Transparency and Tax Evasion:
Evidence from the Foreign Account Tax Compliance Act (FATCA)
De Simone, Lester, and Markle
SAS and STATA Code

```
*********************************************************************

* This program produces the starting STATA (.dta) datasets.  This code
is created in SAS and then used in STATA:
1. Appendix A: dlm_hmt15, dlm_hmt08, data1
2. Figure 1: data1
7. Table 1: data1 (for t-tests)
8. Table 2: data1
9. Table 3: data1
10. Table 4: cpisfull, zucmanfull
11. Table 5: data1
12. Table 6: data1

Figures and Tables are generated in STATA .do file: DLM Figures and
Tables.do

It also produces a SAS dataset, data1, that is used to produce the
descriptives in Table 1 in the program: DLM Table1 code.sas

;



/* Dependent variables:
     1. Logequity - total Equity FPI from Country j
     2. Logdebt - total Debt FPI from Country j
Suffixes:
     1. _us - into the US
     2. _non_us - into every country except the US
     3. _world - into every country (including the US)
Prefixes:
     1. no prefix - CPIS data
     2. tic_ - TIC data
     3. z_ - zucman's augmented
*/


* START TIC monthly data;

* 1984-2011 data ----------------------------------------------------
------;
libname dlm 'C:\Users\kmarkle\Dropbox\FATCA\JAR\DLM_JAR_Code_Data';


* dataset ticdata_ftot pulled from https://www.treasury.gov/resource-
center/data-chart-center/tic/Pages/shlreports.aspx and converted to
SAS using StatTransfer;
```

```
data h1; set dlm.ticdata_ftot; run;

proc sql; create table h2 as select distinct
      country_code as t_country,
      country_name as cname,
      year(date) as year,
      month(date) as month,
      ftot_agcy_est_pos as lta_c,
      ftot_corp_est_pos as ltc_c,
      ftot_stk_est_pos as lte_c,
      ftot_treas_est_pos as ltt_c
      from h1;
quit;
data h2; set h2;
      lte = LTE_c*1 ;
      ltt = ltt_c*1 ;
      lta = lta_c*1 ;
      ltc = ltc_c*1 ;
      drop  lte_c  ltt_c lta_c ltc_c ;
run;

*  2011-2015 data ----------------------------------------------------
-------;

* dataset Sltld_global_forsas pulled from
https://www.treasury.gov/resource-center/data-chart-
center/tic/Pages/ticsec2.aspx and converted to SAS using StatTransfer;
proc sql; create table r2 as select distinct
      ccode as t_country,
      country as cname,
      substr(date,1,4) as year_c,
      substr(date,6,2) as month_c,
      compress(agency,",") as lta_c,
      compress(corpbonds,",") as ltc_c,
      compress(corpstocks,",") as lte_c,
      compress(treasury,",") as ltt_c
      from dlm.Slt1d_global_forsas;
quit;
data r2; set r2;
      year=year_c*1;
      month=month_c*1;
      lte = LTE_c*1 ;
      ltt = ltt_c*1 ;
      lta = lta_c*1 ;
      ltc = ltc_c*1 ;
      drop year_c month_c lte_c  ltt_c lta_c ltc_c ;
run;

proc append data=r2 base=h2 force; run;

* get the country codes matching isos;
```

```
proc sql; create table c1 as select distinct
      t_country,
      cname,
      upcase(scan( cname, 1, '(' )) as upcountry
      from h2;
quit;

proc sql; create table cin as select distinct
      *
      from c1
      where upcountry in (select distinct country from
dlm.countries_2_3);
quit;
proc sql; create table cout as select distinct
      *
      from c1
      where upcountry not in (select distinct country from
dlm.countries_2_3);
quit;


proc sql; create table d1 as select distinct
      *,
      upcase(scan( cname, 1, '(' )) as upcountry
      from h2;
quit;

data d1 ; set d1 ;
if upcountry  = "CHINA, MAINLAND" then upcountry = "CHINA" ;
if upcountry  = "SERBIA AND MONTENEGRO" then upcountry = "SERBIA" ;
if upcountry  = "RUSSIA" then upcountry = "RUSSIAN FEDERATION" ;
if upcountry  = "BONAIRE, SINT EUSTATIUS, & SABA" then upcountry =
"BONAIRE, ST. EUSTATIUS, AND SABA" ;
if upcountry  = "ST. MAARTEN" then upcountry = "SAINT MARTIN" ;
if upcountry  = "FALKLAND ISLANDS" then upcountry = "FALKLAND ISLANDS
(MALVINAS)" ;
if upcountry  = "BRUNEI" then upcountry = "BRUNEI DARUSSALAM" ;
if upcountry  = "BURMA" then upcountry = "MYANMAR" ;
if upcountry  = "KOREA, SOUTH" then upcountry = "SOUTH KOREA" ;
if upcountry  = "LAOS" then upcountry = "LAO PEOPLE'S DEMOCRATIC
REPUBLIC" ;
if upcountry  = "MACAU" then upcountry = "MACAO" ;
if upcountry  = "KOREA, NORTH" then upcountry = "NORTH KOREA" ;
if upcountry  = "EAST TIMOR" then upcountry = "TIMOR-LESTE" ;
if upcountry  = "SYRIA" then upcountry = "SYRIAN ARAB REPUBLIC" ;
if upcountry  = "VIETNAM" then upcountry = "VIET NAM" ;
if upcountry  = "GAMBIA" then upcountry = "GAMBIA, THE" ;
if upcountry  = "COTE D'IVOIRE" then upcountry = "CÔTE D'IVOIRE" ;
if upcountry  = "REUNION" then upcountry = "RÉUNION" ;
if upcountry  = "FEDERATED STATES OF MICRONESIA" then upcountry =
"MICRONESIA" ;
```

```
if upcountry  = "KOREA" then upcountry = "SOUTH KOREA" ;
if upcountry  = "PITCAIRN ISLAND" then upcountry = "PITCAIRN" ;
if upcountry  = "TOKELAU ISLANDS" then upcountry = "TOKELAU" ;

date=mdy(month,1,year);

run ;

* add on iso2 and iso3 and start table;
proc sql; create table d2 as select distinct
      a.country,
      a.iso2,
      a.iso3,
      b.t_country,
      b.*
      from dlm.countries_2_3 (where=(missing(iso2)=0)) a left join d1 b
      on a.country = b.upcountry;
quit;
data d2; set d2;
      if missing(t_country) then delete;
      drop cname upcountry;
run;

proc sort data=d2 nodupkey out=d3 dupout=dups; by iso2 date; run;

data tic_monthly; set d3;
      label
            t_country = "Treasury numeric code for country"
            lte = "FPI in U.S. equities (M USD)"
            ltt = "FPI in U.S. treasury debt (M USD)"
            ltc = "FPI in U.S. corporate debt (M USD)"
            lta = "FPI in U.S. agency debt (M USD)"
      ;
run;

* END TIC data;



* START CPIS data;

* dataset dlm_full_matrices_keyvars was created using separate
program: DLM Data - CPIS and Zucman
That program uses the STATA code from Gabriel Zucman's website and
extends the sample period;

proc sql; create table z1 as select distinct
      a.*,
      b.iso3 as iso3_from,
      b.iso2 as iso2_from,
      b.country as country_from
      from dlm.dlm_full_matrices_keyvars a, dlm.countrycodes b
```

```
      where a.source=b.ifs;
quit;

* calculate total in world and total in US for each country-year;
data zus; set z1; where host=111; if missing (iso3_from) or
missing(year) then delete; run;

proc sql; create table zworld as select distinct
      country_from,
      iso2_from,
      iso3_from,
      year,
      sum(eqasset) as eqasset_world,
      sum(debtasset) as debtasset_world,
      sum(augmeqasset) as augmeq_world,
      sum(augmdebtasset) as augmdebt_world
      from z1
      group by iso3_from, year;
quit;

proc sql; create table zboth as select distinct
      a.country_from,
      a.iso3_from,
      a.iso2_from,
      a.year,
      a.eqasset_world,
      a.debtasset_world,
      log(a.augmeq_world) as logequity,
      log(a.augmeq_world-b.augmeqasset) as logequity_non_us,
      log(b.augmeqasset) as logequity_us,
      log(a.augmdebt_world) as logdebt,
      log(a.augmdebt_world-b.augmdebtasset) as logdebt_non_us,
      log(b.augmdebtasset) as logdebt_us,
      log(a.eqasset_world) as rawlogequity,
      log(a.eqasset_world-b.eqasset) as rawlogequity_non_us,
      log(b.eqasset) as rawlogequity_us,
      log(a.debtasset_world) as rawlogdebt,
      log(a.debtasset_world-b.debtasset) as rawlogdebt_non_us,
      log(b.debtasset) as rawlogdebt_us
      from zworld a, zus b
      where a.iso3_from=b.iso3_from and a.year=b.year;
quit;

data cpis; set zboth;
      rename
      logequity = z_logequity_world
      logequity_non_us = z_logequity_non_us
      logequity_us = z_logequity_us
      logdebt = z_logdebt_world
      logdebt_non_us = z_logdebt_non_us
      logdebt_us = z_logdebt_us;
run;
```

```
* END CPIS data;


* START control variables;

* START World Bank data - 2000 - 2014;

* transpose to long;
proc sort data=dlm.wb_pop out=pop; by iso3; run;
proc transpose data=pop out=popt;
      by iso3;
run;

data pop; set popt;
      year = tranwrd(_NAME_,"v","")*1;
      rename col1=population;
      drop _name_;
run;



* GDP;

proc sort data=dlm.wb_gdp out=gdp; by country_code; where
missing(country_code)=0; run;
proc transpose data=gdp out=gdpt;
      var v:;
      by country_code;
run;

data gdp; set gdpt;
      year = substr(_NAME_,2,4)*1;
      gdp=col1*1;
      drop _name_ col1;
run;


* xrate;

proc sort data=dlm.wb_xrate out=xrate; by country_code; where
missing(country_code)=0; run;
proc transpose data=xrate out=xratet;
      var v:;
      by country_code;
run;

data xrate; set xratet;
      year = substr(_NAME_,2,4)*1;
      xrate=col1*1;
      drop _name_ col1;
```

```
run;


* one dataset of all world bank variables;
proc sql; create table wbdata as select distinct
      a.*,
      e.xrate
      from gdp a,  xrate e
      where a.country_code=e.country_code
      and a.year=e.year;
quit;

proc sql; create table wbdata as select distinct
      a.*,
      b.population
      from wbdata a left join pop b
      on a.country_code=b.iso3
      and a.year=b.year;
quit;

* START world bank data 2015;

* Manually changed variable names in excel before starting this
program... file wdi2015:
Country
iso3
gdp
population
xrate;

data a1;
      set dlm.wdi2015;
      year=2015;
      n_gdp=gdp*1;
      n_population=population*1;
      n_xrate=xrate*1;
      drop
            gdp
population
xrate
;
      rename
      n_gdp=gdp
      n_population=population
      n_xrate=xrate
;

run;

* make sure iso3s match;
data a1; set a1;
      if iso3="TMP" then iso3="TLS";
```

```
      if iso3="ROM" then iso3="ROU";
      if iso3="ADO" then iso3="AND";
      if iso3="IMY" then iso3="IMN";
      if iso3="WBG" then iso3="PSE";
      if iso3="ZAR" then iso3="COD";
      if iso3 in ("CHI" "KSV") then delete;
run;

proc sql; create table a2 as select distinct
      *
      from a1
      where iso3 in (select distinct iso3 from dlm.countries_2_3);
quit;

* no XRATE for Euro countries, so add it;

proc sql; create table fx as select distinct country, iso3, xrate,
year from a1; quit;


proc sort data=dlm.euro out=eur; by country_code; where
country_code="EMU"; run;
proc transpose data=eur out=tempt;
      var v:;
      by country_code;
run;
data eur; set tempt;
      year = substr(_NAME_,2,4)*1;
      fx=round(col1*1,.001);
      rename country_code=iso3;
      drop _name_ col1;
run;

* add 2-digit iso to fx;
proc sql; create table fx2 as select distinct
      a.*,
      b.iso2
      from fx a, dlm.countries_2_3 b
      where a.iso3=b.iso3;
quit;

proc sql; create table junk as select distinct
      *
      from fx2
      where iso2 in ("AT" "BE" "BG" "CY" "CZ" "DK" "EE" "FI" "FR" "DE"
"GR" "HU" "IE" "IT" "LV" "LT" "LU" "MT" "NL" "PL" "PT" "RO" "SK" "SI"
"ES" "SE");
quit;
* tack EURO on;
proc sql; create table junk2 as select distinct
      a.*,
      b.fx as euro
```

```
      from junk a, eur (where=(year=2015)) b
      ;
quit;


* replace ;
proc sql; create table euros as select distinct
      iso3,
      year,
      euro as xrate
      from junk2
      where missing(xrate);
quit;

proc sql; create table fxcut as select distinct
      *
      from fx
      where iso3 not in (select distinct iso3 from euros);
quit;

proc append data=euros base=fxcut force; run;

* add xrate back on;
proc sql; create table a3 as select distinct
      a.*,
      b.xrate
      from a2 (drop=xrate) a, fxcut b
      where a.iso3=b.iso3;
quit;


data wbdata2015; set a3; run;

* END world bank data 2015;



* combine WB datasets;
proc sql; create table k1 as select distinct
      country_code as iso3,
      year,
      gdp,
      xrate,
      population,
      log(population) as logpopulation
      from wbdata where year lt 2015;
quit;

proc sql; create table temp as select distinct
      iso3,
      year,
      gdp,
```

```
      xrate,
      population,
      log(population) as logpopulation
      from wbdata2015;
quit;

proc append data=temp base=k1 force; run;

* fix some iso3s;
data k1; set k1;
      if iso3="TMP" then iso3="TLS";
      if iso3="ROM" then iso3="ROU";
      if iso3="ADO" then iso3="AND";
      if iso3="IMY" then iso3="IMN";
      if iso3="WBG" then iso3="PSE";
      if iso3="ZAR" then iso3="CO";
run;


* Caymans and Bermuda GDP ;
data k2; set k1;
      if iso3="CYM" and year=2007 then gdp=3516000000;
      if iso3="CYM" and year=2008 then gdp=3547600000;
      if iso3="CYM" and year=2009 then gdp=3337100000;
      if iso3="CYM" and year=2010 then gdp=3267000000;
      if iso3="CYM" and year=2011 then gdp=3348900000;
      if iso3="CYM" and year=2012 then gdp=3416900000;
      if iso3="CYM" and year=2013 then gdp=3482100000;
      if iso3="CYM" and year=2014 then gdp=3479800000;
      if iso3="CYM" and year=2015 then gdp=2762200000/.81835;
      if iso3="BMU" and year=2014 then gdp=5699992000;
      if iso3="BMU" and year=2015 then gdp=5927652000;

run;

data k2; set k2;
      loggdp = log(gdp);
run;

* fix the exchange rate for missing years of EURO;
* START -----------------------;
* fx ;
proc sort data=dlm.fx_01_15 out=fx; by country_code; where
missing(country_code)=0; run;
proc transpose data=fx out=tempt;
      var v:;
      by country_code;
run;

data fx; set tempt;
      year = substr(_NAME_,2,4)*1;
      fx=round(col1*1,.001);
```

```
      rename country_code=iso3;
      drop _name_ col1;
run;

proc sort data=dlm.euro out=eur; by country_code; where
country_code="EMU"; run;
proc transpose data=eur out=tempt;
      var v:;
      by country_code;
run;
data eur; set tempt;
      year = substr(_NAME_,2,4)*1;
      fx=round(col1*1,.001);
      rename country_code=iso3;
      drop _name_ col1;
run;

* fix some iso3s;
data fx; set fx;
      if iso3="TMP" then iso3="TLS";
      if iso3="ROM" then iso3="ROU";
      if iso3="ADO" then iso3="AND";
      if iso3="IMY" then iso3="IMN";
      if iso3="WBG" then iso3="PSE";
      if iso3="ZAR" then iso3="COG";
run;
* add 2-digit iso to fx;
proc sql; create table fx2 as select distinct
      a.*,
      b.iso2
      from fx a, dlm.countries_2_3 b
      where a.iso3=b.iso3;
quit;

proc sql; create table junk as select distinct
      *
      from fx2
      where iso2 in ("AT" "BE" "BG" "CY" "CZ" "DK" "EE" "FI" "FR" "DE"
"GR" "HU" "IE" "IT" "LV" "LT" "LU" "MT" "NL" "PL" "PT" "RO" "SK" "SI"
"ES" "SE");
quit;
* tack EURO on;
proc sql; create table junk2 as select distinct
      a.*,
      b.fx as euro
      from junk a, eur b
      where a.year=b.year;
quit;

proc sql; create table junk3 as select distinct
      *,
      sum (missing(fx)) as missloc
```

```
      from junk2
      group by iso3;
quit;

* replace missing euros;
proc sql; create table euros as select distinct
      iso3,
      year,
      euro as fx
      from junk3
      where missing(fx) and iso2 in ("AT" "BE" "BG" "CY" "CZ" "DK" "EE"
"FI" "FR" "DE" "GR" "HU" "IE" "IT" "LV" "LT" "LU" "MT" "NL" "PL" "PT"
"RO" "SK" "SI" "ES" "SE");
quit;

proc sql; create table fxcut as select distinct
      *
      from fx
      where iso3 not in (select distinct iso3 from euros) and
missing(fx)=0;
quit;

proc append data=euros base=fxcut force; run;

data fx; set fxcut; run;


*add to dataset;
proc sql; create table k2a as select distinct
      a.iso3,
      a.year,
      a.gdp,
      a.loggdp,
      a.population,
      a.logpopulation,
      case when missing(a.xrate) then
            b.fx
      else a.xrate
      end
            as xrate label="Official exchange rate (LCU per US$, period
average)"
      from k2 a left join fx b
      on a.iso3=b.iso3 and a.year=b.year;
quit;

* add missing Bermuda and Ecuador exchange rates to data;
data temp; set k2a;
      if iso3="BMU" then xrate=1;
      if iso3="ECU" then xrate=25000;
run;

data wbcontrols; set temp; run;
```

```
* START tech development and exchange controls;

* dataset dlm.wb_techdata was converted from Excel to SAS using
StatTransfer;
* Excel file was pulled from World Bank website;

proc sql; create table biggy as select distinct
      country,
      iso3,
      2000 as year,
      y2000 as IT_NET_SECR_P6
      from dlm.wb_techdata
      where code="IT_NET_SECR_P6";
quit;

* loop through years;

%macro bigloop;%let k=2001;     %do %until (%eval(&k)=2017);

proc sql; create table temp as select distinct
      country,
      iso3,
      &k as year,
      y&k as IT_NET_SECR_P6
      from dlm.wb_techdata
      where code="IT_NET_SECR_P6";
quit;

proc append data=temp base=biggy force; run;

%let k=%eval(&k+1);%end;%mend bigloop;
%bigloop;


* loop through variables;

%let vars=; proc sql noprint; select distinct code into: vars
separated by " " from dlm.wb_techdata where code ne "IT_NET_SECR_P6";
quit;
%put &vars;

%macro loop3(variables=);%let x=1;     %do %until
(%scan(&variables,&x)= );     %let var = %scan(&variables,&x);

proc sql; create table biggy_temp as select distinct
      country,
      iso3,
      2000 as year,
      y2000 as &var
      from dlm.wb_techdata
```

```
     where code="&var";
quit;

* loop through years;

%macro bigloop;%let k=2001;     %do %until (%eval(&k)=2017);

proc sql; create table temp as select distinct
     country,
     iso3,
     &k as year,
     y&k as &var
     from dlm.wb_techdata
     where code="&var";
quit;

proc append data=temp base=biggy_temp force; run;

%let k=%eval(&k+1);%end;%mend bigloop;
%bigloop;

proc sql; create table biggy as select distinct
     a.*,
     b.*
     from biggy a, biggy_temp b
     where a.iso3=b.iso3 and a.year=b.year;
quit;

%let x=%eval(&x+1);%end;%mend loop3;%loop3(variables= &vars);


data wbtech; set biggy;
     label IT_CEL_SETS = "Mobile cellular subscriptions"
                IT_CEL_SETS_P2 = "Mobile cellular subscriptions (per
100 people)"
                IT_NET_BBND = "Fixed broadband subscriptions"
                IT_NET_BBND_P2 = "Fixed broadband subscriptions (per
100 people)"
                IT_NET_SECR = "Secure Internet servers"
                IT_NET_SECR_P6 = "Secure Internet servers (per 1
million people)"
     ;
run;

* fill Mobile for Bermuda 2011 by taking average of 2010 and 2012;
proc sql; create table avg as select distinct
     iso3,
     it_cel_sets as sets_2010,
     it_cel_sets_p2 as sets_p2_2010
     from wbtech
     where  iso3="BMU" and year in (2010);
quit;
```

```
proc sql; create table avg as select distinct
      a.*,
      b.it_cel_sets as sets_2012,
      b.it_cel_sets_p2 as sets_p2_2012,
      mean(a.sets_2010,b.it_cel_sets) as sets_2011,
      mean(a.sets_p2_2010,b.it_cel_sets_p2) as sets_p2_2011
      from avg a, wbtech b
      where  a.iso3=b.iso3 and b.iso3="BMU" and b.year in (2012);
quit;

%let m_2011=; proc sql noprint; select distinct sets_2011 into: m_2011
from avg; quit;
%let m2_2011=; proc sql noprint; select distinct sets_p2_2011 into:
m2_2011 from avg; quit;

data wbtech; set wbtech;
      if iso3="BMU" and year=2011 then do;
            it_cel_sets=&m_2011;
            it_cel_sets_p2=&m2_2011;
            end;
run;

* END tech development and exchange controls;

* START government effectiveness;

* Dataset wgi_sas converted from Excel using StatTransfer;

data junk; set dlm.wgi_sas;
      where code="GE.PER.RNK";
run;
data junk2; set junk;
      keep iso3 v:;
run;
proc sort data=junk2; by iso3; run;
proc transpose data=junk2 out=junk3; by iso3; run;
data junk4; set junk3;
      year=substr(_name_,2,4)*1;
      rename col1=ge_pctile;
      drop _name_;
run;
proc sql; create table junk5 as select distinct
      * from junk4
      where iso3 in (select distinct iso3 from temp)
;quit;
proc sql; create table wgi as select distinct
      *,
      ge_pctile gt 50 as goveff_high_full
      from junk5
      group by year;
quit;
```

```
* need one dummy for the whole sample -- make non-time-varying -- use
2012;
proc sql; create table wgi2 as select distinct
      iso3,
      goveff_high_full ge 0.5 as goveff_high_full
      from wgi where year=2012
      group by iso3;
quit;

data wgi_dummies; set wgi2; run;

* END government effectiveness;




* combine controls into one dataset;

proc sql; create table controls as select distinct
      a.*,
      b.localrate
      from wbcontrols a, dlm.indiv_rates b
      where a.iso3=b.iso3 and a.year=b.year;
quit;

proc sql; create table controls as select distinct
      a.*,
      b.it_cel_sets_p2
      from controls a, wbtech b
      where a.iso3=b.iso3 and a.year=b.year;
quit;

proc sql; create table controls as select distinct
      a.*,
      b.goveff_high_full
      from controls a, wgi_dummies b
      where a.iso3=b.iso3;
quit;

proc sql; create table controls as select distinct
      *,
      iso3 in ("BHS" "BMU" "CHE" "PAN") as tiea4,
      iso3 in ("BHS" "BMU" "CHE") as tiea3
      from controls;
quit;



* add TIC data;
data tic; set tic_monthly;
```

```
      where month=12;
run;

proc sql; create table regdata as select distinct
      a.*,
      b.lte,
      b.ltc,
      b.lta,
      b.ltt
      from controls a, tic b
      where a.iso3=b.iso3 and a.year=b.year;
quit;


* add CPIS data;
proc sql; create table regdata as select distinct
      a.*,
      b.*
      from regdata a, cpis b
      where a.iso3=b.iso3_from and a.year=b.year;
quit;


proc sql; create table regdata as select distinct
      *,
      log(1+lte) as tic_logequity_us,
      log(1+ltc) as tic_logdebt_us,
      log(it_cel_sets_p2) as log_mobile_subs_p100,
      log(xrate) as log_xrate,
      iso2_from in (select distinct c2 from dlm.havenlistx6 where
DL__2009_ = 1) as haven
      from regdata;
quit;

data regdata; set regdata;
      rename
      rawlogequity = logequity_world
      rawlogdebt = logdebt_world
      rawlogequity_us = logequity_us
      rawlogdebt_us = logdebt_us
      rawlogequity_non_us = logequity_non_us
      rawlogdebt_non_us = logdebt_non_us;
run;

proc sql; create table regdata as select distinct
      b.country,
      a.*
      from regdata a, dlm.countries_2_3 b
      where a.iso3=b.iso3;
quit;
```

```
/*
* add region codes;
data regdata; set regdata;
     if iso3 in ("AUS" "CHN" "HKG" "IDN" "JPN" "KOR" "MYS" "NZL" "PHL"
"SGP" "THA") then region_wb = "EAP";
          else if iso3 in ("ARG" "BHS" "BMU" "BRA" "CHL" "COL" "GTM"
"ECU" "JAM" "PAN" "PER" "URY") then region_wb = "LAC";
          else if iso3 in ("EGY" "ISR" "LBN" "MAR") then region_wb =
"MEN";
          else if iso3 in ("CAN" "MEX") then region_wb = "NAM";
          else if iso3 in ("IND") then region_wb = "SAS";
          else if iso3 in ("ZAF") then region_wb = "SSA";
          else region_wb = "ECA";
     if iso3 in ("AUS" "CHN" "HKG" "IDN" "IND" "JPN" "KOR" "MYS" "NZL"
"PHL" "SGP" "THA") then region_wb2 = "EAP";
          else if iso3 in ("ARG" "BHS" "BMU" "BRA" "CHL" "COL" "GTM"
"ECU" "JAM" "PAN" "PER" "URY") then region_wb2 = "LAC";
          else if iso3 in ("EGY" "ISR" "LBN" "MAR" "ZAF") then
region_wb2 = "MEN";
          else if iso3 in ("CAN" "MEX") then region_wb2 = "NAM";
          else region_wb2 = "ECA";
     if iso3 in ("AUT" "BEL" "DEU" "ESP" "FIN" "FRA" "GRC" "IRL" "ITA"
"LUX" "NLD" "PRT") then do; region_tic5 = "EUR"; region_tic7 = "EUR";
end;
     if iso3 in ("CHE" "DNK" "GBR" "HUN" "NOR" "POL" "RUS" "SRB" "SWE"
"ROU" "TUR") then do; region_tic5 = "EUR"; region_tic7 = "EUN"; end;
     if iso3 in ("EGY" "MAR" "ZAF") then do; region_tic5 = "AFR";
region_tic7 = "AFR"; end;
     if iso3 in ("ARG" "BRA" "CAN" "CHL" "COL" "ECU" "GTM" "MEX" "PER"
"URY") then do; region_tic5 = "AME"; region_tic7 = "AME"; end;
     if iso3 in ("BHS" "BMU" "PAN" "JAM") then do; region_tic5 =
"AME"; region_tic7 = "CAR"; end;
     if iso3 in ("CHN" "HKG" "IDN" "IND" "ISR" "JPN" "KOR" "LBN" "MYS"
"PHL" "SGP" "THA") then do; region_tic5 = "ASI"; region_tic7 = "ASI";
end;
     if iso3 in ("AUS" "NZL") then do; region_tic5 = "AUS";
region_tic7 = "AUS"; end;
run;
*/

/* comment this out so that it does not override existing data1.dta
proc export data=regdata outfile="C:\_Research\DLM1\data1.dta"
replace;
run;
*/
data dlm.data1; set regdata; run;

* count obs to use as screens;
proc sql; create table descdata1 as select distinct
     country,
     iso3,
     haven,
```

```
      sum(missing(logequity_world)=0) as n_logequity_world,
      sum(missing(logequity_us)=0) as n_logequity_us,
      sum(missing(logequity_non_us)=0) as n_logequity_non_us,
      sum(missing(logdebt_world)=0) as n_logdebt_world,
      sum(missing(logdebt_us)=0) as n_logdebt_us,
      sum(missing(logdebt_non_us)=0) as n_logdebt_non_us,
      sum(missing(z_logequity_world)=0) as n_z_logequity_world,
      sum(missing(z_logequity_us)=0) as n_z_logequity_us,
      sum(missing(z_logequity_non_us)=0) as n_z_logequity_non_us,
      sum(missing(z_logdebt_world)=0) as n_z_logdebt_world,
      sum(missing(z_logdebt_us)=0) as n_z_logdebt_us,
      sum(missing(z_logdebt_non_us)=0) as n_z_logdebt_non_us
      from regdata
      group by country
      order by country;
quit;
proc sql; create table cpisfull as select distinct
      *
      from regdata
      where iso3 in (select distinct iso3 from descdata1 where
n_logequity_world=10 and n_logequity_us=10 and n_logdebt_us=10);
quit;
proc sql; create table zucmanfull as select distinct
      *
      from regdata
      where iso3 in (select distinct iso3 from descdata1 where
n_z_logequity_world=10 and n_z_logequity_us=10 and n_z_logdebt_us=10
and n_z_logequity_non_us=10);
quit;

/* comment these out to avoid overwrite
proc export data=zucmanfull outfile="C:\_Research\DLM1\zucmanfull.dta"
replace;
run ;
proc export data=cpisfull outfile="C:\_Research\DLM1\cpisfull.dta"
replace;
run ;
*/

* Data for Appendix A -- reconciliation to HMT;


* DLM and HMT data for 1984 - 2008;

%let hmtsample = "ANT" "ARG" "AUS" "AUT" "BHS" "BMU" "BRA" "CHE" "CHL"
"CHN" "COL" "CYM" "DEU" "DNK" "EGY" "ESP"
"FIN" "GBR" "GRC" "GTM" "HKG" "IDN" "IND" "ITA" "JAM" "JPN" "LBN"
"LBR" "NLD" "PAN" "PER" "PHL"
"PRT" "SGP" "SWE" "THA" "TUR" "TWN" "VEN" "ZAF";


* confine later sample to HMT countries;
```

```
proc sql; create table dlm_hmt15 as select distinct
      *
      from regdata
      where iso3 in (&hmtsample)
;quit;

/* comment out to avoid overwrite
proc export data=dlm_hmt15 outfile="C:\_Research\DLM1\dlm_hmt15.dta"
replace;
run ;
*/

* confine to HMT countries;
proc sql; create table dlm_hmt08 as select distinct
      *
      from dlm.dlm_84to15
      where iso3 in (&hmtsample)
      and year le 2008 and month=12 and iso3 ne "CYM";
quit;

/* comment out to avoid overwrite
proc export data=dlm_hmt08 outfile="C:\_Research\DLM1\dlm_hmt08.dta"
replace;
run ;
*/
```

```
* This program produces the file descrip.xls which has one tab:
countries that is Table 1
The t-tests at the bottom of Table 1 are run in Stata in the program
DLM Figures and Tables.do

This program uses tagsets in SAS
;

libname dlm 'c:\_Research\DLM1';

* START Table 1;
* run means by country;
proc sql; create table countries as select distinct
      iso3,
      country,
      haven label="Tax haven per Dyreng Lyndsey 2009"
      from dlm.data1
      group by country
      ;
quit;


data datat1; set dlm.data1;
      cpis_equity_world=    exp(logequity_world);
      cpis_debt_world=exp(logdebt_world);
run;

proc sort data=datat1 out=c1; by country; run;
proc means data=c1; by country; var  lte ltc cpis_equity_world
cpis_debt_world  gdp population it_cel_sets_p2 xrate localrate;
      output out=new(drop=_type_ _freq_) mean= / autoname;
run;

proc sql; create table countries2 as select distinct
      a.country,
      a.haven,
      b.*
      from countries a, new b
      where a.country=b.country;
quit;

ods tagsets.excelxp file="C:\_Research\DLM1\JAR_tables\descrip.XLS"
style=minimal;
ods tagsets.excelxp options(sheet_name="countries");
proc print data=countries2;
run;

ods tagsets.excelxp close;
```

STATA CODE

```
/*
Starting datasets:
1. Appendix A: dlm_hmt08, dlm_hmt15, data1
2. Figure 1: data1
3. Figure 2: none (data manually input)
4. Figure 3: ucits_assets_final.xlsx
5. Figure 4: datacollection_cleaned.csv
6. Figures 5A and 5B: swiss_data_to_import (proprietary)
7. Table 1: data1
8. Table 2: data1
9. Table 3: data1
10. Table 4: cpisfull, zucmanfull
11. Table 5: ucits_assets_final.xlsx, data1
12. Table 6: datacollection_cleaned.csv, data1, chinagdp.dta, russiagdp.dta
*/

* all datasets should be in location specified in this cd statement. All output will go to the same location

cd "C:\Users\kmarkle\Dropbox\FATCA\JAR\DLM_JAR_Code_Data"


* START Appendix A
*******************************************************************************
*************

use dlm_hmt08, clear
rename localrate local_taxrate
gen       post = (year >= 2001)
gen       haven_post = post*haven

encode iso3, gen(niso3)

tsset niso3 year, yearly

xi: reg logequity haven_post local_taxrate logpop loggdp  i.country i.year
local rsq=`e(r2_a)'
xi: newey2 logequity haven_post local_taxrate logpop loggdp  i.country i.year, lag(2) force
   outreg2 using AppA, excel   bdec(3) tdec(3) drop(_I*) addtext (Adj-R2, `rsq' )  replace

use dlm_hmt15, clear
rename localrate local_taxrate
drop if iso2=="KY"
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven

tsset niso3 year, yearly
```

```
xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp  i.country i.year
local rsq=`e(r2_a)'
xi: newey2 tic_logequity_us haven_post local_taxrate logpop loggdp  i.country i.year, lag(2) force
   outreg2 using AppA, excel   bdec(3) tdec(3) drop(_I*) addtext (Adj-R2, `rsq' )  append

use data1, clear
rename localrate local_taxrate
drop if iso2=="KY"
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven
tsset niso3 year, yearly
xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp  i.country i.year
local rsq=`e(r2_a)'
xi: newey2 tic_logequity_us haven_post local_taxrate logpop loggdp  i.country i.year, lag(2) force
   outreg2 using AppA, excel   bdec(3) tdec(3) drop(_I*) addtext (Adj-R2, `rsq' )  append

* END Appendix A
*******************************************************************************
*************



* START Figure 1
*******************************************************************************
*************
use data1, clear
rename localrate local_taxrate
encode iso3, gen(niso3)


forvalues i = 2002/2015 {
   gen haven_`i' = (haven==1) & (year==`i')
}

xi: reg tic_logequity_us  haven_2007 haven_2008 haven_2009 haven_2010 haven_2011 haven_2012
haven_2013 haven_2014 haven_2015 local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)

coefplot, keep(haven_*) vertical    scheme(s1mono) ytitle({it:Log(Equity FPI)}, height(5))  xlabel(1 "2007"
2 "2008" 3 "2009" 4 "2010" 5 "2011" 6 "2012" 7 "2013" 8 "2014" 9 "2015") ylabel(-1.0(0.25)1.0)
yscale(range(-1.0(0.25)1.0)) yline(0, lpattern(dot)) xline(5.5, lpattern(dot))

graph export fig1.pdf, replace

* END Figure 1
*******************************************************************************
*************
```

```
* START Figure 2
*****************************************************************************************
*************

clear
input year expatriationannual
2006 654
2007 470
2008 231
2009 742
2010 1534
2011 1781
2012 932
2013 2999
2014 3415
2015 4279
2016 5409
2017 5132
end

drop if year==2016
drop if year==2017

lgraph expatriationannual year, xline(5.7, lpattern(dash)) scheme(s1mono) xtitle({it:Year}, height(5))
ytitle({it:Number of U.S. Expatriations}, height(5)) xlabel(#10) ylabel(#4, format(%9.0fc))

graph export fig2.pdf, replace
* END Figure 2
*****************************************************************************************
*************

* START Figure 3
*****************************************************************************************
*************

pr drop _all
clear
clear matrix
clear mata

set matsize 1200
set mem 800m
set more off

capture log close
```

```
*****************************************************************************************
*******************************************
*1.  Import dataset for UCITS assets by country
*****************************************************************************************
*******************************************
clear

import excel ucits_assets_final.xlsx, firstrow
reshape long Y,i(Country)
rename _j year
rename Y ucits_assets
rename Country country
drop if missing(ucits_assets)
tab year
drop if year == 2005
drop if year == 2016
sort country year
by country:  gen id = _n
by country:  gen nt = _N
save ucits_assets2, replace

*****************************************************************************************
*******************************************
*2.  Clean dataset and generate graph
*****************************************************************************************
*******************************************

use ucits_assets2, clear
drop if nt != 10
tab country
*Drop countries not in the TIC data used in Tables 2-3 of paper
drop if country == "Czech Republic"
drop if country == "Liechtenstein"
drop if country == "Slovakia"
*Create haven indicator for countries in sample that are also identified as havens in Tables 2-4 of paper
*Indicator in opposite direction for purposes of graphing lines
gen haven_ind = 1
replace haven_ind = 0 if country == "Luxembourg" | country == "Ireland" | country == "Switzerland"
*Scale ucits assets for readability
gen ucits_assets2 = ucits_assets/1000
drop ucits_assets
rename ucits_assets2 ucits_assets
collapse (sum) ucits_assets, by(year haven_ind)
lgraph ucits_assets year haven_ind, scheme(s1mono) xtitle({it:Year}, height(5)) ytitle({it:UCITS Assets by
Country (Euros in billions)}, height(5)) legend(on order(1 "Ireland, Lux., & Switzerland" 2 "Other
Countries")) xlabel(#10) ylabel(#4, format(%9.0fc))

graph export fig3.pdf, replace
```

```
* END Figure 3
*****************************************************************************************
*************

* START Figure 4
*****************************************************************************************
*************

insheet using datacollection_cleaned.csv, clear

destring nom_price, replace force
destring real_price, replace force
destring rent, replace force
destring income, replace force
gen yearstr1 =substr(date, 1,4)
gen yearstr2 =substr(date, 4,7)
g yearstr=yearstr1
replace yearstr=yearstr2 if substr(yearstr1,1,1)=="Q"
drop if yearstr==""
destring yearstr, gen(year) force
replace year=2014 if year==1905
drop yearstr1 yearstr2 yearstr
encode country, gen(countrynum)

* define haven countries
gen haven=0
replace haven=1 if country=="Hong Kong"
replace haven=1 if country=="Ireland"
replace haven=1 if country=="Luxembourg"
replace haven=1 if country=="Malaysia"
replace haven=1 if country=="Singapore"
replace haven=1 if country=="Switzerland"

* define countries with foreign buyer restrictions
gen restricted=0
replace restricted=1 if country=="China"
replace restricted=1 if country=="Denmark"
replace restricted=1 if country=="Greece"
replace restricted=1 if country=="Hong Kong"
replace restricted=1 if country=="India"
replace restricted=1 if country=="Indonesia"
replace restricted=1 if country=="Malaysia"
replace restricted=1 if country=="Poland"
replace restricted=1 if country=="Singapore"
replace restricted=1 if country=="Switzerland"

* highlight China and Hong Kong to exclude
gen exclude=0
```

```
replace exclude=1 if country=="China"
replace exclude=1 if country=="Hong Kong"

* highlight countries without a full panel to exclude
gen notfull=0
replace notfull=1 if country=="Brazil"
replace notfull=1 if country=="India"
replace notfull=1 if country=="Luxembourg"
replace notfull=1 if country=="Morocco"
replace notfull=1 if country=="Poland"
replace notfull=1 if country=="Romania"
replace notfull=1 if country=="Turkey"
replace notfull=1 if country=="Hungary"

sum nom_price real_price haven restricted notfull year rent income countrynum

bysort countrynum year: egen nombycountry=mean(nom_price)
bysort countrynum year: egen realbycountry=mean(real_price)
bysort countrynum year: egen rentbycountry=mean(rent)
bysort countrynum year: egen incomebycountry=mean(income)
bysort year: egen nombyyear=mean(nom_price)
bysort year: egen realbyyear=mean(real_price)
bysort year: egen rentbyyear=mean(rent)
bysort year: egen incomebyyear=mean(income)

save realestate, replace

* restricted vs. unrestricted, drop excluded countries and countries without full panel for nominal and
real prices
use realestate, clear
drop if exclude==1
drop if year<2006
drop if year>2015
drop if notfull==1
bysort restricted year: egen nombyrestricted=mean(nom_price)
bysort restricted year: egen nombyrestrictedstdev=sd(nom_price)
bysort restricted year: egen realbyrestricted=mean(real_price)
bysort restricted year: egen realbyrestrictedstdev=sd(real_price)
bysort restricted year: egen rentbyrestricted=mean(rent)
bysort restricted year: egen incomebyrestricted=mean(income)
by restricted year: keep if _n==1
gen restrictednom = nombyrestricted if restricted==1
gen restrictednomstdev = nombyrestrictedstdev if restricted==1
gen unrestrictednom = nombyrestricted if restricted==0
gen unrestrictednomstdev = nombyrestrictedstdev if restricted==0
gen restrictedreal = realbyrestricted if restricted==1
gen restrictedrealstdev = realbyrestrictedstdev if restricted==1
gen unrestrictedreal = realbyrestricted if restricted==0
```

```
gen unrestrictedrealstdev = realbyrestrictedstdev if restricted==0
gen restrictedrent = rentbyrestricted if restricted==1
gen unrestrictedrent = rentbyrestricted if restricted==0
gen restrictedincome = incomebyrestricted if restricted==1
gen unrestrictedincome = incomebyrestricted if restricted==0

* graph
lgraph realbyrestricted year restricted, scheme(s1mono) xtitle({it:Year}, height(7)) ytitle({it:Global House
Price Index (Real)}, height(5)) legend(on order(1 "No Foreign Buyer Restrictions" 2 "Foreign Buyer
Restrictions")) xlabel(#10) ylabel(#4, format(%9.0fc))
graph export fig4.pdf, replace

* END Figure 4
*******************************************************************************************
*************

* START Figures 5A and 5B
*******************************************************************************************
*************

pr drop _all
clear
clear matrix
clear mata
set matsize 1200
set mem 800m
set more off
capture log close

*******************************************************************************************
*******************************************
*1.  Import dataset
*******************************************************************************************
*******************************************
clear
import excel swiss_data_to_import, firstrow
gen year = year(Period)
gen month = month(Period)
drop if year == 2016
drop Import_kg_pctchg Import_CHF_pctchg Export_kg_pctchg Export_CHF_pctchg
drop if missing(Import_kg)
drop if missing(Import_CHF)
drop if missing(Export_kg)
drop if missing(Export_CHF)
drop if Canton_name == "Unknown"

gen geneva_ind = 0
replace geneva_ind = 1 if Canton_number == 25
```

```
gen Net_import_kg = Import_kg - Export_kg
gen Net_import_CHF = Import_CHF - Export_CHF
tab Canton_number

*Drop cantons that also report large freeports - Ticino and Vaud
drop if Canton_number == 21
drop if Canton_number == 22

*Drop cantons with incomplete coverage
drop if Canton_number == 4
drop if Canton_number == 6 | Canton_number == 7 | Canton_number == 8
drop if Canton_number == 14 | Canton_number == 15 | Canton_number ==16
drop if Canton_number == 26

*Retain cantons with similar high levels of activityL *1 = Zurich; *12 = Basel; *17 = St. Gallen; *19 =
Aargau; *25 = Geneva
sort Canton_number
tabstat Import_kg, by(Canton_number)
keep if Canton_number == 1 | Canton_number == 25 | Canton_number == 12 | Canton_number == 17 |
Canton_number == 19
tabstat Import_kg, by(Canton_number)
tab year
save imported_data, replace

*******************************************************************************************
*******************************************
*2.  Create stock of gross and net imports
*******************************************************************************************
*******************************************

use imported_data, clear
drop if geneva_ind == 0
collapse (sum) Import_kg Import_CHF Net_import_kg Net_import_CHF, by(year)
gen imp_kg_lag = Import_kg[_n-1] if year==year[_n-1]+1
replace imp_kg_lag = 0 if missing(imp_kg_lag)
gen imp_kg_lag2 = Import_kg[_n-2] if year==year[_n-2]+2
replace imp_kg_lag2 = 0 if missing(imp_kg_lag2)
gen imp_kg_lag3 = Import_kg[_n-3] if year==year[_n-3]+3
replace imp_kg_lag3 = 0 if missing(imp_kg_lag3)
gen imp_kg_lag4 = Import_kg[_n-4] if year==year[_n-4]+4
replace imp_kg_lag4 = 0 if missing(imp_kg_lag4)
gen imp_kg_lag5 = Import_kg[_n-5] if year==year[_n-5]+5
replace imp_kg_lag5 = 0 if missing(imp_kg_lag5)
gen imp_kg_lag6 = Import_kg[_n-6] if year==year[_n-6]+6
replace imp_kg_lag6 = 0 if missing(imp_kg_lag6)
gen imp_kg_lag7 = Import_kg[_n-7] if year==year[_n-7]+7
replace imp_kg_lag7 = 0 if missing(imp_kg_lag7)
gen imp_kg_lag8 = Import_kg[_n-8] if year==year[_n-8]+8
```

```
replace imp_kg_lag8 = 0 if missing(imp_kg_lag8)
gen imp_kg_lag9 = Import_kg[_n-9] if year==year[_n-9]+9
replace imp_kg_lag9 = 0 if missing(imp_kg_lag9)
gen imp_kg_lag10 = Import_kg[_n-10] if year==year[_n-10]+10
replace imp_kg_lag10 = 0 if missing(imp_kg_lag10)
gen imp_kg_lag11 = Import_kg[_n-11] if year==year[_n-11]+11
replace imp_kg_lag11 = 0 if missing(imp_kg_lag11)
gen stock_imports_kg = Import_kg + imp_kg_lag + imp_kg_lag2 + imp_kg_lag3 + imp_kg_lag4 +
imp_kg_lag5 + imp_kg_lag6 + imp_kg_lag7 + imp_kg_lag8 + imp_kg_lag9 + imp_kg_lag10 +
imp_kg_lag11 /*+ imp_kg_lag12 + imp_kg_lag13 + imp_kg_lag14 + imp_kg_lag15*/

sort year
gen netimp_kg_lag = Net_import_kg[_n-1] if year==year[_n-1]+1
replace netimp_kg_lag = 0 if missing(netimp_kg_lag)
gen netimp_kg_lag2 = Net_import_kg[_n-2] if year==year[_n-2]+2
replace netimp_kg_lag2 = 0 if missing(netimp_kg_lag2)
gen netimp_kg_lag3 = Net_import_kg[_n-3] if year==year[_n-3]+3
replace netimp_kg_lag3 = 0 if missing(netimp_kg_lag3)
gen netimp_kg_lag4 = Net_import_kg[_n-4] if year==year[_n-4]+4
replace netimp_kg_lag4 = 0 if missing(netimp_kg_lag4)
gen netimp_kg_lag5 = Net_import_kg[_n-5] if year==year[_n-5]+5
replace netimp_kg_lag5 = 0 if missing(netimp_kg_lag5)
gen netimp_kg_lag6 = Net_import_kg[_n-6] if year==year[_n-6]+6
replace netimp_kg_lag6 = 0 if missing(netimp_kg_lag6)
gen netimp_kg_lag7 = Net_import_kg[_n-7] if year==year[_n-7]+7
replace netimp_kg_lag7 = 0 if missing(netimp_kg_lag7)
gen netimp_kg_lag8 = Net_import_kg[_n-8] if year==year[_n-8]+8
replace netimp_kg_lag8 = 0 if missing(netimp_kg_lag8)
gen netimp_kg_lag9 = Net_import_kg[_n-9] if year==year[_n-9]+9
replace netimp_kg_lag9 = 0 if missing(netimp_kg_lag9)
gen netimp_kg_lag10 = Net_import_kg[_n-10] if year==year[_n-10]+10
replace netimp_kg_lag10 = 0 if missing(netimp_kg_lag10)
gen netimp_kg_lag11 = Net_import_kg[_n-11] if year==year[_n-11]+11
replace netimp_kg_lag11 = 0 if missing(netimp_kg_lag11)
gen stock_netimports_kg = Net_import_kg + netimp_kg_lag + netimp_kg_lag2 + netimp_kg_lag3 +
netimp_kg_lag4 + netimp_kg_lag5 + netimp_kg_lag6 + netimp_kg_lag7 + netimp_kg_lag8 +
netimp_kg_lag9 + netimp_kg_lag10 + netimp_kg_lag11  /*+ netimp_kg_lag12 + netimp_kg_lag13 +
netimp_kg_lag14 + netimp_kg_lag15*/
gen geneva_ind = 1
save stock_imports_geneva, replace


use imported_data, clear
drop if geneva_ind == 1
collapse (sum) Import_kg Import_CHF Net_import_kg Net_import_CHF, by(year Canton_number)
sort Canton_number year
collapse (mean) Import_kg Import_CHF Net_import_kg Net_import_CHF, by(year)
```

```
gen imp_kg_lag = Import_kg[_n-1] if year==year[_n-1]+1
replace imp_kg_lag = 0 if missing(imp_kg_lag)
gen imp_kg_lag2 = Import_kg[_n-2] if year==year[_n-2]+2
replace imp_kg_lag2 = 0 if missing(imp_kg_lag2)
gen imp_kg_lag3 = Import_kg[_n-3] if year==year[_n-3]+3
replace imp_kg_lag3 = 0 if missing(imp_kg_lag3)
gen imp_kg_lag4 = Import_kg[_n-4] if year==year[_n-4]+4
replace imp_kg_lag4 = 0 if missing(imp_kg_lag4)
gen imp_kg_lag5 = Import_kg[_n-5] if year==year[_n-5]+5
replace imp_kg_lag5 = 0 if missing(imp_kg_lag5)
gen imp_kg_lag6 = Import_kg[_n-6] if year==year[_n-6]+6
replace imp_kg_lag6 = 0 if missing(imp_kg_lag6)
gen imp_kg_lag7 = Import_kg[_n-7] if year==year[_n-7]+7
replace imp_kg_lag7 = 0 if missing(imp_kg_lag7)
gen imp_kg_lag8 = Import_kg[_n-8] if year==year[_n-8]+8
replace imp_kg_lag8 = 0 if missing(imp_kg_lag8)
gen imp_kg_lag9 = Import_kg[_n-9] if year==year[_n-9]+9
replace imp_kg_lag9 = 0 if missing(imp_kg_lag9)
gen imp_kg_lag10 = Import_kg[_n-10] if year==year[_n-10]+10
replace imp_kg_lag10 = 0 if missing(imp_kg_lag10)
gen imp_kg_lag11 = Import_kg[_n-11] if year==year[_n-11]+11
replace imp_kg_lag11 = 0 if missing(imp_kg_lag11)
gen stock_imports_kg = Import_kg + imp_kg_lag + imp_kg_lag2 + imp_kg_lag3 + imp_kg_lag4 +
imp_kg_lag5 + imp_kg_lag6 + imp_kg_lag7 + imp_kg_lag8 + imp_kg_lag9 + imp_kg_lag10 +
imp_kg_lag11 /*+ imp_kg_lag12 + imp_kg_lag13 + imp_kg_lag14 + imp_kg_lag15*/


sort year
gen netimp_kg_lag = Net_import_kg[_n-1] if year==year[_n-1]+1
replace netimp_kg_lag = 0 if missing(netimp_kg_lag)
gen netimp_kg_lag2 = Net_import_kg[_n-2] if year==year[_n-2]+2
replace netimp_kg_lag2 = 0 if missing(netimp_kg_lag2)
gen netimp_kg_lag3 = Net_import_kg[_n-3] if year==year[_n-3]+3
replace netimp_kg_lag3 = 0 if missing(netimp_kg_lag3)
gen netimp_kg_lag4 = Net_import_kg[_n-4] if year==year[_n-4]+4
replace netimp_kg_lag4 = 0 if missing(netimp_kg_lag4)
gen netimp_kg_lag5 = Net_import_kg[_n-5] if year==year[_n-5]+5
replace netimp_kg_lag5 = 0 if missing(netimp_kg_lag5)
gen netimp_kg_lag6 = Net_import_kg[_n-6] if year==year[_n-6]+6
replace netimp_kg_lag6 = 0 if missing(netimp_kg_lag6)
gen netimp_kg_lag7 = Net_import_kg[_n-7] if year==year[_n-7]+7
replace netimp_kg_lag7 = 0 if missing(netimp_kg_lag7)
gen netimp_kg_lag8 = Net_import_kg[_n-8] if year==year[_n-8]+8
replace netimp_kg_lag8 = 0 if missing(netimp_kg_lag8)
gen netimp_kg_lag9 = Net_import_kg[_n-9] if year==year[_n-9]+9
replace netimp_kg_lag9 = 0 if missing(netimp_kg_lag9)
gen netimp_kg_lag10 = Net_import_kg[_n-10] if year==year[_n-10]+10
replace netimp_kg_lag10 = 0 if missing(netimp_kg_lag10)
```

```
gen netimp_kg_lag11 = Net_import_kg[_n-11] if year==year[_n-11]+11
replace netimp_kg_lag11 = 0 if missing(netimp_kg_lag11)
gen stock_netimports_kg = Net_import_kg + netimp_kg_lag + netimp_kg_lag2 + netimp_kg_lag3 +
netimp_kg_lag4 + netimp_kg_lag5 + netimp_kg_lag6 + netimp_kg_lag7 + netimp_kg_lag8 +
netimp_kg_lag9 + netimp_kg_lag10 + netimp_kg_lag11 /*+ netimp_kg_lag12 + netimp_kg_lag13 +
netimp_kg_lag14 + netimp_kg_lag15*/
gen geneva_ind = 0
save stock_imports_other, replace

*******************************************************************************
*******************************************
*3. Graph gross and net imports for Geneva canton relative to other cantons over sample period
*******************************************************************************
*******************************************

use stock_imports_other, clear
append using stock_imports_geneva
gen stock_imports_kg2 = stock_imports_kg/1000
gen stock_netimports_kg2 = stock_netimports_kg/1000
gen post=0
replace post=1 if year >=2012
gen geneva_post = geneva_ind*post
drop if year == 2005 /*to align with sample in the paper*/
save stock_imports_combined_for_graph, replace
use stock_imports_combined_for_graph, clear

lgraph stock_imports_kg2 year geneva_ind, scheme (s1mono) xtitle({it:Year}, height(5)) ytitle({it:Stock of
Imported Art (kg in thousands)}, height(5)) legend(on order(1 "Other Cantons" 2 "Geneva Canton"))
xlabel(#10) ylabel(#5, format(%9.0fc))
graph export fig5a.pdf, replace
reg stock_imports_kg geneva_ind post geneva_post

lgraph stock_netimports_kg2 year geneva_ind, scheme (s1mono) xtitle({it:Year}, height(5))
ytitle({it:Stock of Imported Art (kg in thousands)}, height(5)) legend(on order(1 "Other Cantons" 2
"Geneva Canton")) xlabel(#10) ylabel(#4, format(%9.0fc))
graph export fig5b.pdf, replace
reg stock_netimports_kg geneva_ind post geneva_post



* END Figures 5A and 5B
*******************************************************************************
*************


* START Table 1 - ttests of means
*******************************************************************************
*************
```

```
* main descriptives produced in SAS

use data1, clear

gen       cpis_equity_world=       exp(logequity_world)
gen       cpis_debt_world=exp(logdebt_world)

putexcel set "T-test.xlsx", sheet("Haven") replace
putexcel A3=("Variable")  B3=("non-HAVEN (N)") C3=("non-HAVEN (Mean)") D3=("non-HAVEN (SD)")
E3=("HAVEN (N)")F3=("HAVEN (Mean)") G3=("HAVEN (SD)") H3=("p value")
local row=4
foreach var of varlist  lte ltc cpis_equity_world cpis_debt_world   gdp population it_cel_sets_p2 xrate
localrate   {
   qui ttest `var',by(haven)
putexcel A`row' = ("`var'")
putexcel B`row' = (r(N_1))
putexcel C`row' = (r(mu_1))
putexcel D`row' = (r(sd_1))
putexcel E`row' = (r(N_2))
putexcel F`row' = (r(mu_2))
putexcel G`row' = (r(sd_2))
putexcel H`row' = (r(p))
local ++row
}

* END Table 1 - ttests if means
*******************************************************************************************
*************

* START Table 2
*******************************************************************************************
*************

use data1, clear

rename localrate local_taxrate
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven
gen       havenwithhighgoveff=haven*goveff_high_full
gen       havenwithlowgoveff=(haven==1 & goveff_high_full==0)
gen       nontiea3 = (tiea3==0 & haven==1)
gen havenwithhighgoveff_post=post*havenwithhighgoveff
gen havenwithlowgoveff_post=post*havenwithlowgoveff
gen tieahaven_post = post*tiea3
gen nontieahaven_post=post*nontiea3
```

```
xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
   outreg2 using t2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster, country,
country fe, yes, year fe, yes )  replace

xi: reg tic_logequity_us tieahaven_post  nontieahaven_post local_taxrate logpop loggdp
log_mobile_subs_p100 log_xrate i.country i.year, robust cl(iso3)
test tieahaven_post= nontieahaven_post
local ftest=round(`r(p)',0.001)
   outreg2 using t2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* tic*) addtext (cluster, country,
country fe, yes, year fe, yes, f-test p-value, `ftest' )  append

xi: reg tic_logequity_us havenwithhighgoveff_post  havenwithlowgoveff_post local_taxrate logpop
loggdp log_mobile_subs_p100 log_xrate i.country i.year, robust cl(iso3)
test havenwithhighgoveff_post= havenwithlowgoveff_post
local ftest=round(`r(p)',0.001)
   outreg2 using t2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* tic*) addtext (cluster, country,
country fe, yes, year fe, yes, f-test p-value, `ftest' )  append


xi: reg tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
   outreg2 using t2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster, country,
country fe, yes, year fe, yes )  append

xi: reg tic_logdebt_us tieahaven_post  nontieahaven_post local_taxrate logpop loggdp
log_mobile_subs_p100 log_xrate i.country i.year, robust cl(iso3)
test tieahaven_post= nontieahaven_post
local ftest=round(`r(p)',0.001)
   outreg2 using t2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* tic*) addtext (cluster, country,
country fe, yes, year fe, yes, f-test p-value, `ftest' )  append

xi: reg tic_logdebt_us havenwithhighgoveff_post  havenwithlowgoveff_post local_taxrate logpop loggdp
log_mobile_subs_p100 log_xrate i.country i.year, robust cl(iso3)
test havenwithhighgoveff_post= havenwithlowgoveff_post
local ftest=round(`r(p)',0.001)
   outreg2 using t2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* tic*) addtext (cluster, country,
country fe, yes, year fe, yes, f-test p-value, `ftest' )  append

* END Table 2
*******************************************************************************
*************


* START Table 3
*******************************************************************************
*************
```

```
use data1, clear

rename localrate local_taxrate
encode iso3, gen(niso3)
drop if year>2009
gen post=(year>=2008)
gen haven_postyear=post*haven

xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t3, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (country fe, yes,
year fe, yes )  replace

xi: reg tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t3, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (country fe, yes,
year fe, yes )  append


use data1, clear
rename localrate local_taxrate
encode iso3, gen(niso3)
drop if year>2011
drop if year<2008
gen post=(year>=2010)
gen haven_postyear=post*haven

tsset niso3 year, yearly

xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t3, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (country fe, yes,
year fe, yes )  append

xi: reg tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t3, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (country fe, yes,
year fe, yes )  append

* END Table 3
*****************************************************************************************
*************


* START Table 4
*****************************************************************************************
*************
```

```
use cpisfull, clear

rename localrate local_taxrate
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven


xi: reg logequity_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* loge* logd*) addtext (country
fe, yes, year fe, yes )  replace

xi: reg logdebt_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* loge* logd*) addtext (country
fe, yes, year fe, yes )  append

xi: reg logequity_non_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* loge* logd*) addtext (country
fe, yes, year fe, yes )  append

xi: reg logdebt_non_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* loge* logd*) addtext (country
fe, yes, year fe, yes )  append


use zucmanfull, clear

rename localrate local_taxrate
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven


xi: reg z_logequity_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* z_log*) addtext (country fe,
yes, year fe, yes )  append

xi: reg z_logdebt_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* z_log*) addtext (country fe,
yes, year fe, yes )  append
```

```
xi: reg z_logequity_non_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* z_log*) addtext (country fe,
yes, year fe, yes )  append

xi: reg z_logdebt_non_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using t4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* z_log*) addtext (country fe,
yes, year fe, yes )  append

* END Table 4
*******************************************************************************
*************


* START Table 5
*******************************************************************************
*************


pr drop _all
clear
clear matrix
clear mata

set matsize 1200
set mem 800m
set more off

capture log close

*******************************************************************************
*******************************************
*1.  Import dataset for UCITS assets by country
*******************************************************************************
*******************************************
clear

import excel ucits_assets_final.xlsx, firstrow
reshape long Y,i(Country)
rename _j year
rename Y ucits_assets
rename Country country
drop if missing(ucits_assets)
tab year
drop if year == 2005
drop if year == 2016
sort country year
```

```
by country:  gen id = _n
by country:  gen nt = _N
save ucits_assets2, replace

*******************************************************************************************
*******************************************
*2.  Regression specification
*******************************************************************************************
*******************************************

*Dataset for Regression
use ucits_assets2, clear
drop if nt != 10
drop if country == "Czech Republic"
drop if country == "Liechtenstein"
drop if country == "Slovakia"
egen country2 = group(country)
gen haven_ind = 0
replace haven_ind = 1 if country == "Luxembourg" | country == "Ireland" | country == "Switzerland"
gen ucits_assets2 = ucits_assets/1000
drop ucits_assets
rename ucits_assets2 ucits_assets
gen log_ucits = log(ucits_assets)
gen post = 0
replace post = 1 if year > 2011
gen post_haven = post*haven_ind
drop id nt
*Merge with country-year control variables used in Tables 2-4
replace country = upper(country)
merge m:1 country year using data1
keep if _merge == 3
drop _merge
keep country country2 year haven_ind ucits_assets log_ucits post post_haven localrate logpop loggdp
log_mobile_subs_p100 log_xrate

tsset country2 year, yearly
save ucits_assets3, replace

*Column 1 results: Obtain coefficients and correct r-squareds
xi: reg ucits_assets post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate i.country
i.year
local rsq= round(`e(r2_a)',0.001)
*Column 1 results: Obtain correct t-statistics with clustering
xi: reg ucits_assets post_haven localrate logpop loggdp log_mobile_subs_p100 log_xrate i.country
i.year, cluster(country)
outreg2 using t5, excel replace stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext ( country fe, yes, year
fe, yes, sample, full )
```

```
*Column 2 results
xi: reg log_ucits post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate i.country i.year
local rsq= round(`e(r2_a)',0.001)
xi: reg log_ucits post_haven localrate logpop loggdp log_mobile_subs_p100 log_xrate i.country i.year,
cluster(country)
outreg2 using t5, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )

*Column 3 results - dropping smallest countries
use ucits_assets3, clear
tabstat ucits_assets, by(country)
drop if country == "GREECE"
drop if country == "PORTUGAL"
drop if country == "HUNGARY"
drop if country == "TURKEY"
drop if country == "POLAND"
xi i.country i.year
xi: reg ucits_assets post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate i.country
i.year
local rsq= round(`e(r2_a)',0.001)
xi: reg ucits_assets post_haven localrate logpop loggdp log_mobile_subs_p100 log_xrate i.country
i.year, cluster(country)
outreg2 using t5, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )

*Column 4 results - dropping smallest countries
xi: reg log_ucits post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate i.country i.year
local rsq= round(`e(r2_a)',0.001)
xi: reg log_ucits post_haven localrate logpop loggdp log_mobile_subs_p100 log_xrate i.country i.year,
cluster(country)
outreg2 using t5, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )

* END Table 5
*****************************************************************************************
*************


* START Table 6
*****************************************************************************************
*************

insheet using datacollection_cleaned.csv, clear

destring nom_price, replace force
destring real_price, replace force
destring rent, replace force
destring income, replace force
```

```
gen yearstr1 =substr(date, 1,4)
gen yearstr2 =substr(date, 4,7)
g yearstr=yearstr1
replace yearstr=yearstr2 if substr(yearstr1,1,1)=="Q"
drop if yearstr==""
destring yearstr, gen(year) force
replace year=2014 if year==1905
drop yearstr1 yearstr2 yearstr source
encode country, gen(countrynum)
* Morocco has wonky iso3 (appear to be month abbreviations)
replace iso3="MAR" if iso2=="MA"

sum nom_price real_price year rent income countrynum
* go from quarterly to annual (country-year observations)
bysort countrynum year: egen nombycountry=mean(nom_price)
bysort countrynum year: egen realbycountry=mean(real_price)
bysort countrynum year: egen rentbycountry=mean(rent)
bysort countrynum year: egen incomebycountry=mean(income)
by countrynum year: keep if _n==1

keep country year nombycountry realbycountry rentbycountry incomebycountry iso2 iso3
save recleaned, replace

* start with tostata
use data1, clear

* merge in cleaned real estate data
* tostata has country, iso2, iso3 and year
* recleaned has country year iso2 iso3
merge 1:1 iso3 year using recleaned

* define countries with foreign buyer restrictions
gen restricted=0
replace restricted=1 if country_from=="China"
replace restricted=1 if country_from=="Denmark"
replace restricted=1 if country_from=="Greece"
replace restricted=1 if country_from=="Hong Kong"
replace restricted=1 if country_from=="India"
replace restricted=1 if country_from=="Indonesia"
replace restricted=1 if country_from=="Malaysia"
replace restricted=1 if country_from=="Poland"
replace restricted=1 if country_from=="Singapore"
replace restricted=1 if country_from=="Switzerland"
* need to clean up restricted and unrestricted
gen unrestricted = 1-restricted

* highlight China and Hong Kong to exclude
gen exclude=0
```

```
replace exclude=1 if country_from=="China"
replace exclude=1 if country_from=="Hong Kong"

* highlight countries without a full panel to exclude
* RE data is also missing for Philippines and Thailand so these will drop out of regressions
* control data is missing for US so this will drop out of regressions
gen notfull=0
replace notfull=1 if country_from=="Brazil"
replace notfull=1 if country_from=="India"
replace notfull=1 if country_from=="Luxembourg"
replace notfull=1 if country_from=="Morocco"
replace notfull=1 if country_from=="Poland"
replace notfull=1 if country_from=="Romania"
replace notfull=1 if country_from=="Turkey"
replace notfull=1 if country_from=="Hungary"

* create regression variables
encode iso3, gen(niso3)
gen post=(year>=2012)
gen unrestricted_post=post*unrestricted
sum nombycountry
gen lognominal = log(nombycountry)
gen logreal = log(realbycountry)
gen logrent = log(rentbycountry)
gen logincome = log(incomebycountry)
* mobile_subs missing for Romania (already notfull==1)

save reregs, replace

use reregs, clear
drop if exclude==1
drop if year<2006
drop if year>2015
drop if notfull==1
drop if iso3=="USA"
drop if iso3=="THA"
drop if iso3=="PHL"
drop _merge
merge m:1 year using chinagdp
drop _merge
merge m:1 year using russiagdp
drop if _merge==1
drop _merge
bysort country: egen mean_realbycountry = mean(realbycountry)
* drop Serbia and Russia because so much smaller than the rest.  Spain would be next
* drop Peru and Malaysia because changes are huge
* drop Ireland because housing crisis in middle of sample, hit harder than US?
* UK is largest in the set, could drop them?
```

```
* drop if country=="Serbia" | country=="Russia" | country=="Peru" | country=="Malaysia" |
country=="Ireland"
* winsor2 realbycountry, trim cuts (5 95)
* gen logreal_t = log(realbycountry_t)

* full sample, country and year FE, clustering by country
eststo clear
* Column 1
rename localrate local_taxrate

xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.country i.year, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using t6, excel replace stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )

* Column 2
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate  i.country
i.year, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using t6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )

* Column 3
* drop countries above or below 10%/90% of mean realbycountry, country and year FE, clustering by
country
drop if country_from=="Serbia" | country_from=="Russian Federation" | country_from=="Spain" |
country_from=="United Kingdom" | country_from=="Peru" | country_from=="Malaysia"
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.country i.year, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using t6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )

* Column 4
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate  i.country
i.year, vce(cluster country)
outreg2 using t6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )


* Column 5
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
logrent logincome i.country i.year, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using t6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )
```

```
* Column 6
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate logrent
logincome i.country i.year, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using t6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq', country
fe, yes, year fe, yes, sample, full )

* END Table 6
*****************************************************************************************
*************
```

STATA CODE FOR ONLINE APPENDIX

```
/*
Starting datasets:

Table 1: data1, cpisfull, ucits_assets_final.xlsx, datacollection_cleaned.csv, chinagdp.dta, russiagdp.dta
Table 2: data1
Table 3: ucits_assets_final.xlsx, data1

*/

* all datasets should be in location specified in this cd statement. All output will go to the same location

cd "C:\Users\kmarkle\Dropbox\FATCA\JAR\DLM_JAR_Code_Data"


* START Table 1 Standard Errors
*******************************************************************************************
*************

* START OA Table 1 Col. 1-2
*******************************************************************************************
*************

use data1, clear

rename localrate local_taxrate
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven
gen      havenwithhighgoveff=haven*goveff_high_full
gen      havenwithlowgoveff=(haven==1 & goveff_high_full==0)
gen      nontiea3 = (tiea3==0 & haven==1)
gen havenwithhighgoveff_post=post*havenwithhighgoveff
gen havenwithlowgoveff_post=post*havenwithlowgoveff
gen tieahaven_post = post*tiea3
gen nontieahaven_post=post*nontiea3

* main result
xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, country fe, yes, year fe, yes )  replace

xi: reg tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, country fe, yes, year fe, yes )  append
```

```
* newey-west
tsset niso3 year, yearly
xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year
local rsq=`e(r2_a)'
xi: newey2 tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, lag(2) force
   outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, country fe, yes, year fe, yes, Adj-R2, `rsq' )  append

xi: reg tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year
local rsq=`e(r2_a)'
xi: newey2 tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, lag(2) force
   outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, country fe, yes, year fe, yes, Adj-R2, `rsq' )  append

* robust
xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, vce(robust)
   outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
none, country fe, yes, year fe, yes )  append

xi: reg tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, vce(robust)
   outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
none, country fe, yes, year fe, yes )  append

* WB region-year FE

gen regionwb="WB1" if country_from=="Argentina" | country_from=="Bahamas" |
country_from=="Bermuda" | country_from=="Brazil" | country_from=="Chile" |
country_from=="Colombia" | country_from=="Ecuador" | country_from== "Guatemala" |
country_from=="Jamaica" | country_from=="Panama" | country_from=="Peru" |
country_from=="Uruguay"
replace regionwb="WB2" if country_from=="Austria" | country_from=="Belgium" |
country_from=="Denmark" | country_from=="Finland" | country_from=="France" |
country_from=="Germany" | country_from=="Greece" | country_from=="Hungary" |
country_from=="Ireland" | country_from=="Italy" | country_from=="Luxembourg" |
country_from=="Netherlands" | country_from=="Norway" | country_from=="Poland" |
country_from=="Portugal" | country_from=="Romania" | country_from=="Russian Federation" |
country_from=="Serbia" | country_from=="Spain" | country_from=="Sweden" |
country_from=="Switzerland" | country_from=="Turkey" | country_from=="United Kingdom"
replace regionwb="WB3" if country_from=="Australia" | country_from=="China" |
country_from=="Hong Kong" | country_from=="Indonesia" | country_from=="Japan" |
```

```
country_from=="Malaysia" | country_from=="New Zealand" | country_from=="Philippines" |
country_from=="Singapore" | country_from=="Korea, Republic of" | country_from=="Thailand"
replace regionwb="WB4" if country_from=="Egypt" | country_from=="Israel" |
country_from=="Lebanon" | country_from=="Morocco"
replace regionwb="WB5" if country_from=="Canada" | country_from=="Mexico"
replace regionwb="WB6" if country_from=="India"
replace regionwb="WB7" if country_from=="South Africa"
tostring year, gen(yearstr)
gen regionwbxyear = regionwb+"_"+yearstr

xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.regionwbxyear, robust cl(iso3)
   outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, wb-year fe, yes, year fe, no )  append

xi: reg tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.regionwbxyear, robust cl(iso3)
   outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, wb-year fe, yes, year fe, no )  append

* TIC region-year FE

use data1, clear

rename localrate local_taxrate
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven
gen       havenwithhighgoveff=haven*goveff_high_full
gen       havenwithlowgoveff=(haven==1 & goveff_high_full==0)
gen       nontiea3 = (tiea3==0 & haven==1)
gen havenwithhighgoveff_post=post*havenwithhighgoveff
gen havenwithlowgoveff_post=post*havenwithlowgoveff
gen tieahaven_post = post*tiea3
gen nontieahaven_post=post*nontiea3

gen regiontic5="TIC5_1" if country_from=="Argentina" | country_from=="Bahamas" |
country_from=="Bermuda" | country_from=="Brazil" | country_from=="Canada" |
country_from=="Chile" | country_from=="Colombia" | country_from=="Ecuador" | country_from==
"Guatemala" | country_from=="Jamaica" | country_from=="Mexico" | country_from=="Panama" |
country_from=="Peru" | country_from=="Uruguay"
replace regiontic5="TIC5_2" if country_from=="Austria" | country_from=="Belgium" |
country_from=="Denmark" | country_from=="Finland" | country_from=="France" |
country_from=="Germany" | country_from=="Greece" | country_from=="Hungary" |
country_from=="Ireland" | country_from=="Italy" | country_from=="Luxembourg" |
country_from=="Netherlands" | country_from=="Norway" | country_from=="Poland" |
country_from=="Portugal" | country_from=="Romania" | country_from=="Russian Federation" |
```

```
country_from=="Serbia" | country_from=="Spain" | country_from=="Sweden" |
country_from=="Switzerland" | country_from=="Turkey" | country_from=="United Kingdom"
replace regiontic5="TIC5_3" if country_from=="Australia" | country_from=="New Zealand"
replace regiontic5="TIC5_4" if country_from=="China" | country_from=="Hong Kong" |
country_from=="India" | country_from=="Indonesia" | country_from=="Israel" |
country_from=="Japan" | country_from=="Lebanon" | country_from=="Malaysia" |
country_from=="Philippines" | country_from=="Singapore" | country_from=="Korea, Republic of" |
country_from=="Thailand"
replace regiontic5="TIC5_5" if country_from=="Egypt" | country_from=="Morocco" |
country_from=="South Africa"
tostring year, gen(yearstr)
gen regiontic5xyear = regiontic5+"_"+yearstr


xi: reg tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.regiontic5xyear, robust cl(iso3)
  outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, tic-year fe, yes, year fe, no )  append

xi: reg tic_logdebt_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.regiontic5xyear, robust cl(iso3)
  outreg2 using oat1c1_2, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, tic-year fe, yes, year fe, no )  append

* END OA Table 1 Col. 1-2
*******************************************************************************
*************

* START OA Table 1 Col. 3-4
*******************************************************************************
*************

use cpisfull, clear

rename localrate local_taxrate
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven


xi: reg logequity_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* loge* logd*) addtext
(country fe, yes, year fe, yes )  replace

xi: reg logdebt_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, robust cl(iso3)
```

```
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* loge* logd*) addtext
(country fe, yes, year fe, yes )  append


* newey-west
tsset niso3 year, yearly
xi: reg logequity_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year
local rsq=`e(r2_a)'
xi: newey2 logequity_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, lag(2) force
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, country fe, yes, year fe, yes, Adj-R2, `rsq' )  append

xi: reg logdebt_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year
local rsq=`e(r2_a)'
xi: newey2 logdebt_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, lag(2) force
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, country fe, yes, year fe, yes, Adj-R2, `rsq' )  append


* robust
xi: reg logequity_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, vce(robust)
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
none, country fe, yes, year fe, yes )  append

xi: reg logdebt_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year, vce(robust)
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
none, country fe, yes, year fe, yes )  append

* WB region-year FE

gen regionwb="WB1" if country_from=="Argentina" | country_from=="Bahamas" |
country_from=="Bermuda" | country_from=="Brazil" | country_from=="Chile" |
country_from=="Colombia" | country_from=="Ecuador" | country_from== "Guatemala" |
country_from=="Jamaica" | country_from=="Panama" | country_from=="Peru" |
country_from=="Uruguay"
replace regionwb="WB2" if country_from=="Austria" | country_from=="Belgium" |
country_from=="Denmark" | country_from=="Finland" | country_from=="France" |
country_from=="Germany" | country_from=="Greece" | country_from=="Hungary" |
country_from=="Ireland" | country_from=="Italy" | country_from=="Luxembourg" |
country_from=="Netherlands" | country_from=="Norway" | country_from=="Poland" |
country_from=="Portugal" | country_from=="Romania" | country_from=="Russian Federation" |
```

```
country_from=="Serbia" | country_from=="Spain" | country_from=="Sweden" |
country_from=="Switzerland" | country_from=="Turkey" | country_from=="United Kingdom"
replace regionwb="WB3" if country_from=="Australia" | country_from=="China" |
country_from=="Hong Kong" | country_from=="Indonesia" | country_from=="Japan" |
country_from=="Malaysia" | country_from=="New Zealand" | country_from=="Philippines" |
country_from=="Singapore" | country_from=="Korea, Republic of" | country_from=="Thailand"
replace regionwb="WB4" if country_from=="Egypt" | country_from=="Israel" |
country_from=="Lebanon" | country_from=="Morocco"
replace regionwb="WB5" if country_from=="Canada" | country_from=="Mexico"
replace regionwb="WB6" if country_from=="India"
replace regionwb="WB7" if country_from=="South Africa"
tostring year, gen(yearstr)
gen regionwbxyear = regionwb+"_"+yearstr

xi: reg logequity_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.regionwbxyear, robust cl(iso3)
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, wb-year fe, yes, year fe, no )  append

xi: reg logdebt_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.regionwbxyear, robust cl(iso3)
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, wb-year fe, yes, year fe, no )  append

* TIC region-year FE

use cpisfull, clear

rename localrate local_taxrate
encode iso3, gen(niso3)
gen post=(year>=2012)
gen haven_post=post*haven

gen regiontic5="TIC5_1" if country_from=="Argentina" | country_from=="Bahamas" |
country_from=="Bermuda" | country_from=="Brazil" | country_from=="Canada" |
country_from=="Chile" | country_from=="Colombia" | country_from=="Ecuador" | country_from==
"Guatemala" | country_from=="Jamaica" | country_from=="Mexico" | country_from=="Panama" |
country_from=="Peru" | country_from=="Uruguay"
replace regiontic5="TIC5_2" if country_from=="Austria" | country_from=="Belgium" |
country_from=="Denmark" | country_from=="Finland" | country_from=="France" |
country_from=="Germany" | country_from=="Greece" | country_from=="Hungary" |
country_from=="Ireland" | country_from=="Italy" | country_from=="Luxembourg" |
country_from=="Netherlands" | country_from=="Norway" | country_from=="Poland" |
country_from=="Portugal" | country_from=="Romania" | country_from=="Russian Federation" |
country_from=="Serbia" | country_from=="Spain" | country_from=="Sweden" |
country_from=="Switzerland" | country_from=="Turkey" | country_from=="United Kingdom"
replace regiontic5="TIC5_3" if country_from=="Australia" | country_from=="New Zealand"
```

```
replace regiontic5="TIC5_4" if country_from=="China" | country_from=="Hong Kong" |
country_from=="India" | country_from=="Indonesia" | country_from=="Israel" |
country_from=="Japan" | country_from=="Lebanon" | country_from=="Malaysia" |
country_from=="Philippines" | country_from=="Singapore" | country_from=="Korea, Republic of" |
country_from=="Thailand"
replace regiontic5="TIC5_5" if country_from=="Egypt" | country_from=="Morocco" |
country_from=="South Africa"
tostring year, gen(yearstr)
gen regiontic5xyear = regiontic5+"_"+yearstr


xi: reg logequity_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.regiontic5xyear, robust cl(iso3)
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, tic-year fe, yes, year fe, no )  append

xi: reg logdebt_world haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.regiontic5xyear, robust cl(iso3)
  outreg2 using oat1c3_4, excel stats(coef tstat)   bdec(3) tdec(3) drop(_I* o._I* tic*) addtext (cluster,
country, tic-year fe, yes, year fe, no )  append


* END OA Table 1 Col. 3-4
*****************************************************************************************
*************

* START OA Table 1 Col. 5-6
*****************************************************************************************
*************

pr drop _all
clear
clear matrix
clear mata

set matsize 1200
set mem 800m
set more off

capture log close

*****************************************************************************************
*********************************************
*1.  Import dataset for UCITS assets by country
*****************************************************************************************
*********************************************
clear
```

```
import excel ucits_assets_final.xlsx, firstrow
reshape long Y,i(Country)
rename _j year
rename Y ucits_assets
rename Country country
drop if missing(ucits_assets)
tab year
drop if year == 2005
drop if year == 2016
sort country year
by country:  gen id = _n
by country:  gen nt = _N
save ucits_assets2, replace

*******************************************************************************************
*********************************************
*2.  Regression specification
*******************************************************************************************
*********************************************

*Dataset for Regression
use ucits_assets2, clear
drop if nt != 10
drop if country == "Czech Republic"
drop if country == "Liechtenstein"
drop if country == "Slovakia"
egen country2 = group(country)
gen haven_ind = 0
replace haven_ind = 1 if country == "Luxembourg" | country == "Ireland" | country == "Switzerland"
gen ucits_assets2 = ucits_assets/1000
drop ucits_assets
rename ucits_assets2 ucits_assets
gen log_ucits = log(ucits_assets)
gen post = 0
replace post = 1 if year > 2011
gen post_haven = post*haven_ind
drop id nt
*Merge with country-year control variables used in Tables 2-4
replace country = upper(country)
merge m:1 country year using data1
keep if _merge == 3
drop _merge
keep country country2  year haven_ind ucits_assets log_ucits post post_haven localrate logpop loggdp
log_mobile_subs_p100 log_xrate

tsset country2 year, yearly
save ucits_assets3, replace
```

```
* main results
xi: reg ucits_assets post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate i.country
i.year
local rsq= round(`e(r2_a)',0.001)

xi: reg ucits_assets post_haven localrate logpop loggdp log_mobile_subs_p100 log_xrate i.country
i.year, cluster(country)
outreg2 using oat1c5_6, excel replace stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext ( country fe,
yes, year fe, yes, sample, full )

xi: reg log_ucits post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate i.country i.year
local rsq= round(`e(r2_a)',0.001)
xi: reg log_ucits post_haven localrate logpop loggdp log_mobile_subs_p100 log_xrate i.country i.year,
cluster(country)
outreg2 using oat1c5_6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

* Newey-West

xi: newey2 ucits_assets post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate i.country
i.year, lag(2) force
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c5_6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

xi: newey2 log_ucits post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate i.country
i.year, lag(2) force
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c5_6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

* Robust
xi: reg ucits_assets post_haven localrate logpop loggdp log_mobile_subs_p100 log_xrate i.country
i.year, vce(robust)
outreg2 using oat1c5_6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext ( country fe,
yes, year fe, yes, sample, full )

xi: reg log_ucits post_haven localrate logpop loggdp log_mobile_subs_p100 log_xrate i.country i.year,
vce(robust)
outreg2 using oat1c5_6, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext ( country fe,
yes, year fe, yes, sample, full )

* END OA Table 1 Col. 5-6
*******************************************************************************
*************
```

```
* START OA Table 1 Col. 7-10
*****************************************************************************************
*************

insheet using datacollection_cleaned.csv, clear

destring nom_price, replace force
destring real_price, replace force
destring rent, replace force
destring income, replace force
gen yearstr1 =substr(date, 1,4)
gen yearstr2 =substr(date, 4,7)
g yearstr=yearstr1
replace yearstr=yearstr2 if substr(yearstr1,1,1)=="Q"
drop if yearstr==""
destring yearstr, gen(year) force
replace year=2014 if year==1905
drop yearstr1 yearstr2 yearstr source
encode country, gen(countrynum)
* Morocco has wonky iso3 (appear to be month abbreviations)
replace iso3="MAR" if iso2=="MA"

sum nom_price real_price year rent income countrynum
* go from quarterly to annual (country-year observations)
bysort countrynum year: egen nombycountry=mean(nom_price)
bysort countrynum year: egen realbycountry=mean(real_price)
bysort countrynum year: egen rentbycountry=mean(rent)
bysort countrynum year: egen incomebycountry=mean(income)
by countrynum year: keep if _n==1

keep country year nombycountry realbycountry rentbycountry incomebycountry iso2 iso3
save recleaned, replace

* start with tostata
use data1, clear

* merge in cleaned real estate data
* tostata has country, iso2, iso3 and year
* recleaned has country year iso2 iso3
merge 1:1 iso3 year using recleaned

* define countries with foreign buyer restrictions
gen restricted=0
replace restricted=1 if country_from=="China"
replace restricted=1 if country_from=="Denmark"
replace restricted=1 if country_from=="Greece"
replace restricted=1 if country_from=="Hong Kong"
replace restricted=1 if country_from=="India"
```

```
replace restricted=1 if country_from=="Indonesia"
replace restricted=1 if country_from=="Malaysia"
replace restricted=1 if country_from=="Poland"
replace restricted=1 if country_from=="Singapore"
replace restricted=1 if country_from=="Switzerland"
* need to clean up restricted and unrestricted
gen unrestricted = 1-restricted

* highlight China and Hong Kong to exclude
gen exclude=0
replace exclude=1 if country_from=="China"
replace exclude=1 if country_from=="Hong Kong"

* highlight countries without a full panel to exclude
* RE data is also missing for Philippines and Thailand so these will drop out of regressions
* control data is missing for US so this will drop out of regressions
gen notfull=0
replace notfull=1 if country_from=="Brazil"
replace notfull=1 if country_from=="India"
replace notfull=1 if country_from=="Luxembourg"
replace notfull=1 if country_from=="Morocco"
replace notfull=1 if country_from=="Poland"
replace notfull=1 if country_from=="Romania"
replace notfull=1 if country_from=="Turkey"
replace notfull=1 if country_from=="Hungary"

* create regression variables
encode iso3, gen(niso3)
gen post=(year>=2012)
gen unrestricted_post=post*unrestricted
sum nombycountry
gen lognominal = log(nombycountry)
gen logreal = log(realbycountry)
gen logrent = log(rentbycountry)
gen logincome = log(incomebycountry)
* mobile_subs missing for Romania (already notfull==1)

save reregs, replace

use reregs, clear
drop if exclude==1
drop if year<2006
drop if year>2015
drop if notfull==1
drop if iso3=="USA"
drop if iso3=="THA"
drop if iso3=="PHL"
drop _merge
```

```
merge m:1 year using chinagdp
drop _merge
merge m:1 year using russiagdp
drop if _merge==1
drop _merge
bysort country: egen mean_realbycountry = mean(realbycountry)
* drop Serbia and Russia because so much smaller than the rest.  Spain would be next
* drop Peru and Malaysia because changes are huge
* drop Ireland because housing crisis in middle of sample, hit harder than US?
* UK is largest in the set, could drop them?
* drop if country=="Serbia" | country=="Russia" | country=="Peru" | country=="Malaysia" |
country=="Ireland"
* winsor2 realbycountry, trim cuts (5 95)
* gen logreal_t = log(realbycountry_t)
rename localrate local_taxrate
save reregsuse, replace

* Cluster by country / Country & Yr FE: row 1
use reregsuse, clear
eststo clear

* Column 7 row 1
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.country i.year, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel replace stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

* Column 8 row 1
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate  i.country
i.year, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

* Column 9 row 1
* drop countries above or below 10%/90% of mean realbycountry, country and year FE, clustering by
country
drop if country_from=="Serbia" | country_from=="Russian Federation" | country_from=="Spain" |
country_from=="United Kingdom" | country_from=="Peru" | country_from=="Malaysia"
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.country i.year, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, 10/90 )

* Column 10 row 1
```

```
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate  i.country
i.year, vce(cluster country)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, 10/90 )


* Newey-West S.Errors / Country & Yr FE: row 2
use reregsuse, clear
eststo clear

* Column 7 row 2
tsset niso3 year, yearly
xi: newey2 realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100
log_xrate  i.country i.year, lag(2) force
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

* Column 8 row 2
xi: newey2 logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.country i.year, lag(2) force
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

* Column 9 row 2
* drop countries above or below 10%/90% of mean realbycountry, country and year FE, clustering by
country
drop if country_from=="Serbia" | country_from=="Russian Federation" | country_from=="Spain" |
country_from=="United Kingdom" | country_from=="Peru" | country_from=="Malaysia"
xi: newey2 realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100
log_xrate  i.country i.year, lag(2) force
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, 10/90 )

* Column 10 row 2
xi: newey2 logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.country i.year, lag(2) force
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, 10/90 )


* Robust S.Errors / Country & Yr FE: row 3
use reregsuse, clear
eststo clear

* Column 7 row 3
```

```
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.country i.year, vce(robust)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

* Column 8 row 3
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate  i.country
i.year, vce(robust)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, full )

* Column 9 row 3
* drop countries above or below 10%/90% of mean realbycountry, country and year FE, clustering by
country
drop if country_from=="Serbia" | country_from=="Russian Federation" | country_from=="Spain" |
country_from=="United Kingdom" | country_from=="Peru" | country_from=="Malaysia"
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.country i.year, vce(robust)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, 10/90 )

* Column 10 row 3
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate  i.country
i.year, vce(robust)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
country fe, yes, year fe, yes, sample, 10/90 )


* Cluster by Country / WB Region-by-Yr FE: row 4
use reregsuse, clear
eststo clear

gen regionwb="WB1" if country_from=="Argentina" | country_from=="Bahamas" |
country_from=="Bermuda" | country_from=="Brazil" | country_from=="Chile" |
country_from=="Colombia" | country_from=="Ecuador" | country_from== "Guatemala" |
country_from=="Jamaica" | country_from=="Panama" | country_from=="Peru" |
country_from=="Uruguay"
replace regionwb="WB2" if country_from=="Austria" | country_from=="Belgium" |
country_from=="Denmark" | country_from=="Finland" | country_from=="France" |
country_from=="Germany" | country_from=="Greece" | country_from=="Hungary" |
country_from=="Ireland" | country_from=="Italy" | country_from=="Luxembourg" |
country_from=="Netherlands" | country_from=="Norway" | country_from=="Poland" |
country_from=="Portugal" | country_from=="Romania" | country_from=="Russian Federation" |
country_from=="Serbia" | country_from=="Spain" | country_from=="Sweden" |
country_from=="Switzerland" | country_from=="Turkey" | country_from=="United Kingdom"
```

```
replace regionwb="WB3" if country_from=="Australia" | country_from=="China" |
country_from=="Hong Kong" | country_from=="Indonesia" | country_from=="Japan" |
country_from=="Malaysia" | country_from=="New Zealand" | country_from=="Philippines" |
country_from=="Singapore" | country_from=="Korea, Republic of" | country_from=="Thailand"
replace regionwb="WB4" if country_from=="Egypt" | country_from=="Israel" |
country_from=="Lebanon" | country_from=="Morocco"
replace regionwb="WB5" if country_from=="Canada" | country_from=="Mexico"
replace regionwb="WB6" if country_from=="India"
replace regionwb="WB7" if country_from=="South Africa"
tostring year, gen(yearstr)
gen regionwbxyear = regionwb+"_"+yearstr

* Column 7 row 4
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.regionwbxyear, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
wb-by-year fe, yes, sample, full)

* Column 8 row 4
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.regionwbxyear, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
wb-by-year fe, yes, sample, full)

* Column 9 row 4
* drop countries above or below 10%/90% of mean realbycountry, country and year FE, clustering by
country, adding china and russia gdp
drop if country_from=="Serbia" | country_from=="Russian Federation" | country_from=="Spain" |
country_from=="United Kingdom" | country_from=="Peru" | country_from=="Malaysia"
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.regionwbxyear, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
wb-by-year fe, yes, sample, 10/90)

* Column 10 row 4
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.regionwbxyear, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
wb-by-year fe, yes, sample, 10/90)



* Cluster by Country / TIC Region-by-Yr FE: row 5
use reregsuse, clear
```

```
eststo clear

gen regiontic5="TIC5_1" if country_from=="Argentina" | country_from=="Bahamas" |
country_from=="Bermuda" | country_from=="Brazil" | country_from=="Canada" |
country_from=="Chile" | country_from=="Colombia" | country_from=="Ecuador" | country_from==
"Guatemala" | country_from=="Jamaica" | country_from=="Mexico" | country_from=="Panama" |
country_from=="Peru" | country_from=="Uruguay"
replace regiontic5="TIC5_2" if country_from=="Austria" | country_from=="Belgium" |
country_from=="Denmark" | country_from=="Finland" | country_from=="France" |
country_from=="Germany" | country_from=="Greece" | country_from=="Hungary" |
country_from=="Ireland" | country_from=="Italy" | country_from=="Luxembourg" |
country_from=="Netherlands" | country_from=="Norway" | country_from=="Poland" |
country_from=="Portugal" | country_from=="Romania" | country_from=="Russian Federation" |
country_from=="Serbia" | country_from=="Spain" | country_from=="Sweden" |
country_from=="Switzerland" | country_from=="Turkey" | country_from=="United Kingdom"
replace regiontic5="TIC5_3" if country_from=="Australia" | country_from=="New Zealand"
replace regiontic5="TIC5_4" if country_from=="China" | country_from=="Hong Kong" |
country_from=="India" | country_from=="Indonesia" | country_from=="Israel" |
country_from=="Japan" | country_from=="Lebanon" | country_from=="Malaysia" |
country_from=="Philippines" | country_from=="Singapore" | country_from=="Korea, Republic of" |
country_from=="Thailand"
replace regiontic5="TIC5_5" if country_from=="Egypt" | country_from=="Morocco" |
country_from=="South Africa"
tostring year, gen(yearstr)
gen regiontic5xyear = regiontic5+"_"+yearstr

* Column 7 row 5
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.regiontic5xyear, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
tic-by-year fe, yes, sample, full)

* Column 8 row 5
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.regiontic5xyear, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
tic-by-year fe, yes, sample, full)

* Column 9 row 5
* drop countries above or below 10%/90% of mean realbycountry, country and year FE, clustering by
country, adding china and russia gdp
drop if country_from=="Serbia" | country_from=="Russian Federation" | country_from=="Spain" |
country_from=="United Kingdom" | country_from=="Peru" | country_from=="Malaysia"
xi: reg realbycountry unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.regiontic5xyear, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
```

```
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
tic-by-year fe, yes, sample, 10/90)

* Column 10 row 5
xi: reg logreal unrestricted_post local_taxrate logpop loggdp  log_mobile_subs_p100 log_xrate
i.regiontic5xyear, vce(cluster country)
local rsq= round(`e(r2_a)',0.001)
outreg2 using oat1c7_10, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*) addtext (adj-r2, `rsq',
tic-by-year fe, yes, sample, 10/90)

* END OA Table 1 Col. 7-10
*******************************************************************************
*************

* END Table 1 Standard Errors
*******************************************************************************
*************



* START Table 2 weighted
*******************************************************************************
*************


clear

use data1

rename localrate local_taxrate
gen post=(year>=2012)
gen haven_post=post*haven

encode iso3, gen(niso3)

save "temp.dta", replace
drop if year != 2011
keep iso2 lte population
gen eq_pop11 = lte/population
rename lte lte11
sort iso2
save "temp2.dta", replace
use "temp.dta"
sort iso2
merge iso2 using "temp2.dta"
drop _merge

tsset niso3 year, yearly
```

```
xi: newey2 tic_logequity_us haven_post local_taxrate logpop loggdp log_mobile_subs_p100 log_xrate
i.country i.year [aweight=eq_pop11], lag(2) force

* END Table 2 weighted
*****************************************************************************************
*************
```

```
* START Table 3 robust reg on UCITS
*****************************************************************************************
*************

pr drop _all
clear
clear matrix
clear mata

set matsize 1200
set mem 800m
set more off

capture log close

*****************************************************************************************
*********************************************
*1.  Import dataset for UCITS assets by country
*****************************************************************************************
*********************************************
clear

import excel ucits_assets_final.xlsx, firstrow
reshape long Y,i(Country)
rename _j year
rename Y ucits_assets
rename Country country
drop if missing(ucits_assets)
tab year
drop if year == 2005
drop if year == 2016
sort country year
by country:  gen id = _n
by country:  gen nt = _N
save ucits_assets2, replace
```

```
*****************************************************************************************
*********************************************
*2.  Regression specification
*****************************************************************************************
*********************************************

*Dataset for Regression
use ucits_assets2, clear
drop if nt != 10
drop if country == "Czech Republic"
drop if country == "Liechtenstein"
drop if country == "Slovakia"
egen country2 = group(country)
gen haven_ind = 0
replace haven_ind = 1 if country == "Luxembourg" | country == "Ireland" | country == "Switzerland"
gen ucits_assets2 = ucits_assets/1000
drop ucits_assets
rename ucits_assets2 ucits_assets
gen log_ucits = log(ucits_assets)
gen post = 0
replace post = 1 if year > 2011
gen post_haven = post*haven_ind
drop id nt
*Merge with country-year control variables used in Tables 2-4
replace country = upper(country)
merge m:1 country year using data1
keep if _merge == 3
drop _merge
keep country country2  year haven_ind ucits_assets log_ucits post post_haven localrate logpop loggdp
log_mobile_subs_p100 log_xrate

xi i.country i.year

rreg ucits post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate _Icountry* _Iyear*
outreg2 using oat3, excel replace stats(coef tstat) bdec(3) tdec(3) drop(_I*)

rreg log_ucits post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate _Icountry* _Iyear*
outreg2 using oat3, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*)

* drop small countries

drop if country == "GREECE"
drop if country == "PORTUGAL"
drop if country == "HUNGARY"
drop if country == "TURKEY"
drop if country == "POLAND"
xi i.country i.year
```

```
rreg ucits post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate _Icountry* _Iyear*
outreg2 using oat3, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*)

rreg log_ucits post_haven localrate logpop loggdp  log_mobile_subs_p100 log_xrate _Icountry* _Iyear*
outreg2 using oat3, excel append stats(coef tstat) bdec(3) tdec(3) drop(_I*)

* END Table 3 robust reg on UCITS
*******************************************************************************
*************
```