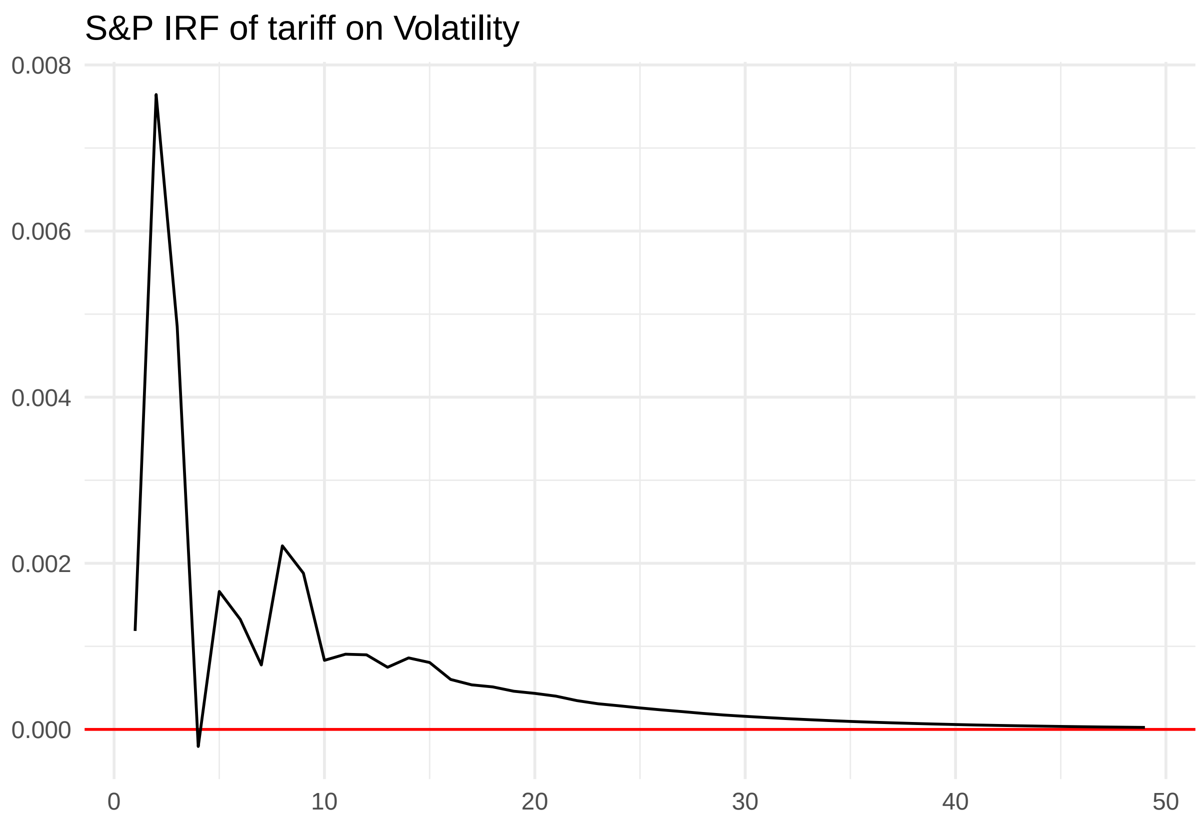
```
y3 = cbind(Vdata$tariff, Vdata$SPY_vol)
colnames(y3)[1:2] <- c("tariff", "vol")
est.VAR3 <- VAR(y3,p=6)

#extract results
mod_vol3 = est.VAR3$varresult$vol
f3 = formula(mod_vol3)
d3 = model.frame(mod_vol3)
lm_clean3 = lm(f3, data= d3)

#apply Newey-West
nw_vcov3 = NeweyWest(lm_clean3, lag=6)
nw_se3 = sqrt(diag(nw_vcov3))

#t-stats
coef3 = coef(lm_clean3)
t_stat3 = coef3/nw_se3

#recalculate p-values
robust3 = 2*(1-pt(abs(t_stat3), df = df.residual(lm_clean3)))
```

IRF 3

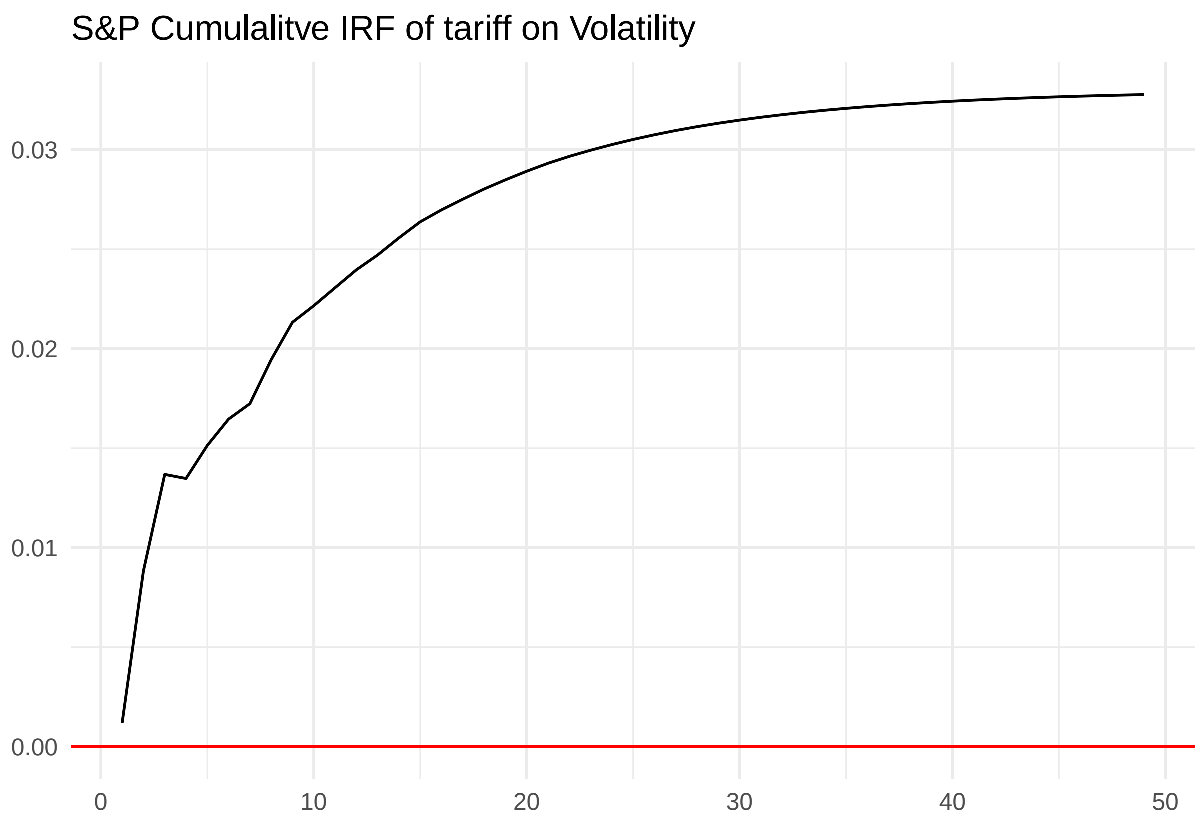
```
#Recreate a Robust Omega Matrix
U3 = residuals(est.VAR3)
T3 = nrow(U3)
Omega3 = matrix(0, ncol(U3), ncol(U3))
for(l in 0:L) {
  weight = 1 - l/(L+1)
  Gamma_l_3 = t(U3[(l+1):T3, , drop=FALSE]) %*% U3[1:(T3-l), , drop=FALSE] /T3
  if (l == 0){
    Omega3 = Omega3 + Gamma_l_3
  } else {
    Omega3 = Omega3 + weight*(Gamma_l_3 + t(Gamma_l_3))
  }
}


#make the B matrix
loss3 <- function(param3){
  #Define the restriction
  B3 <- matrix(c(param3[1], param3[2], 0, param3[3]), ncol = 2)

  #Make BB' approximatively equal to omega
  X3 <- Omega3 - B3 %*% t(B3)

  #loss function
  loss3 <- sum(X3^2)
  return(loss3)
}

res.opt3 <- optim(c(1, 0, 1), loss3, method = "BFGS")
B.hat3 <- matrix(c(res.opt3$par[1], res.opt3$par[2], 0, res.opt3$par[3]), ncol = 2)
```

```
#get back the coefficient of est.VAR
phi3 <- Acoef(est.VAR3)
PHI3 = make.PHI(phi3)

#take the constant
constant3 <- sapply(est.VAR3$varresult, function(eq) coef(eq)["const"])
c3=as.matrix(constant3)

#Simulate the IRF
p3 <- length(phi3)
n3 <- dim(phi3[[1]])[1]

Y3 <- simul.VAR(c=c3, Phi = phi3, B = B.hat3, nb.sim ,y0.star=rep(0, n3*p3),
                  indic.IRF = 1, u.shock = c(1,0))


#Plot the IRF
Yd3 = data.frame(
  period = 1:nrow(Y3),
  response = Y3[,2])
```

```r
ggplot(Yd3,aes(x=period, y=response)) +
  geom_hline(yintercept = 0, color="red") +
  geom_line() +
  theme_light() +
  ggtitle("S&P IRF of tariff on Volatility") +
  ylab("")+
  xlab("") +
  theme_minimal()
```




```r
ggplot(Yd3,aes(x=period, y=cumsum(response))) +
  geom_hline(yintercept = 0, color="red") +
  geom_line() +
  theme_light() +
  ggtitle("S&P Cumulalitve IRF of tariff on Volatility") +
  ylab("")+
  xlab("") +
  theme_minimal()
```



Tables

Dummy

```
y = cbind(Vdata$dummy, Vdata$SPY_vol)
colnames(y)[1:2] <- c("dummy", "vol")
est.VAR <- VAR(y,p=6)

#extract results
mod_vol = est.VAR$varresult$vol
f = formula(mod_vol)
d = model.frame(mod_vol)
lm_clean = lm(f, data= d)

#apply Newey-West
nw_vcov = NeweyWest(lm_clean, lag=6)
nw_se = sqrt(diag(nw_vcov))

#t-stats
coef = coef(lm_clean)
t_stat = coef/nw_se
```

```r
#recalculate p-values
robust = 2*(1-pt(abs(t_stat), df = df.residual(lm_clean)))
```

N

```r
y2 = cbind(Vdata$N, Vdata$SPY_vol)
colnames(y2)[1:2] <- c("N", "vol")
est.VAR2 <- VAR(y2,p=6)

#extract results
mod_vol2 = est.VAR2$varresult$vol
f2 = formula(mod_vol2)
d2 = model.frame(mod_vol2)
lm_clean2 = lm(f2, data= d2)

#apply Newey-West
nw_vcov2 = NeweyWest(lm_clean2, lag=6)
nw_se2 = sqrt(diag(nw_vcov2))

#t-stats
coef2 = coef(lm_clean2)
t_stat2 = coef2/nw_se2

#recalculate p-values
robust2 = 2*(1-pt(abs(t_stat2), df = df.residual(lm_clean2)))
```

Tariff

```r
y3 = cbind(Vdata$tariff, Vdata$SPY_vol)
colnames(y3)[1:2] <- c("tariff", "vol")
est.VAR3 <- VAR(y3,p=6)

#extract results
mod_vol3 = est.VAR3$varresult$vol
f3 = formula(mod_vol3)
d3 = model.frame(mod_vol3)
lm_clean3 = lm(f3, data= d3)

#apply Newey-West
nw_vcov3 = NeweyWest(lm_clean3, lag=6)
nw_se3 = sqrt(diag(nw_vcov3))

#t-stats
coef3 = coef(lm_clean3)
t_stat3 = coef3/nw_se3

#recalculate p-values
robust3 = 2*(1-pt(abs(t_stat3), df = df.residual(lm_clean3)))
```

Trade

```r
y4 = cbind(Vdata$trade, Vdata$SPY_vol)
colnames(y4)[1:2] <- c("trade", "vol")
est.VAR4 <- VAR(y4,p=6)

#extract results
mod_vol4 = est.VAR4$varresult$vol
f4 = formula(mod_vol4)
d4 = model.frame(mod_vol4)
lm_clean4 = lm(f4, data= d4)

#apply Newey-West
nw_vcov4 = NeweyWest(lm_clean4, lag=6)
nw_se4 = sqrt(diag(nw_vcov4))

#t-stats
coef4 = coef(lm_clean4)
t_stat4 = coef4/nw_se4

#recalculate p-values
robust4 = 2*(1-pt(abs(t_stat4), df = df.residual(lm_clean4)))
```

China

```r
ychina = cbind(Vdata$china, Vdata$SPY_vol)
colnames(ychina)[1:2] <- c("china", "vol")
est.VARchina <- VAR(ychina,p=6)

#extract results
mod_volchina = est.VARchina$varresult$vol
fchina = formula(mod_volchina)
dchina = model.frame(mod_volchina)
lm_cleanchina = lm(fchina, data= dchina)

#apply Newey-West
nw_vcovchina = NeweyWest(lm_cleanchina, lag=6)
nw_sechina = sqrt(diag(nw_vcovchina))

#t-stats
coefchina = coef(lm_cleanchina)
t_statchina = coefchina/nw_sechina

#recalculate p-values
robustchina = 2*(1-pt(abs(t_statchina), df = df.residual(lm_cleanchina)))
```

```r
dt_t = d %>%
    rename(X.l1 = dummy.l1,
    X.l2 = dummy.l2,
    X.l3 = dummy.l3,
```

```r
    X.l4 = dummy.l4,
    X.l5 = dummy.l5,
    X.l6 = dummy.l6)


f_t <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                         X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model <- lm(f_t, data = dt_t)

dt_t2 = d2 %>%
    rename(X.l1 = N.l1,
    X.l2 = N.l2,
    X.l3 = N.l3,
    X.l4 = N.l4,
    X.l5 = N.l5,
    X.l6 = N.l6)



f_t2 <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                         X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model2 <- lm(f_t2, data = dt_t2)


dt_t3 = d3 %>%
    rename(X.l1 = tariff.l1,
    X.l2 = tariff.l2,
    X.l3 = tariff.l3,
    X.l4 = tariff.l4,
    X.l5 = tariff.l5,
    X.l6 = tariff.l6)



f_t3 <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                         X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model3 <- lm(f_t3, data = dt_t3)


dt_t4 = d4 %>%
    rename(X.l1 = trade.l1,
    X.l2 = trade.l2,
    X.l3 = trade.l3,
    X.l4 = trade.l4,
    X.l5 = trade.l5,
    X.l6 = trade.l6)



f_t4 <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                         X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model4 <- lm(f_t4, data = dt_t4)
```

```r
dt_tchina = dchina %>%
    rename(X.l1 = china.l1,
    X.l2 = china.l2,
    X.l3 = china.l3,
    X.l4 = china.l4,
    X.l5 = china.l5,
    X.l6 = china.l6)



f_tchina <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
modelchina <- lm(f_tchina, data = dt_tchina)


nw_se_t <- sqrt(diag(sandwich::NeweyWest(model, lag = 6)))
nw_se2_t <- sqrt(diag(sandwich::NeweyWest(model2, lag = 6)))
nw_se3_t <- sqrt(diag(sandwich::NeweyWest(model3, lag = 6)))
nw_se4_t <- sqrt(diag(sandwich::NeweyWest(model4, lag = 6)))
nw_sechina_t <- sqrt(diag(sandwich::NeweyWest(modelchina, lag = 6)))


robust_t <- 2 * (1-pt(abs(coef(model) / nw_se_t), df = df.residual(model)))
robust2_t <- 2 * (1-pt(abs(coef(model2) / nw_se2_t), df = df.residual(model2)))
robust3_t <- 2 * (1-pt(abs(coef(model3) / nw_se3_t), df = df.residual(model3)))
robust4_t <- 2 * (1-pt(abs(coef(model4) / nw_se4_t), df = df.residual(model4)))
robustchina_t <- 2 * (1-pt(abs(coef(modelchina) / nw_sechina_t), df = df.residual(modelchina)))



nw_se_t       <- nw_se_t[names(coef(model))]
robust_t      <- robust_t[names(coef(model))]
nw_se2_t      <- nw_se2_t[names(coef(model2))]
robust2_t     <- robust2_t[names(coef(model2))]
nw_se3_t      <- nw_se3_t[names(coef(model3))]
robust3_t     <- robust3_t[names(coef(model3))]
nw_se4_t      <- nw_se4_t[names(coef(model4))]
robust4_t     <- robust4_t[names(coef(model4))]
nw_sechina_t  <- nw_sechina_t[names(coef(modelchina))]
robustchina_t <- robustchina_t[names(coef(modelchina))]




# Create list for models
models_list <- list(model, model2, model3, model4, modelchina)

# Create list for SE robustes
robust_ses <- list(nw_se_t, nw_se2_t, nw_se3_t, nw_se4_t, nw_sechina_t)

# Create list for p-value
robust_pvals <- list(robust_t, robust2_t, robust3_t, robust4_t, robustchina_t)
```

```r
# Name of Variables
custom_names <- list(
  "vol.l1" = "$AHV_{t-1}$",
  "vol.l2" = "$AHV_{t-2}$",
  "vol.l3" = "$AHV_{t-3}$",
  "vol.l4" = "$AHV_{t-4}$",
  "vol.l5" = "$AHV_{t-5}$",
  "vol.l6" = "$AHV_{t-6}$",
  "X.l1" = "$X_{t-1}$",
  "X.l2" = "$X_{t-2}$",
  "X.l3" = "$X_{t-3}$",
  "X.l4" = "$X_{t-4}$",
  "X.l5" = "$X_{t-5}$",
  "X.l6" = "$X_{t-6}$",
  "const" = "Constant"
)

# Generate table
table_texreg <- texreg(
  l = models_list,
  override.se = robust_ses,
  custom.coef.map = custom_names,
  override.pvalues = robust_pvals,
  custom.model.names = c("TweetDummy", "TweetCount", "Tariff", "Trade", "China"),
  caption = "VAR Models of Average Hourly Volatility",
  label = "tab:VAR_Second_Term",
  caption.above = TRUE,
  digits = 6,
  custom.gof.rows = list("Shock (IRF)" = c(0.0041713, 0.003061, 0.001189, 0.000215, 0.001937)),
  custom.note = "\\parbox[t]{0.9\\textwidth}{\\small\\textit{This table displays VAR regression with onl
)



# print
table_texreg
```

```r
# First and Second Mandate

#first term
Vdata_f = filter(data,between(timestamp, as.Date('2017-01-20'), as.Date('2021-01-20')))

#second term
Vdata_s = filter(data,between(timestamp, as.Date('2025-01-20'), as.Date('2025-05-07')))
```

First Term

Dummy

```r
y_f_d = cbind(Vdata_f$dummy, Vdata_f$SPY_vol)
colnames(y_f_d)[1:2] <- c("dummy", "vol")
```

Table 1: VAR Models of Average Hourly Volatility

	TweetDummy	TweetCount	Tariff	Trade	China
AHV_{t-1}	0.344511***	0.345011***	0.342081***	0.346107***	0.344512***
	(0.103790)	(0.104492)	(0.098665)	(0.101918)	(0.097994)
AHV_{t-2}	0.023714	0.023575	0.027464	0.022949	0.024149
	(0.042739)	(0.043816)	(0.039912)	(0.041538)	(0.043585)
AHV_{t-3}	0.082941***	0.082525***	0.075380***	0.081148***	0.081646***
	(0.007496)	(0.008145)	(0.011695)	(0.008258)	(0.009192)
AHV_{t-4}	0.096948	0.096739	0.088777	0.095797	0.094919
	(0.059298)	(0.060827)	(0.063948)	(0.057082)	(0.058821)
AHV_{t-5}	0.022887***	0.022593**	0.026049***	0.023502**	0.022961**
	(0.006876)	(0.006952)	(0.006815)	(0.007162)	(0.007678)
AHV_{t-6}	0.164034***	0.164442***	0.167546***	0.165323***	0.166695**
	(0.047379)	(0.049763)	(0.049876)	(0.049319)	(0.054194)
X_{t-1}	0.000083	0.000045	0.019718	0.003399	0.006729
	(0.000201)	(0.000037)	(0.018964)	(0.003747)	(0.006694)
X_{t-2}	−0.000473***	−0.000116***	0.005269	0.005600	0.002778
	(0.000071)	(0.000023)	(0.004124)	(0.004809)	(0.004067)
X_{t-3}	−0.000804***	−0.000213***	−0.007797	−0.003904*	−0.004652*
	(0.000088)	(0.000028)	(0.005183)	(0.001726)	(0.002066)
X_{t-4}	−0.000546***	−0.000147***	0.002275	0.000725	−0.002442*
	(0.000088)	(0.000021)	(0.002454)	(0.003458)	(0.001084)
X_{t-5}	−0.000579***	−0.000119**	−0.001145	−0.002363	−0.000607
	(0.000147)	(0.000041)	(0.002634)	(0.001901)	(0.000970)
X_{t-6}	−0.000099	0.000000	−0.002750	−0.001543	0.000596
	(0.000101)	(0.000028)	(0.002450)	(0.001228)	(0.000981)
Constant	0.008726***	0.007587***	0.005770***	0.005939***	0.005857***
	(0.001609)	(0.001578)	(0.001405)	(0.001536)	(0.001612)
Shock (IRF)	0.004171	0.003061	0.001189	0.000215	0.001937
R^2	0.325745	0.325324	0.331931	0.325134	0.326344
Adj. R^2	0.325306	0.324885	0.331496	0.324695	0.325905
Num. obs.	19965	19965	19965	19965	19965

This table displays VAR regression with only two variables : AHV and X regressor. The column names represent the X variable for the selected model.

```r
est.VAR_f_d <- VAR(y_f_d,p=6)

#extract results
mod_vol_f_d = est.VAR_f_d$varresult$vol
f_f_d = formula(mod_vol_f_d)
d_f_d = model.frame(mod_vol_f_d)
lm_clean_f_d = lm(f_f_d, data= d_f_d)

#apply Newey-West
nw_vcov_f_d = NeweyWest(lm_clean_f_d, lag=6)
nw_se_f_d = sqrt(diag(nw_vcov_f_d))

#t-stats
coef_f_d = coef(lm_clean_f_d)
t_stat_f_d = coef_f_d/nw_se_f_d

#recalculate p-values
robust_f_d = 2*(1-pt(abs(t_stat_f_d), df = df.residual(lm_clean_f_d)))
```

N

```r
y_f_n = cbind(Vdata_f$N, Vdata_f$SPY_vol)
colnames(y_f_n)[1:2] <- c("N", "vol")
est.VAR_f_n <- VAR(y_f_n,p=6)

#extract results
mod_vol_f_n = est.VAR_f_n$varresult$vol
f_f_n = formula(mod_vol_f_n)
d_f_n = model.frame(mod_vol_f_n)
lm_clean_f_n = lm(f_f_n, data= d_f_n)

#apply Newey-West
nw_vcov_f_n = NeweyWest(lm_clean_f_n, lag=6)
nw_se_f_n = sqrt(diag(nw_vcov_f_n))

#t-stats
coef_f_n = coef(lm_clean_f_n)
t_stat_f_n = coef_f_n/nw_se_f_n

#recalculate p-values
robust_f_n = 2*(1-pt(abs(t_stat_f_n), df = df.residual(lm_clean_f_n)))
```

Tariff

```r
y_f_ta = cbind(Vdata_f$tariff, Vdata_f$SPY_vol)
colnames(y_f_ta)[1:2] <- c("tariff", "vol")
est.VAR_f_ta <- VAR(y_f_ta,p=6)

#extract results
```

```
mod_vol_f_ta = est.VAR_f_ta$varresult$vol
f_f_ta = formula(mod_vol_f_ta)
d_f_ta = model.frame(mod_vol_f_ta)
lm_clean_f_ta = lm(f_f_ta, data= d_f_ta)

#apply Newey-West
nw_vcov_f_ta = NeweyWest(lm_clean_f_ta, lag=6)
nw_se_f_ta = sqrt(diag(nw_vcov_f_ta))

#t-stats
coef_f_ta = coef(lm_clean_f_ta)
t_stat_f_ta = coef_f_ta/nw_se_f_ta

#recalculate p-values
robust_f_ta = 2*(1-pt(abs(t_stat_f_ta), df = df.residual(lm_clean_f_ta)))
```

Trade

```
y_f_tr = cbind(Vdata_f$trade, Vdata_f$SPY_vol)
colnames(y_f_tr)[1:2] <- c("trade", "vol")
est.VAR_f_tr <- VAR(y_f_tr,p=6)

#extract results
mod_vol_f_tr = est.VAR_f_tr$varresult$vol
f_f_tr = formula(mod_vol_f_tr)
d_f_tr = model.frame(mod_vol_f_tr)
lm_clean_f_tr = lm(f_f_tr, data= d_f_tr)

#apply Newey-West
nw_vcov_f_tr = NeweyWest(lm_clean_f_tr, lag=6)
nw_se_f_tr = sqrt(diag(nw_vcov_f_tr))

#t-stats
coef_f_tr = coef(lm_clean_f_tr)
t_stat_f_tr = coef_f_tr/nw_se_f_tr

#recalculate p-values
robust_f_tr = 2*(1-pt(abs(t_stat_f_tr), df = df.residual(lm_clean_f_tr)))
```

China

```
y_f_ch = cbind(Vdata_f$china, Vdata_f$SPY_vol)
colnames(y_f_ch)[1:2] <- c("china", "vol")
est.VAR_f_ch <- VAR(y_f_ch,p=6)

#extract results
mod_vol_f_ch = est.VAR_f_ch$varresult$vol
f_f_ch = formula(mod_vol_f_ch)
d_f_ch = model.frame(mod_vol_f_ch)
```

```r
lm_clean_f_ch = lm(f_f_ch, data= d_f_ch)

#apply Newey-West
nw_vcov_f_ch = NeweyWest(lm_clean_f_ch, lag=6)
nw_se_f_ch = sqrt(diag(nw_vcov_f_ch))

#t-stats
coef_f_ch = coef(lm_clean_f_ch)
t_stat_f_ch = coef_f_ch/nw_se_f_ch

#recalculate p-values
robust_f_ch = 2*(1-pt(abs(t_stat_f_ch), df = df.residual(lm_clean_f_ch)))
```

Second Term

Dummy

```r
y_s_d = cbind(Vdata_s$dummy, Vdata_s$SPY_vol)
colnames(y_s_d)[1:2] <- c("dummy", "vol")
est.VAR_s_d <- VAR(y_s_d,p=6)

#extract results
mod_vol_s_d = est.VAR_s_d$varresult$vol
f_s_d = formula(mod_vol_s_d)
d_s_d = model.frame(mod_vol_s_d)
lm_clean_s_d = lm(f_s_d, data= d_s_d)

#apply Newey-West
nw_vcov_s_d = NeweyWest(lm_clean_s_d, lag=6)
nw_se_s_d = sqrt(diag(nw_vcov_s_d))

#t-stats
coef_s_d = coef(lm_clean_s_d)
t_stat_s_d = coef_s_d/nw_se_s_d

#recalculate p-values
robust_s_d = 2*(1-pt(abs(t_stat_s_d), df = df.residual(lm_clean_s_d)))
```

N

```r
y_s_n = cbind(Vdata_s$N, Vdata_s$SPY_vol)
colnames(y_s_n)[1:2] <- c("N", "vol")
est.VAR_s_n <- VAR(y_s_n,p=6)

#extract results
mod_vol_s_n = est.VAR_s_n$varresult$vol
f_s_n = formula(mod_vol_s_n)
d_s_n = model.frame(mod_vol_s_n)
```

```
lm_clean_s_n = lm(f_s_n, data= d_s_n)

#apply Newey-West
nw_vcov_s_n = NeweyWest(lm_clean_s_n, lag=6)
nw_se_s_n = sqrt(diag(nw_vcov_s_n))

#t-stats
coef_s_n = coef(lm_clean_s_n)
t_stat_s_n = coef_s_n/nw_se_s_n

#recalculate p-values
robust_s_n = 2*(1-pt(abs(t_stat_s_n), df = df.residual(lm_clean_s_n)))
```

Tariff

```
y_s_ta = cbind(Vdata_s$tariff, Vdata_s$SPY_vol)
colnames(y_s_ta)[1:2] <- c("tariff", "vol")
est.VAR_s_ta <- VAR(y_s_ta,p=6)

#extract results
mod_vol_s_ta = est.VAR_s_ta$varresult$vol
f_s_ta = formula(mod_vol_s_ta)
d_s_ta = model.frame(mod_vol_s_ta)
lm_clean_s_ta = lm(f_s_ta, data= d_s_ta)

#apply Newey-West
nw_vcov_s_ta = NeweyWest(lm_clean_s_ta, lag=6)
nw_se_s_ta = sqrt(diag(nw_vcov_s_ta))

#t-stats
coef_s_ta = coef(lm_clean_s_ta)
t_stat_s_ta = coef_s_ta/nw_se_s_ta

#recalculate p-values
robust_s_ta = 2*(1-pt(abs(t_stat_s_ta), df = df.residual(lm_clean_s_ta)))
```

Trade

```
y_s_tr = cbind(Vdata_s$trade, Vdata_s$SPY_vol)
colnames(y_s_tr)[1:2] <- c("trade", "vol")
est.VAR_s_tr <- VAR(y_s_tr,p=6)

#extract results
mod_vol_s_tr = est.VAR_s_tr$varresult$vol
f_s_tr = formula(mod_vol_s_tr)
d_s_tr = model.frame(mod_vol_s_tr)
lm_clean_s_tr = lm(f_s_tr, data= d_s_tr)

#apply Newey-West
```

```r
nw_vcov_s_tr = NeweyWest(lm_clean_s_tr, lag=6)
nw_se_s_tr = sqrt(diag(nw_vcov_s_tr))

#t-stats
coef_s_tr = coef(lm_clean_s_tr)
t_stat_s_tr = coef_s_tr/nw_se_s_tr

#recalculate p-values
robust_s_tr = 2*(1-pt(abs(t_stat_s_tr), df = df.residual(lm_clean_s_tr)))
```

China

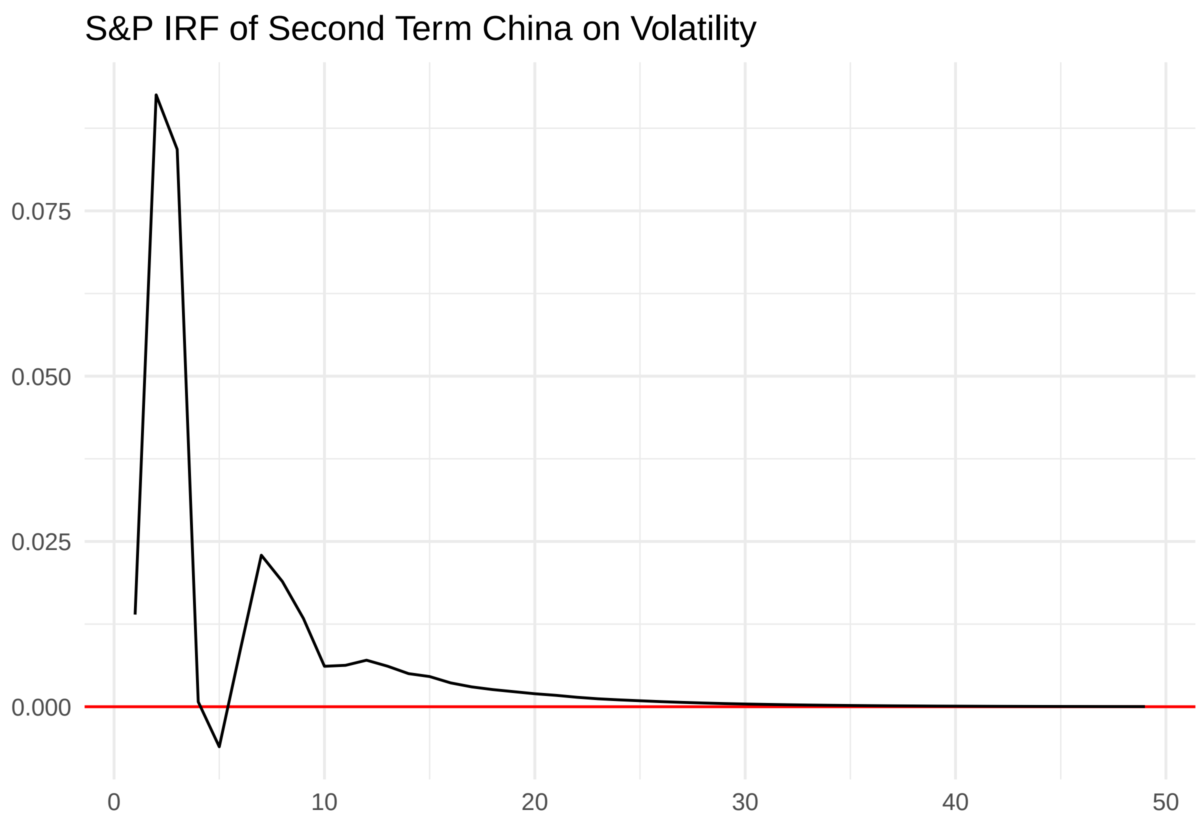
IRF 3

```r
y_s_ch = cbind(Vdata_s$china, Vdata_s$SPY_vol)
colnames(y_s_ch)[1:2] <- c("china", "vol")
est.VAR_s_ch <- VAR(y_s_ch,p=6)

#extract results
mod_vol_s_ch = est.VAR_s_ch$varresult$vol
f_s_ch = formula(mod_vol_s_ch)
d_s_ch = model.frame(mod_vol_s_ch)
lm_clean_s_ch = lm(f_s_ch, data= d_s_ch)

#apply Newey-West
nw_vcov_s_ch = NeweyWest(lm_clean_s_ch, lag=6)
nw_se_s_ch = sqrt(diag(nw_vcov_s_ch))

#t-stats
coef_s_ch = coef(lm_clean_s_ch)
t_stat_s_ch = coef_s_ch/nw_se_s_ch

#recalculate p-values
robust_s_ch = 2*(1-pt(abs(t_stat_s_ch), df = df.residual(lm_clean_s_ch)))
```

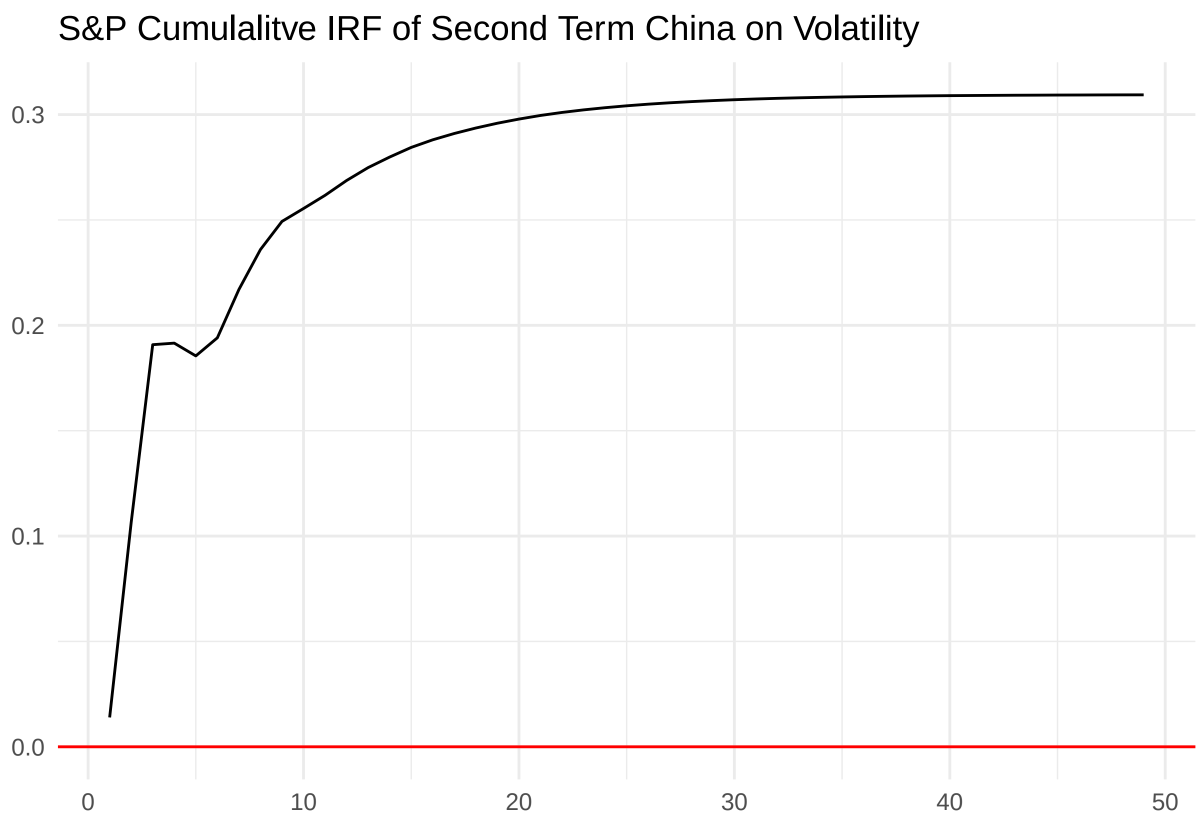
```r
#Construct the Robust Omega Matrix
U_s_ch = residuals(est.VAR_s_ch)
T_s_ch = nrow(U_s_ch)
Omega_s_ch = matrix(0, ncol(U_s_ch), ncol(U_s_ch))
for(l in 0:L) {
  weight = 1 - l/(L+1)
  Gamma_l__s_ch = t(U_s_ch[(l+1):T_s_ch, , drop=FALSE]) %*% U_s_ch[1:(T_s_ch-l), , drop=FALSE] /T_s_ch
  if (l == 0){
    Omega_s_ch = Omega_s_ch + Gamma_l__s_ch
  } else {
    Omega_s_ch = Omega_s_ch + weight*(Gamma_l__s_ch + t(Gamma_l__s_ch))
  }
}


#make the B matrix
```

```r
loss_s_ch <- function(param_s_ch){
  #Define the restriction
  B_s_ch <- matrix(c(param_s_ch[1], param_s_ch[2], 0, param_s_ch[3]), ncol = 2)

  #Make BB' approximatively equal to omega
  X_s_ch <- Omega_s_ch - B_s_ch %*% t(B_s_ch)

  #loss function
  loss_s_ch <- sum(X_s_ch^2)
  return(loss_s_ch)
}

res.opt_s_ch <- optim(c(1, 0, 1), loss_s_ch, method = "BFGS")
B.hat_s_ch <- matrix(c(res.opt_s_ch$par[1], res.opt_s_ch$par[2], 0, res.opt_s_ch$par[3]), ncol = 2)
```

```r
#get back the coefficient of est.VAR
phi_s_ch <- Acoef(est.VAR_s_ch)
PHI_s_ch = make.PHI(phi_s_ch)

#take the constant
constant_s_ch <- sapply(est.VAR_s_ch$varresult, function(eq) coef(eq)["const"])
c_s_ch=as.matrix(constant_s_ch)

#Simulate the IRF
p_s_ch <- length(phi_s_ch)
n_s_ch <- dim(phi_s_ch[[1]])[1]

Y_s_ch <- simul.VAR(c=c_s_ch, Phi = phi_s_ch, B = B.hat_s_ch, nb.sim ,y0.star=rep(0, n_s_ch*p_s_ch),
                  indic.IRF = 1, u.shock = c(1,0))


#Plot the IRF
Yd_s_ch = data.frame(
  period = 1:nrow(Y_s_ch),
  response = Y_s_ch[,2])

ggplot(Yd_s_ch,aes(x=period, y=response)) +
  geom_hline(yintercept = 0, color="red") +
  geom_line() +
  theme_light() +
  ggtitle("S&P IRF of Second Term China on Volatility") +
  ylab("")+
  xlab("") +
  theme_minimal()
```




```r
ggplot(Yd_s_ch,aes(x=period, y=cumsum(response))) +
  geom_hline(yintercept = 0, color="red") +
  geom_line() +
  theme_light() +
  ggtitle("S&P Cumulalitve IRF of Second Term China on Volatility") +
  ylab("")+
  xlab("") +
  theme_minimal()
```



Tables Terms

```r
#first

d_f_d_t = d_f_d %>%
    rename(X.l1 = dummy.l1,
    X.l2 = dummy.l2,
    X.l3 = dummy.l3,
    X.l4 = dummy.l4,
    X.l5 = dummy.l5,
    X.l6 = dummy.l6)

f_t_f_d <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                         X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_f_d <- lm(f_t_f_d, data = d_f_d_t)

d_f_n_t = d_f_n %>%
    rename(X.l1 = N.l1,
    X.l2 = N.l2,
    X.l3 = N.l3,
    X.l4 = N.l4,
    X.l5 = N.l5,
    X.l6 = N.l6)
```

```r
f_t_f_n <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_f_n <- lm(f_t_f_n, data = d_f_n_t)

d_f_ta_t = d_f_ta %>%
    rename(X.l1 = tariff.l1,
    X.l2 = tariff.l2,
    X.l3 = tariff.l3,
    X.l4 = tariff.l4,
    X.l5 = tariff.l5,
    X.l6 = tariff.l6)

f_t_f_ta <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_f_ta <- lm(f_t_f_ta, data = d_f_ta_t)


d_f_tr_t = d_f_tr %>%
    rename(X.l1 = trade.l1,
    X.l2 = trade.l2,
    X.l3 = trade.l3,
    X.l4 = trade.l4,
    X.l5 = trade.l5,
    X.l6 = trade.l6)

f_t_f_tr <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_f_tr <- lm(f_t_f_tr, data = d_f_tr_t)


d_f_ch_t = d_f_ch %>%
    rename(X.l1 = china.l1,
    X.l2 = china.l2,
    X.l3 = china.l3,
    X.l4 = china.l4,
    X.l5 = china.l5,
    X.l6 = china.l6)

f_t_f_ch <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_f_ch <- lm(f_t_f_ch, data = d_f_ch_t)




nw_se_f_d_t <- sqrt(diag(sandwich::NeweyWest(model_f_d, lag = 6)))
nw_se_f_n_t <- sqrt(diag(sandwich::NeweyWest(model_f_n, lag = 6)))
nw_se_f_ta_t <- sqrt(diag(sandwich::NeweyWest(model_f_ta, lag = 6)))
nw_se_f_tr_t <- sqrt(diag(sandwich::NeweyWest(model_f_tr, lag = 6)))
nw_se_f_china_t <- sqrt(diag(sandwich::NeweyWest(model_f_ch, lag = 6)))


robust_f_d_t <- 2 * (1-pt(abs(coef(model_f_d) / nw_se_f_d_t), df = df.residual(model_f_d)))
```

```r
robust_f_n_t <- 2 * (1-pt(abs(coef(model_f_n) / nw_se_f_n_t), df = df.residual(model_f_n)))
robust_f_ta_t <- 2 * (1-pt(abs(coef(model_f_ta) / nw_se_f_ta_t), df = df.residual(model_f_ta)))
robust_f_tr_t <- 2 * (1-pt(abs(coef(model_f_tr) / nw_se_f_tr_t), df = df.residual(model_f_tr)))
robust_f_ch_t <- 2 * (1-pt(abs(coef(model_f_ch) / nw_se_f_china_t), df = df.residual(model_f_ch)))


nw_se_f_d_t       <- nw_se_f_d_t[names(coef(model_f_d))]
robust_f_d_t      <- robust_f_d_t[names(coef(model_f_d))]
nw_se_f_n_t      <- nw_se_f_n_t[names(coef(model_f_n))]
robust_f_n_t     <- robust_f_n_t[names(coef(model_f_n))]
nw_se_f_ta_t      <- nw_se_f_ta_t[names(coef(model_f_ta))]
robust_f_ta_t     <- robust_f_ta_t[names(coef(model_f_ta))]
nw_se_f_tr_t     <- nw_se_f_tr_t[names(coef(model_f_tr))]
robust_f_tr_t     <- robust_f_tr_t[names(coef(model_f_tr))]
nw_se_f_china_t  <- nw_se_f_china_t[names(coef(model_f_ch))]
robust_f_ch_t <- robust_f_ch_t[names(coef(model_f_ch))]


# list of models, SE and p-value
models_list_f <- list(model_f_d, model_f_n, model_f_ta, model_f_tr, model_f_ch)
robust_ses_f <- list(nw_se_f_d_t, nw_se_f_n_t, nw_se_f_ta_t, nw_se_f_tr_t, nw_se_f_china_t)
robust_pvals_f <- list(robust_f_d_t, robust_f_n_t, robust_f_ta_t, robust_f_tr_t, robust_f_ch_t)

# Name of coefficient
custom_names <- list(
  "vol.l1" = "$AHV_{t-1}$",
  "vol.l2" = "$AHV_{t-2}$",
  "vol.l3" = "$AHV_{t-3}$",
  "vol.l4" = "$AHV_{t-4}$",
  "vol.l5" = "$AHV_{t-5}$",
  "vol.l6" = "$AHV_{t-6}$",
  "X.l1" = "$X_{t-1}$",
  "X.l2" = "$X_{t-2}$",
  "X.l3" = "$X_{t-3}$",
  "X.l4" = "$X_{t-4}$",
  "X.l5" = "$X_{t-5}$",
  "X.l6" = "$X_{t-6}$",
  "const" = "Constant"
)

# Generate table
table_texreg_f <- texreg(
  l = models_list_f,
  override.se = robust_ses_f,
  override.pvalues = robust_pvals_f,
  custom.model.names = c("TweetDummy", "TweetCount", "Tariff", "Trade", "China"),
  custom.coef.map = custom_names,
  caption = "First-Term VAR Models of Average Hourly Volatility",
  label = "tab:VAR_First_Term",
  caption.above = TRUE,
  digits = 6,
  custom.gof.rows = list("Shock (IRF)" = c(0.002919, 0.002236, 0.000484, 0.000702, 0.000904)),
  star.cutoffs = c(0.001, 0.01, 0.05),
```

Table 2: First-Term VAR Models of Average Hourly Volatility

	TweetDummy	TweetCount	Tariff	Trade	China
AHV_{t-1}	0.541944***	0.542426***	0.543570***	0.543958***	0.543471***
	(0.074062)	(0.074300)	(0.074985)	(0.075228)	(0.074985)
AHV_{t-2}	−0.113920**	−0.113855**	−0.115106**	−0.115566**	−0.115002**
	(0.038762)	(0.039289)	(0.039650)	(0.039098)	(0.039605)
AHV_{t-3}	0.058050*	0.057592*	0.053635*	0.053636*	0.054382*
	(0.027504)	(0.027211)	(0.027121)	(0.026885)	(0.027488)
AHV_{t-4}	0.188383	0.187417	0.184183	0.184102	0.184610
	(0.132562)	(0.131408)	(0.130566)	(0.133609)	(0.130953)
AHV_{t-5}	−0.088758	−0.089704	−0.091496	−0.091655	−0.091848
	(0.091453)	(0.090665)	(0.090960)	(0.093255)	(0.091105)
AHV_{t-6}	0.336662***	0.337701***	0.343373***	0.343466***	0.343184***
	(0.049019)	(0.048828)	(0.048409)	(0.048454)	(0.048945)
X_{t-1}	−0.000478***	−0.000163**	−0.000454	−0.001838*	−0.000352
	(0.000133)	(0.000055)	(0.000352)	(0.000715)	(0.000375)
X_{t-2}	−0.000184**	−0.000063*	−0.000289	0.000221	−0.000048
	(0.000069)	(0.000029)	(0.000274)	(0.000520)	(0.000233)
X_{t-3}	−0.000693***	−0.000263***	−0.001007***	−0.000949**	−0.001412***
	(0.000160)	(0.000064)	(0.000269)	(0.000301)	(0.000373)
X_{t-4}	−0.000564***	−0.000208**	−0.000274	−0.000612	−0.000202
	(0.000166)	(0.000064)	(0.000400)	(0.000408)	(0.000454)
X_{t-5}	−0.000435***	−0.000125**	−0.000468	−0.000605	−0.000057
	(0.000113)	(0.000044)	(0.000281)	(0.000353)	(0.000352)
X_{t-6}	0.000118	0.000099*	0.000240	−0.000121	0.000275
	(0.000118)	(0.000047)	(0.000344)	(0.000412)	(0.000370)
Constant	0.004020***	0.003079***	0.001510***	0.001657***	0.001593***
	(0.000669)	(0.000520)	(0.000316)	(0.000329)	(0.000312)
Shock (IRF)	0.002919	0.002236	0.000484	0.000702	0.000904
R^2	0.687909	0.687236	0.685341	0.685489	0.685533
Adj. R^2	0.687331	0.686657	0.684758	0.684907	0.684951
Num. obs.	7036	7036	7036	7036	7036

This table displays VAR regression with only two variables : AHV and X regressor. The column names represent the X variable for the selected model.

```
  custom.note = "\\parbox[t]{0.9\\textwidth}{\\small\\textit{This table displays VAR regression with onl
)

# Print table
table_texreg_f
```

```
#Second

d_s_d_t = d_s_d %>%
    rename(X.l1 = dummy.l1,
    X.l2 = dummy.l2,
    X.l3 = dummy.l3,
    X.l4 = dummy.l4,
    X.l5 = dummy.l5,
    X.l6 = dummy.l6)
```

```r
f_t_s_d <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_s_d <- lm(f_t_s_d, data = d_s_d_t)


d_s_n_t = d_s_n %>%
    rename(X.l1 = N.l1,
    X.l2 = N.l2,
    X.l3 = N.l3,
    X.l4 = N.l4,
    X.l5 = N.l5,
    X.l6 = N.l6)

f_t_s_n <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_s_n <- lm(f_t_s_n, data = d_s_n_t)


d_s_ta_t = d_s_ta %>%
    rename(X.l1 = tariff.l1,
    X.l2 = tariff.l2,
    X.l3 = tariff.l3,
    X.l4 = tariff.l4,
    X.l5 = tariff.l5,
    X.l6 = tariff.l6)

f_t_s_ta <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_s_ta <- lm(f_t_s_ta, data = d_s_ta_t)


d_s_tr_t = d_s_tr %>%
    rename(X.l1 = trade.l1,
    X.l2 = trade.l2,
    X.l3 = trade.l3,
    X.l4 = trade.l4,
    X.l5 = trade.l5,
    X.l6 = trade.l6)


f_t_s_tr <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_s_tr <- lm(f_t_s_tr, data = d_s_tr_t)


d_s_ch_t = d_s_ch %>%
    rename(X.l1 = china.l1,
    X.l2 = china.l2,
    X.l3 = china.l3,
    X.l4 = china.l4,
    X.l5 = china.l5,
    X.l6 = china.l6)
```

```r
f_t_s_ch <- as.formula("y ~ -1 + vol.l1 + vol.l2 + vol.l3 + vol.l4 + vol.l5 + vol.l6 +
                        X.l1 + X.l2 + X.l3 + X.l4 + X.l5 + X.l6 + const")
model_s_ch <- lm(f_t_s_ch, data = d_s_ch_t)


nw_se_s_d_t <- sqrt(diag(sandwich::NeweyWest(model_s_d, lag = 6)))
nw_se_s_n_t <- sqrt(diag(sandwich::NeweyWest(model_s_n, lag = 6)))
nw_se_s_ta_t <- sqrt(diag(sandwich::NeweyWest(model_s_ta, lag = 6)))
nw_se_s_tr_t <- sqrt(diag(sandwich::NeweyWest(model_s_tr, lag = 6)))
nw_se_s_china_t <- sqrt(diag(sandwich::NeweyWest(model_s_ch, lag = 6)))


robust_s_d_t <- 2 * (1-pt(abs(coef(model_s_d) / nw_se_s_d_t), df = df.residual(model_s_d)))
robust_s_n_t <- 2 * (1-pt(abs(coef(model_s_n) / nw_se_s_n_t), df = df.residual(model_s_n)))
robust_s_ta_t <- 2 * (1-pt(abs(coef(model_s_ta) / nw_se_s_ta_t), df = df.residual(model_s_ta)))
robust_s_tr_t <- 2 * (1-pt(abs(coef(model_s_tr) / nw_se_s_tr_t), df = df.residual(model_s_tr)))
robust_s_ch_t <- 2 * (1-pt(abs(coef(model_s_ch) / nw_se_s_china_t), df = df.residual(model_s_ch)))

nw_se_s_d_t       <- nw_se_s_d_t[names(coef(model_s_d))]
robust_s_d_t      <- robust_s_d_t[names(coef(model_s_d))]
nw_se_s_n_t      <- nw_se_s_n_t[names(coef(model_s_n))]
robust_s_n_t     <- robust_s_n_t[names(coef(model_s_n))]
nw_se_s_ta_t      <- nw_se_s_ta_t[names(coef(model_s_ta))]
robust_s_ta_t     <- robust_s_ta_t[names(coef(model_s_ta))]
nw_se_s_tr_t     <- nw_se_s_tr_t[names(coef(model_s_tr))]
robust_s_tr_t     <- robust_s_tr_t[names(coef(model_s_tr))]
nw_se_s_china_t  <- nw_se_s_china_t[names(coef(model_s_ch))]
robust_s_ch_t <- robust_s_ch_t[names(coef(model_s_ch))]

# lists of model, robust SE  and p-values
models_list_s <- list(model_s_d, model_s_n, model_s_ta, model_s_tr, model_s_ch)
robust_ses_s <- list(nw_se_s_d_t, nw_se_s_n_t, nw_se_s_ta_t, nw_se_s_tr_t, nw_se_s_china_t)
robust_pvals_s <- list(robust_s_d_t, robust_s_n_t, robust_s_ta_t, robust_s_tr_t, robust_s_ch_t)

# Generate table
table_texreg_s <- texreg(
  l = models_list_s,
  override.se = robust_ses_s,
  override.pvalues = robust_pvals_s,
  custom.model.names = c("TweetDummy", "TweetCount", "Tariff", "Trade", "China"),
  custom.coef.map = custom_names,
  caption = "Second-Term VAR Models of Average Hourly Volatility",
  label = "tab:VAR_Second_Term",
  caption.above = TRUE,
  digits = 6,
  custom.gof.rows = list("Shock (IRF)" = c(0.014659, 0.013315, 0.010752, -0.005665, 0.013937)),
  star.cutoffs = c(0.05, 0.01, 0.001),
  custom.note = "\\parbox[t]{0.9\\textwidth}{\\small\\textit{This table displays VAR regression with onl
)



# Print
```

Table 3: Second-Term VAR Models of Average Hourly Volatility

	TweetDummy	TweetCount	Tariff	Trade	China
AHV_{t-1}	0.299398^{**}	0.299350^{*}	0.294752^{**}	0.301160^{**}	0.274419^{***}
	(0.112369)	(0.115966)	(0.110305)	(0.111228)	(0.078452)
AHV_{t-2}	0.015406	0.013567	0.020667	0.011769	0.031670
	(0.043748)	(0.045193)	(0.038728)	(0.043649)	(0.028307)
AHV_{t-3}	0.076169^{***}	0.076851^{***}	0.068749^{***}	0.072284^{***}	0.052697
	(0.008464)	(0.008788)	(0.016281)	(0.014041)	(0.034520)
AHV_{t-4}	0.084229	0.085108	0.074401	0.080544	0.035573
	(0.067860)	(0.070966)	(0.079828)	(0.066709)	(0.100929)
AHV_{t-5}	0.013424^{**}	0.010406	0.015342^{**}	0.017631^{*}	0.005467
	(0.005076)	(0.005958)	(0.005363)	(0.008269)	(0.032198)
AHV_{t-6}	0.126612^{*}	0.126324^{*}	0.132056^{**}	0.124277^{*}	0.150909^{**}
	(0.050031)	(0.050865)	(0.049243)	(0.051402)	(0.051059)
X_{t-1}	0.006569	0.000947	0.027028	0.020463	0.154584
	(0.010145)	(0.001307)	(0.028623)	(0.029865)	(0.138092)
X_{t-2}	-0.003222^{**}	-0.000736	0.008588	0.047163	0.099315
	(0.001039)	(0.000524)	(0.007211)	(0.041341)	(0.095007)
X_{t-3}	-0.005538^{**}	-0.001637^{*}	-0.010306	-0.026631	-0.047690
	(0.001707)	(0.000717)	(0.007584)	(0.021348)	(0.030038)
X_{t-4}	0.002474	0.000136	0.002002	0.019925	-0.020669
	(0.004957)	(0.000914)	(0.002962)	(0.031571)	(0.012428)
X_{t-5}	-0.008527^{*}	-0.001651	-0.002649	-0.012965	-0.004470
	(0.003989)	(0.001068)	(0.004280)	(0.014973)	(0.018255)
X_{t-6}	-0.003594	-0.000627	-0.004279	-0.011100	0.008027
	(0.003191)	(0.000760)	(0.003854)	(0.010317)	(0.022286)
Constant	0.072524^{**}	0.068423^{**}	0.049265^{**}	0.052127^{***}	0.044027^{*}
	(0.024156)	(0.021308)	(0.015262)	(0.014290)	(0.017471)
Shock (IRF)	0.014659	0.013315	0.010752	-0.005665	0.013937
R^2	0.244117	0.240788	0.251263	0.244406	0.285165
Adj. R^2	0.224424	0.221009	0.231757	0.224721	0.266543
Num. obs.	512	512	512	512	512

This table displays VAR regression with only two variables : AHV and X regressor. The column names represent the X variable for the selected model.

```
table_texreg_s
```