

Annex A9: SPSS SYNTAX TO PREPARE DATA FILE FOR multilevel regression analysis

```
*****  
*****Multilevel analysis*****  
*****Data preparation*****  
*****
```

*** RE-CODING AND IMPUTATION OF STUDENT AND SCHOOL-LEVEL VARIABLES FOR MULTI-LEVEL-ANALYSIS.

GET FILE = 'C:\PISA\DATA2006\All.sav'.
EXE.

***** 1 INDIVIDUAL LEVEL VARIABLES *****

*** FEMALE note: as there is only one female missing therefore no missing dummy***.

RECODE ST04Q01 (2=0)(1=1)(ELSE=SYSMIS) INTO female.
EXE.

IF (MISSING(female)) female=0.
EXE.

*** ESCS***.

WEIGHT BY W_FSTUWT .

SORT CASES BY CNT SCHOOLID STIDSTD.

AGGREGATE
/OUTFILE=*
MODE=ADDVARIABLES
/PRESORTED
/BREAK=CNT SCHOOLID
/XESCS = MEAN(ESCS).

AGGREGATE
/OUTFILE=*
MODE=ADDVARIABLES
/PRESORTED
/BREAK=CNT
/YESCS = MEAN(ESCS).

** IMPUTE student level and school level ESCS***.

COMPUTE MESCS=0.
COMPUTE MXESCS=0.
EXE.

```
DO IF (MISSING(ESCS)).  
  COMPUTE MESCS=1.  
  COMPUTE ESCS=XESCS.  
END IF.  
EXE.
```

```
IF (MISSING(ESCS)) ESCS=YESCS.  
EXE.
```

```
DO IF (MISSING(XESCS)).  
  COMPUTE MXESCS=1.  
  COMPUTE XESCS=YESCS.  
END IF.  
EXE.
```

```
*** ESCS2***.
```

```
compute ESCS2 = escs**2.  
EXE.
```

```
*** NATIVE***.
```

```
recode immig (1 = 1) (2 3= 0) (else = sysmiss) into native.  
COMPUTE MNATIVE=0.  
EXE.
```

```
DO IF (MISSING(native)).  
  COMPUTE MNATIVE=1.  
  COMPUTE NATIVE=0.  
END IF.  
EXE.
```

```
*** SAMELANG***.
```

```
recode st12q01 (1 2 = 1) (3= 0) (else = sysmiss) into samelang.  
EXE.
```

```
COMPUTE MSAMELANG=0.  
EXE.
```

```
DO IF (MISSING(SAMELANG)).  
  COMPUTE MSAMELANG=1.  
  COMPUTE SAMELANG=0.  
END IF.  
EXE.
```

```
*****2. School level variables*****.
```

```
****compute school and country means for imputation**.
```

```
WEIGHT
  BY W_FSTUWT .
```

```
AGGREGATE
/OUTFILE=*
MODE=ADDVARIABLES
/PRESORTED
/BREAK=CNT
/YIRATCOMP = MEAN(IRATCOMP)
/YSTRATIO = MEAN(STRATIO)
/YSCIPROM = MEAN(SCIPROM)
/YSCMATEDU = MEAN(SCMATEDU)
/YTCSHORT = MEAN(TCSHORT)
/YSCHSIZE=MEAN(SCHSIZE).
```

```
**** 2.1 BACKGROUND VARIABLES****.
```

```
***XRURAL and XCITY***.
```

```
recode sc07q01 (1 2 = 1) (3 4 5 = 0) (else = sysmiss) into xrural.
recode sc07q01 (4 5 = 1) (1 2 3 = 0) (else = sysmiss) into xcity.
EXE.
```

```
If (CNT="HKG") xcity=1.
If (CNT="HKG") xrural=0.
compute mxrural=0.
exe.
```

```
do if (missing(xrural)).
  compute mxrural=1.
  compute xrural=0.
  compute xcity=0.
end if.
exe.
```

```
*** XSCHSIZE and XSCHSIZ2***.
```

```
COMPUTE XSCHSIZE=(SCHSIZE/100).
COMPUTE MXSCHSIZE=0.
EXE.
```

```
DO IF (MISSING(SCHSIZE)).
  COMPUTE XSCHSIZE=(YSCHSIZE/100).
  COMPUTE MXSCHSIZE=1.
END IF.
EXE.
```

```
compute XSCHSIZ2 = XSCHSIZE**2.
EXE.
```

*** XESCS***.

** was computed from individual file**.

*** BLOCK 1: SELECTIVITY***.

****Selectivity****.

recode selsch (1=1) (4,3,2 = 0) (ELSE =SYSMIS) into losele.
recode selsch (4=1) (3,2,1 = 0) (ELSE=SYSMIS) into hisele.
exe.

COMPUTE xlosele=losele.
COMPUTE xhisele=hisele.
EXE.

recode selsch (sysmiss = 1) (else = 0) into mxhisele.

IF (MISSING(losele)) xlosele=0.
IF (MISSING(hisele)) xhisele=0.
exe.

format xlosele xhisele mxhisele (f1.0).

****Ability grouping****.

RECODE ABGROUP (3=1) (2,1=0) (ELSE=SYSMIS) INTO abgr.
EXE.

COMPUTE xabgr=abgr.
EXE.

COMPUTE mxabgr=0.
EXE.

DO IF (MISSING(abgr)).
 COMPUTE mxabgr=1.
 COMPUTE xabgr=0.
END IF.
EXE.

*** BLOCK 2: SCHOOL MANAGEMENT/ GOV FUNDING***.

*** Private management and public funding***.

RECODE SC02q01 (2=1)(1=0) (ELSE=SYSMIS) INTO PRIVMAN.

```
COMPUTE GOVFUND=SC03Q01.  
EXE.
```

```
WEIGHT  
BY W_FSTUWT .
```

```
AGGREGATE  
/OUTFILE=*  
MODE=ADDVARIABLES OVERWRITE=YES  
/PRESORTED  
/BREAK=CNT  
/YPRIVMAN = MEAN(PRIVMAN)  
/YGOVFUND = MEAN(GOVFUND).
```

```
COMPUTE MXPRIVMAN=0.  
COMPUTE XPRIVMAN=PRIVMAN.  
EXE.
```

```
DO IF (MISSING(PRIVMAN)).  
  COMPUTE MXPRIVMAN=1.  
  COMPUTE XPRIVMAN=0.  
END IF.  
EXE.
```

```
COMPUTE MXGOVFUND=0.  
COMPUTE XGOVFUND=GOVFUND.  
EXE.
```

```
DO IF (MISSING(GOVFUND)).  
  COMPUTE MXGOVFUND=1.  
  COMPUTE XGOVFUND=YGOVFUND.  
END IF.  
EXE.
```

** substitute the international mean for the country where we have no information on gov funding***.

```
COMPUTE HELP=1.
```

```
WEIGHT OFF.
```

```
AGGREGATE /OUTFILE=* MODE=ADDVARIABLES overwrite = yes  
/PRESORTED  
/BREAK=CNT  
/stpopwgt=SUM(w_fstuwt).
```

```
COMPUTE cntfac_equal_wgt=1000/stpopwgt.  
EXE.
```

```
COMPUTE newwgt = cntfac_equal_wgt*w_fstuwt.  
EXE.
```

```
WEIGHT BY newwgt.
```

```
AGGREGATE  
/OUTFILE=*  
MODE=ADDVARIABLES OVERWRITE=YES  
/BREAK=help  
/ZGOVFUND = MEAN(GOVFUND)  
/ZPRIVMAN = MEAN(PRIVMAN).
```

```
weight off.  
IF (MISSING(XGOVFUND)) XGOVFUND=ZGOVFUND.  
IF (MISSING(YPRIVMAN)) YPRIVMAN=ZPRIVMAN.  
EXE.
```

```
**** BLOCK 3: Parental Pressure and Choice****.
```

```
**** SCHOOL COMPETITION****.
```

```
RECODE sc18q01 (1=1) (2=1) (3=0) (else = sysmiss) into schlcomp.  
EXE.
```

```
COMPUTE mxschlcomp=0.  
COMPUTE XSCHLCOMP=schlcomp.  
exe.
```

```
DO IF (MISSING (schlcomp)).  
  COMPUTE xschlcomp=0.  
  COMPUTE MXSCHLCOMP=1.  
END IF.  
EXE.
```

```
*** PRESSURE FROM PARENTS***.
```

```
RECODE sc16q01 (1=1) (2=1) (3=0) (else = sysmiss) into presspa.  
EXE.
```

```
COMPUTE MXPRESSPA=0.  
COMPUTE xpresspa=presspa.
```

EXE.

```
DO IF (MISSING(presspa)).  
  COMPUTE XPRESSPA=0.  
  COMPUTE MXPRESSPA=1.  
END IF.  
EXE.
```

WEIGHT BY W_FSTUWT.

SORT CASES BY CNT.

```
AGGREGATE  
/OUTFILE=*  
MODE=ADDVARIABLES  
/PRESORTED  
/BREAK=CNT  
/YSCHLCOMP=MEAN(schlcomp).
```

*** BLOCK 4: Accountability***.

```
RECODE  
Sc15Q01 (2=0) (1=1) (ELSE=SYSMIS) into xacc1  
/Sc15Q02 (2=0) (1=1) (ELSE=SYSMIS) into xacc2  
/Sc15Q03 (2=0) (1=1) (ELSE=SYSMIS) into xacc3.  
EXE.
```

```
RECODE  
Sc17Q01 (2=0) (1=1) (ELSE=SYSMIS) into xacc4  
/Sc17Q02 (2=0) (1=1) (ELSE=SYSMIS) into xacc5  
/Sc17Q03 (2=0) (1=1) (ELSE=SYSMIS) into xacc6  
/Sc17Q04 (2=0) (1=1) (ELSE=SYSMIS) into xacc7  
/Sc17Q05 (2=0) (1=1) (ELSE=SYSMIS) into xacc8.  
EXE.
```

```
If (CNT="HRV") xacc2=0.  
If (CNT="LTU") xacc2=0.  
If (CNT="JPN") xacc3=0.  
If (CNT="SVN") xacc7=0.  
COMPUTE mxacc1=0.  
COMPUTE mxacc2=0.  
COMPUTE mxacc3=0.  
COMPUTE mxacc4=0.  
COMPUTE mxacc5=0.  
COMPUTE mxacc6=0.  
COMPUTE mxacc7=0.  
COMPUTE mxacc8=0.  
COMPUTE mxacc=0.
```

EXE.

```
IF (SYSMIS(xacc1)) mxacc1=1.
IF (SYSMIS(xacc2)) mxacc2=1.
IF (SYSMIS(xacc3)) mxacc3=1.
IF (SYSMIS(xacc4)) mxacc4=1.
IF (SYSMIS(xacc5)) mxacc5=1.
IF (SYSMIS(xacc6)) mxacc6=1.
IF (SYSMIS(xacc7)) mxacc7=1.
IF (SYSMIS(xacc8)) mxacc8=1.
EXE.
```

```
RECODE xacc1 To xacc8 (SYSMIS=0) (0=0) (1=1).
If ((mxacc1=1) or (mxacc2=1) or (mxacc3=1) or (mxacc4=1) or (mxacc5=1) or (mxacc6=1) or
(mxacc7=1) or (mxacc8=1)) mxacc=1.
EXE.
```

*** BLOCK 5: AUTONOMY VARIABLES***.

```
DO REPEAT a=mxauthir mxautfir mxautstsal mxautsalin mxautfbud mxautabud mxautdisp mxautassp
mxautadm mxauttextb mxautccont mxautcoff.
COMPUTE a=0.
END REPEAT.
EXE.
```

```
IF (MISSING(sc11qA1))&(MISSING(sc11qA2))&(MISSING(sc11qA3))&(MISSING(sc11qA4))
mxauthir=1.
IF (MISSING(sc11qB1))&(MISSING(sc11qB2))&(MISSING(sc11qB3))&(MISSING(sc11qB4))
mxautfir=1.
IF (MISSING(sc11qC1))&(MISSING(sc11qC2))&(MISSING(sc11qC3))&(MISSING(sc11qC4))
mxautstsal=1.
IF (MISSING(sc11qD1))&(MISSING(sc11qD2))&(MISSING(sc11qD3))&(MISSING(sc11qD4))
mxautsalin=1.
IF (MISSING(sc11qE1))&(MISSING(sc11qE2))&(MISSING(sc11qE3))&(MISSING(sc11qE4))
mxautfbud=1.
IF (MISSING(sc11qF1))&(MISSING(sc11qF2))&(MISSING(sc11qF3))&(MISSING(sc11qF4))
mxautabud=1.
IF (MISSING(sc11qG1))&(MISSING(sc11qG2))&(MISSING(sc11qG3))&(MISSING(sc11qG4))
mxautdisp=1.
IF (MISSING(sc11qH1))&(MISSING(sc11qH2))&(MISSING(sc11qH3))&(MISSING(sc11qH4))
mxautassp=1.
IF (MISSING(sc11qI1))&(MISSING(sc11qI2))&(MISSING(sc11qI3))&(MISSING(sc11qI4))
mxautadm=1.
IF (MISSING(sc11qJ1))&(MISSING(sc11qJ2))&(MISSING(sc11qJ3))&(MISSING(sc11qJ4))
mxauttextb=1.
IF (MISSING(sc11qK1))&(MISSING(sc11qK2))&(MISSING(sc11qK3))&(MISSING(sc11qK4))
mxautccont=1.
```



```
IF (MISSING(sc11qL1))&(MISSING(sc11qL2))&(MISSING(sc11qL3))&(MISSING(sc11qL4))
mxautcoff=1.
EXE.
```

COMPUTE

```
mxaut=(mxauthir|mxautfir|mxautstsal|mxautsalin|mxautfbud|mxautabud|mxautassp|mxauttextb|mxautccont
|mxautcoff).
EXE.
```

```
IF (mxauthir~=1) xauthir=0.
IF (mxautfir~=1) xautfir=0.
IF (mxautstsal~=1) xautstsal=0.
IF (mxautsalin~=1) xautsalin=0.
IF (mxautfbud~=1) xautfbud=0.
IF (mxautabud~=1) xautabud=0.
IF (mxautdisp~=1) xautdisp=0.
IF (mxautassp~=1) xautassp=0.
IF (mxautadm~=1) xautadm=0.
IF (mxauttextb~=1) xauttextb=0.
IF (mxautccont~=1) xautccont=0.
IF (mxautcoff~=1) xautcoff=0.
EXE.
```

```
IF (mxauthir~=1) xcenthir=0.
IF (mxautfir~=1) xcentfir=0.
IF (mxautstsal~=1) xcentstsal=0.
IF (mxautsalin~=1) xcentsalin=0.
IF (mxautfbud~=1) xcentfbud=0.
IF (mxautabud~=1) xcentabud=0.
IF (mxautdisp~=1) xcentdisp=0.
IF (mxautassp~=1) xcentassp=0.
IF (mxautadm~=1) xcentadm=0.
IF (mxauttextb~=1) xcenttextb=0.
IF (mxautccont~=1) xcentccont=0.
IF (mxautcoff~=1) xcentcoff=0.
EXE.
```

```
IF (sc11qa3=1|sc11qa4=1) xcenthir=1.
IF (sc11qb3=1|sc11qb4=1) xcentfir=1.
IF (sc11qc3=1|sc11qc4=1) xcentstsal=1.
IF (sc11qd3=1|sc11qd4=1) xcentsalin=1.
IF (sc11qe3=1|sc11qe4=1) xcentfbud=1.
IF (sc11qf3=1|sc11qf4=1) xcentabud=1.
```

```
IF (sc11qg3=1|sc11qg4=1) xcentdisp=1.
IF (sc11qh3=1|sc11qh4=1) xcentassp=1.
IF (sc11qi3=1|sc11qi4=1) xcentadm=1.
IF (sc11qj3=1|sc11qj4=1) xcenttextb=1.
IF (sc11qk3=1|sc11qk4=1) xcentccont=1.
IF (sc11ql3=1|sc11ql4=1) xcentcoff=1.
EXE.
```

```
IF (sc11qa1=1|sc11qa2=1) xauthir=1.
IF (sc11qb1=1|sc11qb2=1) xautfir=1.
IF (sc11qc1=1|sc11qc2=1) xautstsal=1.
IF (sc11qd1=1|sc11qd2=1) xautsalin=1.
IF (sc11qe1=1|sc11qe2=1) xautfbud=1 .
IF (sc11qf1=1|sc11qf2=1) xautabud=1.
IF (sc11qg1=1|sc11qg2=1) xautdisp=1.
IF (sc11qh1=1|sc11qh2=1) xautassp=1.
IF (sc11qi1=1|sc11qi2=1) xautadm=1.
IF (sc11qj1=1|sc11qj2=1) xauttextb=1.
IF (sc11qk1=1|sc11qk2=1) xautccont=1.
IF (sc11ql1=1|sc11ql2=1) xautcoff=1.
EXE.
```

```
COMPUTE xauthir= xauthir-xcenthir.
COMPUTE xautfir= xautfir-xcentfir.
COMPUTE xautstsal=xautstsal-xcentstsal.
COMPUTE xautsalin=xautsalin-xcentsalin.
COMPUTE xautfbud= xautfbud-xcentfbud.
COMPUTE xautabud=xautabud-xcentabud.
COMPUTE xautdisp=xautdisp-xcentdisp.
COMPUTE xautassp=xautassp-xcentassp.
COMPUTE xautadm=xautadm-xcentadm.
COMPUTE xauttextb=xauttextb-xcenttextb.
COMPUTE xautccont=xautccont-xcentccont.
COMPUTE xautcoff=xautcoff-xcentcoff.
EXE.
```

* principal component analysis to combine different autonomy variables**.

```
WEIGHT BY newwgt.
```

```
FACTOR
```

```
/VARIABLES xauthir xautfir xautstsal xautsalin xautfbud xautabud
xauttextb xautccont xautcoff /MISSING LISTWISE /ANALYSIS xauthir xautfir
xautstsal xautsalin xautfbud xautabud xauttextb xautccont xautcoff
/PRINT INITIAL EXTRACTION ROTATION
/CRITERIA MINEIGEN(1) ITERATE(25)
```

```
/EXTRACTION PC  
/CRITERIA ITERATE(25)  
/ROTATION VARIMAX  
/METHOD=CORRELATION .
```

*** as can be seen in the output, three components can be extracted**.

*** create combined variables for the three extracted components**.

WEIGHT BY newwgt.

FACTOR

```
/VARIABLES xauthir xautfir xautstsal xautsalin /MISSING LISTWISE /ANALYSIS  
xauthir xautfir xautstsal xautsalin  
/PRINT INITIAL EXTRACTION ROTATION  
/CRITERIA MINEIGEN(1) ITERATE(25)  
/EXTRACTION PC  
/CRITERIA ITERATE(25)  
/ROTATION VARIMAX  
/SAVE REG(all)  
/METHOD=CORRELATION .
```

RENAME VARIABLES (FAC1_1 = FACS).

VARIABLE LABELS FACS 'School autonomy in staffing (hiring, firing, starting salary and salary increases)'.

WEIGHT BY newwgt.

FACTOR

```
/VARIABLES xautfbud xautabud /MISSING LISTWISE /ANALYSIS  
xautfbud xautabud  
/PRINT INITIAL EXTRACTION ROTATION  
/CRITERIA MINEIGEN(1) ITERATE(25)  
/EXTRACTION PC  
/CRITERIA ITERATE(25)  
/ROTATION VARIMAX  
/SAVE REG(all)  
/METHOD=CORRELATION .
```

RENAME VARIABLES (FAC1_1 = FACB).

VARIABLE LABELS FACB 'School autonomy in budgeting (formulating and allocating the school budget)'.

WEIGHT BY newwgt.

FACTOR

```
/VARIABLES xautccont xautcoff xauttextb /MISSING LISTWISE /ANALYSIS  
xautccont xautcoff xauttextb
```

```
/PRINT INITIAL EXTRACTION ROTATION
/CRITERIA MINEIGEN(1) ITERATE(25)
/EXTRACTION PC
/CRITERIA ITERATE(25)
/ROTATION VARIMAX
/SAVE REG(all)
/METHOD=CORRELATION .
```

RENAME VARIABLES (FAC1_1 = FACC).

VARIABLE LABELS FACC 'School autonomy in curriculum (courses offered, course content, textbooks used)'.
'

*** impute the three components.

WEIGHT BY W_FSTUWT.

```
AGGREGATE
/OUTFILE=*
MODE=ADDVARIABLES
/BREAK=CNT
/YFACS = MEAN(FACS) /YFACB = MEAN(FACB) /YFACC = MEAN(FACC).
```

```
COMPUTE XFACS=FACS.
COMPUTE XFACB=FACB.
COMPUTE XFACC=FACC.
COMPUTE MXFACS=0.
COMPUTE MXFACB=0.
COMPUTE MXFACC=0.
EXE.
```

```
DO IF (MISSING(FACS)).
  COMPUTE MXFACS=1.
  COMPUTE XFACS=YFACS.
END IF.
EXE.
```

```
DO IF (MISSING(FACB)).
  COMPUTE MXFACB=1.
  COMPUTE XFACB=YFACB.
END IF.
EXE.
```

```
DO IF (MISSING(FACC)).
  COMPUTE MXFACC=1.
```

```
    COMPUTE XFACC=YFACC.  
END IF.  
EXE.
```

```
COMPUTE MXFAC=(MXFACS|MXFACB|MXFACC).  
EXE.
```

```
***** BLOCK 6: RESOURCES
```

```
*** Human Resources***.
```

```
*** vacant science positions ***
```

```
COMPUTE xvscpos=(Sc10Q01=3).  
EXE.
```

```
COMPUTE mxvscpos=0.  
EXE.
```

```
DO IF (MISSING(Sc10q01)).  
    COMPUTE mxvscpos=1.  
    COMPUTE xvscpos=0.  
END IF.  
EXE.
```

```
*** STRATIO***.
```

```
COMPUTE MXSTRATIO=0.  
COMPUTE XSTRATIO=STRATIO.  
EXE.
```

```
DO IF (MISSING(STRATIO)) .  
    COMPUTE MXSTRATIO=1.  
    COMPUTE XSTRATIO=YSTRATIO.  
END IF.  
EXE.
```

```
*** TCSHORT***.
```

```
COMPUTE XTCSHORT=TCSHORT.  
COMPUTE MXTCSHORT=0.  
EXE.
```

```
DO IF (MISSING(TCSHORT)).  
    COMPUTE MXTCSHORT=1.
```

```
COMPUTE XTCSHORT=YTCSHORT.  
END IF.  
EXE.
```

```
** MATERIAL RESOURCES**.
```

```
*** XIRATCOM***.
```

```
COMPUTE MXIRATCOMP=0.  
COMPUTE XIRATCOMP=IRATCOMP.  
EXE.
```

```
DO IF (MISSING(IRATCOMP)) .  
  COMPUTE MXIRATCOMP=1.  
  COMPUTE XIRATCOMP=YIRATCOMP.  
END IF.  
EXE.
```

```
*** XSCMATED***.
```

```
COMPUTE XSCMATEDU=SCMATEDU.  
COMPUTE MXSCMATEDU=0.  
EXE.
```

```
DO IF (MISSING(SCMATEDU)).  
  COMPUTE MXSCMATEDU=1.  
  COMPUTE XSCMATEDU=YSCMATEDU.  
END IF.  
EXE.
```

```
***** EDUCATIONAL RESOURCES**.
```

```
*** XSCIPROM***.
```

```
COMPUTE XSCIPROM=SCIPROM.  
COMPUTE MXSCIPROM=0.  
EXE.
```

```
DO IF (MISSING(SCIPROM)).  
  COMPUTE MXSCIPROM=1.  
  COMPUTE XSCIPROM=YSCIPROM.  
END IF.  
EXE.
```

```
***LTSTOT, LTOSTOT, LTSTTOT Learning time****.
```

```
recode st31q01 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt1.
```

```

recode st31q02 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt2.
recode st31q03 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt3.
recode st31q04 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt4.
recode st31q05 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt5.
recode st31q06 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt6.
recode st31q07 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt7.
recode st31q08 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt8.
recode st31q09 (1 = 0) (2 = 1) (3 = 3) (4 = 5) (5 = 7) (sysmiss=sysmiss) into lt9.
exe.

```

```

compute ltsctot = lt1 + lt4 + lt7.
compute ltsstot = lt3 + lt6 + lt9.
compute ltostot = lt2 + lt5 + lt8.
exe.

```

WEIGHT BY W_FSTUWT.

```

AGGREGATE
/OUTFILE=*
MODE=ADDVARIABLES OVERWRITE=YES
/PRESORTED
/BREAK=CNT SCHOOLID
/xltsstot = MEAN(ltsstot)
/xltsctot = MEAN (ltsctot)
/xltostot = MEAN (ltostot).

```

WEIGHT
BY W_FSTUWT .

```

AGGREGATE
/OUTFILE=*
MODE=ADDVARIABLES OVERWRITE=YES
/PRESORTED
/BREAK=CNT
/yltsctot = MEAN (ltsctot)
/yltsstot = MEAN(ltsstot)
/yltostot = MEAN (ltostot).

```

```

COMPUTE mxltsctot=0.
COMPUTE mxltostot=0.
COMPUTE mxltsstot=0.
EXE.

```

```

IF (MISSING(xltsctot)) mxltsctot = 1 .
IF (MISSING(xltostot)) mxltostot = 1 .
IF (MISSING(xltsstot)) mxltsstot = 1 .
EXE.

```

```
IF (MISSING(xltsctot)) xltsctot=yltsctot.
IF (MISSING(xltostot)) xltostot=yltostot.
IF (MISSING(xltsstot)) xltsstot=yltsstot.
EXE.
```

```
COMPUTE mxlittot = mxltsstot | mxltostot | mxltsctot .
EXECUTE .
```

```
*** ANY SCIENCE COURSE***.
```

```
recode ST33Q12 (1 = 1) (2 = 0) (SYSMIS=SYSMIS) into scie1.
recode ST33Q22 (1 = 1) (2 = 0) (SYSMIS=SYSMIS) into scie2.
recode ST33Q32 (1 = 1) (2 = 0) (SYSMIS=SYSMIS) into scie3.
recode ST33Q42 (1 = 1) (2 = 0) (SYSMIS=SYSMIS) into scie4.
recode ST33Q52 (1 = 1) (2 = 0) (SYSMIS=SYSMIS) into scie5.
recode ST33Q62 (1 = 1) (2 = 0) (SYSMIS=SYSMIS) into scie6.
recode ST33Q72 (1 = 1) (2 = 0) (SYSMIS=SYSMIS) into scie7.
recode ST33Q82 (1 = 1) (2 = 0) (SYSMIS=SYSMIS) into scie8.
exe.
```

```
***Change countries all missing into "not taking" #####needs to be checked***.
```

```
AGGREGATE
/OUTFILE=*
MODE=ADDVARIABLES OVERWRITE=YES
/PRESORTED
/BREAK=CNT
/cs1= N(scie1)
/cs2= N(scie2)
/cs3 =N(scie3)
/cs4= N(scie4)
/cs5= N(scie5)
/cs6= N(scie6)
/cs7= N(scie7)
/cs8= N(scie8).
```

```
if cs1=0 scie1=0.
if cs2=0 scie2=0.
if cs3=0 scie3=0.
if cs4=0 scie4=0.
if cs5=0 scie5=0.
if cs6=0 scie6=0.
if cs7=0 scie7=0.
if cs8=0 scie8=0.
exe.
```



```
compute scieall = SUM(scie1,scie2,scie3,scie4,scie5,scie6,scie7,scie8).  
EXE.
```

```
COMPUTE anyscie=(scieall>=1).  
EXE.
```

* aggregate up to the school level***.

```
WEIGHT  
BY W_FSTUWT .
```

```
AGGREGATE  
/OUTFILE=*  
MODE=ADDVARIABLES  
/PRESORTED  
/BREAK=CNT SCHOOLID  
/XANYSCIE = MEAN(anyscie).
```

```
WEIGHT  
BY W_FSTUWT .
```

```
AGGREGATE  
/OUTFILE=*  
MODE=ADDVARIABLES  
/PRESORTED  
/BREAK=CNT  
/YANYSCIE = MEAN(anyscie).
```

```
COMPUTE MXANYSCIE=0.  
EXE.
```

```
DO IF (MISSING(XANYSCIE)).  
  COMPUTE MXANYSCIE=1.  
  COMPUTE XANYSCIE=YANYSCIE.  
END IF.  
EXE.
```

*** rescale ratio variables into percentages for better interpretability

```
COMPUTE XANYSCIE=XANYSCIE*100.  
COMPUTE YPRIVMAN=YPRIVMAN*100.  
COMPUTE YSCHLCOMP=YSCHLCOMP*100.  
EXE.
```

Compute standardised student weights.

WEIGHT OFF.

```
AGGREGATE /OUTFILE=* MODE=ADDVARIABLES overwrite = yes
/PRESORTED
/BREAK=CNT
/stpopwgt=SUM(w_fstuwt)
/stsmpsize=N .
```

```
compute st_wgt=(w_fstuwt/stpopwgt)*stsmpsize.
compute pisa2006=1.
exe.
```

```
AGGREGATE /OUTFILE=* MODE=ADDVARIABLES overwrite = yes
/PRESORTED
/BREAK=pisa2006
/totstsmpsize=N.
```

```
compute cntfac55=(totstsmpsize/55)/stsmpsize.
compute st_wgt55=st_wgt*cntfac55.
exe.
```

SAVE OUTFILE='C:\PISA\DATA2006\All.sav'.

*** SAVE STUDENT FILE FOR HLM (55 COUNTRIES)

```
GET FILE = 'C:\PISA\DATA2006\All.sav'.
EXE.
```

```
SAVE OUTFILE='C:\PISA\DATA2006\student55.sav'
/KEEP CNT SCHOOLID STIDSTD ST_WGT55 PV1SCIE PV2SCIE PV3SCIE PV4SCIE PV5SCIE
ESCS MESCS ESCS2 female NATIVE MNATIVE SAMELANG MSAMELANG
/COMPRESS.
```

*** SAVE SCHOOL FILE FOR HLM (55 COUNTRIES)

SORT CASES BY CNT SCHOOLID .

```
AGGREGATE
/OUTFILE='C:\PISA\DATA2006\school55.sav'
/PRESORTED
/BREAK=CNT SCHOOLID
/XESCS=MEAN(XESCS) /MXESCS=MEAN(MXESCS)
/xrural = MEAN(xrural) /xcity = MEAN(xcity) /mxrural = MEAN(mxrural)
/XSCHSIZE = MEAN(XSCHSIZE) /XSCHSIZ2 = MEAN(XSCHSIZ2) /MXSCHSIZE =
MEAN(MXSCHSIZE)
```

```

/XSTRATIO = MEAN(XSTRATIO) /MXSTRATIO = MEAN(MXSTRATIO)
/XTCSHORT =MEAN(XTCSHORT) /MXTCSHORT = MEAN(MXTCSHORT)
/XIRATCOMP = MEAN(XIRATCOMP) /MXIRATCOMP = MEAN(MXIRATCOMP)
/XSCMATEDU = MEAN(XSCMATEDU) /MXSCMATEDU = MEAN(MXSCMATEDU)
/xltsttot = MEAN(xltsttot) /xltstctot = MEAN(xltstctot) /xltostot = MEAN(xltostot) /mxltsttot =
MEAN(mxltsttot) /mxltostot = MEAN(mxltostot)
/XANYSCIE =MEAN(XANYSCIE) /MXANYSCIE = MEAN(MXANYSCIE)
/XSCIPROM = MEAN(XSCIPROM) /MXSCIPROM =MEAN(MXSCIPROM)
/XFACS=MEAN(XFACS) /XFACB=MEAN(XFACB) /XFACC=MEAN(XFACC)
/MXFACS=MEAN(MXFACS) /MXFACB=MEAN(MXFACB) /MXFACC=MEAN(MXFACC)
/MXFAC=MEAN(MXFAC)
/xacc1 = MEAN(xacc1) /xacc2 =MEAN(xacc2) /xacc3 = MEAN(xacc3) /xacc4 = MEAN(xacc4) /xacc5 =
MEAN(xacc5) /xacc6 = MEAN(xacc6)
/xacc7 = MEAN(xacc7) /xacc8 =MEAN(xacc8) /mxacc1 = MEAN(mxacc1) /mxacc2 =MEAN(mxacc2)
/mxacc3 = MEAN(mxacc3) /mxacc4 = MEAN(mxacc4)
/mxacc5 = MEAN(mxacc5) /mxacc6 = MEAN(mxacc6) /mxacc7 = MEAN(mxacc7) /mxacc8
=MEAN(mxacc8) /mxacc = MEAN(mxacc)
/xlosele = MEAN(xlosele) /xhisele = MEAN(xhisele) /mxhisele = MEAN(mxhisele)
/xabgr=MEAN(xabgr) /mxabgr=MEAN(mxabgr)
/XPRIVMAN = MEAN(XPRIVMAN) /MXPRIVMAN = MEAN(MXPRIVMAN) /XGOVFUND =
MEAN(XGOVFUND) /MXGOVFUND = MEAN(MXGOVFUND)
/XSCHLCOMP = MEAN(XSCHLCOMP) /mxschlcomp = MEAN(mxschlcomp)
/xpresspa = MEAN(xpresspa) /MXPRESSPA = MEAN(MXPRESSPA) .

```

** Note: country file for HLM is created in a different SPSS syntax file.

Updaed on March 20, 2008.

```

*****END*****
*****END*****
*****END*****

```