### Drawing Samples of 30 Households (Using Segment Structure) Macro

libname prac "H:\Thesis\_Practicum";

/\*creating variable for number of gotts per woreda\*/

proc sql;

create table t1

as select \*, max(gott) as numgott

from prac.population

group by woreda;

/\*creating variable for number of segment in each gott\*/

proc sql;

create table t2

as select \*, max(segment) as numseg

from t1

group by unique\_gott;

/\*tabulating woreda by gott by segment\*/

proc freq data=t2 noprint;

   tables woreda\*gott\*segment / out=x;

run;

proc sort data=x;

   by woreda gott segment;

run;

/\*creating data set with one observation for each gott, with variable that is total number of segments per gott\*/

data frame;

   set x;

   by woreda gott segment;

   drop count percent;

   if last.gott;

run;

/\*The following macro draws 1000 samples from the population data set created by the previous macro. It also calculates 95% uncertainty intervals. It uses the segment structure, so only one segment or 30 households can be selected using this macro. The user inputs the cluster number desired.\*/

%macro sample(cluster=14);

/\*selecting user input number of clusters (gotts) within each of the 17 woredas\*/

proc surveyselect data=frame method=sys N=&cluster out=sample\_&cluster seed=6212018 rep=1000;

samplingunit gott;

strata woreda;

run;

proc sort data=sample\_&cluster;

   by woreda gott segment;

run;

/\*expanding data set to the segment level so that we can randomly select a segment\*/

data sampseg\_&cluster;

   set sample\_&cluster;

   do segment=1 to segment; output;

   end;

run;

/\*creating random number for each segment\*/

data sseg\_&cluster;

set sampseg\_&cluster;

rand=ranuni(123456);

run;

/\*sorting by random number within replicate, woreda, and gott\*/

proc sort data=sseg\_&cluster;

   by replicate woreda gott rand;

run;

/\*selecting only the first segment within each replicate, woreda, and gott to get 1 segment per cluster\*/

data full\_&cluster;

   set sseg\_&cluster;

   by replicate woreda gott;

   if first.gott;

   do household=1 to 30; /\*expanding to the household level for future merging\*/

      output;

   end;

   drop samplingweight;

run;

proc sort data=full\_&cluster;

   by woreda gott segment household;

run;

proc sort data=t2;

   by woreda gott segment household;

run;

/\*merging population clinical data to sample data from above\*/

proc sql;

create table sql\_sample\_&cluster as

select f.replicate, f.woreda, p.woreda, f.gott, p.gott, f.segment, p.segment, f.household, p.household,

p.member\_1to9, p.tf\_ind, f.selectionprob, p.numseg, p.numgott

from work.full\_&cluster f

left join work.t2 p

on f.woreda=p.woreda AND f.gott=p.gott AND f.segment=p.segment AND f.household=p.household

;

quit;

proc sort data=sql\_sample\_&cluster;

by replicate woreda gott segment household member\_1to9;

run;

/\*calculating true weights by multiplying selection prob from both stages\*/

data prac.sample\_&cluster;

set sql\_sample\_&cluster;

finalprob=(1/numseg)\*selectionprob;

final\_weight=1/finalprob;

run;

/\*getting bootstrap confidence intervals for each woreda\*/

/\*calculating prevalence for each replicate and each woreda weighted based on sampling weights\*/

proc means data=prac.sample\_&cluster noprint;

weight final\_weight;

class replicate woreda;

var tf\_ind;

output out=means\_&cluster;

run;

proc sort data=prac.sample\_&cluster;

by gott segment;

run;

proc descript data=prac.sample\_&cluster filetype=SAS design=uneqwor conf\_lim=95; /\*won't calculate variance with only 1 segment available - need to change sample design?\*/

nest gott segment;

totcnt numgott numseg;

weight final\_weight;

jointprob \_one\_;

var tf\_ind;

subgroup woreda;

levels      30;

print mean semean lowmean upmean;

run;

/\*manipulating data set of prevalences\*/

data mn\_&cluster;

set means\_&cluster;

if \_STAT\_ ne "MEAN" then delete;

drop \_TYPE\_ \_FREQ\_ \_STAT\_;

rename tf\_ind=Mean\_prev;

if woreda=. then delete;

if replicate=. then delete;

run;

/\*calculating percentiles within replicates for 95% confidence intervals\*/

proc univariate data=mn\_&cluster noprint;

class woreda;

var mean\_prev;

output out=percentile\_&cluster mean=avg\_est\_mean pctlpre=replicate pctlpts=2.5,97.5;

run;

/\*creating dataset with lower and upper bounds, and length on confidence interval\*/

data prac.ci\_one\_&cluster;

set percentile\_&cluster;

rename replicate2\_5=lower replicate97\_5=upper;

label replicate2\_5=" " replicate97\_5=" ";

clusters=&cluster;

length=replicate97\_5-replicate2\_5;

run;

%mend sample;