Final_VAR

Contents

ARMAX

Once we have our final dataframe, we could then finally start on some analysis. We first thought of a simple ARMA-X type specification, taking the AHV as our "y variable" and taking any of the social media variables as the exogenous regressors. The assumption here is that, while the market reacts to Trump posts, Trump's posts are chaotic, nonsensical, and random enough to be considered exogenous.

We of course first start by checking stationarity of our variables (ADF), where we find p-values of 0.01 suggesting that the processes are not explosive. Then, we use a custom function in order to choose the number of lags based on the AIC criterion. This however, while often choose a very high number of lags, which could be explained by our data being hourly. As such we decided to put a limit of 3 lags, which sees minimal AIC loss and simplifying our models considerably.

SVAR

Additionally, 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. The SVAR approach offers the advantage of accounting for structural endogeneity.In this regards, we assume that the volatility does not contemporaneously affect Trump's posting activity - neither quantitatively nor qualitatively. Consequently, we impose a short-run restriction on the shock of volatility for all the Trump tweet's variables.

Based on the information criteria, we found similar results across all specification, 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 - up to unrealistic levels, i.e., 150 days for Tweet Count, which is implausible. Therefore, we chose to fix the number of lags at 6. Finally, given the presence of heteroscedasticity and serial correlation in the residuals, we use the Newey-West estimator to compute a robust covariance matrix.

Results

Full Timeframe

We run models with the following exogenous regressors: $TweetDummy$, $TweetCount$, and the mentions of words $Tariff$, $Trade$, and $China$. We first note on *bigfigX* that all the x-regressors are significant, apart from trade. Notice also that all the coefficients (apart from $Tariff_{t-3}$) are positive, in line with our main hypothesis. The effect of $Tariff_{t-1}$ and $Tariff_{t-2}$ are especially large, given the usual size of the volatility (*smalldescfigX*). We in fact predict that an extra mention of tariffs one hour ago, leads to a whopping extra 0.02 in volatility which is just about the average size for the full timeframe. We can see the impulse response function (IRF) for this shock, in *IRFtarif*. Notice that there is a large response in the first periods, and then a graduate decline over time. Something to note is that in our analysises of IRF's, when including MA terms, the decline shows up gradual while being much sharper when only including AR terms. Note that we ran all these models on the VGK and ASHR ETF's as well, though no signficant results appear apart from a small but statistically significant effect of the tariff variable for VGK.

Split Samples

We then split our sample for the first and second term of the Trump presidancy. We only run models on tariff, trade and china this time. As seen on table *bigtableX2*, the first interesting result is in the coefficients of tariff being significant and very large in the second term, while being small and not statistically significant in the first. A similar story goes for the China variable. This may lend some evidence to support the claim that investors are much more reactive to Trump's social media presence now than before. Finally, we can check the residuals of all these models to test them somewhat. On *smalldescfigX*, the pvalues being zero for the full timeframe and first term indicate that there is autocorrelation in the residuals, thus suggesting that these estimations have problems. Note however, that the pvalues for the second term are quite high, lending support to our models on the split sample. These results suggest that perhaps ARMA-X models are not right in this context, as it is not unreasonable to think that Trump does in fact react to market movements. With this information, we decided to run a VAR model to deepen our understanding of these variables.

Full Timeframe

As in the ARMA-X framework, we initially run an estimation for each of our five main variables where we find similar results on the effect of the posts. For all estimations, taking the US ETF market, the significant coefficient of post's variables were all negative, and the positive one where large but insignificant. Except for TweetCount, the first and sometimes second coefficient were positive while the rest not. In the mean time, we find that the contemporaneously effect of the shocks were all positive and high, up to 0.2 for TweetCount. This lead to the IRFs to explode when the shocks occur and tend to decrease, even going negative some hours after the shock. But, for Tariff, China and Trade, the effect show a clear cumulative positive effect of the post. Finally, except for Tariff, all Granger test manifests that Trump post has an impact on Volatility. This model clearly shows that Trump's post seem to have a positive instantaneous effect on volatility, but encompass very low persistence. Finally, taking the European and Chinese ETF market, we observe similar pattern, except for the impact of Tariff and China variables on the ASHR market, where the cumulative effect show no positive impact. impact on Trump and granger test.

Split Sample

For the splitting framework, the results were similar and striking. While we found rather small shock effect and almost all negative coefficient leading to cumulative IRF to show negative impact of posts, we found shock effect in the second term that were between 5 times (for TweetCount and TweetDummy) to 25 times higher (for Tariff), except for Trade were we found a negative impact of a shock in the second term. Once again we found positive coefficient of lag variables in the second term, mainly on the first, second and fourth lag, but non of them was significant. Grangertest and impact on Trump

```
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)))

#table
screenreg(lm_clean, override.se = nw_se, override.pvalues = robust, digits = 6)
```

```
##
## ============================
##             Model 1
## ----------------------------
## dummy.l1      0.000083
##              (0.000201)
## vol.l1        0.344511 ***
##              (0.103790)
## dummy.l2     -0.000473 ***
##              (0.000071)
## vol.l2        0.023714
##              (0.042739)
## dummy.l3     -0.000804 ***
##              (0.000088)
## vol.l3        0.082941 ***
##              (0.007496)
## dummy.l4     -0.000546 ***
##              (0.000088)
## vol.l4        0.096948
##              (0.059298)
## dummy.l5     -0.000579 ***
##              (0.000147)
## vol.l5        0.022887 ***
##              (0.006876)
## dummy.l6     -0.000099
##              (0.000101)
## vol.l6        0.164034 ***
```

```
##                (0.047379)
## const           0.008726 ***
##                (0.001609)
## ---------------------------
## R^2             0.325745
## Adj. R^2        0.325306
## Num. obs.   19965
## ===========================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

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)))

#table
screenreg(lm_clean2, override.se = nw_se2, override.pvalues = robust2, digits = 6)
```

```
##
## ===========================
##            Model 1
## ---------------------------
## N.l1            0.000045
##                (0.000037)
## vol.l1          0.345011 ***
##                (0.104492)
## N.l2           -0.000116 ***
##                (0.000023)
## vol.l2          0.023575
##                (0.043816)
## N.l3           -0.000213 ***
##                (0.000028)
## vol.l3          0.082525 ***
##                (0.008145)
```

```
## N.14          -0.000147 ***
##               (0.000021)
## vol.14         0.096739
##               (0.060827)
## N.15          -0.000119 **
##               (0.000041)
## vol.15         0.022593 **
##               (0.006952)
## N.16           0.000000
##               (0.000028)
## vol.16         0.164442 ***
##               (0.049763)
## const          0.007587 ***
##               (0.001578)
## ---------------------------
## R^2            0.325324
## Adj. R^2       0.324885
## Num. obs.  19965
## ===========================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

Tariff

```
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)))

#table
screenreg(lm_clean3, override.se = nw_se3, override.pvalues = robust3, digits = 6)
```

```
##
## ===========================
##            Model 1
## ---------------------------
## tariff.l1      0.019718
```

```
##                (0.018964)
## vol.l1          0.342081 ***
##                (0.098665)
## tariff.l2       0.005269
##                (0.004124)
## vol.l2          0.027464
##                (0.039912)
## tariff.l3      -0.007797
##                (0.005183)
## vol.l3          0.075380 ***
##                (0.011695)
## tariff.l4       0.002275
##                (0.002454)
## vol.l4          0.088777
##                (0.063948)
## tariff.l5      -0.001145
##                (0.002634)
## vol.l5          0.026049 ***
##                (0.006815)
## tariff.l6      -0.002750
##                (0.002450)
## vol.l6          0.167546 ***
##                (0.049876)
## const           0.005770 ***
##                (0.001405)
## ---------------------------
## R^2             0.331931
## Adj. R^2        0.331496
## Num. obs.   19965
## ===========================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

Trade

```
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
```

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

#table
screenreg(lm_clean4, override.se = nw_se4, override.pvalues = robust4, digits = 6)
```

```
##
## ===========================
##             Model 1
## ---------------------------
## trade.l1      0.003399
##              (0.003747)
## vol.l1        0.346107 ***
##              (0.101918)
## trade.l2      0.005600
##              (0.004809)
## vol.l2        0.022949
##              (0.041538)
## trade.l3     -0.003904 *
##              (0.001726)
## vol.l3        0.081148 ***
##              (0.008258)
## trade.l4      0.000725
##              (0.003458)
## vol.l4        0.095797
##              (0.057082)
## trade.l5     -0.002363
##              (0.001901)
## vol.l5        0.023502 **
##              (0.007162)
## trade.l6     -0.001543
##              (0.001228)
## vol.l6        0.165323 ***
##              (0.049319)
## const         0.005939 ***
##              (0.001536)
## ---------------------------
## R^2           0.325134
## Adj. R^2      0.324695
## Num. obs.  19965
## ===========================
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

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
```

```r
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)))

#table
screenreg(lm_cleanchina, override.se = nw_sechina, override.pvalues = robustchina, digits = 6)
```

```
##
## ===========================
##              Model 1
## ---------------------------
## china.l1       0.006729
##               (0.006694)
## vol.l1         0.344512 ***
##               (0.097994)
## china.l2       0.002778
##               (0.004067)
## vol.l2         0.024149
##               (0.043585)
## china.l3      -0.004652 *
##               (0.002066)
## vol.l3         0.081646 ***
##               (0.009192)
## china.l4      -0.002442 *
##               (0.001084)
## vol.l4         0.094919
##               (0.058821)
## china.l5      -0.000607
##               (0.000970)
## vol.l5         0.022961 **
##               (0.007678)
## china.l6       0.000596
##               (0.000981)
## vol.l6         0.166695 **
##               (0.054194)
## const          0.005857 ***
##               (0.001612)
## ---------------------------
## R^2            0.326344
## Adj. R^2       0.325905
## Num. obs.  19965
## ===========================
```

```
## *** p < 0.001; ** p < 0.01; * p < 0.05
```

mean_day_filtered <- Vdata %>% mutate(day = as.Date(timestamp), year = year(day), month = month(day)) %>% filter(!(year == 2025 & month %in% c(4, 5))) %>% # exclut avril et mai 2025 group_by(day) %>% summarise(mean_vol_day = mean(SPY_vol, na.rm = TRUE))

mean(mean_day_filtered$mean_vol_day)

```
dt_t = 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 <- 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,
```

```r
    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)

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


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))]
```

```r
# Créer la liste des modèles
models_list <- list(model, model2, model3, model4, modelchina)

# Créer la liste des SE robustes
robust_ses <- list(nw_se_t, nw_se2_t, nw_se3_t, nw_se4_t, nw_sechina_t)

# Créer la liste des p-values
robust_pvals <- list(robust_t, robust2_t, robust3_t, robust4_t, robustchina_t)

# Nom des variables (affichées dans le tableau)
custom_names <- list(
  "vol.l1" = "$Vol_{t-1}$",
  "vol.l2" = "$Vol_{t-2}$",
  "vol.l3" = "$Vol_{t-3}$",
  "vol.l4" = "$Vol_{t-4}$",
  "vol.l5" = "$Vol_{t-5}$",
  "vol.l6" = "$Vol_{t-6}$",
  "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"
)

# Générer le tableau
table_texreg <- texreg(
  l = models_list,
  override.se = robust_ses,
  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))
)



# Afficher dans le Viewer
table_texreg
```

0.004171303 0.003061411 0.001189108 0.0002157857 0.0019376

stargazer(model, model2, model3, model4, modelchina, type = “text”, se = list(nw_se_t, nw_se2_t, nw_se3_t, nw_se4_t, nw_sechina_t), p = list(robust_t, robust2_t, robust3_t, robust4_t, robustchina_t), column.labels = c(“Model 1”, “Model 2”, “Model 3”, “Model 4”, “Model 5”), dep.var.labels = “ “, dep.var.labels.include = FALSE, model.names = FALSE, model.numbers = FALSE, title =”Table 1 : Trump’s Post on Volatility (Newey-West robust SE)“, digits = 6, no.space = TRUE, omit.stat = c(”f”, “ser”, “rsq”, “adj.rsq”), star.cutoffs = c(0.05, 0.01, 0.001), flip = TRUE, add.lines=list(c(“shock”, “0.004171303”, “0.003061411”, “0.001189108”, “0.0002157857”, “0.0019376”)))

Table 1: VAR Models of Average Hourly Volatility

	TweetDummy	TweetCount	Tariff	Trade	China
vol.l1	0.344511***	0.345011***	0.342081***	0.346107***	0.344512***
	(0.103329)	(0.103473)	(0.100397)	(0.103007)	(0.102386)
vol.l2	0.023714	0.023575	0.027464	0.022949	0.024149
	(0.047239)	(0.047379)	(0.042571)	(0.047267)	(0.046561)
vol.l3	0.082941***	0.082525***	0.075380***	0.081148***	0.081646***
	(0.010963)	(0.011004)	(0.013612)	(0.011336)	(0.011026)
vol.l4	0.096948	0.096739	0.088777	0.095797	0.094919
	(0.065612)	(0.065614)	(0.068856)	(0.064948)	(0.066347)
vol.l5	0.022887	0.022593	0.026049*	0.023502	0.022961
	(0.012328)	(0.012316)	(0.011859)	(0.012242)	(0.012546)
vol.l6	0.164034**	0.164442**	0.167546**	0.165323**	0.166695**
	(0.061085)	(0.061129)	(0.060128)	(0.061328)	(0.061192)
X.l1	0.000083	0.000045	0.019718	0.003399	0.006729
	(0.000231)	(0.000040)	(0.019004)	(0.004067)	(0.006313)
X.l2	−0.000473***	−0.000116***	0.005269	0.005600	0.002778
	(0.000087)	(0.000024)	(0.004162)	(0.005050)	(0.003938)
X.l3	−0.000804***	−0.000213***	−0.007797	−0.003904*	−0.004652*
	(0.000093)	(0.000028)	(0.005041)	(0.001717)	(0.001998)
X.l4	−0.000546***	−0.000147***	0.002275	0.000725	−0.002442*
	(0.000101)	(0.000023)	(0.002654)	(0.003504)	(0.001044)
X.l5	−0.000579***	−0.000119**	−0.001145	−0.002363	−0.000607
	(0.000146)	(0.000041)	(0.002728)	(0.001717)	(0.000993)
X.l6	−0.000099	0.000000	−0.002750	−0.001543	0.000596
	(0.000117)	(0.000033)	(0.002441)	(0.001170)	(0.000973)
const	0.008726***	0.007587***	0.005770***	0.005939***	0.005857**
	(0.001825)	(0.001707)	(0.001695)	(0.001706)	(0.001806)
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

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

```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")
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)))
```

```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)))
```

```
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)))
```

```
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)))
```

```
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)))
```

```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)))
```

```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
```

```r
robust_s_n = 2*(1-pt(abs(t_stat_s_n), df = df.residual(lm_clean_s_n)))
```

```r
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)))
```

```r
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
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)))
```

```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)
```

```r
#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
#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)

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,
```

```r
    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, prewhite = FALSE)))
nw_se_f_n_t <- sqrt(diag(sandwich::NeweyWest(model_f_n, lag = 6, prewhite = FALSE)))
nw_se_f_ta_t <- sqrt(diag(sandwich::NeweyWest(model_f_ta, lag = 6, prewhite = FALSE)))
nw_se_f_tr_t <- sqrt(diag(sandwich::NeweyWest(model_f_tr, lag = 6, prewhite = FALSE)))
nw_se_f_china_t <- sqrt(diag(sandwich::NeweyWest(model_f_ch, lag = 6, prewhite = FALSE)))


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


# Listes modèles, SE robustes et p-values robustes pour first
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)

# Noms personnalisés des coefficients
custom_names <- list(
  "vol.l1" = "$Vol_{t-1}$",
  "vol.l2" = "$Vol_{t-2}$",
  "vol.l3" = "$Vol_{t-3}$",
```

```r
  "vol.l4" = "$Vol_{t-4}$",
  "vol.l5" = "$Vol_{t-5}$",
  "vol.l6" = "$Vol_{t-6}$",
  "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"
)

# Générer tableau texreg pour first
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)
)

# Afficher le tableau
table_texreg_f
```

```r
#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)

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)
```

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

	TweetDummy	TweetCount	Tariff	Trade	China
Vol_{t-1}	0.541944^{***}	0.542426^{***}	0.543570^{***}	0.543958^{***}	0.543471^{***}
	(0.080972)	(0.080477)	(0.079273)	(0.079048)	(0.079428)
Vol_{t-2}	-0.113920^{**}	-0.113855^{**}	-0.115106^{**}	-0.115566^{**}	-0.115002^{**}
	(0.040758)	(0.040843)	(0.041031)	(0.040968)	(0.040995)
Vol_{t-3}	0.058050	0.057592	0.053635	0.053636	0.054382
	(0.030414)	(0.030474)	(0.030529)	(0.030534)	(0.030495)
Vol_{t-4}	0.188383	0.187417	0.184183	0.184102	0.184610
	(0.118235)	(0.117967)	(0.117238)	(0.117118)	(0.117369)
Vol_{t-5}	-0.088758	-0.089704	-0.091496	-0.091655	-0.091848
	(0.079651)	(0.079584)	(0.079702)	(0.079683)	(0.079634)
Vol_{t-6}	0.336662^{***}	0.337701^{***}	0.343373^{***}	0.343466^{***}	0.343184^{***}
	(0.048176)	(0.048104)	(0.047473)	(0.047512)	(0.047665)
X_{t-1}	-0.000478^{***}	-0.000163^{**}	-0.000454	-0.001838^{**}	-0.000352
	(0.000140)	(0.000057)	(0.000353)	(0.000702)	(0.000385)
X_{t-2}	-0.000184^{**}	-0.000063^{*}	-0.000289	0.000221	-0.000048
	(0.000070)	(0.000030)	(0.000271)	(0.000513)	(0.000233)
X_{t-3}	-0.000693^{***}	-0.000263^{***}	-0.001007^{***}	-0.000949^{**}	-0.001412^{***}
	(0.000153)	(0.000062)	(0.000267)	(0.000308)	(0.000359)
X_{t-4}	-0.000564^{***}	-0.000208^{***}	-0.000274	-0.000612	-0.000202
	(0.000159)	(0.000062)	(0.000392)	(0.000411)	(0.000452)
X_{t-5}	-0.000435^{***}	-0.000125^{**}	-0.000468	-0.000605	-0.000057
	(0.000118)	(0.000046)	(0.000274)	(0.000361)	(0.000354)
X_{t-6}	0.000118	0.000099^{*}	0.000240	-0.000121	0.000275
	(0.000122)	(0.000049)	(0.000344)	(0.000395)	(0.000371)
Constant	0.004020^{***}	0.003079^{***}	0.001510^{***}	0.001657^{***}	0.001593^{***}
	(0.000661)	(0.000520)	(0.000353)	(0.000371)	(0.000343)
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

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$

```r
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)

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


robust_s_d_t <- 2 * (1-pt(abs(coef(model_s_d) / nw_se_s_d_t), df = df.residual(model_s_d)))
```

```r
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))]

# Listes modèles, SE robustes et p-values robustes pour second
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)

# Générer tableau texreg pour second
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(0.016739, 0.015714, 0.011582, -0.004131, 0.015569)),
  star.cutoffs = c(0.05, 0.01, 0.001)
)



# Afficher le tableau
table_texreg_s
```

browseURL(file.path(getwd(), "table_trump.html"))

0.002919823 0.002236589 0.0004843789 0.0007026808 0.0009049532

0.01673941 0.01571499 0.01158212 -0.004131086 0.01556975

stargazer(model_f_d, model_f_n, model_f_ta, model_f_tr, model_f_ch, type = "text", se = 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), p = list(robust_f_d_t, robust_f_n_t, robust_f_ta_t, robust_f_tr_t, robust_f_ch_t), column.labels = c("Dummy", "TweetCount", "Tariff", "Trade", "China"), dep.var.labels = " ", dep.var.labels.include = FALSE, covariate.labels = c("volt-1","volt-2","volt-3","volt-4","volt-5","volt-6","Xt-1","Xt-2","Xt-3","Xt-4","Xt-5","Xt-6","const"), model.names = FALSE, model.numbers = FALSE, title = "Table 1 : Trump's Post on Volatility (Newey-West robust SE)", digits = 6, no.space = TRUE, omit.stat = c("f", "ser", "rsq", "adj.rsq"), star.cutoffs = c(0.001, 0.01, 0.0), flip = TRUE, add.lines=list(c("shock", "1", "2", "3", "4", "5")))

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

	TweetDummy	TweetCount	Tariff	Trade	China
Vol_{t-1}	0.299398**	0.299350**	0.294752**	0.301160**	0.274419***
	(0.112417)	(0.114098)	(0.108853)	(0.111015)	(0.081386)
Vol_{t-2}	0.015406	0.013567	0.020667	0.011769	0.031670
	(0.045643)	(0.046702)	(0.039243)	(0.045795)	(0.031719)
Vol_{t-3}	0.076169***	0.076851***	0.068749***	0.072284***	0.052697
	(0.010099)	(0.010266)	(0.016672)	(0.015313)	(0.033941)
Vol_{t-4}	0.084229	0.085108	0.074401	0.080544	0.035573
	(0.073259)	(0.073217)	(0.080381)	(0.069965)	(0.107841)
Vol_{t-5}	0.013424	0.010406	0.015342	0.017631	0.005467
	(0.009477)	(0.009620)	(0.009167)	(0.011320)	(0.031442)
Vol_{t-6}	0.126612*	0.126324*	0.132056*	0.124277*	0.150909*
	(0.058486)	(0.057687)	(0.057043)	(0.057466)	(0.059264)
X_{t-1}	0.006569	0.000947	0.027028	0.020463	0.154584
	(0.010877)	(0.001389)	(0.029078)	(0.031549)	(0.140076)
X_{t-2}	−0.003222**	−0.000736	0.008588	0.047163	0.099315
	(0.001221)	(0.000535)	(0.007247)	(0.041734)	(0.097425)
X_{t-3}	−0.005538**	−0.001637*	−0.010306	−0.026631	−0.047690
	(0.001707)	(0.000726)	(0.007507)	(0.021408)	(0.028342)
X_{t-4}	0.002474	0.000136	0.002002	0.019925	−0.020669
	(0.005119)	(0.000924)	(0.003272)	(0.031241)	(0.013733)
X_{t-5}	−0.008527*	−0.001651	−0.002649	−0.012965	−0.004470
	(0.004029)	(0.001070)	(0.004407)	(0.014516)	(0.020539)
X_{t-6}	−0.003594	−0.000627	−0.004279	−0.011100	0.008027
	(0.003213)	(0.000745)	(0.003857)	(0.010029)	(0.024100)
Constant	0.072524**	0.068423***	0.049265**	0.052127***	0.044027*
	(0.023894)	(0.020607)	(0.015173)	(0.013712)	(0.018061)
Shock (IRF)	0.016739	0.015714	0.011582	−0.004131	0.015569
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

$^{***}p < 0.001$; $^{**}p < 0.01$; $^{*}p < 0.05$