

```

1  /*****
2  ****
3  **Datasheet - Code Description Stata File
4
5  **Component of Journal of Accounting Research Data Policy Requirements,
6      Code and description for
7      Made in the U.S.A.? A Study of Firm Responses to Domestic Production Incentives
8      Rebecca Lester, 2019
9
10 **Data primarily comes from four sources:
11 1. Historical Financial Information obtained from Compustat (Annual, Quarterly, and
12    Segments) databases.
13 2. Jurisdiction-specific confidential information on multinational firms obtained from the
14    Bureau of Economic Analysis as part of the Special Sworn
15    Employee program.
16 3. Data collected from financial statement searches used to (i) identify DPAD firms, (ii)
17    identify ETI claimants, (iii) measure positive performance
18    shocks, (iv) and measure tax NOLs.
19 4. Data obtained from other researchers, including foreign subsidiary data (Scott Dyreng's
20    website), repatriating firms (Michelle Hanlon), and R&D
21    tax credit data (Jeff Hoopes).
22
23 A brief description of each of these files and the source of the information (along with
24 any additional needed detail) is provided below. After these
25 descriptions, I provide Stata code that details variable construction and the process of
26 linking the various data sets together. See the datasheet
27 description for details of when the data was downloaded from the source, keeping in mind
28 that variable definitions and labels change over time.
29 ****
30 ****
31 *ALPHABETICAL LISTING OF DATASETS*
32
33 bea_dom_data: Data on domestic activity of multinational firms as reported on the BEA
34 surveys for U.S. Direct Investment Abroad
35
36 bea_for_data: Data on foreign activities of U.S. multinational firms as reported on the
37 BEA surveys for U.S. Direct Investment Abroad
38
39 compustat_dataset_1995_2013: Annual data downloaded from Compustat with key variables.
40 Includes data from 1995 through 2013 so as to construct appropriate
41 lagged values needed for some tests.
42
43 compustat_data_2013_2017: Annual data downloaded fro Compustat with key variables for
44 Cross-border Income Shifting tests (H2).
45
46 compustat_quarterly_data: Quarterly data downloaded from Compustat with key variables
47 needed for testing inter-temporal shifting (H1). Includes data from 1995
48 through 2013 so as to construct appropriate lagged values needed for some tests.
49
50 DPAD_firms: Dataset identifying firms that either discuss or disclose the DPAD benefit in
51 their financial statements. Dataset was created by first providing
52 search terms to Jeff Hoopes in 2014 to include in a Perl program. All hits were
53 reviewed to determine if they did indeed related to the DPAD benefit.
54
55 exhibit21: Data on material foreign subsidiaries disclosed in firm financial statements;
56 obtained from Scott Dyreng's website.
57
58 nol_data: Data collected from firm financial statements on the amount of a firm's total
59 NOL, as well as the federal portion of NOL. See Data Description Sheet.
60
61 rdchand: R&D tax credit data provided by Jeff Hoopes from his paper "The Effect of
62 Temporary Tax Laws on Understanding and Predicting Corporate Earnings."
63
64 segment_dataset: Includes segment data downloaded from Compustat for 1995 through 2013.
65
66 segment_data_2013_2017: Includes Compustat segment data downloaded for 2013 through 2017
67 for Cross-border Income Shifting tests (H2).
68
69
70

```

```

51 *****
52 *****
53
54 1. Create dataset of merged DPAD-Compustat data
55 2. Match DPAD firms to control firms
56 3. Create dataset for tests of Intertemporal Shifting (H1)
57 4. Create dataset for tests of Cross-border Shifting (H2)
58 5. Create dataset for tests of Domestic Investment (H3a)
59 6. Create dataset for tests of Domestic Employment (H3b)
60
61 *****
62 *****
63 *** Step 1 - Create dataset of DPAD firms and Compustat data
64 *****/
65 /*IMPORT INFORMATION TO IDENTIFY DPAD FIRMS*/
66 clear
67 import excel using DPAD_firms, firstrow
68 drop if match_text == 0
69 drop if missing(match_text)
70
71 /*FORMAT DATES TO ENSURE CORRECT FISCAL-TAX YEAR MATCH*/
72 todate datadate, generate(datadate2) pattern(yyyymmdd)
73 gen year=year(datadate2)
74 gen month=month(datadate2)
75 gen str10 cik2 = string(cik, "%010.0f")
76 drop cik
77 rename cik2 cik
78 sort cik datadate month
79 sort cik fiscal_year
80 by cik fiscal_year: keep if _n==1
81 rename fiscal_year fyear
82 drop datadate datadate2 year month
83 save 199_fy, replace
84
85 /*IMPORT COMPUSTAT DATA AND MERGE WITH 199 DATASET*/
86 use compustat_dataset_1995_2013, clear
87 sort gvkey fyear
88 by gvkey fyear: keep if _n==1
89 save compustat_all, replace
90
91 use compustat_all, clear
92 keep if missing(cik)
93 save compustat_all_nocik, replace
94 use compustat_all, clear
95 drop if missing(cik)
96 sort cik fyear
97 by cik fyear: keep if _n==1
98 merge m:1 cik fyear using 199_fy
99 drop if _merge==2
100 drop _merge
101 save compustat_all_cik, replace
102 use compustat_all_cik, clear
103 append using compustat_all_nocik
104 sort gvkey fyear
105 by gvkey fyear: keep if _n==1
106 keep if fic == "USA"
107
108 /*CREATE LAGGED VARIABLES FOR LATER TESTS*/
109 by gvkey: gen at_lag = at[_n-1] if fyear==fyear[_n-1]+1
110 by gvkey: gen at_lag2 = at[_n-2] if fyear==fyear[_n-2]+2
111 by gvkey: gen ch_lag = ch[_n-1] if fyear==fyear[_n-1]+1
112 by gvkey: gen che_lag = che[_n-1] if fyear==fyear[_n-1]+1
113 by gvkey: gen cogs_lag = cogs[_n-1] if fyear==fyear[_n-1]+1
114 by gvkey: gen csho_lag = csho[_n-1] if fyear==fyear[_n-1]+1
115 by gvkey: gen dlc_lag = dlc[_n-1] if fyear==fyear[_n-1]+1
116 by gvkey: gen dltd_lag = dltd[_n-1] if fyear==fyear[_n-1]+1
117 by gvkey: gen do_lag = do[_n-1] if fyear==fyear[_n-1]+1

```

```

118 by gvkey: gen invt_lag = invt[_n-1] if fyear==fyear[_n-1]+1
119 by gvkey: gen oancf_lag = oancf[_n-1] if fyear==fyear[_n-1]+1
120 by gvkey: gen oiadp_lag = oiadp[_n-1] if fyear==fyear[_n-1]+1
121 by gvkey: gen pi_lag = pi[_n-1] if fyear==fyear[_n-1]+1
122 by gvkey: gen pidom_lag = pidom[_n-1] if fyear==fyear[_n-1]+1
123 by gvkey: gen pifo_lag = pifo[_n-1] if fyear==fyear[_n-1]+1
124 by gvkey: gen ppent_lag = ppent[_n-1] if fyear==fyear[_n-1]+1
125 by gvkey: gen sale_lag = sale[_n-1] if fyear==fyear[_n-1]+1
126 by gvkey: gen seq_lag = seq[_n-1] if fyear==fyear[_n-1]+1
127 by gvkey: gen spi_lag = spi[_n-1] if fyear==fyear[_n-1]+1
128 by gvkey: gen tlcf_lag = tlcf[_n-1] if fyear==fyear[_n-1]+1
129 by gvkey: gen txfed_lag = txfed[_n-1] if fyear==fyear[_n-1]+1
130 by gvkey: gen txfo_lag = txfo[_n-1] if fyear==fyear[_n-1]+1
131 by gvkey: gen xad_lag = xad[_n-1] if fyear==fyear[_n-1]+1
132 by gvkey: gen xi_lag = xi[_n-1] if fyear==fyear[_n-1]+1
133 by gvkey: gen xido_lag = xido[_n-1] if fyear==fyear[_n-1]+1
134 *calculate three lags for R&D credit calculation
135 by gvkey: gen xrd_lag = xrd[_n-1] if fyear==fyear[_n-1]+1
136 by gvkey: gen xrd_lag2 = xrd[_n-2] if fyear==fyear[_n-2]+2
137 by gvkey: gen xrd_lag3 = xrd[_n-3] if fyear==fyear[_n-3]+3
138 by gvkey: gen xrd_lag4 = xrd[_n-4] if fyear==fyear[_n-4]+4
139 by gvkey: gen xsga_lag = xsga[_n-1] if fyear==fyear[_n-1]+1
140 by gvkey: gen prcc_f_lag = prcc_f[_n-1] if fyear==fyear[_n-1]+1
141 *by gvkey: gen capx_lag = capx[_n-1] if fyear==fyear[_n-1]+1
142 *by gvkey: gen emp_lag = emp[_n-1] if fyear==fyear[_n-1]+1
143 sort gvkey fyear
144 by gvkey fyear: keep if _n==1
145 save compustat_all3, replace
146
147 /*JOIN SEGMENT DATA*/
148 use segment_dataset, clear
149 sort gvkey datadate sid srcdate
150 by gvkey datadate sid: keep if _n==1
151 drop if stype != "GEOSEG" /*DROP BUSINESS SEGMENTS*/
152 drop if geotp == "1" /*TYPE 1 = NO LONGER USED PER WRDS*/
153 gen fyear=year(datadate)
154 sort gvkey sid fyear
155 by gvkey sid: gen atlls_lag = atlls[_n-1] if fyear==fyear[_n-1]+1
156 by gvkey sid: gen capxs_lag = capxs[_n-1] if fyear==fyear[_n-1]+1
157 by gvkey sid: gen cogss_lag = cogss[_n-1] if fyear==fyear[_n-1]+1
158 by gvkey sid: gen emps_lag = emps[_n-1] if fyear==fyear[_n-1]+1
159 by gvkey sid: gen ias_lag = ias[_n-1] if fyear==fyear[_n-1]+1
160 by gvkey sid: gen intseg_lag = intseg[_n-1] if fyear==fyear[_n-1]+1
161 by gvkey sid: gen oelim_lag = oelim[_n-1] if fyear==fyear[_n-1]+1
162 by gvkey sid: gen ppents_lag = ppents[_n-1] if fyear==fyear[_n-1]+1
163 by gvkey sid: gen rds_lag = rds[_n-1] if fyear==fyear[_n-1]+1
164 by gvkey sid: gen revts_lag = revts[_n-1] if fyear==fyear[_n-1]+1
165 by gvkey sid: gen sales_lag = sales[_n-1] if fyear==fyear[_n-1]+1
166 by gvkey sid: gen salexg_lag = salexg[_n-1] if fyear==fyear[_n-1]+1
167 by gvkey sid: gen txts_lag = txts[_n-1] if fyear==fyear[_n-1]+1
168 by gvkey sid: gen xsgas_lag = xsgas[_n-1] if fyear==fyear[_n-1]+1
169 save segment_all, replace
170
171 /*CLEAN DOMESTIC SEGMENT DATA*/
172 use segment_all, clear
173 drop if geotp == "3" /*DROP THOSE SEGMENTS CODED FOREIGN AND SEGMENTS WITH NON-US NAMES*/
174 drop if snms=="Africa" | snms=="Anguilla" | snms=="Antigua" | snms=="Antigua and Barbuda" |
snms=="Antigua, West Indies" | snms=="Argentina" | snms=="Asia" | snms=="Asia Pacific" |
snms=="Asia and Others" | snms=="Asia pacific" | snms=="Asia,Pacific,Australia" | snms=="
Australasia"
175 drop if snms=="Australia" | snms=="Australia & New Zealand" | snms=="Australia & S.E. Asia"
| snms=="Australia and New Zealand" | snms=="Australia, Asia and Pacific" | snms=="
Australia, New Zealand"
176 drop if snms=="Austria" | snms=="Bahamas" | snms=="Barbados" | snms=="Belgium" | snms=="
Belize" | snms=="Bermuda" | snms=="Brazil" | snms=="Brazil and Others" | snms=="British
Columbia"
177 drop if snms=="British Virgin Islands" | snms=="Canada" | snms=="Canada and CALA" | snms=="
Canada and Other" | snms=="Canada and Other Foreign" | snms=="
Canada/Mexico/Caribbean/South America" | snms=="Canada/Mexico/Caribbean"
178 drop if snms=="Canada/Other Americas/Caribbean" | snms=="Canadian" | snms=="Caribbean" |

```

```

snms=="Cayman Islands" | snms=="Cayman Islands/Bermuda" | snms=="Central" | snms=="Central
179 drop if snms=="Canada" | snms=="Central and Eastern Europe, Middle East and Africa" | snms
=="Central and Northern Europe" | snms=="Channel Islands" | snms=="Chile" | snms=="Chile
& Other" | snms=="China" | snms=="China/Hong Kong"
180 drop if snms=="Continental Europe" | snms=="Curacao" | snms=="Cyprus" | snms=="Denmark" |
snms=="Dominican Republic" | snms=="Dutch Antilles" | snms=="El Sitio Inc" | snms=="Euro
Area"
181 drop if snms=="Europe" | snms=="Europe & AME" | snms=="Europe & Central Asia" | snms==
"Europe & Middle East" | snms=="Europe Region" | snms=="Europe and Australia" | snms==
"Europe and Middle East"
182 drop if snms=="Europe and Other" | snms=="Europe and Pacific" | snms=="Europe regions" |
snms=="Europe, Africa & Middle East" | snms=="Europe, Africa, Middle East" | snms=="Europe,
CIS & West Africa"
183 drop if snms=="Europe, Middle East & Africa" | snms=="Europe, Middle East and Africa" | snms
=="Europe, Middle East and Asia" | snms=="Europe, Middle-East and Africa" | snms=="Europe,
North Africa"
184 drop if snms=="Europe, including France" | snms=="Europe/ West Africa/ FSU" | snms==
"Europe/Australia" | snms=="Europe/Island of Guernsey" | snms=="Europe/SSA/Russia" | snms==
"European Countries"
185 drop if snms=="European Russia" | snms=="European Union" | snms=="Excise Taxes" | snms==
"FR, DE, NL and UK" | snms=="Far East and Other" | snms=="Finland" | snms=="Foreign
Countries" | snms=="France" | snms=="France/Western Europe"
186 drop if snms=="Germany" | snms=="Germany and Ireland" | snms=="Ghana" | snms=="Great
Britain" | snms=="Greater China" | snms=="Greece" | snms=="Hong Kong" | snms=="Hong Kong
and Mainland China"
187 drop if snms=="Hong kong" | snms=="Hungary" | snms=="India" | snms=="India & Other" | snms==
"Indonesia" | snms=="International" | snms=="Ireland" | snms=="Israel" | snms=="Israel and
Other"
188 drop if snms=="Israel and Others" | snms=="Isreal and Others" | snms=="Isreal" | snms==
"Italy" | snms=="Italy & Other European Countries" | snms=="Italy/Europe" | snms=="Japan" |
snms=="Jersey" | snms=="Jersey, Channel Islands"
189 drop if snms=="Korea" | snms=="Latin America" | snms=="Latin America North" | snms==
"Liberia" | snms=="Luxembourg" | snms=="Mainland China" | snms=="Malaysia & Singapore" |
snms=="Marshall Islands"
190 drop if snms=="Mauritius" | snms=="Metropolitan France" | snms=="Mexico" | snms=="Mexico
and Central America" | snms=="Middle East, Central Europe & Other" | snms=="Moscow" | snms==
"Moscow License Area"
191 drop if snms=="NALA" | snms=="NASA" | snms=="NOLAD" | snms=="Netherlands" | snms==
"Netherlands Antilles" | snms=="Netherlands Antillies" | snms=="New Zealand" | snms==
"Non-U.S. or Non-U.K." | snms=="Non-US" | snms=="Non-US Americas"
192 drop if snms=="Non-operating" | snms=="Nordic" | snms=="Nordic Countries" | snms=="Nordic
Region" | snms=="North & Central Europe" | snms=="North Region" | snms=="North West Europe"
| snms=="Northern Europe"
193 drop if snms=="Norway" | snms=="Other Countries" | snms=="Other European Countries" | snms==
"Other Foreign" | snms=="Other Geographical Areas" | snms=="Other Markets" | snms=="Other
Regions"
194 drop if snms=="PRC" | snms=="Panama" | snms=="Papua New Guinea" | snms=="People's Republic
of China" | snms=="Peoples Republic of China" | snms=="The Peoples Republic of China (PRC)"
| snms=="Peru" | snms=="Philippines" | snms=="Phillippines"
195 drop if snms=="Poland" | snms=="Portugal" | snms=="RUSSIA" | snms=="Republic of Ireland" |
snms=="Republic of Korea" | snms=="Republic of Liberia" | snms=="Republic of Panama"
196 drop if snms=="Republic of The Marshall Islands" | snms=="Republic of the Marshall Islands"
| snms=="Rest of Europe" | snms=="Rest of World" | snms=="Russia" | snms=="Russian
Federation"
197 drop if snms=="Scandinavia" | snms=="Singapore" | snms=="South Africa" | snms=="South
America" | snms=="South Korea" | snms=="Southern Africa" | snms=="Spain" | snms=="Spain &
Portugal"
198 drop if snms=="Sweden" | snms=="Switzerland" | snms=="Taiwan" | snms=="Taiwan, ROC" | snms==
"Thailand" | snms=="The Netherlands" | snms=="The PRC" | snms=="The People's Republic of
China"
199 drop if snms=="The Peoples Republic of China: Hong Kong" | snms=="Turkey" | snms=="UK" |
snms=="UK & Ireland" | snms=="UK and Ireland" | snms=="Ukraine" | snms=="United Kingdom" |
snms=="United Kingdom & Denmark"
200 drop if snms=="United Kingdom & Europe" | snms=="United Kingdom & Ireland" | snms=="United
Kingdom and Europe" | snms=="United Kingdom and Ireland" | snms=="United Kingdom, Europe &
Middle East"
201 drop if snms=="United Kingdom, Europe and Other" | snms=="United Kingdom, Europe and middle
east" | snms=="United Kingdom, Ireland" | snms=="United Kingdom/Ireland" | snms=="United
Kingdom/Republic of Ireland"

```

```

202 drop if snms=="Venezuela" | snms=="Western Canada" | snms=="Western Europe" | snms==
    "Western Hemisphere"
203
204 bysort gvkey fyear: egen seq = seq()
205 foreach v in atlls capxs cogss emps ias intseg oelim ppents rds revts sales salexg txts
    xsgas atlls_lag capxs_lag cogss_lag emps_lag ias_lag intseg_lag oelim_lag ppents_lag rds_lag
    revts_lag sales_lag salexg_lag txts_lag xsgas_lag {
206     bysort gvkey fyear: egen c`v' = count(`v') if(`v'==.)
207     replace c`v' = 1 if(c`v'==0)
208     bysort gvkey fyear: egen c2`v' = count(c`v')
209     bysort gvkey fyear: egen sum`v' = sum(`v')
210     bysort gvkey fyear: egen mn`v' = mean(`v')
211     replace sum`v' = -99 if(mn`v' == .)
212     replace sum`v' = . if(seq > 1)
213     replace sum`v' = -99 if (c2`v'~=0)
214     drop c`v' c2`v' mn`v' `v'
215     rename sum`v' `v'
216 }
217 drop seq
218
219 collapse (sum) DOMLLASSET=atlls DOMCAPX=capxs DOMCOGS=cogss DOMEMP=emps DOMASSET=ias DOMINT
    =intseg DOMOELIM=oelim DOMPPE=ppents DOMRD=rds DOMREV=revts DOMSEGSale=sales DOMEXPORT=
    salexg DOMTXT=txts DOMSGA=xsgas DOMLLASSET_LAG=atlls_lag DOMCAPX_LAG=capxs_lag DOMCOGS_LAG=
    cogss_lag DOMEMP_LAG=emps_lag DOMASSET_LAG=ias_lag DOMINT_LAG=intseg_lag DOMOELIM_LAG=
    oelim_lag DOMPPE_LAG=ppents_lag DOMRD_LAG=rds_lag DOMREV_LAG=revts_lag DOMSEGSale_LAG=
    sales_lag DOMEXPORT_LAG=salexg_lag DOMTXT_LAG=txts_lag DOMSGA_LAG=xsgas_lag, by(gvkey fyear)
220
221 foreach v in DOMLLASSET DOMCAPX DOMCOGS DOMEMP DOMASSET DOMINT DOMOELIM DOMPPE DOMRD DOMREV
    DOMSEGSale DOMEXPORT DOMTXT DOMSGA DOMLLASSET_LAG DOMCAPX_LAG DOMCOGS_LAG DOMEMP_LAG
    DOMASSET_LAG DOMINT_LAG DOMOELIM_LAG DOMPPE_LAG DOMRD_LAG DOMREV_LAG DOMSEGSale_LAG
    DOMEXPORT_LAG DOMTXT_LAG DOMSGA_LAG {
222     replace `v' = . if(`v' < 0)
223 }
224
225 save segment_domestic, replace
226
227 /*CLEAN FOREIGN SEGMENT DATA*/
228 use segment_all, clear
229 drop if geotp == "2" /*DROP THOSE SEGMENTS CODED DOMESTIC AND SEGMENTS WITH US OR DOMESTIC
    LABELS*/
230
231 drop if snms=="North America" | snms=="North America (mainly the United States)" | snms==
    "North America,United States" | snms=="North America,United States,Canada"
232 drop if snms=="North America,United States,Other Foreign" | snms=="North America/United
    States" | snms=="North American" | snms=="Other Domestic" | snms=="Parent"
233 drop if snms=="Puerto Rico" | snms=="Puerto Rico and Barbados" | snms=="Puerto Rico,
    Mexico, Canada and Europe" | snms=="The Americas" | snms=="U.S." | snms=="U.S. , Canada &
    Mexico"
234 drop if snms=="U.S. Gulf of Mexico" | snms=="U.S. Gulf-Mexico" | snms=="U.S. and Canada" |
    snms=="U.S.A" | snms=="U.S.A." | snms=="US" | snms=="US & Canada" | snms=="USA"
235 drop if snms=="USA & Canada" | snms=="USA & Europe" | snms=="USA & North America" | snms==
    "USA Region" | snms=="USA and Canada" | snms=="USA and Mexico" | snms=="USA and North
    America"
236 drop if snms=="USA and Other" | snms=="USA region" | snms=="USA(including the Americas)" |
    snms=="USA/South America" | snms=="United States" | snms=="United States & Canada"
237 drop if snms=="United States & Europe" | snms=="United States & India" | snms=="United
    States & Latin America" | snms=="United States & Mexico" | snms=="United States & Other"
238 drop if snms=="United States & Puerto Rico" | snms=="United States America" | snms=="United
    States Northeast" | snms=="United States Of America" | snms=="United States Region"
239 drop if snms=="United States South" | snms=="United States Territories and Foreign
    Countries" | snms=="United States and Canada" | snms=="United States and Europe"
240 drop if snms=="United States and Other" | snms=="United States of America" | snms=="United
    States& North America" | snms=="United States, Asia Pacific" | snms=="United States, Canada"
241 drop if snms=="United States, Europe, other regions" | snms=="United States, Latin America
    & Canada" | snms=="United States,Canada" | snms=="United States,Canada,Mexico"
242 drop if snms=="United States,Canada,Other Foreign" | snms=="United States,Mexico" | snms==
    "United States,Other Foreign" | snms=="United States/Canada" | snms=="United
    States/Mexico/Indonesia/China"
243 drop if snms=="United states" | snms=="United states of America" | snms=="Unted States" |

```

```

snms=="Untied States" | snms=="Usa and Canada"

244
245 bysort gvkey fyear: egen seq = seq()
246 foreach v in atlls capxs cogss emps ias intseg oelim ppents rds revts sales salexg txts
xsgas atlls_lag capxs_lag cogss_lag emps_lag ias_lag intseg_lag oelim_lag ppents_lag rds_lag
revts_lag sales_lag salexg_lag txts_lag xsgas_lag {
247     bysort gvkey fyear: egen c`v' = count(`v') if(`v'==.)
248     replace c`v' = 1 if(c`v'==0)
249     bysort gvkey fyear: egen c2`v' = count(c`v')
250     bysort gvkey fyear: egen sum`v' = sum(`v')
251     bysort gvkey fyear: egen mn`v' = mean(`v')
252     replace sum`v' = -99 if(mn`v' == .)
253     replace sum`v' = . if(seq > 1)
254     replace sum`v' = -99 if (c2`v'~=0)
255     drop c`v' c2`v' mn`v' `v'
256     rename sum`v' `v'
257 }
258 drop seq
259
260 collapse (sum) FORLLASSET=atlls FORCAPX=capxs FORCOGS=cogss FOREMP=emps FORASSET=ias FORINT
=intseg FOROELIM=oelim FORPPE=ppents FORRD=rds FORREV=revts FORSEGSale=sales FOREXPOR=
salexg FORTXT=txts FORSGA=xsgas FORLLASSET_LAG=atlls_lag FORCAPX_LAG=capxs_lag FORCOGS_LAG=
cogss_lag FOREMP_LAG=emps_lag FORASSET_LAG=ias_lag FORINT_LAG=intseg_lag FOROELIM_LAG=
oelim_lag FORPPE_LAG=ppents_lag FORRD_LAG=rds_lag FORREV_LAG=revts_lag FORSEGSale_LAG=
sales_lag FOREXPOR_LAG=salexg_lag FORTXT_LAG=txts_lag FORSGA_LAG=xsgas_lag, by(gvkey fyear)
261 foreach v in FORLLASSET FORCAPX FORCOGS FOREMP FORASSET FORINT FOROELIM FORPPE FORRD FORREV
FORSEGSale FOREXPOR FORTXT FORSGA FORLLASSET_LAG FORCAPX_LAG FORCOGS_LAG FOREMP_LAG
FORASSET_LAG FORINT_LAG FOROELIM_LAG FORPPE_LAG FORRD_LAG FORREV_LAG FORSEGSale_LAG
FOREXPOR_LAG FORTXT_LAG FORSGA_LAG {
262     replace `v' = . if(`v' < 0)
263 }
264
265 gen for_sale_ind = 0
266 replace for_sale_ind = 1 if FORSEGSale > 0
267 replace for_sale_ind = 0 if missing(FORSEGSale)
268 gen for_asset_ind = 0
269 replace for_asset_ind = 1 if FORASSET > 0
270 replace for_asset_ind = 0 if missing(FORASSET)
271 save segment_foreign, replace
272
273 /*MERGE DOMESTIC/FOREIGN SEGMENT DATA*/
274 use segment_all, clear
275 sort gvkey fyear
276 keep gvkey fyear
277 by gvkey fyear: keep if _n==1
278 save clean_segment, replace
279 merge 1:1 gvkey fyear using segment_domestic
280 drop _merge
281 merge 1:1 gvkey fyear using segment_foreign
282 drop _merge
283 save clean_segment, replace
284
285 use compustat_all3, clear
286 merge 1:1 gvkey fyear using clean_segment
287 sort gvkey fyear
288 drop if _merge == 2
289 drop _merge
290 save compustat_all_segments, replace
291
292 /*ADD EXHIBIT21 DATA FROM DYRENG AND LINDSAY (2009)*/
293 use exhibit21, clear
294 gen fyear=year(datadate)
295 rename ciknumber cik
296 save exhibit21_v2, replace
297 collapse (sum) taxhaven, by (cik fyear)
298 save exhibit21_havens, replace
299
300 use exhibit21_v2, clear
301 sort cik fyear
302 by cik fyear: keep if _n==1

```



```

303   gen total_for_sub = totalcount
304   replace total_for_sub = 0 if totalcount == .
305   gen for_sub_ind = 0
306   replace for_sub_ind = 1 if total_for_sub > 0
307   keep cik fyear total_for_sub for_sub_ind
308   merge 1:1 cik fyear using exhibit21_havens
309   drop _merge
310   gen str10 cik2 = string(cik, "%010.0f")
311   drop cik
312   rename cik2 cik
313   save exhibit21_summary, replace
314
315   use compustat_all_segments, clear
316   merge 1:1 cik fyear using exhibit21_summary
317   drop if _merge == 2
318   drop _merge
319   save compustat_seg_ex21, replace
320
321   /*ADD IN QUARTERLY DATA FOR INTER-TEMPORAL SHIFTING TESTS (H1)*/
322   use compustat_quarterly_data, clear
323   rename fyearq fyear
324   sort gvkey fyear fqtr
325   by gvkey fyear fqtr: keep if _n==1
326   keep if fic == "USA"
327   gen yr_qtr=fyear+((fqtr-1)/100)
328   replace yr_qtr=fyear+.25 if fqtr == 2
329   replace yr_qtr=fyear+.50 if fqtr == 3
330   replace yr_qtr=fyear+.75 if fqtr == 4
331   by gvkey: gen atq_lag = atq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
332   by gvkey: gen cogsq_lag = cogsq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
333   by gvkey: gen cshoq_lag = cshoq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
334   by gvkey: gen dlcq_lag = dlcq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
335   by gvkey: gen dlттq_lag = dlттq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
336   by gvkey: gen doq_lag = doq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
337   by gvkey: gen invтq_lag = invтq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
338   by gvkey: gen oiadpq_lag = oiadpq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
339   by gvkey: gen saleq_lag = saleq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
340   by gvkey: gen seqq_lag = seqq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
341   by gvkey: gen spiq_lag = spiq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
342   by gvkey: gen xidoq_lag = xidoq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
343   by gvkey: gen xiq_lag = xiq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
344   by gvkey: gen xrdq_lag = xrdq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
345   by gvkey: gen xsgaq_lag = xsgaq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
346   by gvkey: gen oancfy_lag = oancfy[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
347   by gvkey: gen piy_lag = piy[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
348   by gvkey: gen prccq_lag = prccq[_n-1] if yr_qtr==yr_qtr[_n-.25]+.25
349   by gvkey: gen cogsq_4lag = cogsq[_n-4] if yr_qtr==yr_qtr[_n-4]+1
350   by gvkey: gen saleq_4lag = saleq[_n-4] if yr_qtr==yr_qtr[_n-4]+1
351   by gvkey: gen xrdq_4lag = xrdq[_n-4] if yr_qtr==yr_qtr[_n-4]+1
352   by gvkey: gen xsgaq_4lag = xsgaq[_n-4] if yr_qtr==yr_qtr[_n-4]+1
353   by gvkey: gen cogsq_5lag = cogsq[_n-5] if yr_qtr==yr_qtr[_n-5]+1.25
354   by gvkey: gen saleq_5lag = saleq[_n-5] if yr_qtr==yr_qtr[_n-5]+1.25
355   by gvkey: gen xrdq_5lag = xrdq[_n-5] if yr_qtr==yr_qtr[_n-5]+1.25
356   by gvkey: gen xsgaq_5lag = xsgaq[_n-5] if yr_qtr==yr_qtr[_n-5]+1.25
357   save quarterly_data, replace
358
359   use compustat_seg_ex21, clear
360   merge 1:m gvkey fyear using quarterly_data
361   drop if _merge==2
362   drop _merge
363   drop if fyear < 1997
364   drop if fyear > 2013
365   save comp_seg_ex21_quarter, replace
366
367   /*CONSTRUCT H1 CONTROL VARIABLES*/
368   use comp_seg_ex21_quarter, clear
369   drop if at <= 0
370   drop if missing(at)
371   drop if pi <= 0
372   drop if missing(pi)

```

```

373 drop if at_lag <= 0
374 drop if missing(at_lag)
375 drop if missing(prcc_f_lag)
376 drop if prcc_f_lag == 0
377 drop if csho_lag == 0
378 drop if missing(csho_lag)
379 drop if seq_lag <= 0
380 drop if missing(seq_lag)
381 drop if missing(oiadp_lag)
382 drop if sale <= 0
383 drop if missing(sale)
384 drop if sale_lag <= 0
385 drop if missing(sale_lag)
386 drop if missing(dltt_lag)
387 drop if missing(dlc_lag)
388 drop if missing(pi_lag)
389 gen size=log(at_lag)
390 gen mtb = (prcc_f_lag*csho_lag)/seq_lag
391 gen roa = oiadp_lag/at_lag
392 gen roa2 = pi_lag/at_lag
393 gen sales_growth = (sale-sale_lag)/sale_lag
394 gen lev = (dltt_lag+dlc_lag)/(csho_lag*prcc_f_lag)
395 gen nol_ind = 1
396 replace nol_ind = . if missing(tlcf_lag)
397 replace nol_ind = 0 if tlcf_lag == 0
398 gen nol_ind2 = 1
399 replace nol_ind2 = 0 if missing(tlcf_lag)
400 replace nol_ind2 = 0 if tlcf_lag == 0
401
402 /*CONSTRUCT H1 SHIFTING VARIABLES*/
403 drop if saleq < 0
404 drop if missing(saleq)
405 drop if cogsq < 0
406 drop if missing(cogsq)
407 drop if xsgaq < 0
408 drop if xrdq < 0
409 gen xsgaq_zero = xsgaq
410 replace xsgaq_zero = 0 if missing(xsgaq)
411 gen xrdq_zero = xrdq
412 replace xrdq_zero = 0 if missing(xrdq)
413 gen gross_margin = saleq-cogsq
414 gen op_income = gross_margin-xsgaq
415 gen op_income_sgard_zero = gross_margin-xsgaq_zero-xrdq_zero /*sets missing sga and rd to
zero*/
416
417 drop if saleq_lag < 0
418 drop if missing(saleq_lag)
419 drop if cogsq_lag < 0
420 drop if missing(cogsq_lag)
421 drop if xsgaq_lag < 0
422 drop if xrdq_lag < 0
423 gen xsgaq_zero_lag = xsgaq_lag
424 replace xsgaq_zero_lag = 0 if missing(xsgaq_lag)
425 gen xrdq_zero_lag = xrdq_lag
426 replace xrdq_zero_lag = 0 if missing(xrdq_lag)
427 gen gross_margin_lag = saleq_lag-cogsq_lag
428 gen op_income_lag = gross_margin_lag-xsgaq_lag
429 gen op_income_sgard_zero_lag = gross_margin_lag-xsgaq_zero_lag-xrdq_zero_lag
430
431 gen shift_gross_margin = ((gross_margin-gross_margin_lag)/2)/at_lag
432 gen shift_gross_margin_pct = shift_gross_margin*100
433 gen shift_op_income = ((op_income-op_income_lag)/2)/at_lag
434 gen shift_op_income_pct = shift_op_income*100
435 gen shift_op_income_sgard_zero = ((op_income_sgard_zero-op_income_sgard_zero_lag)/2)/at_lag
436 gen shift_op_income_sgard_zero_pct = shift_op_income_sgard_zero*100
437
438 /*ABBREVIATED CODE: CREATE INDICATOR "TEST QUARTER" TO IDENTIFY QUARTERS FOR SHIFTING
439 *CODE FOR EACH YEAR 1997 THROUGH 2013; 1997 PROVIDED AS EXAMPLE*/
440 replace test_quarter = 1997 if fyear == 1997 & fyr == 1 & fqtr == 1
441 replace test_quarter = 1997 if fyear == 1997 & fyr == 2 & fqtr == 1

```



```

442 replace test_quarter = 1997 if fyear == 1997 & fyr == 3 & fqtr == 1
443 replace test_quarter = 1997 if fyear == 1997 & fyr == 4 & fqtr == 1
444 replace test_quarter = 1997 if fyear == 1997 & fyr == 5 & fqtr == 1
445 replace test_quarter = 1997 if fyear == 1997 & fyr == 6 & fqtr == 1
446 replace test_quarter = 1997 if fyear == 1997 & fyr == 12 & fqtr == 1
447 replace test_quarter = 1997 if fyear == 1998 & fyr == 7 & fqtr == 1
448 replace test_quarter = 1997 if fyear == 1998 & fyr == 8 & fqtr == 1
449 replace test_quarter = 1997 if fyear == 1998 & fyr == 9 & fqtr == 1
450 replace test_quarter = 1997 if fyear == 1998 & fyr == 10 & fqtr == 1
451 replace test_quarter = 1997 if fyear == 1998 & fyr == 11 & fqtr == 1
452 drop if test_quarter==0
453 gen treat_quarter = 0
454 replace treat_quarter = 1 if test_quarter==2005
455 replace treat_quarter = 2 if test_quarter==2007
456 replace treat_quarter = 3 if test_quarter==2010
457
458 /*MERGE HAND-COLLECTED NOL DATA*/
459 merge m:1 gvkey fyear using nol_data
460 drop if _merge==2
461 drop _merge
462 save all_data_hlvar_qtr, replace
463
464 /*FILL IN INDUSTRY INFORMATION IF AVAILABLE IN DIFFERENT PERIOD*/
465 use all_data_hlvar_qtr, clear
466 drop if missing(sich)
467 sort gvkey
468 by gvkey: keep if _n==1
469 keep gvkey sich
470 rename sich sich1
471 save sich1, replace
472 use all_data_hlvar_qtr, clear
473 merge m:1 gvkey using sich1
474 drop _merge
475 gen sich_final = sich
476 replace sich_final = sich1 if missing(sich)
477 drop sich1
478 drop if missing(sich_final)
479 nsplit sich_final, digits(1) generate(sic)
480 drop sic2 sic3 sic4
481 nsplit sich_final, digits(2) generate(sic2_)
482 drop sic2_2
483 rename sic2_1 sic2
484 save all_data_sich1, replace
485
486 /*REQUIRE FIRMS TO HAVE GREATER THAN $100M IN ASSETS AS OF YEAR PRIOR TO DPAD*/
487 use all_data_sich1, clear
488 drop if fyear < 2003
489 collapse (mean) avgat=at, by(gvkey)
490 gen at100m = 1
491 replace at100m = 0 if avgat < 100
492 gen at_cat = 1
493 replace at_cat = 2 if avgat >= 100
494 replace at_cat = 3 if avgat >= 250
495 replace at_cat = 4 if avgat >= 500
496 replace at_cat = 5 if avgat >= 1000
497 keep gvkey at100m at_cat
498 save 100m_indicator, replace
499 use all_data_sich1, clear
500 merge m:1 gvkey using 100m_indicator
501 drop _merge
502 drop if at100m == 0
503 sort gvkey fyear fqtr
504 by gvkey fyear fqtr: keep if _n==1
505 save all_data_hlvar_100m, replace
506
507 /*CREATE INDICATOR FOR 199 FIRM*/
508 use all_data_hlvar_100m, clear
509 drop if missing(match_text)
510 sort gvkey
511 by gvkey: keep if _n==1

```

```

512 gen DPADfirm = 1
513 keep gvkey DPADfirm
514 save DPAD_id, replace
515 use all_data_hlvar_100m, clear
516 merge m:1 gvkey using DPAD_id
517 drop _merge
518 replace DPADfirm = 0 if missing(DPADfirm)
519 gen DPADfirmyear = 0
520 replace DPADfirmyear = 1 if DPADfirm == 1 & match_text == 1
521 save all_data_hlvar_100m_dpadind, replace
522
523
524 *****
525 *** Step 2 - Mahalanobis Matching with Replacement on size and performance in year prior
526 to the DPAD, as well as exact match on industry and year
527 *****/
528 /*MATCH ON SIZE, PERFORMANCE, INDUSTRY, AND YEAR USING LAGGED SIZE/PERFORMANCE VALUES*/
529 use all_data_hlvar_100m_dpadind, clear
530 tostring fyear, generate(fyear2)
531 gen ID_YR = gvkey + fyear2
532 save match, replace
533 use match, clear
534 drop if DPADfirmyear == 0
535 sort gvkey
536 by gvkey: keep if _n==1
537 save match1_pretreated, replace
538 use match, clear
539 drop if DPADfirm == 1
540 save match1_precontrol, replace
541 use match1_pretreated, replace
542 append using match1_precontrol
543 save prematch1, replace
544
545 mahapick size roa, idvar(ID_YR) treated(DPADfirm) genfile(maha_match1) replace prime_id(
546 DPADfirm) matchnum(NO_MATCHES) nummatches(1) matchon(sic1 fyear)
547
548 use maha_match1, clear
549 drop if NO_MATCHES == 0
550 gen gvkey = substr(ID_YR, 1, 6)
551 gen fyear = substr(ID_YR, 7, 4)
552 sort gvkey fyear
553 save match1_control, replace
554
555 use match1_control, clear
556 gen obs = 1
557 collapse (sum) obs, by (gvkey)
558 tab obs
559 save match2_control, replace
560
561 /*IDENTIFY FIRST YEAR IN SAMPLE FOR CONTROL FIRMS*/
562 use match1_control, clear
563 merge m:1 gvkey using match2_control
564 drop _merge
565 sort gvkey fyear
566 gen fyear_number = real(fyear)
567 drop fyear
568 rename fyear_number fyear
569 sort gvkey fyear
570 by gvkey: gen n1 = _n
571 keep gvkey obs fyear n1
572 reshape wide fyear, i(gvkey) j(n1)
573 by gvkey: keep if _n==1
574 gen CONTROL = 1
575 save match3_control, replace
576
577 /*IDENTIFY THE TREATED FIRMS FOR DATASET OF DPAD AND MATCHED CONTROLS*/
578 use match1_control, clear

```

```

578 drop ID_YR NO_MATCHES gvkey fyear
579 gen gvkey = substr(DPADfirm, 1, 6)
580 gen fyear = substr(DPADfirm, 7, 4)
581 gen DPAD_FIRSTYEAR = real(fyear)
582 gen DPAD_T2 = DPAD_FIRSTYEAR - 1
583 gen DPAD_T3 = DPAD_FIRSTYEAR - 2
584 sort gvkey
585 by gvkey: keep if _n==1
586 keep gvkey DPAD_FIRSTYEAR DPAD_T2 DPAD_T3
587 save match1_treated, replace
588
589 use match, clear
590 drop fyear2
591 sort gvkey
592 merge m:1 gvkey using match3_control
593 drop _merge
594 merge m:1 gvkey using match1_treated
595 drop _merge
596
597 drop if DPADfirm == 0 & missing(CONTROL)
598 replace CONTROL = 0 if missing(CONTROL)
599
600 /*EXPAND CONTROL FIRMS SINCE MATCHED WITH REPLACEMENT*/
601 expand obs
602 sort gvkey fyear
603 by gvkey fyear: generate n1 = _n
604 gen repeat_gvkey = gvkey
605 replace repeat_gvkey = gvkey + "_n2" if n1 == 2
606 replace repeat_gvkey = gvkey + "_n3" if n1 == 3
607 replace repeat_gvkey = gvkey + "_n4" if n1 == 4
608 replace repeat_gvkey = gvkey + "_n5" if n1 == 5
609 replace repeat_gvkey = gvkey + "_n6" if n1 == 6
610 replace repeat_gvkey = gvkey + "_n7" if n1 == 7
611 replace repeat_gvkey = gvkey + "_n8" if n1 == 8
612 gen CONTROL_FIRSTYEAR = fyear1
613 replace CONTROL_FIRSTYEAR = fyear2 if n1 == 2
614 replace CONTROL_FIRSTYEAR = fyear3 if n1 == 3
615 replace CONTROL_FIRSTYEAR = fyear4 if n1 == 4
616 replace CONTROL_FIRSTYEAR = fyear5 if n1 == 5
617 replace CONTROL_FIRSTYEAR = fyear6 if n1 == 6
618 replace CONTROL_FIRSTYEAR = fyear7 if n1 == 7
619 replace CONTROL_FIRSTYEAR = fyear8 if n1 == 8
620 gen CONTROL_T2 = CONTROL_FIRSTYEAR - 1
621 gen CONTROL_T3 = CONTROL_FIRSTYEAR - 2
622
623 gen first_year_ind = 0
624 replace first_year_ind = 1 if fyear == DPAD_FIRSTYEAR
625 replace first_year_ind = 1 if fyear == CONTROL_FIRSTYEAR
626 gen postdpad = 0
627 replace postdpad = 1 if DPADfirm == 1 & fyear >= DPAD_FIRSTYEAR
628 replace postdpad = 1 if CONTROL == 1 & fyear >= CONTROL_FIRSTYEAR
629 save matched_sample, replace
630
631 *****
632 *** Step 3 - Create dataset for tests of Intertemporal Shifting (H1)
633 *****
634
635 /*CREATE R&D TAX CREDIT CONTROL VARIABLE: FIRST USING DATA FROM JEFF HOOPES (THROUGH 2010)*/
636 use rdchand, clear
637 rename year fyear
638 keep cik rcxm fyear
639 gen new_cik = string(cik, "%010.0f")
640 drop cik
641 rename new_cik cik
642 sort cik
643 by cik: gen py_rdc = rcxm[_n-1]
644 by cik: gen py_rdc2 = rcxm[_n-2]
645 gen sum_rd = rcxm+py_rdc+py_rdc2

```

```

646 replace sum_rd = rcxm+py_rdc if missing(py_rdc2)
647 replace sum_rd = rcxm if missing(py_rdc2) & missing(py_rdc)
648 drop if fyear == 2011
649 save rd_credit, replace
650
651 /*ESTIMATE R&D TAX CREDIT FOR 2011 THROUGH 2013*/
652 use matched_sample, clear
653 merge m:1 cik fyear using rd_credit
654 drop if _merge == 2
655 drop _merge
656 gen fullpostpd_interaction = postdpad*DPADfirm
657
658 sort repeat_gvkey fyear
659 by repeat_gvkey: gen py_txfed = txfed[_n-1]
660 by repeat_gvkey: gen py_txfed2 = txfed[_n-2]
661 gen sum_tx = txfed+py_txfed+py_txfed2
662 replace sum_tx = txfed+py_txfed if missing(py_txfed2)
663 replace sum_tx = txfed if missing(py_txfed) & missing(py_txfed2)
664 gen avg_rd = ((xrd_lag+xrd_lag2+xrd_lag3)/3)*.50
665 replace avg_rd = 0 if missing(avg_rd)
666 gen calc_RD_credit = (xrd - avg_rd)*.14
667 replace calc_RD_credit = 0 if calc_RD_credit < 0
668 sort repeat_gvkey fyear
669 by repeat_gvkey: gen calc_RD_credit_lag = calc_RD_credit[_n-1]
670 replace calc_RD_credit_lag = . if calc_RD_credit_lag < 0
671 by repeat_gvkey: gen calc_RD_credit_lag2 = calc_RD_credit[_n-2]
672 replace calc_RD_credit_lag2 = 0 if calc_RD_credit_lag2 < 0
673 gen rd_credit = rcxm
674 replace rd_credit = calc_RD_credit if fyear == 2011 | fyear == 2012 | fyear == 2013
675 gen rd_credit_lag = py_rdc
676 replace rd_credit_lag = calc_RD_credit_lag if fyear == 2011 | fyear == 2012 | fyear == 2013
677 gen rd_credit_lag2 = py_rdc2
678 replace rd_credit_lag2 = calc_RD_credit_lag2 if fyear == 2011 | fyear == 2012 | fyear == 2013
679 gen rd_credit_ratio = (rcxm+py_rdc+py_rdc2)/(txfed+py_txfed+py_txfed2)
680 replace rd_credit_ratio = . if rd_credit_ratio < 0
681 gen rd_credit_ind = 1
682 replace rd_credit_ind = 0 if missing(rd_credit_ratio)
683 sort repeat_gvkey fyear
684 by repeat_gvkey: gen rd_credit_ind_lag = rd_credit_ind[_n-1] if fyear==fyear[_n-1]+1
685 by repeat_gvkey: gen rd_credit_ind_lag2 = rd_credit_ind[_n-2] if fyear==fyear[_n-2]+2
686 save timeshiftdiff1, replace
687
688 /*WINSORIZE & FIXED EFFECTS*/
689 use timeshiftdiff1, clear
690 xi i.fyear i.sic1 i.sic2
691
692 capture program drop clean2
693 program define clean2
694 sort fyear
695 by fyear: egen p99 = pctlile (`1'), p(99)
696 by fyear: egen p1 = pctlile (`1'), p(1)
697 replace `1'=p99 if `1'>p99&`1'~= .
698 replace `1'=p1 if `1'<p1&`1'~= .
699 drop p99 p1
700 end
701 clean2 shift_gross_margin_pct
702 clean2 shift_op_income_pct
703 clean2 shift_op_income_sgard_zero_pct
704 clean2 size
705 clean2 mtb
706 clean2 roa2
707 clean2 roa
708 clean2 sales_growth
709 clean2 lev
710 clean2 lev_at
711 save timeshiftdiff3, replace
712
713 /*IDENTIFY THE CORRECT TREATMENT YEARS FOR EACH DPAD YEAR - EXAMPLES PROVIDED FOR 2004,
2007, AND 2010*/
714 use timeshiftdiff3, clear

```

```

715 keep if DPAD_FIRSTYEAR == 2004 | CONTROL_FIRSTYEAR == 2004
716 gen treat_yr = 0
717 replace treat_yr = 1 if fyear == 2004 | fyear == 2007 | fyear == 2010
718 gen DPAD_treatyr_interact = 0
719 replace DPAD_treatyr_interact = DPADfirm*treat_yr if DPADfirm == 1
720 save hlpooled2_2004, replace
721
722 use timeshiftdiff3, clear
723 keep if DPAD_FIRSTYEAR == 2007 | CONTROL_FIRSTYEAR == 2007
724 tab fyear
725 gen treat_yr = 0
726 replace treat_yr = 1 if fyear == 2007 | fyear == 2010
727 gen DPAD_treatyr_interact = 0
728 replace DPAD_treatyr_interact = DPADfirm*treat_yr if DPADfirm == 1
729 save hlpooled2_2007, replace
730
731 use timeshiftdiff3, clear
732 keep if DPAD_FIRSTYEAR == 2010 | CONTROL_FIRSTYEAR == 2010
733 tab fyear
734 gen treat_yr = 0
735 replace treat_yr = 1 if fyear == 2010
736 gen DPAD_treatyr_interact = 0
737 replace DPAD_treatyr_interact = DPADfirm*treat_yr if DPADfirm == 1
738 save hlpooled2_2010, replace
739
740 use hlpooled2_2004, clear
741 foreach num of numlist 2005/2013 {
742 append using hlpooled2_`num'
743 }
744
745 /*CREATE INDICATORS BASED ON PERIOD IN WHICH INITIALLY DISCUSS THE DPAD*/
746 gen treat_yr_05_06 = 0
747 replace treat_yr_05_06 = 1 if fyear == 2004 & DPAD_FIRSTYEAR == 2004 | fyear == 2004 &
CONTROL_FIRSTYEAR == 2004
748 replace treat_yr_05_06 = 1 if fyear == 2005 & DPAD_FIRSTYEAR == 2005 | fyear == 2005 &
CONTROL_FIRSTYEAR == 2005
749 replace treat_yr_05_06 = 1 if fyear == 2006 & DPAD_FIRSTYEAR == 2006 | fyear == 2006 &
CONTROL_FIRSTYEAR == 2006
750 gen treat_yr_07_09 = 0
751 replace treat_yr_07_09 = 1 if fyear == 2007 & DPAD_FIRSTYEAR == 2007 | fyear == 2007 &
CONTROL_FIRSTYEAR == 2007
752 replace treat_yr_07_09 = 1 if fyear == 2008 & DPAD_FIRSTYEAR == 2008 | fyear == 2008 &
CONTROL_FIRSTYEAR == 2008
753 replace treat_yr_07_09 = 1 if fyear == 2009 & DPAD_FIRSTYEAR == 2009 | fyear == 2009 &
CONTROL_FIRSTYEAR == 2009
754 replace treat_yr_07_09 = 1 if fyear == 2007 & DPAD_FIRSTYEAR == 2004 | fyear == 2007 &
CONTROL_FIRSTYEAR == 2004
755 replace treat_yr_07_09 = 1 if fyear == 2007 & DPAD_FIRSTYEAR == 2005 | fyear == 2007 &
CONTROL_FIRSTYEAR == 2005
756 replace treat_yr_07_09 = 1 if fyear == 2007 & DPAD_FIRSTYEAR == 2006 | fyear == 2007 &
CONTROL_FIRSTYEAR == 2006
757 gen treat_yr_10_13 = 0
758 replace treat_yr_10_13 = 1 if fyear == 2010 & DPAD_FIRSTYEAR == 2010 | fyear == 2010 &
CONTROL_FIRSTYEAR == 2010
759 replace treat_yr_10_13 = 1 if fyear == 2011 & DPAD_FIRSTYEAR == 2011 | fyear == 2011 &
CONTROL_FIRSTYEAR == 2011
760 replace treat_yr_10_13 = 1 if fyear == 2012 & DPAD_FIRSTYEAR == 2012 | fyear == 2012 &
CONTROL_FIRSTYEAR == 2012
761 replace treat_yr_10_13 = 1 if fyear == 2013 & DPAD_FIRSTYEAR == 2013 | fyear == 2013 &
CONTROL_FIRSTYEAR == 2013
762 replace treat_yr_10_13 = 1 if fyear == 2010 & DPAD_FIRSTYEAR == 2004 | fyear == 2010 &
CONTROL_FIRSTYEAR == 2004
763 replace treat_yr_10_13 = 1 if fyear == 2010 & DPAD_FIRSTYEAR == 2005 | fyear == 2010 &
CONTROL_FIRSTYEAR == 2005
764 replace treat_yr_10_13 = 1 if fyear == 2010 & DPAD_FIRSTYEAR == 2006 | fyear == 2010 &
CONTROL_FIRSTYEAR == 2006
765 replace treat_yr_10_13 = 1 if fyear == 2010 & DPAD_FIRSTYEAR == 2007 | fyear == 2010 &
CONTROL_FIRSTYEAR == 2007
766 replace treat_yr_10_13 = 1 if fyear == 2010 & DPAD_FIRSTYEAR == 2008 | fyear == 2010 &
CONTROL_FIRSTYEAR == 2008

```

```

767 replace treat_yr_10_13 = 1 if fyear == 2010 & DPAD_FIRSTYEAR == 2009 | fyear == 2010 &
CONTROL_FIRSTYEAR == 2009
768 gen DPAD_05_06 = 0
769 replace DPAD_05_06 = DPADfirm*treat_yr_05_06
770 gen DPAD_07_09 = 0
771 replace DPAD_07_09 = DPADfirm*treat_yr_07_09
772 gen DPAD_10_13 = 0
773 replace DPAD_10_13 = DPADfirm*treat_yr_10_13
774 save h1_dataset, replace
775
776
777 *****
778 *** Step 4 - Create dataset for tests of Cross-border Shifting (H2)
779 *****/
780
781 /*OBTAIN SEGMENT AND COMPUSTAT DATA TO CONSTRUCT 2- AND 5- YEAR INCOME SHIFTING MEASURES*/
782
783 use segment_data_2013_2017, clear
784 sort gvkey datadate sid srcdate
785 by gvkey datadate sid: keep if _n==1
786 drop if geotp == "1"
787 gen fyear = year(datadate)
788 tab fyear
789 drop if geotp == "3"
790
791 drop if snms=="Africa" | snms=="Anguilla" | snms=="Antigua" | snms=="Antigua and Barbuda" |
snms=="Antigua, West Indies" | snms=="Argentina" | snms=="Asia" | snms=="Asia Pacific" |
snms=="Asia and Others" | snms=="Asia pacific" | snms=="Asia,Pacific,Australia" | snms==
"Australasia"
792 drop if snms=="Australia" | snms=="Australia & New Zealand" | snms=="Australia & S.E. Asia"
| snms=="Australia and New Zealand" | snms=="Australia, Asia and Pacific" | snms==
"Australia, New Zealand"
793 drop if snms=="Austria" | snms=="Bahamas" | snms=="Barbados" | snms=="Belgium" | snms==
"Belize" | snms=="Bermuda" | snms=="Brazil" | snms=="Brazil and Others" | snms=="British
Columbia"
794 drop if snms=="British Virgin Islands" | snms=="Canada" | snms=="Canada and CALA" | snms==
"Canada and Other" | snms=="Canada and Other Foreign" | snms==
"Canada/Mexico/Caribbean/South America" | snms=="Canada/Mexico/Caribbean"
795 drop if snms=="Canada/Other Americas/Caribbean" | snms=="Canadian" | snms=="Caribbean" |
snms=="Cayman Islands" | snms=="Cayman Islands/Bermuda" | snms=="Central" | snms=="Central
America & Caribbean"
796 drop if snms=="Cananda" | snms=="Central and Eastern Europe, Middle East and Africa" | snms
=="Central and Northern Europe" | snms == "Channel Islands" | snms=="Chile" | snms=="Chile
& Other" | snms=="China" | snms=="China/Hong Kong"
797 drop if snms=="Continental Europe" | snms=="Curacao" | snms=="Cyprus" | snms=="Denmark" |
snms=="Dominican Republic" | snms=="Dutch Antilles" | snms=="El Sitio Inc" | snms=="Euro
Area"
798 drop if snms=="Europe" | snms=="Europe & AME" | snms=="Europe & Central Asia" | snms==
"Europe & Middle East" | snms=="Europe Region" | snms=="Europe and Australia" | snms==
"Europe and Middle East"
799 drop if snms=="Europe and Other" | snms=="Europe and Pacific" | snms=="Europe regions" |
snms=="Europe, Africa & Middle East" | snms=="Europe, Africa, Middle East" | snms=="Europe,
CIS & West Africa"
800 drop if snms=="Europe, Middle East & Africa" | snms=="Europe, Middle East and Africa" | snms
=="Europe, Middle East and Asia" | snms=="Europe, Middle-East and Africa" | snms=="Europe,
North Africa"
801 drop if snms=="Europe, including France" | snms=="Europe/ West Africa/ FSU" | snms==
"Europe/Australia" | snms=="Europe/Island of Guernsey" | snms=="Europe/SSA/Russia" | snms==
"European Countries"
802 drop if snms=="European Russia" | snms=="European Union" | snms=="Excise Taxes" | snms==
"FR, DE, NL and UK" | snms=="Far East and Other" | snms=="Finland" | snms == "Foreign
Countries" | snms=="France" | snms=="France/Western Europe"
803 drop if snms=="Germany" | snms=="Germany and Ireland" | snms=="Ghana" | snms=="Great
Britain" | snms=="Greater China" | snms=="Greece" | snms=="Hong Kong" | snms=="Hong Kong
and Mainland China"
804 drop if snms=="Hong kong" | snms=="Hungary" | snms=="India" | snms=="India & Other" | snms==
"Indonesia" | snms=="International" | snms=="Ireland" | snms=="Israel" | snms=="Israel and
Other"

```



```

805 drop if snms=="Israel and Others" | snms=="Isreal and Others" | snms=="Isreal" | snms=="
  "Italy" | snms=="Italy & Other European Countries" | snms=="Italy/Europe" | snms=="Japan" |
  snms=="Jersey" | snms=="Jersey, Channel Islands"
806 drop if snms=="Korea" | snms=="Latin America" | snms=="Latin America North" | snms=="
  "Liberia" | snms=="Luxembourg" | snms=="Mainland China" | snms=="Malaysia & Singapore" |
  snms=="Marshall Islands"
807 drop if snms=="Mauritius" | snms=="Metropolitan France" | snms=="Mexico" | snms=="Mexico
  and Central America" | snms=="Middle East, Central Europe & Other" | snms=="Moscow" | snms=="
  "Moscow License Area"
808 drop if snms=="NALA" | snms=="NASA" | snms=="NOLAD" | snms=="Netherlands" | snms=="
  "Netherlands Antilles" | snms=="Netherlands Antillies" | snms=="New Zealand" | snms=="
  "Non-U.S. or Non-U.K." | snms=="Non-US" | snms=="Non-US Americas"
809 drop if snms=="Non-operating" | snms=="Nordic" | snms=="Nordic Countries" | snms=="Nordic
  Region" | snms=="North & Central Europe" | snms=="North Region" | snms=="North West Europe"
  | snms=="Northern Europe"
810 drop if snms=="Norway" | snms=="Other Countries" | snms=="Other European Countries" | snms=="
  "Other Foreign" | snms=="Other Geographical Areas" | snms=="Other Markets" | snms=="Other
  Regions"
811 drop if snms=="PRC" | snms=="Panama" | snms=="Papua New Guinea" | snms=="People's Republic
  of China" | snms=="Peoples Republic of China" | snms=="The Peoples Republic of China (PRC)"
  | snms=="Peru" | snms=="Philippines" | snms=="Phillippines"
812 drop if snms=="Poland" | snms=="Portugal" | snms=="RUSSIA" | snms=="Republic of Ireland" |
  snms=="Republic of Korea" | snms=="Republic of Liberia" | snms=="Republic of Panama"
813 drop if snms=="Republic of The Marshall Islands" | snms=="Republic of the Marshall Islands"
  | snms=="Rest of Europe" | snms=="Rest of World" | snms=="Russia" | snms=="Russian
  Federation"
814 drop if snms=="Scandinavia" | snms=="Singapore" | snms=="South Africa" | snms=="South
  America" | snms=="South Korea" | snms=="Southern Africa" | snms=="Spain" | snms=="Spain &
  Portugal"
815 drop if snms=="Sweden" | snms=="Switzerland" | snms=="Taiwan" | snms=="Taiwan, ROC" | snms=="
  "Thailand" | snms=="The Netherlands" | snms=="The PRC" | snms=="The People's Republic of
  China"
816 drop if snms=="The Peoples Republic of China: Hong Kong" | snms=="Turkey" | snms=="UK" |
  snms=="UK & Ireland" | snms=="UK and Ireland" | snms=="Ukraine" | snms=="United Kingdom" |
  snms=="United Kingdom & Denmark"
817 drop if snms=="United Kingdom & Europe" | snms=="United Kingdom & Ireland" | snms=="United
  Kingdom and Europe" | snms=="United Kingdom and Ireland" | snms=="United Kingdom, Europe &
  Middle East"
818 drop if snms=="United Kingdom, Europe and Other" | snms=="United Kingdom, Europe and middle
  east" | snms=="United Kingdom, Ireland" | snms=="United Kingdom/Ireland" | snms=="United
  Kingdom/Republic of Ireland"
819 drop if snms=="Venezuela" | snms=="Western Canada" | snms=="Western Europe" | snms=="
  "Western Hemisphere" | snms=="Canada and Other" | snms=="Colombia" | snms=="EUROPE -
  Other" | snms=="England and Wales"
820 drop if snms=="Europe /Middle East / Africa" | snms=="Europe excl. CIS" | snms=="Europe,
  Israel and Japan" | snms=="Europe, Russia & Central Asia" | snms=="Iberia" | snms=="
  "Mainland of China"
821 drop if snms=="Northern Europe & CA" | snms=="Other EU Countries" | snms=="Republic of
  China (ROC)" | snms=="Switzerland." | snms=="The Cayman Islands"
822
823 bysort gvkey fyear: egen seq = seq()
824 foreach v in sales {
825     bysort gvkey fyear: egen c`v' = count(`v') if(`v'==.)
826     replace c`v' = 1 if(c`v'==0)
827     bysort gvkey fyear: egen c2`v' = count(c`v')
828     bysort gvkey fyear: egen sum`v' = sum(`v')
829     bysort gvkey fyear: egen mn`v' = mean(`v')
830     replace sum`v' = -99 if(mn`v' == .)
831     replace sum`v' = . if(seq > 1)
832     replace sum`v' = -99 if (c2`v'~=0)
833     drop c`v' c2`v' mn`v' `v'
834     rename sum`v' `v'
835 }
836 drop seq
837
838 collapse (sum) DOMSEGSale=sales , by(gvkey fyear)
839
840 foreach v in DOMSEGSale {
841     replace `v' = . if(`v' < 0)
842 }

```

```

843
844     sort gvkey fyear
845     by gvkey fyear:  keep if _n==1
846     drop if missing(fyear)
847     save segment_domestic_13_17, replace
848
849     use compustat_data_2013_2017, clear
850     drop if fyear == 2012
851     drop if fyear == 2017
852     sort gvkey fyear
853     by gvkey fyear:  keep if _n==1
854     merge 1:1 gvkey fyear using segment_domestic_13_17
855     drop if _merge == 2
856     drop _merge
857     drop if missing(fyear)
858     by gvkey fyear:  keep if _n==1
859     save data_13_16_with_segment, replace
860
861     use compustat_all_segments, clear
862     append using data_13_16_with_segment
863     sort gvkey fyear
864     by gvkey fyear:  keep if _n==1
865     drop if missing(fyear)
866     keep gvkey fyear pidom DOMSEGSale pi sale xrd at che  dlтт dlc csho prcc_f pifo txfo
867     sort gvkey fyear
868     by gvkey:  gen pidom_1 = pidom[_n+1] if fyear==fyear[_n+1]-1
869     by gvkey:  gen pidom_2 = pidom[_n+2] if fyear==fyear[_n+2]-2
870     by gvkey:  gen pidom_3 = pidom[_n+3] if fyear==fyear[_n+3]-3
871     by gvkey:  gen pidom_4 = pidom[_n+4] if fyear==fyear[_n+4]-4
872
873     by gvkey:  gen DOMSEGSale_1 = DOMSEGSale[_n+1] if fyear==fyear[_n+1]-1
874     by gvkey:  gen DOMSEGSale_2 = DOMSEGSale[_n+2] if fyear==fyear[_n+2]-2
875     by gvkey:  gen DOMSEGSale_3 = DOMSEGSale[_n+3] if fyear==fyear[_n+3]-3
876     by gvkey:  gen DOMSEGSale_4 = DOMSEGSale[_n+4] if fyear==fyear[_n+4]-4
877
878     by gvkey:  gen pi_1 = pi[_n+1] if fyear==fyear[_n+1]-1
879     by gvkey:  gen pi_2 = pi[_n+2] if fyear==fyear[_n+2]-2
880     by gvkey:  gen pi_3 = pi[_n+3] if fyear==fyear[_n+3]-3
881     by gvkey:  gen pi_4 = pi[_n+4] if fyear==fyear[_n+4]-4
882
883     by gvkey:  gen sale_1 = sale[_n+1] if fyear==fyear[_n+1]-1
884     by gvkey:  gen sale_2 = sale[_n+2] if fyear==fyear[_n+2]-2
885     by gvkey:  gen sale_3 = sale[_n+3] if fyear==fyear[_n+3]-3
886     by gvkey:  gen sale_4 = sale[_n+4] if fyear==fyear[_n+4]-4
887
888     by gvkey:  gen xrd_1 = xrd[_n+1] if fyear==fyear[_n+1]-1
889     by gvkey:  gen xrd_2 = xrd[_n+2] if fyear==fyear[_n+2]-2
890     by gvkey:  gen xrd_3 = xrd[_n+3] if fyear==fyear[_n+3]-3
891     by gvkey:  gen xrd_4 = xrd[_n+4] if fyear==fyear[_n+4]-4
892
893     by gvkey:  gen atlag_1 = at
894     by gvkey:  gen atlag_2 = at[_n+1] if fyear==fyear[_n+1]-1
895     by gvkey:  gen atlag_3 = at[_n+2] if fyear==fyear[_n+2]-2
896     by gvkey:  gen atlag_4 = at[_n+3] if fyear==fyear[_n+3]-3
897
898     by gvkey:  gen che_1 = che
899     by gvkey:  gen che_2 = che[_n+1] if fyear==fyear[_n+1]-1
900     by gvkey:  gen che_3 = che[_n+2] if fyear==fyear[_n+2]-2
901     by gvkey:  gen che_4 = che[_n+3] if fyear==fyear[_n+3]-3
902     tab fyear
903
904     by gvkey:  gen at_1 = at[_n+1] if fyear==fyear[_n+1]-1
905     by gvkey:  gen at_2 = at[_n+2] if fyear==fyear[_n+2]-2
906     by gvkey:  gen at_3 = at[_n+3] if fyear==fyear[_n+3]-3
907     by gvkey:  gen at_4 = at[_n+4] if fyear==fyear[_n+4]-4
908
909     by gvkey:  gen dlтт_1 = dlтт
910     by gvkey:  gen dlтт_2 = dlтт[_n+1] if fyear==fyear[_n+1]-1
911     by gvkey:  gen dlтт_3 = dlтт[_n+2] if fyear==fyear[_n+2]-2
912     by gvkey:  gen dlтт_4 = dlтт[_n+3] if fyear==fyear[_n+3]-3

```

```

913
914 by gvkey: gen dlc_1 = dlc
915 by gvkey: gen dlc_2 = dlc[_n+1] if fyear==fyear[_n+1]-1
916 by gvkey: gen dlc_3 = dlc[_n+2] if fyear==fyear[_n+2]-2
917 by gvkey: gen dlc_4 = dlc[_n+3] if fyear==fyear[_n+3]-3
918
919 by gvkey: gen csho_1 = csho
920 by gvkey: gen csho_2 = csho[_n+1] if fyear==fyear[_n+1]-1
921 by gvkey: gen csho_3 = csho[_n+2] if fyear==fyear[_n+2]-2
922 by gvkey: gen csho_4 = csho[_n+3] if fyear==fyear[_n+3]-3
923
924 by gvkey: gen prcc_f_1 = prcc_f
925 by gvkey: gen prcc_f_2 = prcc_f[_n+1] if fyear==fyear[_n+1]-1
926 by gvkey: gen prcc_f_3 = prcc_f[_n+2] if fyear==fyear[_n+2]-2
927 by gvkey: gen prcc_f_4 = prcc_f[_n+3] if fyear==fyear[_n+3]-3
928
929 by gvkey: gen pifo_1 = pifo[_n+1] if fyear==fyear[_n+1]-1
930 by gvkey: gen pifo_2 = pifo[_n+2] if fyear==fyear[_n+2]-2
931 by gvkey: gen pifo_3 = pifo[_n+3] if fyear==fyear[_n+3]-3
932 by gvkey: gen pifo_4 = pifo[_n+4] if fyear==fyear[_n+4]-4
933
934 by gvkey: gen txfo_1 = txfo[_n+1] if fyear==fyear[_n+1]-1
935 by gvkey: gen txfo_2 = txfo[_n+2] if fyear==fyear[_n+2]-2
936 by gvkey: gen txfo_3 = txfo[_n+3] if fyear==fyear[_n+3]-3
937 by gvkey: gen txfo_4 = txfo[_n+4] if fyear==fyear[_n+4]-4
938
939 sort gvkey fyear
940 by gvkey fyear: keep if _n==1
941 drop if missing(fyear)
942 save future_ros_data, replace
943
944 /*CONSTRUCT VARIABLES FOR INCOME SHIFTING TESTS*/
945 use timeshiftdiff1, clear
946 drop pidom
947 sort repeat_gvkey fyear
948 by repeat_gvkey fyear: keep if _n==1
949 sort gvkey fyear
950 merge m:1 gvkey fyear using future_ros_data
951 keep if _merge == 3
952 drop _merge
953 save dataset_for_h2, replace
954
955 use dataset_for_h2, clear
956 gen future_pidom = pidom+pidom_1/*+pidom_2+pidom_3+pidom_4*/
957 gen future_domsegsale = DOMSEGSALE+DOMSEGSALE_1/*+DOMSEGSALE_2+DOMSEGSALE_3+DOMSEGSALE_4*/
958 gen future_pi = pi+pi_1/*+pi_2+pi_3+pi_4*/
959 gen future_sale = sale+sale_1/*+sale_2+sale_3+sale_4*/
960
961 drop if missing(future_pidom)
962 drop if future_pidom < 0
963 drop if missing(future_domsegsale)
964 drop if future_domsegsale <= 0
965 gen future_dom_ros = future_pidom/future_domsegsale
966
967 drop if missing(future_pi)
968 drop if missing(future_sale)
969 drop if future_sale == 0
970 gen future_ww_ros = future_pi/future_sale
971
972 replace xrd = 0 if missing(xrd)
973 replace xrd_1 = 0 if missing(xrd_1)
974 replace xrd_2 = 0 if missing(xrd_2)
975 replace xrd_3 = 0 if missing(xrd_3)
976 replace xrd_4 = 0 if missing(xrd_4)
977
978 gen future_xrd = xrd+xrd_1/*+xrd_2+xrd_3+xrd_4*/
979 gen future_lagasset = at_lag+atlag_1/*+atlag_2+atlag_3+atlag_4*/
980 gen future_rd = future_xrd/future_lagasset
981 gen future_che = che_lag+che_1/*+che_2+che_3+che_4*/
982 gen future_cash = future_che/future_lagasset

```

```

983 gen future_pct_dom = future_domsegsale/future_sale
984 gen future_asset = (at+at_1/*+at_2+at_3+at_4*/)/2
985 gen future_size = ln(future_asset)
986 gen future_debt = dlтт_lag+dlтт_1/*+dlтт_2+dlтт_3+dlтт_4*/+dlc_lag+dlc_1
/*+dlc_2+dlc_3+dlc_4*/
987 gen future_mkt_cap = (csho_lag*prcc_f_lag)+(csho_1*prcc_f_1)
/*+(csho_2*prcc_f_2)+(csho_3*prcc_f_3)+(csho_4*prcc_f_4)*/
988 gen future_lev = future_debt/future_mkt_cap
989 drop if missing(future_lev)
990 sort repeat_gvkey fyear
991 by repeat_gvkey fyear: keep if _n==1
992
993 gen future_pifo = pifo + pifo_1/* + pifo_2 + pifo_3 + pifo_4*/
994 drop if future_pifo < 0
995 drop if missing(future_pifo)
996 gen future_txfo = txfo + txfo_1/* + txfo_2 + txfo_3 + txfo_4*/
997 drop if missing(future_txfo)
998 gen future_for_etr = future_txfo/future_pifo
999 gen future_for_etr2 = future_for_etr
1000 replace future_for_etr2 = 1 if future_for_etr2 > 1
1001 gen future_rate_diff = .35-future_for_etr2
1002
1003 gen post_rate_diff = postdpad*future_rate_diff
1004 gen post_DPAD = postdpad*DPADfirm
1005 gen DPAD_rate = DPADfirm*future_rate_diff
1006 gen triple_interact = DPADfirm*postdpad*future_rate_diff
1007 save cbshift1_5yr, replace
1008
1009 /*REQUIRE TREATMENT-CONTROL PAIRS WITHIN SAMPLE*/
1010 use cbshift1_5yr, clear
1011 keep if DPADfirm == 1
1012 save cbshift1_DPAD_5yr, replace
1013
1014 use cbshift1_5yr, clear
1015 keep if DPADfirm == 0
1016 save cbshift1_CONTROL_5yr, replace
1017
1018 use match1_control, clear
1019 rename gvkey CONTROL_gvkey
1020 rename fyear CONTROL_fyear
1021 gen DPAD_gvkey = substr(DPADfirm, 1, 6)
1022 gen DPAD_fyear = substr(DPADfirm, 7, 4)
1023 sort DPAD_gvkey DPAD_fyear
1024 sort CONTROL_gvkey CONTROL_fyear
1025 by CONTROL_gvkey: gen n1 = _n
1026 gen repeat_gvkey = CONTROL_gvkey
1027 replace repeat_gvkey = CONTROL_gvkey + "_n2" if n1 == 2
1028 replace repeat_gvkey = CONTROL_gvkey + "_n3" if n1 == 3
1029 replace repeat_gvkey = CONTROL_gvkey + "_n4" if n1 == 4
1030 replace repeat_gvkey = CONTROL_gvkey + "_n5" if n1 == 5
1031 replace repeat_gvkey = CONTROL_gvkey + "_n6" if n1 == 6
1032 replace repeat_gvkey = CONTROL_gvkey + "_n7" if n1 == 7
1033 replace repeat_gvkey = CONTROL_gvkey + "_n8" if n1 == 8
1034 rename DPAD_gvkey gvkey
1035 rename DPAD_fyear match_year
1036 keep gvkey repeat_gvkey match_year
1037 save treat_control_pair, replace
1038
1039 use cbshift1_DPAD_5yr, clear
1040 sort gvkey fyear
1041 by gvkey: keep if _n==1
1042 keep gvkey fyear
1043 merge 1:1 gvkey using treat_control_pair
1044 keep if _merge == 3
1045 drop _merge
1046 rename gvkey treat_gvkey
1047 keep treat_gvkey repeat_gvkey
1048 gen control_h2 = 1
1049 save control_firms_for_h2_5yr, replace
1050

```

```

1051 use cbshift1_CONTROL_5yr, clear
1052 merge m:1 repeat_gvkey using control_firms_for_h2_5yr
1053 drop if _merge==2
1054 drop _merge
1055 keep if control_h2 == 1
1056 save control_firms_h2_5yr, replace
1057
1058 use control_firms_h2_5yr, clear
1059 keep treat_gvkey
1060 sort treat_gvkey
1061 by treat_gvkey: keep if _n==1
1062 gen treat_for_h2_5yr = 1
1063 rename treat_gvkey gvkey
1064 save id_treat_5yr, replace
1065
1066 use cbshift1_DPAD_5yr, clear
1067 merge m:1 gvkey using id_treat_5yr
1068 keep if _merge == 3
1069 append using control_firms_h2_5yr
1070 save cbshift1_final_5yr, replace
1071
1072 /*FIXED EFFECTS AND WINSORIZE*/
1073 xi i.fyear i.sic
1074
1075
1076 capture program drop clean2
1077 program define clean2
1078 sort fyear
1079 by fyear: egen p99 = pctlile (`1'), p(99)
1080 by fyear: egen p1 = pctlile (`1'), p(1)
1081 replace `1'=p99 if `1'>p99&`1'~=.
1082 replace `1'=p1 if `1'<p1&`1'~=.
1083 drop p99 p1
1084 end
1085
1086 clean2 future_dom_ros
1087 clean2 future_ww_ros
1088 clean2 future_rd
1089 clean2 future_cash
1090 clean2 future_pct_dom
1091 clean2 future_size
1092 clean2 future_lev
1093 clean2 post_rate_diff
1094 clean2 future_rate_diff
1095 clean2 post_DPAD
1096 clean2 DPAD_rate
1097 clean2 triple_interact
1098
1099 save cbshift2, replace
1100
1101
1102 *****
1103 *** Step 5 - Create dataset for tests of Domestic Investment (H3a)
1104 *****
1105
1106 /*MERGE BEA DATA WITH COMPUSTAT DATA*/
1107 use bea_dom_data_97_10, clear
1108 append using bea_dom_data_11_13
1109 drop for_capex
1110 rename period_year fyear
1111 sort gvkey fyear us_id
1112 by gvkey fyear: keep if _n==1
1113 by gvkey: gen dom_exp_ppe_lag = dom_exp_ppe[_n-1] if fyear==fyear[_n-1]+1
1114 save bea_dom_data, replace
1115
1116 use bea_for_data, clear
1117 rename period_year fyear
1118 sort gvkey fyear us_id

```

```

1119 by gvkey fyear: keep if _n==1
1120 keep us_id fyear for_capex for_rd gvkey
1121 save bea_for_data, replace
1122
1123 use timeshiftdiff1, clear
1124 merge m:1 gvkey fyear using bea_dom_data
1125 drop if _merge==2
1126 drop _merge
1127 drop for_rd
1128 merge m:1 gvkey fyear using bea_for_data
1129 drop if _merge==2
1130 drop _merge
1131 save merged_bea_dpad1, replace
1132
1133 use capex, clear
1134 drop if missing(fyear)
1135 sort gvkey fyear
1136 by gvkey fyear: keep if _n==1
1137 by gvkey: gen capx_lag = capx[_n-1] if fyear==fyear[_n-1]+1
1138 save capex2, replace
1139
1140 use merged_bea_dpad1, clear
1141 merge m:1 gvkey fyear using capex2
1142 drop _merge
1143 drop if missing(at)
1144 save merged_bea_dpad2, replace
1145
1146 use merged_bea_dpad2, clear
1147 gen bea_ind = 1
1148 replace bea_ind = 0 if missing(dom_ein_1)
1149 drop if bea_ind == 1 & missing(dom_exp_ppe)
1150 drop if bea_ind == 1 & missing(ppent_lag)
1151 drop if bea_ind == 1 & ppent_lag == 0
1152 drop if bea_ind == 0
1153 save bea_capex_sample, replace
1154
1155
1156 use merged_bea_dpad2, clear
1157 keep if missing(us_id)
1158 gen for_sub_ind2 = 1
1159 replace for_sub_ind2 = 0 if for_sub_ind == 0
1160 replace for_sub_ind2 = 0 if missing(for_sub_ind)
1161 gen for_sale_ind2 = 1
1162 replace for_sale_ind2 = 0 if for_sale_ind == 0
1163 replace for_sale_ind2 = 0 if missing(for_sale_ind)
1164 gen for_asset_ind2 = 1
1165 replace for_asset_ind2 = 0 if for_asset_ind == 0
1166 replace for_asset_ind2 = 0 if missing(for_asset_ind)
1167 gen mnc_ind = for_sub_ind2 + for_sale_ind2 + for_asset_ind2
1168
1169 drop if missing(capx)
1170 drop if missing(ppent_lag)
1171 drop if ppent_lag == 0
1172 save dom_only_sample, replace
1173
1174 use dom_only_sample, clear
1175 append using bea_capex_sample
1176 save capex_sample, replace
1177
1178 /*CREATE VARIABLES*/
1179 use capex_sample, clear
1180 gen dom_capex = .
1181 replace dom_capex = (dom_exp_ppe/1000)/ppent_lag if bea_ind == 1
1182 replace dom_capex = capx/ppent_lag if missing(bea_ind)
1183 gen ocf = oancf/at_lag
1184 gen cash = che_lag/at_lag
1185 drop if missing(ocf)
1186 drop if missing(cash)
1187 sort repeat_gvkey fyear
1188 by repeat_gvkey fyear: keep if _n==1

```



```

1189   save invla, replace
1190
1191
1192   /*REQUIRE TREATMENT-CONTROL PAIRS WITHIN MNC AND DOMESTIC SAMPLE*/
1193
1194   use invla, clear
1195   keep if bea_ind == 1
1196   keep if DPADfirm == 1
1197   save invla_DPAD_BEA, replace
1198
1199   use invla, clear
1200   keep if bea_ind == 1
1201   keep if DPADfirm == 0
1202   save invla_CONTROL_BEA, replace
1203
1204   use match1_control, clear
1205   rename gvkey CONTROL_gvkey
1206   rename fyear CONTROL_fyear
1207   gen DPAD_gvkey = substr(DPADfirm, 1, 6)
1208   gen DPAD_fyear = substr(DPADfirm, 7, 4)
1209   sort CONTROL_gvkey CONTROL_fyear
1210   by CONTROL_gvkey: gen n1 = _n
1211   gen repeat_gvkey = CONTROL_gvkey
1212   replace repeat_gvkey = CONTROL_gvkey + "_n2" if n1 == 2
1213   replace repeat_gvkey = CONTROL_gvkey + "_n3" if n1 == 3
1214   replace repeat_gvkey = CONTROL_gvkey + "_n4" if n1 == 4
1215   replace repeat_gvkey = CONTROL_gvkey + "_n5" if n1 == 5
1216   replace repeat_gvkey = CONTROL_gvkey + "_n6" if n1 == 6
1217   replace repeat_gvkey = CONTROL_gvkey + "_n7" if n1 == 7
1218   replace repeat_gvkey = CONTROL_gvkey + "_n8" if n1 == 8
1219   rename DPAD_gvkey gvkey
1220   rename DPAD_fyear match_year
1221   keep gvkey repeat_gvkey match_year
1222   save treat_control_pair, replace
1223
1224   use invla_DPAD_BEA, clear
1225   sort gvkey fyear
1226   by gvkey: keep if _n==1
1227   keep gvkey fyear
1228   merge 1:1 gvkey using treat_control_pair
1229   keep if _merge == 3
1230   drop _merge
1231   rename gvkey treat_gvkey
1232   keep treat_gvkey repeat_gvkey
1233   gen control_h3inv = 1
1234   save control_firms_for_h3inv_BEA, replace
1235
1236   use invla_CONTROL_BEA, clear
1237   merge m:1 repeat_gvkey using control_firms_for_h3inv_BEA
1238   drop if _merge==2
1239   drop _merge
1240   keep if control_h3inv == 1
1241   save control_firms_h3inv_BEA, replace
1242
1243   use control_firms_h3inv_BEA, clear
1244   keep treat_gvkey
1245   sort treat_gvkey
1246   by treat_gvkey: keep if _n==1
1247   gen treat_for_h3inv = 1
1248   rename treat_gvkey gvkey
1249   save id_treat_BEA, replace
1250
1251   use invla_DPAD_BEA, clear
1252   merge m:1 gvkey using id_treat_BEA
1253   keep if _merge == 3
1254   append using control_firms_h3inv_BEA
1255   save invl_final_BEA, replace
1256
1257   use invla, clear
1258   drop if bea_ind == 1

```

```

1259 keep if DPADfirm == 1
1260 save invla_DPAD_domestic, replace
1261
1262 use invla, clear
1263 drop if bea_ind == 1
1264 keep if DPADfirm == 0
1265 save invla_CONTROL_domestic, replace
1266
1267 use match1_control, clear
1268 rename gvkey CONTROL_gvkey
1269 rename fyear CONTROL_fyear
1270 gen DPAD_gvkey = substr(DPADfirm, 1, 6)
1271 gen DPAD_fyear = substr(DPADfirm, 7, 4)
1272 sort DPAD_gvkey DPAD_fyear
1273 sort CONTROL_gvkey CONTROL_fyear
1274 by CONTROL_gvkey: gen n1 = _n
1275 tab n1
1276 gen repeat_gvkey = CONTROL_gvkey
1277 replace repeat_gvkey = CONTROL_gvkey + "_n2" if n1 == 2
1278 replace repeat_gvkey = CONTROL_gvkey + "_n3" if n1 == 3
1279 replace repeat_gvkey = CONTROL_gvkey + "_n4" if n1 == 4
1280 replace repeat_gvkey = CONTROL_gvkey + "_n5" if n1 == 5
1281 replace repeat_gvkey = CONTROL_gvkey + "_n6" if n1 == 6
1282 replace repeat_gvkey = CONTROL_gvkey + "_n7" if n1 == 7
1283 replace repeat_gvkey = CONTROL_gvkey + "_n8" if n1 == 8
1284 rename DPAD_gvkey gvkey
1285 rename DPAD_fyear match_year
1286 keep gvkey repeat_gvkey match_year
1287 save treat_control_pair, replace
1288
1289 use invla_DPAD_domestic, clear
1290 sort gvkey fyear
1291 by gvkey: keep if _n==1
1292 keep gvkey fyear
1293 merge 1:1 gvkey using treat_control_pair
1294 keep if _merge == 3
1295 drop _merge
1296 rename gvkey treat_gvkey
1297 keep treat_gvkey repeat_gvkey
1298 gen control_h3inv = 1
1299 save control_firms_for_h3inv_domestic, replace
1300
1301 use invla_CONTROL_domestic, clear
1302 merge m:1 repeat_gvkey using control_firms_for_h3inv_domestic
1303 drop if _merge==2
1304 drop _merge
1305 keep if control_h3inv == 1
1306 save control_firms_h3inv_domestic, replace
1307
1308 use control_firms_h3inv_domestic, clear
1309 keep treat_gvkey
1310 sort treat_gvkey
1311 by treat_gvkey: keep if _n==1
1312 gen treat_for_h3inv = 1
1313 rename treat_gvkey gvkey
1314 save id_treat_domestic, replace
1315
1316 use invla_DPAD_domestic, clear
1317 merge m:1 gvkey using id_treat_domestic
1318 keep if _merge == 3
1319 append using control_firms_h3inv_domestic
1320 save invl_final_domestic, replace
1321
1322 use invl_final_domestic, clear
1323 append using invl_final_BEA
1324 sort repeat_gvkey fyear
1325 by repeat_gvkey fyear: keep if _n==1
1326
1327 /*FIXED EFFECTS AND WINSORIZE*/
1328 xi i.fyear i.sic

```

```

1329
1330   capture program drop clean2
1331   program define clean2
1332   sort fyear
1333   by fyear: egen p99 = pctlile (`1'), p(99)
1334   by fyear: egen p1 = pctlile (`1'), p(1)
1335   replace `1'=p99 if `1'>p99&`1'~= .
1336   replace `1'=p1 if `1'<p1&`1'~= .
1337   drop p99 p1
1338   end
1339
1340   clean2 dom_capex
1341   clean2 size
1342   clean2 roa
1343   clean2 sales_growth
1344   clean2 ocf
1345   clean2 mtb
1346   clean2 lev
1347   clean2 cash
1348   clean2 dom_exp_ppe
1349   clean2 ppent_lag
1350   clean2 capx
1351   clean2 gross_capx
1352
1353   save invl_final, replace
1354
1355
1356   ****
1357   *** Step 6 - Create dataset for tests of Domestic Employment (H3b)
1358   ****
1359
1360   /*MERGE BEA DATA WITH COMPUSTAT DATA*/
1361   use bea_dom_data_97_10, clear
1362   append using bea_dom_data_11_13
1363   rename period_year fyear
1364   sort gvkey fyear us_id
1365   by gvkey fyear: keep if _n==1
1366   by gvkey: gen dom_emp_lag = dom_emp[_n-1] if fyear==fyear[_n-1]+1
1367   by gvkey: gen dom_emp_compen_lag = dom_emp_compen[_n-1] if fyear==fyear[_n-1]+1
1368   save bea_dom_data, replace
1369
1370   use timeshiftdiff1, clear
1371   tab fyear
1372   merge m:1 gvkey fyear using bea_dom_data
1373   drop if _merge==2
1374   drop _merge
1375   save merged_bea_dpad1_emp, replace
1376
1377   use employees, clear
1378   sort gvkey fyear
1379   by gvkey fyear: keep if _n==1
1380   by gvkey: gen emp_lag = emp[_n-1] if fyear==fyear[_n-1]+1
1381   by gvkey: gen xlr_lag = xlr[_n-1] if fyear==fyear[_n-1]+1
1382   drop dcom stkco xstf xstfo xstfws
1383   save employees2, replace
1384
1385   use merged_bea_dpad1_emp, clear
1386   merge m:1 gvkey fyear using employees2
1387   drop _merge
1388   drop if missing(at)
1389   save merged_bea_dpad2_emp, replace
1390
1391   use merged_bea_dpad2_emp, clear
1392   gen bea_ind = 1
1393   replace bea_ind = 0 if missing(dom_ein_1)
1394   drop if bea_ind == 1 & missing(dom_emp)
1395   replace dom_emp = 1 if dom_emp == 0
1396   drop if bea_ind == 0

```

```

1397   save bea_emp_sample, replace
1398
1399   use merged_bea_dpad2_emp, clear
1400   keep if missing(us_id)
1401   gen for_sub_ind2 = 1
1402   replace for_sub_ind2 = 0 if for_sub_ind == 0
1403   replace for_sub_ind2 = 0 if missing(for_sub_ind)
1404   gen for_sale_ind2 = 1
1405   replace for_sale_ind2 = 0 if for_sale_ind == 0
1406   replace for_sale_ind2 = 0 if missing(for_sale_ind)
1407   gen for_asset_ind2 = 1
1408   replace for_asset_ind2 = 0 if for_asset_ind == 0
1409   replace for_asset_ind2 = 0 if missing(for_asset_ind)
1410   gen mnc_ind = for_sub_ind2 + for_sale_ind2 + for_asset_ind2
1411
1412   replace emp = 1 if emp == 0
1413   drop if missing(emp)
1414   save dom_only_sample_emp, replace
1415
1416   use dom_only_sample_emp, clear
1417   append using bea_emp_sample
1418   save emp_sample, replace
1419
1420   /*CREATE VARIABLES*/
1421   use emp_sample, clear
1422   gen log_emp = .
1423   replace log_emp = log(dom_emp) if bea_ind == 1
1424   replace log_emp = log(emp*1000) if missing(bea_ind)
1425   gen dom_emp2 = dom_emp
1426   replace dom_emp2 = emp*1000 if missing(bea_ind)
1427   gen inventory = invt_lag/at_lag
1428   gen tangibility = ppent_lag/at_lag
1429   gen ocf = oancf/at_lag
1430   drop if missing(inventory)
1431   drop if missing(tangibility)
1432   drop if missing(ocf)
1433   save empla, replace
1434
1435   /*REQUIRE TREATMENT-CONTROL PAIRS WITHIN MNC AND DOMESTIC SAMPLE*/
1436   use empla, clear
1437   tab fyear DPADfirm
1438   keep if bea_ind == 1
1439   keep if DPADfirm == 1
1440   save empla_DPAD_BEA, replace
1441
1442   use empla, clear
1443   keep if bea_ind == 1
1444   keep if DPADfirm == 0
1445   save empla_CONTROL_BEA, replace
1446
1447   use match1_control, clear
1448   rename gvkey CONTROL_gvkey
1449   rename fyear CONTROL_fyear
1450   gen DPAD_gvkey = substr(DPADfirm, 1, 6)
1451   gen DPAD_fyear = substr(DPADfirm, 7, 4)
1452   sort DPAD_gvkey DPAD_fyear
1453   sort CONTROL_gvkey CONTROL_fyear
1454   by CONTROL_gvkey: gen n1 = _n
1455   gen repeat_gvkey = CONTROL_gvkey
1456   replace repeat_gvkey = CONTROL_gvkey + "_n2" if n1 == 2
1457   replace repeat_gvkey = CONTROL_gvkey + "_n3" if n1 == 3
1458   replace repeat_gvkey = CONTROL_gvkey + "_n4" if n1 == 4
1459   replace repeat_gvkey = CONTROL_gvkey + "_n5" if n1 == 5
1460   replace repeat_gvkey = CONTROL_gvkey + "_n6" if n1 == 6
1461   replace repeat_gvkey = CONTROL_gvkey + "_n7" if n1 == 7
1462   replace repeat_gvkey = CONTROL_gvkey + "_n8" if n1 == 8
1463   rename DPAD_gvkey gvkey
1464   rename DPAD_fyear match_year
1465   keep gvkey repeat_gvkey match_year
1466   save treat_control_pair, replace

```

```

1467
1468 use empla_DPAD_BEa, clear
1469 sort gvkey fyear
1470 by gvkey: keep if _n==1
1471 keep gvkey fyear
1472 merge 1:1 gvkey using treat_control_pair
1473 keep if _merge == 3
1474 drop _merge
1475 rename gvkey treat_gvkey
1476 keep treat_gvkey repeat_gvkey
1477 gen control_h3emp = 1
1478 save control_firms_for_h3emp_BEa, replace
1479
1480 use empla_CONTROL_BEa, clear
1481 merge m:1 repeat_gvkey using control_firms_for_h3emp_BEa
1482 drop if _merge==2
1483 drop _merge
1484 keep if control_h3emp == 1
1485 save control_firms_h3emp_BEa, replace
1486
1487 use control_firms_h3emp_BEa, clear
1488 keep treat_gvkey
1489 sort treat_gvkey
1490 by treat_gvkey: keep if _n==1
1491 gen treat_for_h3emp = 1
1492 rename treat_gvkey gvkey
1493 save id_treat_BEa, replace
1494
1495 use empla_DPAD_BEa, clear
1496 merge m:1 gvkey using id_treat_BEa
1497 keep if _merge == 3
1498 append using control_firms_h3emp_BEa
1499 save empl_final_BEa, replace
1500
1501 use empla, clear
1502 drop if bea_ind == 1
1503 keep if DPADfirm == 1
1504 save empla_DPAD_domestic, replace
1505
1506 use empla, clear
1507 drop if bea_ind == 1
1508 keep if DPADfirm == 0
1509 save empla_CONTROL_domestic, replace
1510
1511 use match1_control, clear
1512 rename gvkey CONTROL_gvkey
1513 rename fyear CONTROL_fyear
1514 gen DPAD_gvkey = substr(DPADfirm, 1, 6)
1515 gen DPAD_fyear = substr(DPADfirm, 7, 4)
1516 sort DPAD_gvkey DPAD_fyear
1517 sort CONTROL_gvkey CONTROL_fyear
1518 by CONTROL_gvkey: gen n1 = _n
1519 gen repeat_gvkey = CONTROL_gvkey
1520 replace repeat_gvkey = CONTROL_gvkey + "_n2" if n1 == 2
1521 replace repeat_gvkey = CONTROL_gvkey + "_n3" if n1 == 3
1522 replace repeat_gvkey = CONTROL_gvkey + "_n4" if n1 == 4
1523 replace repeat_gvkey = CONTROL_gvkey + "_n5" if n1 == 5
1524 replace repeat_gvkey = CONTROL_gvkey + "_n6" if n1 == 6
1525 replace repeat_gvkey = CONTROL_gvkey + "_n7" if n1 == 7
1526 replace repeat_gvkey = CONTROL_gvkey + "_n8" if n1 == 8
1527 rename DPAD_gvkey gvkey
1528 rename DPAD_fyear match_year
1529 keep gvkey repeat_gvkey match_year
1530 save treat_control_pair, replace
1531
1532 use empla_DPAD_domestic, clear
1533 sort gvkey fyear
1534 by gvkey: keep if _n==1
1535 keep gvkey fyear
1536 merge 1:1gvkey using treat_control_pair

```

```

1537 keep if _merge == 3
1538 drop _merge
1539 rename gvkey treat_gvkey
1540 keep treat_gvkey repeat_gvkey
1541 gen control_h3emp = 1
1542 save control_firms_for_h3emp_domestic, replace
1543
1544 use empla_CONTROL_domestic, clear
1545 merge m:1 repeat_gvkey using control_firms_for_h3emp_domestic
1546 drop if _merge==2
1547 drop _merge
1548 keep if control_h3emp == 1
1549 save control_firms_h3emp_domestic, replace
1550
1551 use control_firms_h3emp_domestic, clear
1552 keep treat_gvkey
1553 sort treat_gvkey
1554 by treat_gvkey: keep if _n==1
1555 gen treat_for_h3emp = 1
1556 rename treat_gvkey gvkey
1557 save id_treat_domestic, replace
1558
1559 use empla_DPAD_domestic, clear
1560 merge m:1 gvkey using id_treat_domestic
1561 keep if _merge == 3
1562 append using control_firms_h3emp_domestic
1563 save empl_final_domestic, replace
1564
1565 use empl_final_domestic, clear
1566 append using empl_final_BEA
1567 sort repeat_gvkey fyear
1568 by repeat_gvkey fyear: keep if _n==1
1569 save empl_final, replace
1570
1571 /*FIXED EFFECTS AND WINSORIZE*/
1572 xi i.fyear i.sic
1573
1574 capture program drop clean2
1575 program define clean2
1576 sort fyear
1577 by fyear: egen p99 = pctlile (`1'), p(99)
1578 by fyear: egen p1 = pctlile (`1'), p(1)
1579 replace `1'=p99 if `1'>p99&`1'~=.
1580 replace `1'=p1 if `1'<p1&`1'~=.
1581 drop p99 p1
1582 end
1583
1584 clean2 log_emp
1585 clean2 log_compen
1586 clean2 size
1587 clean2 roa
1588 clean2 ocf
1589 clean2 inventory
1590 clean2 tangibility
1591 clean2 sales_growth
1592 clean2 dom_emp2
1593 clean2 log_for_emp
1594
1595 save empl_final, replace
1596
1597

```