```
/* STATA code for the STANDARD approach for the laboratory-based acquired HIV drug resistance survey */

/* This code assumes patient- and laboratory-level information follow the configuration of the Excel data upload template and are stored in a data file labeled "patient_data_A1.xlsx". */

/* HIV drug resistance sequences are assumed to be in FASTA file format and stored in a data file labeled "FASTA_A1.xlsx". Sequence identification numbers and eligible case specimen identifiers must be identical and follow WHO convention. */

/* Both data files should be in the same directory, e.g., "C:/Documents" */



/* Clear any previous output */

clear


/* Set working directory to the directory containing all data files. For example, if directory is "C:/Documents", run the following code */

cd "C:/Documents"


/* Import each sheet of the Excel Upload Template, storing the first row as headings and changing all header names to uppercase. Save each sheet as its own .dta file. Given the Excel file name "patient_data_A1.xlsx", run the following*/

import excel using "patient_data_A1.xlsx", describe

forvalues sheet=1/`=r(N_worksheet)' {

	local sheetname = r(worksheet_`sheet')

	import excel using patient_data_A1, sheet("`sheetname'") firstrow case(upper)

	local sheetname = upper(subinstr("`sheetname'", " ", "_", .))

	save "`sheetname'", replace

	clear

}
```

```
/* This code prepares the resistance data imported from the Stanford HIVdb database */


/* Import the HIVdb resistance data, storing the first row as heading and changing all header names to uppercase. Given the file name "FASTA_A1.xlsx", run */

import excel using "FASTA_A1.xlsx", sheet("ResistanceSummary") firstrow case(upper)

/* Rename SEQUENCENAME as PARTICIPANTID */

rename SEQUENCENAME PARTICIPANTID

/* Drop all cells without a subject ID */

drop if missing(PARTICIPANTID)

/* Drop all unnecessary variables */

drop *SCORE ALGORITHM* STRAIN GENES PI* NRTI* NNRTI* INSTI*

/* Replace NA's with the missing symbol */

destring, ignore("NA") replace

/* For each of the resistance level variables, classify as a binary resistance indicator, with levels 1-2 corresponding to susceptible (no HIVDR), and levels 3-5 corresponding to HIVDR. */

ds *LEVEL

local plist = r(varlist)

foreach i of local plist {

        replace `i' = 0 if `i' < 3 & !missing(`i')

        replace `i' = 1 if `i' >= 3 & !missing(`i')

}

/* Rename resistance type variables */

rename *LEVEL *_RES

/* Generate variable for DTG-specific resistance */

gen DTG_ADR = DTG_RES

/* Generate variable for Any Boosted PI resistance, defined as any resistance to ATV_r, DRV_r, or LPV_r */

egen ANY_PI = rowmax(ATVR_RES DRVR_RES LPVR_RES)

/* Generate variable for Any NRTI resistance */

egen ANY_NRTI = rowmax(ABC_RES AZT_RES D4T_R DDI_RES FTC_RES TDF_RES)
```

```
/* Generate variable for Any NNRTI resistance, defined as any resistance to NVP or EFV */

egen ANY_NNRTI = rowmax(EFV_RES NVP_RES)

/* Generate variable for Any INI resistance */

egen ANY_INI = rowmax(BIC_RES DTG_RES EVG_RES RAL_RES)

/* Generate variable for Any ADR resistance, as defined in the protocol */

egen ANY_ADR = rowmax(ANY_PI ANY_NRTI ANY_NNRTI ANY_INI)

/* save changes to the modified dataset */

save RESISTANCE_SUMMARY, replace



/* This code prepares the VL laboratory data */

/* Remove previous dataset, then load the VL lab data and rename the variables for total DTG and non-DTG case specimens */

clear

use VL_LAB_INFORMATION.dta

rename NELIGIBLEDTGCASESPECIMENS NDTG

rename NELIGIBLENONDTGCASESPECIMENS NNONDTG

/* Save changes */

save VL_LAB_INFORMATION, replace



/* This code prepares the patient-level data on ARV treatment regimen */


/* Remove previous dataset, then load the treatment regimen data */

clear

use PARTICIPANT_TREATMENTS.dta

/* Exclude observations missing a subject ID or corresponding to past ART */

drop if missing(PARTICIPANTID) | upper(CURRENTARTYN) == "N"

/* Drop unnecessary variables */
```

```
drop OTHERARVDRUG CURRENTARTYN

/* Rename ARV drug types so that all variable names start with a letter */

replace ARVDRUG = "ARV_" + ARVDRUG

/* Generate binary DTG variable corresponding to 1 if patient is on a DTG-containing regimen and 0 if patient is on a non-DTG-containing regimen */

gen TEMP_DTG = cond(inlist(ARVDRUG, "ARV_DTG", "ARV_TLD", "ARV_JUL"), 1, 0)

by PARTICIPANTID, sort: egen DTG = max(TEMP_DTG)

drop TEMP_DTG

/* Reformat the ARVDRUG variable so that each ARV drug type corresponds to a new binary variable, set to 1 if the patient is taking that drug, and set to 0 if not */

gen ON = 1

reshape wide ON, i(PARTICIPANTID) j(ARVDRUG) string

rename ON* *

/* Save changes */

save PARTICIPANT_TREATMENTS, replace



/* This code prepares the patient-level data on other variables */


/* Remove previous dataset, then load the patient-level data */

clear

use SURVEY_PARTICIPANTS.dta

/* Drop if missing subject ID or patient is in the paediastric population */

drop if missing(PARTICIPANTID) | substr(PARTICIPANTID, 23, 1) != "a"

/* Generate variable corresponding to the VL laboratory for each patient */

gen VLLABCODE = substr(PARTICIPANTID, 14, 3)

/* Recode unknown values as the missing symbol */

recode DATE* (9999 = .)

recode AGE (-9 = .)
```

```
foreach var of varlist PRIORART BREASTFEEDINGSTATUS PREGNANCYSTATUS CURRENTART {
        replace `var' = "." if `var' == "UNK"
}
/* Save changes */
save SURVEY_PARTICIPANTS.dta, replace



/* Merge all the datasets and save the combined data into one .dta file */


/* Use a many-to-one merge to merge the VL laboratory data on total case specimens */
merge m:1 VLLABCODE using VL_LAB_INFORMATION, keepusing(NDTG NNONDTG) keep(match) nogenerate
/* Merge in the treatment regimen data by subject ID */
merge 1:1 PARTICIPANTID using PARTICIPANT_TREATMENTS, keep(match) nogenerate
/* Merge in the HIVDR data by subject ID */
merge 1:1 PARTICIPANTID using RESISTANCE_SUMMARY, keep(match) nogenerate
save ALL_DATA.dta, replace



/* This code is for declaring the survey design and analyzing data for HIVDR */


/* Remove previous dataset and load in the combined data */
clear
use ALL_DATA.dta
/* Generate the variable for total DTG and non-DTG case specimens per laboratory. This corresponds to the stratum populations. */
gen STRAT_POP = cond(DTG == 1, NDTG, NNONDTG)
/* Generate the stratum weights, calculated as the stratum totals divided by the number of sampled case specimens per stratum */
by VLLABCODE DTG, sort: gen WEIGHTS = STRAT_POP/_N
```

```
/* Stratify the data by laboratory and DTG/non-DTG regimen */

egen STRATA = group(VLLABCODE DTG)

/* Set the stratified one-stage survey design with finite population correction. Standard errors for single-unit strata will be set to the average of the standard errors for other strata */

svyset [pweight = WEIGHTS], strata(STRATA) fpc(STRAT_POP) singleunit(scaled)



/* Analyses for Outcomes 1 & 2 */

/* Obtain estimates and confidence intervals for the prevalence of overall drug resistance among all patients */

svy: proportion ANY_ADR

/* If Wald confidence intervals are desired instead, uncomment the following */

/* svy: proportion ANY_ADR, citype(wald)*/

/* Obtain estimates and confidence intervals for the prevalence of overall drug resistance among patients on DTG-containing regimens */

svy, subpop(if DTG==1): proportion ANY_ADR

/* Obtain estimates and confidence intervals for the prevalence of overall drug resistance among patients on non-DTG-containing regimens */

svy, subpop(if DTG==0): proportion ANY_ADR

/* Obtain estimates and confidence intervals for the prevalence of DTG-specific drug resistance among individuals on DTG regimens */

svy, subpop(if DTG==1): proportion DTG_ADR



/* Analyses for Outcome 3 */
```

```
/* Example: Obtain prevalence and variance estimates of overall drug resistance among men */

svy, subpop(if GENDER == "M"): proportion ANY_ADR



/* Example: Obtain prevalence and variance estimates of Any NRTI drug resistance among individuals on DTG regimens */

svy, subpop(if DTG==1): proportion ANY_NRTI



/* Analyses for Outcome 4 */


/* Remove previous dataset, then load the VL lab data */

clear

use VL_LAB_INFORMATION

/* drop irrelevant variables and collapse data into only the column sums, which are the totals across all laboratories */

drop NAMEOFVLLAB VLLABCODE

collapse (sum) _all

/* Obtain estimate for the prevalence of VL suppression amongst individuals on DTG-based ART. No confidence intervals are necessary */

di TDTGVS/TDTG


/* Obtain estimate for the prevalence of VL suppression amongst individuals on non-DTG-based ART. No confidence intervals are necessary */

di TNONDTGVS/TNONDTG


/* Obtain estimate for the prevalence of VL suppression amongst all individuals on ART. No confidence intervals are necessary */

di (TDTGVS + TNONDTGVS)/(TDTG + TNONDTG)
```

```
/* Analyses for Outcome 5 */


/* Example: obtain prevalence of VL suppression among men on DTG-based ART. No confidence
intervals are necessary */

di TDTGVSMEN/TDTGMEN


/* Example: obtain prevalence of VL suppression among all individuals age 0-9 on ART. No confidence
intervals are necessary */

di (TDTGVSAGE09 + TNONDTGVSAGE09)/(TDTGAGE09 + TNONDTGAGE09)
```