

```

*****Download Data*****;
rsubmit;
*download COMPUSTAT data;
data Comp00;
set comp.funda;
where indfmt = 'INDL' and datafmt = 'STD' and consol = 'C' and popsrc = 'D';
endfyr=datadate;
begfyr=intnx('month',datadate,-11,'beg');
format endfyr YYMMDDN8.;informat endfyr 8.;
format begfyr YYMMDDN8.;informat begfyr 8.;
run;
*download and clean segment data;
PROC SQL;
CREATE TABLE geo
AS SELECT    gvkey, datadate, geotp, sales
FROM        comp.WRDS_SEGMERGED
WHERE       stype='GEOSEG'
Group BY    gvkey, datadate having srcdate = min(srcdate)
order by    gvkey, datadate;

create table geo1
as select distinct gvkey, datadate, geotp, sum(sales) as total_sale
from geo
group by gvkey, datadate, geotp;
quit;
proc transpose data=geo1 out=geo2a
(rename=(_1=obsolete _2=dsale _3=fsale));
    by gvkey datadate;
    id geotp;
    var total_sale;
run;
*Merge segment data with COMPUSTAT data;
proc sql;
create table Comp01
as select a.*, b.dsale, b.fsale
from Comp00 as a left join geo2a as b
on a.gvkey = b.gvkey and a.datadate=b.datadate;
quit;
*download and merge with company data;
data company;
set comp.company;
keep gvkey fic sic;
run;
proc sql;
create table Comp02 as
select a.*, b.fic, b.sic
from Comp01 a left join Company b
on a.gvkey = b.gvkey
order by a.gvkey, a.fyear;
quit;
*Obtain CRSP permno;
proc sql;
create table Comp03
as select a.*, b.lpermno as permno
from Comp02 as a left join crsp.ccmxpf_linktable as b
on a.gvkey = b.gvkey and b.linktype in ("LU","LC","LD","LN","LS","LX") and
b.usedflag = 1

```

```

        and (b.LINKDT <= a.endfyr or missing(b.LINKDT) = 1) and (b.LINKENDDT >=
a.begfyr or missing(b.LINKENDDT)= 1);
quit;
*fill in missing permnos to the extent possible;
proc sort data = Comp03; by gvkey fyear; run;
data Comp03a;
set Comp03;
by gvkey fyear;
retain permno2;
if first.gvkey then permno2=permno; else do;
if missing(permno) then permno=permno2;
else permno2=permno; end;
drop permno2;
run;
*identify Permno linked to multiple gvkeys, or vice versa;
proc sql;
create table Comp03b as
select *, count(unique(gvkey)) as count1
from Comp03a
group by permno
order by gvkey, datadate;
create table Comp03c as
select *, count(unique(permno)) as count2
from Comp03b
group by gvkey
order by gvkey, datadate;
quit;
data Comp03c;
set Comp03c;
if count1=1 & count2=1 then unique_link=1; else unique_link=0;
drop count1 count2;
run;
*Fill in missing data;
data Comp04;
set Comp03c;
if pifo=. then pifo=pi-pidom;
if pidom=. then pidom=pi-pifo;
if fsale=. then fsale=sale-dsale;
if dsale=. then dsale=sale-fsale;
ciknumber=cik+1-1;
format ciknumber BEST12.;
informat ciknumber BEST32.;
run;
PROC download data=Comp04 out=fx.Original_Data;
run;
endrsubmit;

*Import Scott Dyreng's Exhibit 21 data here;

*Clean Exhibit 21 data;
data fx.sub1;
set country31may2015;
format datadate yymmddn8. ;
informat datadate 8.;
if not missing(ncountries);
if not missing(ciknumber);
drop INCSTATE HQSTATE IRSNUMBER FILEDATE;

```

```

run;
proc sort data=fx.sub1 nodupkey out=ncountries; by ciknumber datadate
ncountries; run;

*Obtain multi-period income shifting incentive variables here using the
AVGFTR macro (output=fx.avgftr);
*****End of Download*****;

*****Organize Data*****;
%let identifier = gvkey cik ciknumber datadate fyear permno unique_link sic
fic;
%let financial = at xrd xad intan ch dt sale pi pifo pidom dsale fsale;

data fx.Comp00;
set fx.Original_Data;
keep &identifier &financial;
run;

*Create lags (t-4);
PROC SQL;
CREATE TABLE fx.Comp01 AS
SELECT a.*, b.xrd as l4xrd 'l4xrd', b.xad as l4xad 'l4xad', b.intan as
l4intan 'l4intan',
           b.ch as l4ch 'l4ch', b.dt as l4dt 'l4dt', b.at as l4at
'l4at'
FROM fx.Comp00 as a left join fx.Comp00 as b
on a.gvkey = b.gvkey and a.fyear = b.fyear+4;
quit;
proc sort data=fx.Comp01 nodupkey; by gvkey fyear at; run;

*Add multi-period income shifting incentive variables;
proc sql;
create table fx.Comp02
as select a.*, b.avgfetr, b.avgftr, b.lowavgftr, b.avgros, b.avgfros
from fx.Comp01 as a left join fx.avgftr as b
on a.gvkey = b.gvkey and a.fyear=b.fyear;
quit;

*Clean data;
data fx.Comp03_all;
set fx.Comp02;
if 6000<=sic<=6999 then delete; *Financial;
if 4900<=sic<=4999 then delete; *Utilities;
sic2=input(substr(sic,1,2),12.);
if fic='USA';
if cik ne '';
if not missing(avgfros+avgros+AvgFTR);
ciknumber=cik+1-1;
format ciknumber BEST12.;
informat ciknumber BEST32.;
if l4xrd=. then l4xrd=0;
if l4xad=. then l4xad=0;
if l4intan=. then l4intan=0;
if l4ch=. then l4ch=0;
if l4dt=. then l4dt=0;
l4rd_at=l4xrd/l4at; if l4rd_at=. then l4rd_at=0;
l4ads_at=l4xad/l4at; if l4ads_at=. then l4ads_at=0;

```

```

l4intan_at=l4intan/l4at; if l4intan_at=. then l4intan_at=0;
l4cash_at=l4ch/l4at; if l4cash_at=. then l4cash_at=0;
l4debt_at=l4dt/l4at; if l4debt_at=. then l4debt_at=0;
l4size=log(1+l4at);
run;

*Final Outbound Income Shifter Sample;
data fx.Comp03;
set fx.Comp03_all;
if avgftr<0;
if 1997<=fyear<=2017;
run;

*winsorize all continuous variables here (output=fx.Comp03_win);

*****ICRG Score*****;
*Import ICRG FX risk data here (z1);

*Clean ICRG data;
proc sort data=z1; by Country Country_Code; run;
proc transpose data=z1 out=z2 ;
by Country Country_Code;
run;

data z3 (keep=Country Country_Code year score);
set z2;
year=input(_LABEL_,12.);
score=COL1;
if Country_code ne "";
run;

*Merge with tax haven data (EX-21);
proc sql;
create table z4 as
select a.ciknumber, a.datadate, a.ISO3, a.country, a.countrycount,
a.totalcount, a.ncountries, a.TAXHAVEN, b.score
from fx.sub1 as a left join z3 as b
on a.ISO3=b.Country_code and year(a.datadate)=b.year
order by a.ciknumber, a.datadate;
quit;

*Clean merged data;
data z5;
set z4;
if totalcount ne 0;
if ISO3 ne "USA";
if not missing(score);
drop totalcount;
run;

proc sql;
create table z6
as select *, sum(countrycount) as totalcount
from z5
group by ciknumber, datadate;
quit;

```

```

*Create firm-level FX risk score;
proc sql;
create table z7
as select distinct ciknumber, datadate, sum(score*countrycount/totalcount) as
score
from z6
group by ciknumber, datadate;
quit;

*match FX risk to COMPUSTAT variables at t+4;
proc sql;
create table z8 as
select distinct a.*, b.datadate as matched_date 'matched_date', b.score as
l4score 'l4score',
from fx.Comp03 as a, z7 as b
where a.ciknumber=b.ciknumber and -12<=intck('month',
intnx('year',b.datadate,+5), a.datadate)<=12
order by a.gvkey, a.fyear;
quit;

*Make FXRisk variable increase in volatility and delete duplicates;
data z9;
set z8;
l4score=l4score*-1;
mo=intck('month',matched_date,datadate);
ftr_score=l4score*avgftr;
run;
proc sort data=z9; by gvkey fyear avgros mo; run;
proc sort data=z9 nodupkey out=z10; by gvkey fyear avgros; run;

*winsorize all continuous variables here (output=z11);

*****Political Stability Data*****;
*Import ICRG Political Stability risk point dataset here (fx.ps);

*Clean Political Stability data;
proc sort data=fx.ps; by Country_Code; run;
proc transpose data=fx.ps out=ps2 ;
by Country_Code;
run;
data ps3 (keep=Country_Code year PSIndex);
set ps2;
year=input(_LABEL_,12.);
PSIndex=COL1;
if not missing (PSIndex);
if Country_Code ne '';
run;

*Merge with EX-21 data;
proc sql;
create table ps4 as
select a.ciknumber, a.datadate, a.ISO3, a.country, a.countrycount,
a.ncountries, b.PSIndex
from fx.sub1 as a left join ps3 as b
on a.ISO3=b.Country_code and year(a.datadate)=b.year
order by a.ciknumber, a.datadate;
quit;

```

```

data ps5;
set ps4;
if ncountries ne 0;
if not missing (PSIndex);
drop ncountries;
run;

proc sql;
create table ps6
as select *, sum(countrycount) as totalcount
from ps5
group by ciknumber, datadate;
quit;

*Create firm-level PS variable;
proc sql;
create table ps7
as select distinct ciknumber, datadate, sum(PSIndex*countrycount/totalcount)
as PSIndex
from ps6
group by ciknumber, datadate;
quit;

*Merge with COMPUSTAT data at t+4, then remove duplicates;
proc sql;
create table ps8 as
select distinct a.*, b.PSIndex as l4PSIndex 'l4PSIndex', b.datadate as
matched_date_ps
from fx.Comp03 as a, ps7 as b
where a.ciknumber=b.ciknumber and -12<=intck('month',
intnx('year',b.datadate,+5), a.datadate)<=12
order by a.gvkey, a.fyear;
quit;

data ps9;
set ps8;
mo=intck('month',matched_date_ps,datadate);
run;
proc sort data=ps9; by gvkey fyear avgros mo; run;
proc sort data=ps9 nodupkey; by gvkey fyear avgros; run;

*****Capital Controls Data*****;
*Import Capital Controls dataset here (fx.cc);

*Merge with EX-21 data;
proc sql;
create table cc1 as
select a.ciknumber, a.datadate, a.ISO3, a.country, a.countrycount,
a.ncountries, b.kao
from fx.sub1 as a left join fx.cc as b
on a.ISO3=b.ISO3 and year(a.datadate)=b.year
order by a.ciknumber, a.datadate;
quit;

data cc2;
set cc1;

```

```

if ncountries ne 0;
if not missing (kao);
drop ncountries;
run;

proc sql;
create table cc3
as select *, sum(countrycount) as totalcount
from cc2
group by ciknumber, datadate;
quit;

*Create firm-level CC variable;
proc sql;
create table cc4
as select distinct ciknumber, datadate, sum(kao*countrycount/totalcount) as
cc
from cc3
group by ciknumber, datadate;
quit;

*Merge with COMPUSTAT data at t+4, then remove duplicates;
proc sql;
create table cc5 as
select distinct a.*, b.cc as l4cc 'l4cc', b.datadate as matched_date_cc
from fx.Comp03 as a, cc4 as b
where a.ciknumber=b.ciknumber and -12<=intck('month',
intnx('year',b.datadate,+5), a.datadate)<=12
order by a.gvkey, a.fyear;
quit;

data cc6;
set cc5;
mo=intck('month',matched_date_cc,datadate);
run;
proc sort data=cc6; by gvkey fyear avgros mo; run;
proc sort data=cc6 nodupkey; by gvkey fyear avgros; run;

*****Merge All Datasets Above*****;
proc sql;
create table fx.Comp04 as
select a.*, b.l4PSIndex, c.l4cc
from z11 as a left join ps9 as b
on a.gvkey=b.gvkey and a.fyear=b.fyear
left join cc6 as c
on a.gvkey=c.gvkey and a.fyear=c.fyear;
quit;

*****Cross-section: Functional Currency*****;
*Create functional currency indicator variable;
proc sql;
create table fx.func1 as
select a.gvkey, a.fyear, a.datadate, a.at, a.recta, b.recta as lrecta
'lrecta'
from fx.original_data as a, fx.original_data as b
where a.gvkey=b.gvkey and a.fyear=b.fyear+1;
quit;

```

```

proc sort data=fx.func1 nodupkey; by gvkey fyear; run;

data fx.func2;
set fx.func1;
if not missing(recta+lrecta);
if recta-lrecta=0 then foreign_funcur=0; else foreign_funcur=1;
run;

*Match to COMPUSTAT variables at t+4;
proc sql;
create table fx.func3 as
select a.*, b.foreign_funcur as l4foreign_funcur 'l4foreign_funcur'
from fx.Comp04 as a left join fx.func2 as b
on a.gvkey=b.gvkey and a.fyear=b.fyear+4;
quit;

*Clean data;
data fx.func4;
set fx.func3;
if l4foreign_funcur=. then l4foreign_funcur=0;
run;
proc sort data=fx.func4; by l4foreign_funcur gvkey fyear; run;

*****Financial Hedging*****;
*Import hand-collected hedging data here (fx.hedge);

*Merge with COMPUSTAT data;
PROC SQL;
CREATE TABLE fx.hedge1 AS
SELECT a.*, b.matched_date_hedge, b.l4FX_contract, b.l4Hedge
FROM fx.Comp03 as a left join fx.hedge as b
on a.cik = b.cik and a.fyear = b.fyear;
QUIT;

*Merge with PS and CC data;
proc sql;
create table fx.hedge1a as
select a.*, b.l4PSIndex, c.l4cc
from fx.hedge1 as a left join ps9 as b
on a.gvkey=b.gvkey and a.fyear=b.fyear
left join cc6 as c
on a.gvkey=c.gvkey and a.fyear=c.fyear;
quit;

*Clean data;
data fx.hedge2;
set fx.hedge1a;
l4fcd_at=l4FX_contract/l4at;
if not missing(l4Hedge);
run;

*winsorize all continuous variables here (output=fx.hedge3) then export;

*****Operational Hedging*****;
*Create breadth variable;
proc sql;
create table breadth

```



```

as select distinct ciknumber, datadate, totalcount, ncountries as breadth
from fx.sub1
order by ciknumber, datadate;
quit;

*Match to COMPUSTAT variables at t+4 then clean;
proc sql;
create table breadth1 as
select distinct a.*, b.breadth as l4breadth 'l4breadth', b.datadate as
matched_date_br
from fx.Comp03 as a, breadth as b
where a.ciknumber=b.ciknumber and -12<=intck('month',
intnx('year',b.datadate,+5), a.datadate)<=12
order by a.gvkey, a.fyear;
quit;

data breadth2;
set breadth1;
mo=intck('month',matched_date_br,datadate);
run;
proc sort data=breadth2; by gvkey fyear avgros mo; run;
proc sort data=breadth2 nodupkey; by gvkey fyear avgros; run;

*Create depth variable;
proc sql;
create table depth
as select distinct ciknumber, datadate, ISO3, countrycount, ncountries,
totalcount, countrycount/totalcount as depth
from fx.sub1
order by ciknumber, datadate;
quit;
proc sort data=depth; by ciknumber datadate descending COUNTRYCOUNT ;run;

data depth1;*generate a random firm-year group code in order to rank
countrynumber;
set depth;
group=ciknumber*sqrt(datadate)+sqrt(ciknumber);
if ncountries ne 0;
run;
proc sort data=depth1; by group descending COUNTRYCOUNT ;run;

data depth2;
set depth1;
by group;
if first.group then rank=0;
rank+1;
if rank=1 | rank=2;*top two countries;
run;

proc sql;
create table depth3 as
select ciknumber, datadate, sum(depth) as depth
from depth2
group by ciknumber, datadate
order by ciknumber, datadate;
quit;

```

```

*Match to COMPUSTAT variables at t+4 then clean;
proc sql;
create table depth4 as
select distinct a.*, b.depth as l4depth 'l4depth', b.datadate as
matched_date_dp
from fx.Comp03 as a, depth3 as b
where a.ciknumber=b.ciknumber and -12<=intck('month',
intnx('year',b.datadate,+5), a.datadate)<=12
order by a.gvkey, a.fyear;
quit;

data depth5;
set depth4;
mo=intck('month',matched_date_dp,datadate);
run;
proc sort data=depth5; by gvkey fyear avgros mo; run;
proc sort data=depth5 nodupkey; by gvkey fyear avgros; run;

*Merge all datasets above and create ln_breadth;
proc sql;
create table bd as
select a.*, b.l4depth, c.l4breadth
from fx.Comp04 as a left join depth5 as b
on a.gvkey=b.gvkey and a.fyear=b.fyear
left join breadth2 as c
on a.gvkey=c.gvkey and a.fyear=c.fyear;
quit;

data bd1;
set bd;
l4ln_breadth=log(1+l4breadth);
run;

*winsorize all continuous variables here (output=fx.breadth_depth) then
export;

```