New Hope

Yimang Zhou

4/18/2021

```r
if (!require("pacman")) install.packages("pacman")
```

```
## Loading required package: pacman
```

```r
p_load(data.table, tidyverse, Hmisc, Matrix, lfe, plm, dynlm, car, lmtest, tseries, broom, knitr)
```

0. Data Preprocessing

Keep only 1960-2018 variables.

Instead of a selected bunch of countries, the dataframe df here includes all countries available.

```r
df <- read.csv("data_processing/financialization_df.csv")
df$X <- NULL
df <- df %>%
      filter(Year >= 1960)
colnames(df)
```

```
##  [1] "Country"                   "Year"
##  [3] "C_M2"                      "C_cpi"
##  [5] "IV_fdi_outflow"            "IV_lending"
##  [7] "C_gdp"                     "IV_trade_balance"
##  [9] "C_REER"                    "DV_VA"
## [11] "DV_nfc_ls"                 "DV_hh_ls"
## [13] "IV_gov_exp"                "IV_gini"
## [15] "C_gdp_log"                 "IV_trade_balance_log"
## [17] "DV_VA_lag1"                "DV_nfc_ls_lag1"
## [19] "DV_hh_ls_lag1"             "IV_lending_lag1"
## [21] "IV_gov_exp_lag1"           "IV_gini_lag1"
## [23] "IV_trade_balance_loglag1"  "IV_fdi_outflow_lag1"
## [25] "IV_lendingplus_lag2"       "IV_gov_expplus_lag2"
## [27] "IV_gini_lag2"              "IV_trade_balance_loglag2"
## [29] "IV_fdi_outflow_lag2"       "DV_VAplus"
## [31] "DV_nfc_lsplus"             "DV_hh_lsplus"
## [33] "DV_VAplus_lag1"            "DV_nfc_lsplus_lag1"
## [35] "DV_hh_lsplus_lag1"         "IV_lendingplus"
## [37] "IV_gov_expplus"            "IV_trade_balanceplus"
## [39] "IV_fdi_outflowplus"        "IV_lendingplus_lag1"
## [41] "IV_gov_expplus_lag1"       "IV_trade_balanceplus_lag1"
## [43] "IV_fdi_outflowplus_lag1"   "IV_trade_balanceplus_lag2"
## [45] "IV_fdi_outflowplus_lag2"   "C_REERplus"
## [47] "C_cpiplus"                 "C_gdpplus"
## [49] "C_m2plus"
```

Then create lagged variables and plus variables. Here I only have to proceed the variable of IV__gini.

```r
df <- df %>%
  group_by(Country) %>%
  mutate(IV_gini_lag1 = dplyr::lag(IV_gini,k=1)) %>%
  mutate(IV_giniplus = (IV_gini-IV_gini_lag1)/IV_gini_lag1*100) %>%
  mutate(IV_giniplus_lag1 = dplyr::lag(IV_giniplus,k=1)) %>%
  mutate(IV_giniplus_lag2 = dplyr::lag(IV_giniplus,k=2)) %>%
  ungroup()
```

In calculating "plus", it is possible to get inf results and they should be eliminated.

```r
turn_na <- function(a){
  a[abs(a)>10^10] <- NA
  a
}
for(i in 2:length(df)){
  df[,i] <- lapply(df[,i], turn_na)
}
```

Statistic description It is weird that gdp has a lot of NAs.

```r
for(i in 3:length(df)){
result <- paste0(colnames(df[,i]), " has ", sum(is.na(df[,i])), " NAs.")
print(result)
}
```

```
## [1] "C_M2 has 5609 NAs."
## [1] "C_cpi has 13204 NAs."
## [1] "IV_fdi_outflow has 5270 NAs."
## [1] "IV_lending has 14243 NAs."
## [1] "C_gdp has 13215 NAs."
## [1] "IV_trade_balance has 9602 NAs."
## [1] "C_REER has 13564 NAs."
## [1] "DV_VA has 14076 NAs."
## [1] "DV_nfc_ls has 13092 NAs."
## [1] "DV_hh_ls has 13087 NAs."
## [1] "IV_gov_exp has 12535 NAs."
## [1] "IV_gini has 14400 NAs."
## [1] "C_gdp_log has 13215 NAs."
## [1] "IV_trade_balance_log has 8643 NAs."
## [1] "DV_VA_lag1 has 14079 NAs."
## [1] "DV_nfc_ls_lag1 has 13104 NAs."
## [1] "DV_hh_ls_lag1 has 13098 NAs."
## [1] "IV_lending_lag1 has 14247 NAs."
## [1] "IV_gov_exp_lag1 has 12515 NAs."
## [1] "IV_gini_lag1 has 14420 NAs."
## [1] "IV_trade_balance_loglag1 has 8660 NAs."
## [1] "IV_fdi_outflow_lag1 has 5293 NAs."
## [1] "IV_lendingplus_lag2 has 412 NAs."
## [1] "IV_gov_expplus_lag2 has 284 NAs."
## [1] "IV_gini_lag2 has 14414 NAs."
## [1] "IV_trade_balance_loglag2 has 8679 NAs."
## [1] "IV_fdi_outflow_lag2 has 5315 NAs."
## [1] "DV_VAplus has 429 NAs."
## [1] "DV_nfc_lsplus has 35 NAs."
## [1] "DV_hh_lsplus has 35 NAs."
```

```
## [1] "DV_VAplus_lag1 has 430 NAs."
## [1] "DV_nfc_lsplus_lag1 has 36 NAs."
## [1] "DV_hh_lsplus_lag1 has 36 NAs."
## [1] "IV_lendingplus has 410 NAs."
## [1] "IV_gov_expplus has 282 NAs."
## [1] "IV_trade_balanceplus has 20 NAs."
## [1] "IV_fdi_outflowplus has 111 NAs."
## [1] "IV_lendingplus_lag1 has 411 NAs."
## [1] "IV_gov_expplus_lag1 has 283 NAs."
## [1] "IV_trade_balanceplus_lag1 has 21 NAs."
## [1] "IV_fdi_outflowplus_lag1 has 112 NAs."
## [1] "IV_trade_balanceplus_lag2 has 22 NAs."
## [1] "IV_fdi_outflowplus_lag2 has 113 NAs."
## [1] "C_REERplus has 412 NAs."
## [1] "C_cpiplus has 400 NAs."
## [1] "C_gdpplus has 326 NAs."
## [1] "C_m2plus has 1 NAs."
## [1] "IV_giniplus has 14424 NAs."
## [1] "IV_giniplus_lag1 has 14441 NAs."
## [1] "IV_giniplus_lag2 has 14441 NAs."
```

1. Do the three levels of financialization co-occur and in the US only ? (Hypothesis 1 & 2)

This part attempts to select countries experience financialization since 1960. We determine it by test if an indicator of financialization (i.e., a dependent variable) is stationary by ADF tests. If the indicator is stationary in a country, it indicates that the country is not financialized during the period in this dimension.

"The Dickey-Fuller test tests the null hypothesis that a unit root is present in an autoregressive model. The alternative hypothesis is different depending on which version of the test is used, but is usually stationarity and trend-stationary." (wikipeida)

A unit root is present if $\rho = 1$ in $y_t = \rho y_{t-1} + u_t$. If the null hypothesis is not rejected, a unit root is present and the variable is not stationary. Instead, a variable is stationary if the null hypothesis is reject.

Define adf function (http://www.econ.uiuc.edu/~econ508/R/e-ta8_R.html)

```r
"adf" <- function(x,k = 0, int = TRUE, trend = FALSE){
# NB:  returns conventional lm summary so p-values for adf test are wrong!
    require(dynlm)
    dx <- diff(x)
    formula <- paste("dx ~ L(x)")
    if(k > 0)
        formula <- paste(formula," + L(dx,1:k)")
    if(trend){
        s <- time(x)
        t <- ts(s - s[1],start = s[1],freq = frequency(x))
        formula <- paste(formula," + t")
        }
    if(!int) formula <- paste(formula," - 1")
    summary(dynlm(as.formula(formula)))
}
```

ADF test for each country Generate adf_test() function, the strategy is: 1. extract the column of independent variable X from df 2. for a certain country C, use adf() function to calculate the augmented Dickey-Fuller statistic for rejecting non-stationarity 3. combine the results from all countries together and report

```r
adf_test <- function(df, x, k = k){
  result <- data.frame(country = c(),
                       lx_t = c(),
                       lx_p = c(),
                       stationarity = c())
  df %>%
    select(Country, Year, x) -> iv
  iv <- na.omit(iv)
  iv$Country <- as.character(iv$Country)
  country <- data.frame(table(iv$Country))[,1]
  for(i in 1:length(country)){
    countryname = as.character(country[i])
    temp <- iv %>%
      filter(Country == countryname)
    adf_iv <- ts(temp[,3])
    adf(adf_iv, k = k, int = T, trend = T) -> adf_model
    adf_model$coefficient[2,3] -> lx_t
    adf_model$coefficient[2,4] -> lx_p
    stationarity = c()
    stationarity[lx_p < 0.05] <- "stationarity"
    stationarity[lx_p > 0.05] <- "non-stationarity"
    newrow <- c(countryname, lx_t, lx_p, stationarity)
    result <- rbind(result, newrow)
  }
  colnames(result) <- c("country","lx_t","lx_p","stationarity")
  print(result)
}
```

Adf test for DV_VA

```r
VA_adf <- adf_test(df,"DV_VA",1)
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(x)` instead of `x` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
##                        country              lx_t                 lx_p
## 1                    Australia  -4.12355521658247 0.000413755444938922
## 2                      Austria  -2.19779626575383   0.0342994311490942
## 3                      Belgium  -2.77492918103848   0.0124893025442422
## 4                       Brazil  -1.53601573510329    0.144072150829083
## 5                       Canada  -2.08721843292199   0.0571214341715726
## 6                        Chile   -3.386259957052  0.00351062057896628
## 7  China (People's Republic of)  -1.90263646416831   0.0641271581630633
## 8               Czech Republic  -3.69191139594907  0.00144416500039166
## 9                      Denmark  -1.93510673870152    0.059571936411496
## 10                     Estonia  -2.10585447930474   0.0495188902894878
## 11                     Finland  -1.53456975439003    0.133174433112129
## 12                      France  -2.42229596092349   0.0197105350801031
## 13                     Germany  -2.65052554792073   0.0146092663625594
## 14                      Greece -0.628789138655818    0.537385585542487
## 15                     Hungary  -1.63027036891104    0.120417968941695
## 16                     Iceland -0.401200522238363     0.69432916478238
## 17                     Ireland -0.889246187505682    0.385601095420194
```

```
## 18                  Israel  -1.66260836813039    0.114716713499623
## 19                   Italy  -2.19757573925472   0.0383147904212295
## 20                   Japan  -1.54612199606178    0.139475756168772
## 21                   Korea   -1.7466841380986   0.0880024183690864
## 22                  Latvia  -2.85001829923415   0.0106313268518471
## 23               Lithuania   -1.8978560138482   0.0738720933448326
## 24              Luxembourg  -3.65259261801807  0.00182109407927302
## 25                  Mexico  -3.74542125493545  0.00137021536723159
## 26             Netherlands  -3.49949650442143  0.00109830780604386
## 27             New Zealand  -2.22680911635395   0.0316550660962191
## 28                  Norway  -2.57104363167739   0.0136823552078991
## 29                  Poland  -4.81254671113279 0.000139525001171999
## 30                Portugal  -1.16009749608356    0.261165793905357
## 31         Slovak Republic  -1.85990451550057   0.0793212627204507
## 32                Slovenia  -2.81136735133403   0.0115517494383717
## 33            South Africa -0.355472561575209    0.725957203962754
## 34                   Spain  -1.99312343020578   0.0616304429916874
## 35                  Sweden  -2.88689089676355   0.0068139637436216
## 36             Switzerland  -1.78050194550234   0.0882089149858961
## 37                  Turkey  -1.92789177579955   0.0744032027581534
## 38          United Kingdom -0.887211206593032    0.384148980336483
## 39           United States  -1.81731494276961   0.0906306334530486
##        stationarity
## 1      stationarity
## 2      stationarity
## 3      stationarity
## 4  non-stationarity
## 5  non-stationarity
## 6      stationarity
## 7  non-stationarity
## 8      stationarity
## 9  non-stationarity
## 10     stationarity
## 11 non-stationarity
## 12     stationarity
## 13     stationarity
## 14 non-stationarity
## 15 non-stationarity
## 16 non-stationarity
## 17 non-stationarity
## 18 non-stationarity
## 19     stationarity
## 20 non-stationarity
## 21 non-stationarity
## 22     stationarity
## 23 non-stationarity
## 24     stationarity
## 25     stationarity
## 26     stationarity
## 27     stationarity
## 28     stationarity
## 29     stationarity
## 30 non-stationarity
## 31 non-stationarity
```

```
## 32     stationarity
## 33 non-stationarity
## 34 non-stationarity
## 35     stationarity
## 36 non-stationarity
## 37 non-stationarity
## 38 non-stationarity
## 39 non-stationarity
```

ADF test for DV_hh_ls

```
hh_adf <- adf_test(df,"DV_hh_ls",1)
```

```
##                         country              lx_t                 lx_p
## 1                   Afghanistan   -3.0201253171714   0.0193828849587408
## 2                       Albania  -4.51482467617898  0.00111723242361956
## 3                     Argentina  -2.51675968926763   0.0209849332843938
## 4                     Australia  -2.15413570253539   0.0379981811084128
## 5                       Austria -0.607896087586097     0.55085001948297
## 6                    Bangladesh  -2.32840273790975   0.0587690555345349
## 7                       Belgium  -2.40861685094199   0.0217574387046835
## 8                        Brazil  -1.52367395090069    0.144061952764702
## 9                      Bulgaria  -2.24742136655743   0.0373883018743738
## 10                       C.A.R.  -1.71870786190353    0.111333687616232
## 11                     Cameroon  -1.90829750359924   0.0805586563367421
## 12                       Canada  -1.76612220189034   0.0843130205207385
## 13                         Chad  -2.71518599729673   0.0187743423650393
## 14                        Chile  -4.26229903365938  0.00133783700395264
## 15 China (People's Republic of) -0.355292915911001    0.732834885617317
## 16                     Colombia   -2.8880495295413   0.0102180071143074
## 17           Congo, Republic of  -1.52010780400154     0.15438561866301
## 18                   Costa Rica  -3.26223775511168  0.00680036773826405
## 19                      Croatia  -1.15361511692079    0.271117893474132
## 20                       Cyprus -0.949940698385644     0.35472378206319
## 21               Czech Republic  -1.50128728454845    0.152757489068002
## 22                      Denmark  -0.43718376841241    0.666906683965674
## 23                  El Salvador  -2.61288593598299   0.0226805890609531
## 24                      Estonia  -1.66365221336151    0.113494280513303
## 25                      Finland  -1.70144818871511   0.0960802207477342
## 26                       France  -1.43717043097343    0.159304884038289
## 27                      Germany   -1.4220181117649    0.162234418234517
## 28                       Greece  -1.11241198690889     0.27983842978802
## 29                     Honduras  -1.69409986669001     0.11601887830855
## 30                Hong Kong SAR   -2.0020984694989    0.057203179466305
## 31                      Hungary  -1.97614855666909   0.0537819737969108
## 32                      Iceland -0.937480518403866    0.353747399180564
## 33                        India  -2.67022853584853    0.017470202341253
## 34                    Indonesia  -2.88521611166655   0.0136972782868505
## 35                      Ireland  -1.89618310460846   0.0844943656968838
## 36                       Israel  -2.53058228736257   0.0194470161588574
## 37                        Italy  -1.63577257947978     0.10781294134532
## 38                        Japan  -0.76223459679673    0.449573667313601
## 39                   Kazakhstan  -3.51468357324225   0.0055877939062445
## 40                        Korea   -2.5008495421585   0.0156436511577492
## 41                       Latvia  -1.53686742286949    0.141718021589315
```

```
## 42                   Lesotho  -3.52922732210854  0.00415231263829475
## 43                 Lithuania   -2.2515708323146   0.0370779972962977
## 44                Luxembourg  -1.20318103487776    0.254162462826772
## 45            Macedonia, FYR  -8.05304107653214 4.16383502998213e-05
## 46                  Malaysia  -1.49675049195467    0.178120804067313
## 47                     Malta -0.187600724771688    0.853288194923162
## 48                 Mauritius  -1.87144279987493     0.11045766365952
## 49                    Mexico  -4.22564491590357   0.0004577420865306
## 50                   Morocco -0.492082982593466    0.631539553084179
## 51                   Myanmar  -1.98181373627982   0.0708781580026256
## 52                     Nepal  -2.38442544464736   0.0362166481393264
## 53               Netherlands  0.833185994959936    0.413308849587127
## 54               New Zealand  -1.59212599137328    0.125008262561431
## 55                 Nicaragua  -2.67466724500273   0.0202354389684712
## 56                    Norway  -1.93114019858952   0.0609505022817982
## 57                  Pakistan  -1.48556623443699    0.180977671766866
## 58                      Peru  -1.88070520674276   0.0844946428250696
## 59                    Poland  -1.40076205051191    0.178289091933429
## 60                  Portugal   -2.1016552328276   0.0430684866730964
## 61                   Romania  -1.78535187596752   0.0994808720330459
## 62                    Russia  -4.42047182543944 0.000496154149371979
## 63                     Samoa  -2.33661938579354   0.0312197712884131
## 64     São Tomé and Príncipe  -1.95121246243614   0.0920030867618418
## 65              Saudi Arabia   -2.2156011705695     0.0426065624001
## 66              Sierra Leone  -2.27205792135708   0.0422805780111362
## 67                 Singapore  -2.60628468279045    0.016119468056354
## 68           Slovak Republic  -2.52501579361588   0.0211760392143503
## 69                  Slovenia  -1.28759538251919    0.214196372420553
## 70           Solomon Islands  -2.94820343031683   0.0121848294420994
## 71              South Africa  -2.13174563642278    0.086213368964815
## 72                     Spain  -3.05861632895165  0.00439012283021471
## 73                 Sri Lanka  -1.64267758560979    0.128700070866996
## 74                    Sweden  -1.27214327070916    0.208875604904622
## 75               Switzerland  -2.01771200189772   0.0632107894365635
## 76                Tajikistan  -1.55546996104784    0.145801349250332
## 77                  Thailand  -0.82184598235322    0.430323517480786
## 78                    Turkey  -1.60830018905099    0.119401787197779
## 79                    U.A.E.  -2.80493266622572   0.0377731988860208
## 80                   Ukraine  -2.28461147502927   0.0346907318330252
## 81            United Kingdom  -2.71346643064133   0.0092774558676321
## 82             United States  -2.04594852563032   0.0457374539932064
## 83                   Vanuatu -0.405344780500349    0.692357551422037
##        stationarity
## 1      stationarity
## 2      stationarity
## 3      stationarity
## 4      stationarity
## 5  non-stationarity
## 6  non-stationarity
## 7      stationarity
## 8  non-stationarity
## 9      stationarity
## 10 non-stationarity
## 11 non-stationarity
```

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

```
## 66     stationarity
## 67     stationarity
## 68     stationarity
## 69 non-stationarity
## 70     stationarity
## 71 non-stationarity
## 72     stationarity
## 73 non-stationarity
## 74 non-stationarity
## 75 non-stationarity
## 76 non-stationarity
## 77 non-stationarity
## 78 non-stationarity
## 79     stationarity
## 80     stationarity
## 81     stationarity
## 82     stationarity
## 83 non-stationarity
```

ADF test for DV_hh_ls

```r
hh_adf <- adf_test(df,"DV_hh_ls",1)
```

```
##                        country              lx_t                 lx_p
## 1                  Afghanistan   -3.0201253171714   0.0193828849587408
## 2                      Albania  -4.51482467617898  0.00111723242361956
## 3                    Argentina  -2.51675968926763   0.0209849332843938
## 4                    Australia  -2.15413570253539   0.0379981811084128
## 5                      Austria -0.607896087586097     0.55085001948297
## 6                   Bangladesh  -2.32840273790975   0.0587690555345349
## 7                      Belgium  -2.40861685094199   0.0217574387046835
## 8                       Brazil  -1.52367395090069    0.144061952764702
## 9                     Bulgaria  -2.24742136655743   0.0373883018743738
## 10                      C.A.R.  -1.71870786190353    0.111333687616232
## 11                    Cameroon  -1.90829750359924   0.0805586563367421
## 12                      Canada  -1.76612220189034   0.0843130205207385
## 13                        Chad  -2.71518599729673   0.0187743423650393
## 14                       Chile  -4.26229903365938  0.00133783700395264
## 15 China (People's Republic of) -0.355292915911001    0.732834885617317
## 16                    Colombia   -2.8880495295413   0.0102180071143074
## 17           Congo, Republic of  -1.52010780400154     0.15438561866301
## 18                  Costa Rica  -3.26223775511168  0.00680036773826405
## 19                     Croatia  -1.15361511692079    0.271117893474132
## 20                      Cyprus -0.949940698385644     0.35472378206319
## 21              Czech Republic  -1.50128728454845    0.152757489068002
## 22                     Denmark  -0.43718376841241    0.666906683965674
## 23                 El Salvador  -2.61288593598299   0.0226805890609531
## 24                     Estonia  -1.66365221336151    0.113494280513303
## 25                     Finland  -1.70144818871511   0.0960802207477342
## 26                      France  -1.43717043097343    0.159304884038289
## 27                     Germany   -1.4220181117649    0.162234418234517
## 28                      Greece  -1.11241198690889     0.27983842978802
## 29                    Honduras  -1.69409986669001     0.11601887830855
## 30                Hong Kong SAR   -2.0020984694989    0.057203179466305
## 31                     Hungary  -1.97614855666909   0.0537819737969108
```

```
## 32                  Iceland -0.937480518403866    0.353747399180564
## 33                    India  -2.67022853584853    0.017470202341253
## 34                Indonesia  -2.88521611166655   0.0136972782868505
## 35                  Ireland  -1.89618310460846   0.0844943656968838
## 36                   Israel  -2.53058228736257   0.0194470161588574
## 37                    Italy  -1.63577257947978     0.10781294134532
## 38                    Japan  -0.76223459679673    0.449573667313601
## 39               Kazakhstan  -3.51468357324225   0.0055877939062445
## 40                    Korea   -2.5008495421585   0.0156436511577492
## 41                   Latvia  -1.53686742286949    0.141718021589315
## 42                  Lesotho  -3.52922732210854  0.00415231263829475
## 43                Lithuania   -2.2515708323146   0.0370779972962977
## 44               Luxembourg  -1.20318103487776    0.254162462826772
## 45           Macedonia, FYR  -8.05304107653214 4.16383502998213e-05
## 46                 Malaysia  -1.49675049195467    0.178120804067313
## 47                    Malta -0.187600724771688    0.853288194923162
## 48                Mauritius  -1.87144279987493     0.11045766365952
## 49                   Mexico  -4.22564491590357   0.0004577420865306
## 50                  Morocco -0.492082982593466    0.631539553084179
## 51                  Myanmar  -1.98181373627982   0.0708781580026256
## 52                    Nepal  -2.38442544464736   0.0362166481393264
## 53              Netherlands  0.833185994959936    0.413308849587127
## 54              New Zealand  -1.59212599137328    0.125008262561431
## 55                Nicaragua  -2.67466724500273   0.0202354389684712
## 56                   Norway  -1.93114019858952   0.0609505022817982
## 57                 Pakistan  -1.48556623443699    0.180977671766866
## 58                     Peru  -1.88070520674276   0.0844946428250696
## 59                   Poland  -1.40076205051191    0.178289091933429
## 60                 Portugal   -2.1016552328276   0.0430684866730964
## 61                  Romania  -1.78535187596752   0.0994808720330459
## 62                   Russia  -4.42047182543944 0.000496154149371979
## 63                    Samoa  -2.33661938579354   0.0312197712884131
## 64    São Tomé and Príncipe  -1.95121246243614   0.0920030867618418
## 65             Saudi Arabia   -2.2156011705695      0.0426065624001
## 66             Sierra Leone  -2.27205792135708   0.0422805780111362
## 67                Singapore  -2.60628468279045    0.016119468056354
## 68          Slovak Republic  -2.52501579361588   0.0211760392143503
## 69                 Slovenia  -1.28759538251919    0.214196372420553
## 70          Solomon Islands  -2.94820343031683   0.0121848294420994
## 71             South Africa  -2.13174563642278    0.086213368964815
## 72                    Spain  -3.05861632895165  0.00439012283021471
## 73                Sri Lanka  -1.64267758560979    0.128700070866996
## 74                   Sweden  -1.27214327070916    0.208875604904622
## 75              Switzerland  -2.01771200189772   0.0632107894365635
## 76               Tajikistan  -1.55546996104784    0.145801349250332
## 77                 Thailand  -0.82184598235322    0.430323517480786
## 78                   Turkey  -1.60830018905099    0.119401787197779
## 79                   U.A.E.  -2.80493266622572   0.0377731988860208
## 80                  Ukraine  -2.28461147502927   0.0346907318330252
## 81           United Kingdom  -2.71346643064133   0.0092774558676321
## 82            United States  -2.04594852563032   0.0457374539932064
## 83                  Vanuatu -0.405344780500349    0.692357551422037
##         stationarity
## 1       stationarity
```

```
## 2      stationarity
## 3      stationarity
## 4      stationarity
## 5  non-stationarity
## 6  non-stationarity
## 7      stationarity
## 8  non-stationarity
## 9      stationarity
## 10 non-stationarity
## 11 non-stationarity
## 12 non-stationarity
## 13     stationarity
## 14     stationarity
## 15 non-stationarity
## 16     stationarity
## 17 non-stationarity
## 18     stationarity
## 19 non-stationarity
## 20 non-stationarity
## 21 non-stationarity
## 22 non-stationarity
## 23     stationarity
## 24 non-stationarity
## 25 non-stationarity
## 26 non-stationarity
## 27 non-stationarity
## 28 non-stationarity
## 29 non-stationarity
## 30 non-stationarity
## 31 non-stationarity
## 32 non-stationarity
## 33     stationarity
## 34     stationarity
## 35 non-stationarity
## 36     stationarity
## 37 non-stationarity
## 38 non-stationarity
## 39     stationarity
## 40     stationarity
## 41 non-stationarity
## 42     stationarity
## 43     stationarity
## 44 non-stationarity
## 45     stationarity
## 46 non-stationarity
## 47 non-stationarity
## 48 non-stationarity
## 49     stationarity
## 50 non-stationarity
## 51 non-stationarity
## 52     stationarity
## 53 non-stationarity
## 54 non-stationarity
## 55     stationarity
```

```
## 56 non-stationarity
## 57 non-stationarity
## 58 non-stationarity
## 59 non-stationarity
## 60     stationarity
## 61 non-stationarity
## 62     stationarity
## 63     stationarity
## 64 non-stationarity
## 65     stationarity
## 66     stationarity
## 67     stationarity
## 68     stationarity
## 69 non-stationarity
## 70     stationarity
## 71 non-stationarity
## 72     stationarity
## 73 non-stationarity
## 74 non-stationarity
## 75 non-stationarity
## 76 non-stationarity
## 77 non-stationarity
## 78 non-stationarity
## 79     stationarity
## 80     stationarity
## 81     stationarity
## 82     stationarity
## 83 non-stationarity
```

Hypothesis 1: Financialization occurs in many other countries than the U.S. Hypothesis 2: The three levels of financialization do not co-occur in all countries.

Compare the varieties of financialization among different countries

First, many countries find some kind of financialization. Only a few countries do not witness financialization in all the three levels (Austrialia, Belgium, Chile, Mexico). This supports Hypothesis 1.

Second, only a few countries find all the three levels of financialization (Denmark, Greece, Hungary, Ireland, Japan, Turkey). Even in the U.S., financialization shows only in the level of the market. In most countries, the three levels of financialization do not co-occur.

```
variety <- merge(VA_adf, hh_adf, by = "country", all.y = T)
variety <- merge(variety, hh_adf, by = "country", all.y = T)
variety <- select(variety,
                  country, stationarity.x, stationarity.y, stationarity)
variety <- variety[,c("country", "stationarity.x", "stationarity.y", "stationarity")]
colnames(variety) <- c("country", "va", "hh", "hh")
variety
```

```
##                        country              va               hh
## 1                  Afghanistan            <NA>     stationarity
## 2                      Albania            <NA>     stationarity
## 3                    Argentina            <NA>     stationarity
## 4                    Australia    stationarity     stationarity
## 5                      Austria    stationarity non-stationarity
## 6                   Bangladesh            <NA> non-stationarity
## 7                      Belgium    stationarity     stationarity
```

```
## 8                        Brazil non-stationarity non-stationarity
## 9                      Bulgaria             <NA>     stationarity
## 10                       C.A.R.             <NA> non-stationarity
## 11                     Cameroon             <NA> non-stationarity
## 12                       Canada non-stationarity non-stationarity
## 13                         Chad             <NA>     stationarity
## 14                        Chile     stationarity     stationarity
## 15 China (People's Republic of) non-stationarity non-stationarity
## 16                     Colombia             <NA>     stationarity
## 17          Congo, Republic of              <NA> non-stationarity
## 18                   Costa Rica             <NA>     stationarity
## 19                      Croatia             <NA> non-stationarity
## 20                       Cyprus             <NA> non-stationarity
## 21               Czech Republic     stationarity non-stationarity
## 22                      Denmark non-stationarity non-stationarity
## 23                  El Salvador             <NA>     stationarity
## 24                      Estonia     stationarity non-stationarity
## 25                      Finland non-stationarity non-stationarity
## 26                       France     stationarity non-stationarity
## 27                      Germany     stationarity non-stationarity
## 28                       Greece non-stationarity non-stationarity
## 29                     Honduras             <NA> non-stationarity
## 30                Hong Kong SAR             <NA> non-stationarity
## 31                      Hungary non-stationarity non-stationarity
## 32                      Iceland non-stationarity non-stationarity
## 33                        India             <NA>     stationarity
## 34                    Indonesia             <NA>     stationarity
## 35                      Ireland non-stationarity non-stationarity
## 36                       Israel non-stationarity     stationarity
## 37                        Italy     stationarity non-stationarity
## 38                        Japan non-stationarity non-stationarity
## 39                   Kazakhstan             <NA>     stationarity
## 40                        Korea non-stationarity     stationarity
## 41                       Latvia     stationarity non-stationarity
## 42                      Lesotho             <NA>     stationarity
## 43                    Lithuania non-stationarity     stationarity
## 44                   Luxembourg     stationarity non-stationarity
## 45               Macedonia, FYR             <NA>     stationarity
## 46                     Malaysia             <NA> non-stationarity
## 47                        Malta             <NA> non-stationarity
## 48                    Mauritius             <NA> non-stationarity
## 49                       Mexico     stationarity     stationarity
## 50                      Morocco             <NA> non-stationarity
## 51                      Myanmar             <NA> non-stationarity
## 52                        Nepal             <NA>     stationarity
## 53                  Netherlands     stationarity non-stationarity
## 54                  New Zealand     stationarity non-stationarity
## 55                    Nicaragua             <NA>     stationarity
## 56                       Norway     stationarity non-stationarity
## 57                     Pakistan             <NA> non-stationarity
## 58                         Peru             <NA> non-stationarity
## 59                       Poland     stationarity non-stationarity
## 60                     Portugal non-stationarity     stationarity
## 61                      Romania             <NA> non-stationarity
```

```
## 62                     Russia              <NA>     stationarity
## 63                      Samoa              <NA>     stationarity
## 64       São Tomé and Príncipe             <NA> non-stationarity
## 65                Saudi Arabia             <NA>     stationarity
## 66                Sierra Leone             <NA>     stationarity
## 67                   Singapore             <NA>     stationarity
## 68             Slovak Republic non-stationarity     stationarity
## 69                    Slovenia     stationarity non-stationarity
## 70             Solomon Islands             <NA>     stationarity
## 71                South Africa non-stationarity non-stationarity
## 72                       Spain non-stationarity     stationarity
## 73                   Sri Lanka             <NA> non-stationarity
## 74                      Sweden     stationarity non-stationarity
## 75                 Switzerland non-stationarity non-stationarity
## 76                  Tajikistan             <NA> non-stationarity
## 77                    Thailand             <NA> non-stationarity
## 78                      Turkey non-stationarity non-stationarity
## 79                      U.A.E.             <NA>     stationarity
## 80                     Ukraine             <NA>     stationarity
## 81              United Kingdom non-stationarity     stationarity
## 82               United States non-stationarity     stationarity
## 83                     Vanuatu             <NA> non-stationarity
##                  hh
## 1      stationarity
## 2      stationarity
## 3      stationarity
## 4      stationarity
## 5  non-stationarity
## 6  non-stationarity
## 7      stationarity
## 8  non-stationarity
## 9      stationarity
## 10 non-stationarity
## 11 non-stationarity
## 12 non-stationarity
## 13     stationarity
## 14     stationarity
## 15 non-stationarity
## 16     stationarity
## 17 non-stationarity
## 18     stationarity
## 19 non-stationarity
## 20 non-stationarity
## 21 non-stationarity
## 22 non-stationarity
## 23     stationarity
## 24 non-stationarity
## 25 non-stationarity
## 26 non-stationarity
## 27 non-stationarity
## 28 non-stationarity
## 29 non-stationarity
## 30 non-stationarity
## 31 non-stationarity
```

```
## 32 non-stationarity
## 33     stationarity
## 34     stationarity
## 35 non-stationarity
## 36     stationarity
## 37 non-stationarity
## 38 non-stationarity
## 39     stationarity
## 40     stationarity
## 41 non-stationarity
## 42     stationarity
## 43     stationarity
## 44 non-stationarity
## 45     stationarity
## 46 non-stationarity
## 47 non-stationarity
## 48 non-stationarity
## 49     stationarity
## 50 non-stationarity
## 51 non-stationarity
## 52     stationarity
## 53 non-stationarity
## 54 non-stationarity
## 55     stationarity
## 56 non-stationarity
## 57 non-stationarity
## 58 non-stationarity
## 59 non-stationarity
## 60     stationarity
## 61 non-stationarity
## 62     stationarity
## 63     stationarity
## 64 non-stationarity
## 65     stationarity
## 66     stationarity
## 67     stationarity
## 68     stationarity
## 69 non-stationarity
## 70     stationarity
## 71 non-stationarity
## 72     stationarity
## 73 non-stationarity
## 74 non-stationarity
## 75 non-stationarity
## 76 non-stationarity
## 77 non-stationarity
## 78 non-stationarity
## 79     stationarity
## 80     stationarity
## 81     stationarity
## 82     stationarity
## 83 non-stationarity
```

2. Explain financialization in the level of the market

Hypothesis 3.1 (Statecraft model): VA increases as governmental spending increases. Hypothesis 4.1 (development model): VA increases as fdi inflows

2.0 panel ADF test

https://rdrr.io/rforge/punitroots/man/pCADFtest.html

```
p_load(fBasics,fUnitRoots)
if (!require("punitroots"))install.packages("punitroots", repos="http://R-Forge.R-project.org")
```

```
## Loading required package: punitroots

## Loading required package: CADFtest

## Loading required package: sandwich

## Loading required package: urca

##
## Attaching package: 'urca'

## The following objects are masked from 'package:fUnitRoots':
##
##     punitroot, qunitroot, unitrootTable

## Registered S3 methods overwritten by 'CADFtest':
##   method      from
##   bread.mlm   sandwich
##   estfun.mlm  sandwich
```

```
if (!require("CADFtest"))install.packages("CADFtest")
#install.packages("ua")
library(punitroots)
```

```
padf <- function(df, x){
  df %>%
    select(Country, Year, x) -> iv
  iv <- na.omit(iv)
  result = pCADFtest(Y=iv, max.lag.y = 5, criterion = "AIC", crosscorr=0.10)
  print(result)
}

VA_padf <- padf(df,"DV_VA")
```

```
##
##  Panel-ADF test
##
## data:
## test statistic.Ht = -12.153, mean.rho2 = NA, p-value < 2.2e-16
```

```
hh_padf <- padf(df,"DV_hh_ls")
```

```
##
##  Panel-ADF test
##
## data:
## test statistic.Ht = -12.433, mean.rho2 = NA, p-value < 2.2e-16
```

```r
hh_padf <- padf(df,"DV_hh_ls")
```

```
##
##  Panel-ADF test
##
## data:
## test statistic.Ht = -12.433, mean.rho2 = NA, p-value < 2.2e-16
```

```r
lending_padf <- padf(df, "IV_lending")
```

```
##
##  Panel-ADF test
##
## data:
## test statistic.Ht = -11.666, mean.rho2 = NA, p-value < 2.2e-16
```

```r
govexp_padf <- padf(df, "IV_gov_exp")
```

```
##
##  Panel-ADF test
##
## data:
## test statistic.Ht = -12.549, mean.rho2 = NA, p-value < 2.2e-16
```

```r
gini_padf <- padf(df, "IV_gini")
```

```
##
##  Panel-ADF test
##
## data:
## test statistic.Ht = -7.0165, mean.rho2 = NA, p-value = 1.137e-12
```

```r
trade_balance_padf <- padf(df, "IV_trade_balance")
```

```
##
##  Panel-ADF test
##
## data:
## test statistic.Ht = -16.732, mean.rho2 = NA, p-value < 2.2e-16
```

```r
fdi_outflow_padf <- padf(df, "IV_fdi_outflow")
```

```
##
##  Panel-ADF test
##
## data:
## test statistic.Ht = -14.87, mean.rho2 = NA, p-value < 2.2e-16
```

2.1 sub-database

Get the countries of non-stationarity in DV_VA as the dataframe for financialization at the market level (DV_VA)

```r
#VA_country <- na.omit(variety$country[variety$va == "non-stationarity"])

df <- df %>%
  group_by(Country) %>%
  mutate(IV_lending_lag2 = dplyr::lag(IV_lending_lag1)) %>%
```

```r
  ungroup()

df %>%
dplyr::select(Country, Year, DV_VA, DV_VA_lag1,
              IV_lending,  IV_trade_balance_log, IV_fdi_outflow, IV_gini,
              IV_lending_lag1, IV_lending_lag2,
              IV_trade_balance_loglag1, IV_trade_balance_loglag2, IV_fdi_outflow_lag1, IV_fdi_outflow_la
              IV_gini_lag1, IV_gini_lag2,
              C_M2, C_REER, C_cpi, C_gdp_log) -> va
#va[va$Country %in% VA_country,] -> va
va[is.na(va)] <- 0
aggregate(. ~ Country + Year, data = va, sum) -> va
va[va==0] <- NA
va <- va[!is.na(va$DV_VA),]
```

2.3 Cointegration: Phillips-Ouliaris test

```r
library(tseries)
po.test(as.matrix(cbind(va$DV_VA, va$IV_lending), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(va$DV_VA, va$IV_lending), demean = FALSE)):
## p-value smaller than printed p-value

##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(va$DV_VA, va$IV_lending), demean = FALSE)
## Phillips-Ouliaris demeaned = -1073.4, Truncation lag parameter = 9,
## p-value = 0.01
```

```r
po.test(as.matrix(cbind(va$DV_VA, va$IV_trade_balance_log), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(va$DV_VA, va$IV_trade_balance_log), demean =
## FALSE)): p-value smaller than printed p-value

##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(va$DV_VA, va$IV_trade_balance_log), demean = FALSE)
## Phillips-Ouliaris demeaned = -1303.4, Truncation lag parameter = 10,
## p-value = 0.01
```

```r
po.test(as.matrix(cbind(va$DV_VA, va$IV_fdi_outflow), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(va$DV_VA, va$IV_fdi_outflow), demean =
## FALSE)): p-value smaller than printed p-value

##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(va$DV_VA, va$IV_fdi_outflow), demean = FALSE)
## Phillips-Ouliaris demeaned = -1400.7, Truncation lag parameter = 11,
## p-value = 0.01
```

```r
po.test(as.matrix(cbind(va$DV_VA, va$IV_gini), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(va$DV_VA, va$IV_gini), demean = FALSE)): p-
```

```
## value smaller than printed p-value
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(va$DV_VA, va$IV_gini), demean = FALSE)
## Phillips-Ouliaris demeaned = -759.45, Truncation lag parameter = 5,
## p-value = 0.01
```

2.4 FE test and Hausman test for va

Here we use Hausman test to determine whether an FE model or a RE model fit the data better. The null hypothesis is to use the RE model.(No correlation between the unique errors and the regressors in the model).

RE model cannot include lag2 variables, otherwise it reports the error: (I doubt this is because there are only 19 countries in the model).

Error in swar_Between_check(estm[[2L]], method) : model not estimable: 20 coefficient(s) (incl. intercept) to be estimated for the between model but only 11 individual(s)

Hausman test supports null hypothesis.

```
library(Matrix)
library(plm)
library(car)
library(lmtest)

# m1 is an FE model
m1  <- plm(DV_VA ~  DV_VA_lag1 +
                        IV_lending + IV_lending_lag1 + #IV_lending_lag2 +
                        IV_gini + IV_gini_lag1 + #IV_gini_lag2 +
                        IV_trade_balance_log + IV_trade_balance_loglag1 + #IV_trade_balance_loglag2 +
                        IV_fdi_outflow + IV_fdi_outflow_lag1 + #IV_fdi_outflow_lag2 +
                        C_M2 + C_REER + C_gdp_log,
          data = va, model = 'within',
          effect = 'twoways', index = c('Country', 'Year'))
summary(m1)
```

```
## Twoways effects Within Model
##
## Call:
## plm(formula = DV_VA ~ DV_VA_lag1 + IV_lending + IV_lending_lag1 +
##     IV_gini + IV_gini_lag1 + IV_trade_balance_log + IV_trade_balance_loglag1 +
##     IV_fdi_outflow + IV_fdi_outflow_lag1 + C_M2 + C_REER + C_gdp_log,
##     data = va, effect = "twoways", model = "within", index = c("Country",
##         "Year"))
##
## Unbalanced Panel: n = 19, T = 8-28, N = 368
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -1.878747 -0.259479 -0.014779  0.252341  2.066954
##
## Coefficients:
##                            Estimate Std. Error t-value  Pr(>|t|)
## DV_VA_lag1                0.7268129  0.0366653 19.8229 < 2.2e-16 ***
## IV_lending               -0.0075559  0.0158477 -0.4768 0.6338544
```

```
## IV_lending_lag1          -0.0117998  0.0164751 -0.7162 0.4743962
## IV_gini                   6.0697955  2.4068818  2.5219 0.0121764 *
## IV_gini_lag1             -3.5639013  2.3781608 -1.4986 0.1349995
## IV_trade_balance_log     -0.0019764  0.0021012 -0.9406 0.3476261
## IV_trade_balance_loglag1 -0.0026882  0.0020785 -1.2934 0.1968488
## IV_fdi_outflow            0.0149027  0.0040314  3.6967 0.0002583 ***
## IV_fdi_outflow_lag1      -0.0011199  0.0044136 -0.2537 0.7998704
## C_M2                      0.0027573  0.0021549  1.2796 0.2016493
## C_REER                   -0.0006733  0.0026873 -0.2506 0.8023264
## C_gdp_log                -0.7187213  0.4227746 -1.7000 0.0901352 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    206.67
## Residual Sum of Squares: 71.03
## R-Squared:      0.6563
## Adj. R-Squared: 0.59179
## F-statistic: 49.1709 on 12 and 309 DF, p-value: < 2.22e-16
```

```r
# PCSE
summary(m1, vcovBK)
```

```
## Twoways effects Within Model
##
## Note: Coefficient variance-covariance matrix supplied: vcovBK
##
## Call:
## plm(formula = DV_VA ~ DV_VA_lag1 + IV_lending + IV_lending_lag1 +
##     IV_gini + IV_gini_lag1 + IV_trade_balance_log + IV_trade_balance_loglag1 +
##     IV_fdi_outflow + IV_fdi_outflow_lag1 + C_M2 + C_REER + C_gdp_log,
##     data = va, effect = "twoways", model = "within", index = c("Country",
##         "Year"))
##
## Unbalanced Panel: n = 19, T = 8-28, N = 368
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -1.878747 -0.259479 -0.014779  0.252341  2.066954
##
## Coefficients:
##                            Estimate Std. Error t-value  Pr(>|t|)
## DV_VA_lag1                0.7268129  0.0366085 19.8537 < 2.2e-16 ***
## IV_lending               -0.0075559  0.0160991 -0.4693  0.639160
## IV_lending_lag1          -0.0117998  0.0190159 -0.6205  0.535371
## IV_gini                   6.0697955  2.3039005  2.6346  0.008849 **
## IV_gini_lag1             -3.5639013  2.5679084 -1.3879  0.166179
## IV_trade_balance_log     -0.0019764  0.0018418 -1.0731  0.284076
## IV_trade_balance_loglag1 -0.0026882  0.0017854 -1.5056  0.133179
## IV_fdi_outflow            0.0149027  0.0048725  3.0585  0.002419 **
## IV_fdi_outflow_lag1      -0.0011199  0.0035038 -0.3196  0.749474
## C_M2                      0.0027573  0.0027122  1.0166  0.310123
## C_REER                   -0.0006733  0.0031685 -0.2125  0.831855
## C_gdp_log                -0.7187213  0.5040709 -1.4258  0.154926
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Total Sum of Squares:    206.67
## Residual Sum of Squares: 71.03
## R-Squared:      0.6563
## Adj. R-Squared: 0.59179
## F-statistic: 50.4199 on 12 and 18 DF, p-value: 2.461e-11
```

```r
# Durbin-Watson Statistics
pdwtest(m1)
```

```
##
##  Durbin-Watson test for serial correlation in panel models
##
## data:  DV_VA ~ DV_VA_lag1 + IV_lending + IV_lending_lag1 + IV_gini +     IV_gini_lag1 + IV_trade_bala
## DW = 2.102, p-value = 0.7795
## alternative hypothesis: serial correlation in idiosyncratic errors
```

```r
# m1_re is a RE model
m1_re  <- plm(DV_VA ~  DV_VA_lag1 +
                        IV_lending + IV_lending_lag1 + #IV_lending_lag2 +
                        IV_gini + IV_gini_lag1 + #IV_gini_lag2 +
                        IV_trade_balance_log + IV_trade_balance_loglag1 + #IV_trade_balance_loglag2 +
                        IV_fdi_outflow + IV_fdi_outflow_lag1 + #IV_fdi_outflow_lag2 +
                        C_M2 + C_REER + C_gdp_log,
          data = va, model = 'random')
phtest(m1, m1_re)
```

```
##
##  Hausman Test
##
## data:  DV_VA ~ DV_VA_lag1 + IV_lending + IV_lending_lag1 + IV_gini +  ...
## chisq = 54.195, df = 12, p-value = 2.522e-07
## alternative hypothesis: one model is inconsistent
```

3. Explain financialization in the level of corporate

Hypothesis 3.2 (Statecraft model): hh increases as governmental spending increases. Hypothesis 4.2 (development model): hh increases as fdi inflows

3.1 sub-database

Get the countries of non-stationarity in DV_hh_ls as the dataframe for financialization at the market level

```r
df %>%
dplyr::select(Country, Year, DV_hh_ls, DV_hh_ls_lag1,
              IV_lending,  IV_trade_balance_log, IV_fdi_outflow, IV_gini,
              IV_lending_lag1,IV_trade_balance_loglag1, IV_fdi_outflow_lag1,IV_gini_lag1,
              C_M2, C_REER, C_cpi, C_gdp_log) -> hh
#hh[hh$Country %in% hh_country,] -> hh
hh[is.na(hh)] <- 0
aggregate(. ~ Country + Year, data = hh, sum) -> hh
hh[hh==0] <- NA
hh <- hh[!is.na(hh$DV_hh_ls),]
```

3.3 Cointegration: Phillips-Ouliaris test

```
library(tseries)
po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_lending), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_lending), demean =
## FALSE)): p-value smaller than printed p-value
```

```
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(hh$DV_hh_ls, hh$IV_lending), demean = FALSE)
## Phillips-Ouliaris demeaned = -1035.2, Truncation lag parameter = 9,
## p-value = 0.01
```

```
#po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_gov_exp), demean=FALSE))
po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_trade_balance_log), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_trade_balance_log), : p-
## value smaller than printed p-value
```

```
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(hh$DV_hh_ls, hh$IV_trade_balance_log), demean = FALSE)
## Phillips-Ouliaris demeaned = -2090.5, Truncation lag parameter = 18,
## p-value = 0.01
```

```
po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_fdi_outflow), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_fdi_outflow), demean =
## FALSE)): p-value smaller than printed p-value
```

```
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(hh$DV_hh_ls, hh$IV_fdi_outflow), demean = FALSE)
## Phillips-Ouliaris demeaned = -2166.9, Truncation lag parameter = 19,
## p-value = 0.01
```

```
po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_gini), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_gini), demean = FALSE)):
## p-value smaller than printed p-value
```

```
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(hh$DV_hh_ls, hh$IV_gini), demean = FALSE)
## Phillips-Ouliaris demeaned = -905.63, Truncation lag parameter = 5,
## p-value = 0.01
```

3.4 Hausman Test

```
library(Matrix)
library(plm)
library(car)
library(lmtest)
```

```r
# m2 is an FE model
m2  <- plm(DV_hh_ls ~  DV_hh_ls_lag1 +
                        IV_lending + IV_lending_lag1 + #IV_lending_lag2 +
                        IV_gini + IV_gini_lag1 + #IV_gini_lag2 +
                        IV_trade_balance_log + IV_trade_balance_loglag1 + #IV_trade_balance_loglag2 +
                        IV_fdi_outflow + IV_fdi_outflow_lag1 + #IV_fdi_outflow_lag2 +
                        C_M2 + C_REER + C_gdp_log,
          data = hh, model = 'within',
          effect = 'twoways', index = c('Country', 'Year'))
summary(m2)
```

```
## Twoways effects Within Model
##
## Call:
## plm(formula = DV_hh_ls ~ DV_hh_ls_lag1 + IV_lending + IV_lending_lag1 +
##     IV_gini + IV_gini_lag1 + IV_trade_balance_log + IV_trade_balance_loglag1 +
##     IV_fdi_outflow + IV_fdi_outflow_lag1 + C_M2 + C_REER + C_gdp_log,
##     data = hh, effect = "twoways", model = "within", index = c("Country",
##         "Year"))
##
## Unbalanced Panel: n = 19, T = 8-28, N = 354
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -7.865581 -1.124894  0.013479  1.160442  6.705462
##
## Coefficients:
##                            Estimate  Std. Error t-value  Pr(>|t|)
## DV_hh_ls_lag1             0.9437132   0.0199630 47.2731 < 2.2e-16 ***
## IV_lending                0.0137158   0.0747521  0.1835 0.8545442
## IV_lending_lag1           0.1630556   0.0754562  2.1609 0.0315054 *
## IV_gini                 -12.4490636  11.3440199 -1.0974 0.2733563
## IV_gini_lag1             10.4317559  11.0500484  0.9440 0.3459186
## IV_trade_balance_log     -0.0353706   0.0098482 -3.5916 0.0003849 ***
## IV_trade_balance_loglag1 -0.0085729   0.0098347 -0.8717 0.3840848
## IV_fdi_outflow            0.0659400   0.0185322  3.5581 0.0004351 ***
## IV_fdi_outflow_lag1      -0.0244772   0.0200384 -1.2215 0.2228663
## C_M2                      0.0125913   0.0098160  1.2827 0.2005891
## C_REER                    0.0421001   0.0128157  3.2850 0.0011426 **
## C_gdp_log                 4.9166176   1.9782931  2.4853 0.0134989 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    15383
## Residual Sum of Squares: 1412
## R-Squared:      0.90821
## Adj. R-Squared: 0.89017
## F-statistic: 243.244 on 12 and 295 DF, p-value: < 2.22e-16
```

```r
# PCSE
summary(m2, vcovBK)
```

```
## Twoways effects Within Model
##
```

```
## Note: Coefficient variance-covariance matrix supplied: vcovBK
##
## Call:
## plm(formula = DV_hh_ls ~ DV_hh_ls_lag1 + IV_lending + IV_lending_lag1 +
##     IV_gini + IV_gini_lag1 + IV_trade_balance_log + IV_trade_balance_loglag1 +
##     IV_fdi_outflow + IV_fdi_outflow_lag1 + C_M2 + C_REER + C_gdp_log,
##     data = hh, effect = "twoways", model = "within", index = c("Country",
##         "Year"))
##
## Unbalanced Panel: n = 19, T = 8-28, N = 354
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -7.865581 -1.124894  0.013479  1.160442  6.705462
##
## Coefficients:
##                             Estimate  Std. Error t-value  Pr(>|t|)
## DV_hh_ls_lag1              0.9437132   0.0267330 35.3014 < 2.2e-16 ***
## IV_lending                 0.0137158   0.0950968  0.1442  0.885417
## IV_lending_lag1            0.1630556   0.0934790  1.7443  0.082148 .
## IV_gini                  -12.4490636  12.0579094 -1.0324  0.302712
## IV_gini_lag1              10.4317559  10.7178748  0.9733  0.331199
## IV_trade_balance_log      -0.0353706   0.0091488 -3.8662  0.000136 ***
## IV_trade_balance_loglag1  -0.0085729   0.0102480 -0.8365  0.403526
## IV_fdi_outflow             0.0659400   0.0210618  3.1308  0.001918 **
## IV_fdi_outflow_lag1       -0.0244772   0.0216133 -1.1325  0.258341
## C_M2                       0.0125913   0.0139220  0.9044  0.366510
## C_REER                     0.0421001   0.0157892  2.6664  0.008090 **
## C_gdp_log                  4.9166176   3.0585649  1.6075  0.109016
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    15383
## Residual Sum of Squares: 1412
## R-Squared:      0.90821
## Adj. R-Squared: 0.89017
## F-statistic: 145.242 on 12 and 18 DF, p-value: 2.3293e-15
```

```r
# Durbin-Watson Statistics
pdwtest(m2)
```

```
##
##  Durbin-Watson test for serial correlation in panel models
##
## data:  DV_hh_ls ~ DV_hh_ls_lag1 + IV_lending + IV_lending_lag1 + IV_gini +     IV_gini_lag1 + IV_trad
## DW = 1.22, p-value = 2.108e-14
## alternative hypothesis: serial correlation in idiosyncratic errors
```

```r
# m2_re is a RE model
m2_re  <- plm(DV_hh_ls ~  DV_hh_ls_lag1 +
                        IV_lending + IV_lending_lag1 + #IV_lending_lag2 +
                        IV_gini + IV_gini_lag1 + #IV_gini_lag2 +
                        IV_trade_balance_log + IV_trade_balance_loglag1 + #IV_trade_balance_loglag2 +
                        IV_fdi_outflow + IV_fdi_outflow_lag1 + #IV_fdi_outflow_lag2 +
                        C_M2 + C_REER + C_gdp_log,
```

```
         data = hh, model = 'random')
phtest(m2, m2_re)
```

```
##
##  Hausman Test
##
## data:  DV_hh_ls ~ DV_hh_ls_lag1 + IV_lending + IV_lending_lag1 + IV_gini +  ...
## chisq = 69.341, df = 12, p-value = 4.253e-10
## alternative hypothesis: one model is inconsistent
```

4. Explain financialization in the level of corporate

Hypothesis 3.3 (Statecraft model): hh increases as governmental spending increases. Hypothesis 4.3 (development model): hh increases as fdi inflows

4.1 sub-database

Get the countries of non-stationarity in DV_hh_ls as the dataframe for financialization at the market

```
df %>%
dplyr::select(Country, Year, DV_hh_ls, DV_hh_ls_lag1,
              IV_lending,  IV_trade_balance_log, IV_fdi_outflow, IV_gini,
              IV_lending_lag1,IV_trade_balance_loglag1, IV_fdi_outflow_lag1,IV_gini_lag1,
              C_M2, C_REER, C_cpi, C_gdp_log) -> hh
#hh[hh$Country %in% hh_country,] -> hh
hh[is.na(hh)] <- 0
aggregate(. ~ Country + Year, data = hh, sum) -> hh
hh[hh==0] <- NA
hh <- hh[!is.na(hh$DV_hh_ls),]
```

4.3 Cointegration: Phillips-Ouliaris test

```
library(tseries)
po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_lending), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_lending), demean =
## FALSE)): p-value smaller than printed p-value
```

```
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(hh$DV_hh_ls, hh$IV_lending), demean = FALSE)
## Phillips-Ouliaris demeaned = -1035.2, Truncation lag parameter = 9,
## p-value = 0.01
```

```
#po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_gov_exp), demean=FALSE))
po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_trade_balance_log), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_trade_balance_log), : p-
## value smaller than printed p-value
```

```
##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(hh$DV_hh_ls, hh$IV_trade_balance_log), demean = FALSE)
## Phillips-Ouliaris demeaned = -2090.5, Truncation lag parameter = 18,
```

```
## p-value = 0.01
```

```r
po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_fdi_outflow), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_fdi_outflow), demean =
## FALSE)): p-value smaller than printed p-value

##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(hh$DV_hh_ls, hh$IV_fdi_outflow), demean = FALSE)
## Phillips-Ouliaris demeaned = -2166.9, Truncation lag parameter = 19,
## p-value = 0.01
```

```r
po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_gini), demean=FALSE))
```

```
## Warning in po.test(as.matrix(cbind(hh$DV_hh_ls, hh$IV_gini), demean = FALSE)):
## p-value smaller than printed p-value

##
##  Phillips-Ouliaris Cointegration Test
##
## data:  as.matrix(cbind(hh$DV_hh_ls, hh$IV_gini), demean = FALSE)
## Phillips-Ouliaris demeaned = -905.63, Truncation lag parameter = 5,
## p-value = 0.01
```

4.4 Hausman Test

```r
library(Matrix)
library(plm)
library(car)
library(lmtest)

# m3 is an FE model
m3  <- plm(DV_hh_ls ~  DV_hh_ls_lag1 +
                        IV_lending + IV_lending_lag1 + #IV_lending_lag2 +
                        IV_gini + IV_gini_lag1 + #IV_gini_lag2 +
                        IV_trade_balance_log + IV_trade_balance_loglag1 + #IV_trade_balance_loglag2 +
                        IV_fdi_outflow + IV_fdi_outflow_lag1 + #IV_fdi_outflow_lag2 +
                        C_M2 + C_REER + C_gdp_log,
          data = hh, model = 'within',
          effect = 'twoways', index = c('Country', 'Year'))
summary(m3)
```

```
## Twoways effects Within Model
##
## Call:
## plm(formula = DV_hh_ls ~ DV_hh_ls_lag1 + IV_lending + IV_lending_lag1 +
##     IV_gini + IV_gini_lag1 + IV_trade_balance_log + IV_trade_balance_loglag1 +
##     IV_fdi_outflow + IV_fdi_outflow_lag1 + C_M2 + C_REER + C_gdp_log,
##     data = hh, effect = "twoways", model = "within", index = c("Country",
##         "Year"))
##
## Unbalanced Panel: n = 19, T = 8-28, N = 354
##
## Residuals:
```

```
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -7.865581 -1.124894  0.013479  1.160442  6.705462
##
## Coefficients:
##                           Estimate  Std. Error t-value  Pr(>|t|)
## DV_hh_ls_lag1            0.9437132   0.0199630 47.2731 < 2.2e-16 ***
## IV_lending               0.0137158   0.0747521  0.1835 0.8545442
## IV_lending_lag1          0.1630556   0.0754562  2.1609 0.0315054 *
## IV_gini                -12.4490636  11.3440199 -1.0974 0.2733563
## IV_gini_lag1            10.4317559  11.0500484  0.9440 0.3459186
## IV_trade_balance_log    -0.0353706   0.0098482 -3.5916 0.0003849 ***
## IV_trade_balance_loglag1 -0.0085729  0.0098347 -0.8717 0.3840848
## IV_fdi_outflow           0.0659400   0.0185322  3.5581 0.0004351 ***
## IV_fdi_outflow_lag1     -0.0244772   0.0200384 -1.2215 0.2228663
## C_M2                     0.0125913   0.0098160  1.2827 0.2005891
## C_REER                   0.0421001   0.0128157  3.2850 0.0011426 **
## C_gdp_log                4.9166176   1.9782931  2.4853 0.0134989 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    15383
## Residual Sum of Squares: 1412
## R-Squared:      0.90821
## Adj. R-Squared: 0.89017
## F-statistic: 243.244 on 12 and 295 DF, p-value: < 2.22e-16
```

```r
# PCSE
summary(m3, vcovBK)
```

```
## Twoways effects Within Model
##
## Note: Coefficient variance-covariance matrix supplied: vcovBK
##
## Call:
## plm(formula = DV_hh_ls ~ DV_hh_ls_lag1 + IV_lending + IV_lending_lag1 +
##     IV_gini + IV_gini_lag1 + IV_trade_balance_log + IV_trade_balance_loglag1 +
##     IV_fdi_outflow + IV_fdi_outflow_lag1 + C_M2 + C_REER + C_gdp_log,
##     data = hh, effect = "twoways", model = "within", index = c("Country",
##         "Year"))
##
## Unbalanced Panel: n = 19, T = 8-28, N = 354
##
## Residuals:
##      Min.   1st Qu.    Median   3rd Qu.      Max.
## -7.865581 -1.124894  0.013479  1.160442  6.705462
##
## Coefficients:
##                           Estimate  Std. Error t-value  Pr(>|t|)
## DV_hh_ls_lag1            0.9437132   0.0267330 35.3014 < 2.2e-16 ***
## IV_lending               0.0137158   0.0950968  0.1442  0.885417
## IV_lending_lag1          0.1630556   0.0934790  1.7443  0.082148 .
## IV_gini                -12.4490636  12.0579094 -1.0324  0.302712
## IV_gini_lag1            10.4317559  10.7178748  0.9733  0.331199
## IV_trade_balance_log    -0.0353706   0.0091488 -3.8662  0.000136 ***
## IV_trade_balance_loglag1 -0.0085729  0.0102480 -0.8365  0.403526
```

```
## IV_fdi_outflow             0.0659400   0.0210618  3.1308  0.001918 **
## IV_fdi_outflow_lag1       -0.0244772   0.0216133 -1.1325  0.258341
## C_M2                       0.0125913   0.0139220  0.9044  0.366510
## C_REER                     0.0421001   0.0157892  2.6664  0.008090 **
## C_gdp_log                  4.9166176   3.0585649  1.6075  0.109016
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    15383
## Residual Sum of Squares: 1412
## R-Squared:      0.90821
## Adj. R-Squared: 0.89017
## F-statistic: 145.242 on 12 and 18 DF, p-value: 2.3293e-15
```

```r
# Durbin-Watson Statistics
pdwtest(m3)
```

```
##
##  Durbin-Watson test for serial correlation in panel models
##
## data:  DV_hh_ls ~ DV_hh_ls_lag1 + IV_lending + IV_lending_lag1 + IV_gini +     IV_gini_lag1 + IV_trac
## DW = 1.22, p-value = 2.108e-14
## alternative hypothesis: serial correlation in idiosyncratic errors
```

```r
# m3_re is a RE model
m3_re  <- plm(DV_hh_ls ~  DV_hh_ls_lag1 +
                        IV_lending + IV_lending_lag1 + #IV_lending_lag2 +
                        IV_gini + IV_gini_lag1 + #IV_gini_lag2 +
                        IV_trade_balance_log + IV_trade_balance_loglag1 + #IV_trade_balance_loglag2 +
                        IV_fdi_outflow + IV_fdi_outflow_lag1 + #IV_fdi_outflow_lag2 +
                        C_M2 + C_REER + C_gdp_log,
          data = hh, model = 'random')
phtest(m3, m3_re)
```

```
##
##  Hausman Test
##
## data:  DV_hh_ls ~ DV_hh_ls_lag1 + IV_lending + IV_lending_lag1 + IV_gini +  ...
## chisq = 69.341, df = 12, p-value = 4.253e-10
## alternative hypothesis: one model is inconsistent
```