Kyrgyzstan Poverty Measurement Survey

Administered Fall 1996

Data Dictionary

Table of Contents

Introduction

The following is intended as a explanatory guide to the constructed variables used the analysis of the Living Standards and Measurement Survey conducted in Kyrgyzstan in the fall of 1996. The dictionary is divided into sections that correspond with the section divisions of the accompanying analysis report ("Kyrgystan Poverty Measurement Survey: Report 1997"), rather than the order followed in the survey itself. This is important to note, as they do differ.

General Variables

Variable	*Short Description and Code*
AGE0104	defines respondents who are between the ages of 1 and 4 (included) *if* S1AQ3 >= 1 *and* S1AQ3 <= 4 *then* AGE0104 = 1
AGE1517	defines respondents who are between the ages of 15 and 17 (included) *if* S1AQ3 >= 15 *and* S1AQ3 <= 17 *then* AGE1517 = 1
AGE1825	defines respondents who are between the ages of 18 and 25 (included) *if* S1AQ3 >= 18 *and* S1AQ3 <= 25 *then* AGE1825 = 1
AGE2639	defines respondents who are between the ages of 26 and 39 (included) *if* S1AQ3 >= 26 *and* S1AQ3 <= 39 *then* AGE2639 = 1
AGE4050	defines respondents who are between the ages of 40 and 50 (included) *if* S1AQ3 >= 40 *and* S1AQ3 <= 50 *then* AGE4050 = 1
AGE5160	defines respondents who are between the ages of 51 and 60 (included) *if* S1AQ3 >= 51 *and* S1AQ3 <= 60 *then* AGE5160 = 1
AGE60PL	defines respondents who are over the age of 60 *if* S1AQ3 > 60 *then* AGE60PL = 1
DAYSDIFF	difference between the date of the last visit to the household and the date of completion of Section 11 of the questionnaire. If this yields a negative value, or is greater than 100, then the variable is given the value of the

difference of the date the first round was completed and the date the second round was completed.

if {(S11_QD1, S11_QM1, S11_QY1) - (S11_QD2, S11_QM2, S11_QY2)} > 100 *or* < 0
then DAYSDIFF = (DD1, MM1, YY1) - (DD2, MM2, YY2)
otherwise DAYSDIFF = (S11_QD1, S11_QM1, S11_QY1) - (S11_QD2, S11_QM2, S11_QY2)

DEFLATOR computation of monthly (1996) and yearly (1997) deflators

if MM2 = 10 *then* DEFLATOR = 1.11397
if MM2 = 11 *then* DEFLATOR = 1.045
if MM2 = 1 *then* DEFLATOR = 0.967118
otherwise DEFLATOR = 1

ETH_KRYG defines those respondents who are of Kyrgyz ethnicity

if S1AQ_9 = 1 *then* ETH_KRYG =1

ETH_RUSS defines those respondents who are of Russian ethnicity

if S1AQ_9 = 2 *then* ETH_RUSS =1

ETH_SLAV defines those respondents who are of Slav ethnicity

if S1AQ_9 = 3 *or* S1AQ_9 = 5 *or* S1AQ_9 = 6
or S1AQ_9 = 7 *then* ETH_SLAV = 8

EXCHRT value of the soms/US dollar exchange rate for December 1996.

= 16.8325

Section 1: Poverty and Inequality

Variable	*Short Description and Code*
ADLTEQVL	aggregate of per capita adult equivalency caloric requirements for each household
ADULEQUI	adult equivalent caloric requirement per respondent *if* S1AQ4 = 1 *and* AGE > 17 *and* AGE > 60 *then* ADULEQUI = 1 *if* S1AQ4 = 2 *and* AGE > 17 *and* AGE > 55 *then* ADULEQUI = 8 *if* AGE >= 0 *and* AGE > 4 *then* ADULEQUI = 49 *if* AGE > 3 *and* AGE < 7 *then* ADULEQUI = 64 *if* AGE > 6 *and* AGE < 14 *then* ADULEQUI = 78 *if* AGE > 13 *and* AGE < 18 *then* ADULEQUI = 89 *if* AGE > 13 *and* AGE < 18 *then* ADULEQUI = 89 *if* S1AQ4 = 1 *and* AGE > 59 *or* S1AQ4 = 2 *and* AGE > 54 *then* ADULEQUI = 78
LNPLPPC	(LNRQKCPC - 4.913)/0.943
LNPVLPPC	(LNRQKCPC - 7.345)/0.698
LNRQKCPC	the log of required household kilo calories LN (REQKCPC)
NONPOORE	determination of the non-poor respondents *if* (TOTEXPPC >= PLINEPPC) *then* NONPOORE = 1 *otherwise* NONPOORE = 0
PLINVPPC	determination of the poverty line using total expenditure on *food* EXP (LNPLVPPC)
PLINEPPC	EXP (LNPLPPC)

POORE determination of the poor respondents

if (TOTEXPPC < PLINEPPC)
then POORE = 1
otherwise POORE = 0

REQKCPC required kilo calories per capita calculation

2750 * 30 * ADLTEQVL/HHCOMP

VPOOR determination of the very-poor respondents

if (TOTEXPPC < PLINVPPC)
then VPOOR = 1
otherwise VPOOR = 0

Section 2: Expenditure[1]

Variable	*Short Description and Code*
ALCTOB	monthly expenditure alcohol and tobacco products {SUM (A1110_77 TO A1110_80, A1202_03, B1103_07 TO B11203_09)} * DEFLATOR * 30/DAYSDIFF
BREADS	monthly expenditure on grains and grain derivatives such as breads, pasta and flour SUM (A1110_01 TO A1110_08) * DEFLATOR * 30/DAYSDIFF
DAIRY	monthly expenditure on dairy products {SUM (A1110_64 TO A1110_70, A1110_72)} * DEFLATOR * 30/DAYSDIFF
EATNGOUT	monthly expenditure on food consumption outside the home {SUM (B1103_01 TO B1103_06)} * DEFLATOR * 30/DAYSDIFF
EGGS	monthly expenditure on eggs (A1110_63) * DEFLATOR * 30/DAYSDIFF
FATS	monthly expenditure on fat products (oils, lards etc.) {SUM (A1110_71, A1110_73, A1110_74)} * DEFLATOR * 30/DAYSDIFF
FISHBIRD	monthly expenditure on fish, and wild game and birds, and preserved meat and fish products {SUM (A1110_59 TO A1110_62)} * DEFLATOR * 30/DAYSDIFF
FOODGIF	monthly value of food received as gifts corrected for outliers (replaced by median food consumption) *if* FOODRECE > 388 *then* FOODGIF = 388 *otherwise* FOODGIF = FOODRECE
FOODRECE	monthly value of food received as gifts

[1] This section contains many of the constructed variables used in the calculation of expenditure. However, not ALL constructed expenditure variables can be found in this section. Those variables that correspond to specific sections can be found in those sections. For example, expenditure in education variables, are documented in Section 6: Education.

	S11BQ23 * DEFLATOR * 30/DAYSDIFF
FOODT	total monthly expenditures on all home consumption, eating out, alcohol and tobacco and the value of food received as a gift
	SUM (FOOD, EATNGOUT, ALCTOB, FOODGIF)
FRUITS	monthly expenditure on fruits
	{SUM (A1110_28 TO A1110_42, A1110_44, A1110_47)} * DEFLATOR * 30/DAYSDIFF
MEAT	monthly expenditure on meat
	{SUM (A1110_48 TO A1110_58)} * DEFLATOR * 30/DAYSDIFF
OTHERF	monthly expenditure on misc. food items (various drinks and salt)
	{SUM (A1110_75, A1110_76, A1110_81, A1110_82, A1110_85)} * DEFLATOR * 30/DAYSDIFF
POTATOES	monthly expenditure on potatoes
	A1110_09 * DEFLATOR * 30/DAYSDIFF
SUGAR	monthly expenditure on sweet items
	{SUM (A1110_45, A1110_46, A1110_83, A1110_84, A1110_86 TO A1110_89)} * DEFLATOR * 30/DAYSDIFF
VEGET	monthly expenditure on vegetables
	{SUM (A1110_10 TO A1110_27, A1110_43)} * DEFLATOR * 30/DAYSDIFF
CLOTHING	expenditures on clothing (ready-made, fabric costs and tailoring)
	SUM (B1202_01 TO B1202_06) * DEFLATOR/12
VALDU	group of variables that replaces C1204_01 TO C1204_22, replacing the top and bottom 10 percentiles with the median values
DURABLE	expenditure in the last year on durable goods
	SUM (B1202_07 TO B1202_10, VALDU_01 TO VALDU_22) * DEFLATOR/12
SOTHNF	expenditure on non-food items for the last month
	SUM (A1202_05 TO A1202_07, A1202_11) * DEFLATOR * 30/DAYSDIFF

BRTHCONT expenditure on birth control for the last month

S8AQ72 * DEFLATOR

OTHNF total expenditure on non-food times for the last month

SUM (SOTHNF, BRTHCONT)

SSERVICE expenditure for those services reported on a monthly basis

SUM (A1202_09, A1202_10, A1202_12 TO A1202_15, A1202_19 TO A1202_21) * DELFATOR * 30/DAYSDIFF

SSERVANN expenditure for those services reported on an annual basis, calculated monthly

SUM (B1202_13, B1202_17, B1202_19, B1202_24) * DEFLATOR/12

SERVICE total expenditure on services for the last month

SUM (SSERVICE, SSERVANN)

REMIANN the vector of variables from E1203_01 TO E1203_14

REMIMTH the vector of variables from E1204_01 TO E1204_14

REMICOMP the vector of variables from REMIC_01 TO REMIC_14, where the following applies

if REMIANN/12 > REMIMTH
then REMICOMP = REMIMTH/12
otherwise REMICOMP = REMIMTH

REMITTAN total value of remittances

SUM (REMIC_01 TO REMIC_14) * DEFLATOR

SAVINGS money saved during the last month

S4CQ2 * DEFLATOR

OTHEREX1 other household expenditures reported on a monthly basis

SUM (A1202_04, A1202_08, A1202_16, A1202_17, A1202_18) * DEFLATOR

OTHEREX2 other household expenditures reported on an annual basis

SUM (B1202_16, B1202_18, B1202_20TO B1202_23,B1202_25 TO B1202_31, S3CQ11) * DEFLATOR/12

OTHEREX3 other household expenditures reported on a monthly basis

SUM (S3AQ6, S3AQ10, S5CQ_38) * DEFLATOR

OTHEREX total other household expenditures for the last month

SUM (OTHEREX1, OTHEREX2, OTHEREX3)

TAX expenditure on non-agricultural taxes

SUM (B1202_14, B12020_15) * DEFLATOR

NFOODT total non-food expenditures

SUM (CLOTHING, DURABLE, EDUCAT, HEALTH, OTHNF, SERVICE, RENT, UTILITY, REMITTAN, SAVINGS, OTHEREX, TAX)

TEXPEND total monthly household expenditures

SUM (FOODT, NFOODT)

TOTEXP total monthly expenditures is the sum of monthly household expenditures, the value of the consumption of home produced agricultural products and the consumption of home produced animal products

SUM (TEXPEND, HARVHC, ANPRODHC)

TOTEXPPC total monthly expenditures per capita

TOTEXP/HHCOMP

FOODTPC total monthly food expenditures per capita

FOODT/HHCOMP

EX1NF_01-18 vector calculations to define monthly non-farm expenditures

if (C1002201 TO C1002218) = 3
then EX1NF_01 TO EX1NF_18 = (C1002101 TO C1002118) * 30
if (C1002201 TO C1002218) = 4
then EX1NF_01TO EX1NF_18 = (C1002101 TO C1002118) * 4.3
if (C1002201 TO C1002218) = 6
then EX1NF_01TO EX1NF_18 = (C1002101 TO C1002118)/3
if (C1002201 TO C1002218) = 7
then EX1NF_01TO EX1NF_18 = (C1002101 TO C1002118)/6

if (C1002201 TO C1002218) = 8
then EX1NF_01TO EX1NF_18 = (C1002101 TO C1002118)/12
otherwise EX1NF_01TO EX1NF_18 = C1002101 TO C1002118

EXPNTOT total monthly non-farm expenditures

SUM (EX1NF_01 TO EX1NF_18) * DEFLATOR

Section 3: Income

Variable	*Short Description and Code*
YPCASH	defines respondents' monthly salary S5CQ_14 * DEFLATOR
YBONUS	defines respondents' monthly "other" pay (such as trips and bonuses) *if* S5CQ20_2 = 3 *then* YBONUS = S5CQ20_1 * 30 *if* S5CQ20_2 = 4 *then* YBONUS = S5CQ20_1 * 4.3 *if* S5CQ20_2 = 8 *then* YBONUS = S5CQ20_1 * 1/12 *otherwise* YBONUS = S5CQ20_1
YFOOD	value of food/goods subsidy received monthly through work *if* S5CQ23_2 = 3 *then* YFOOD = S5CQ23_1 * 30 *if* S5CQ23_2 = 4 *then* YFOOD = S5CQ23_1 * 4.3 *if* S5CQ23_2 = 8 *then* YFOOD = S5CQ23_1 * 1/12 *otherwise* YFOOD = S5CQ23_1
YTRANS	value of transport subsidy received monthly through work *if* S5CQ25_2 = 3 *then* YTRANS = S5CQ25_1 * 30 *if* S5CQ25_2 = 4 *then* YTRANS = S5CQ25_1 * 4.3 *if* S5CQ25_2 = 8 *then* YTRANS = S5CQ25_1 * 1/12 *otherwise* YTRANS = S5CQ25_1
YHOUSING	value of housing subsidy received monthly through work *if* S5CQ27_2 = 3 *then* YHOUSING = S5CQ27_1 * 30 *if* S5CQ27_2 = 4 *then* YHOUSING = S5CQ27_1 * 4.3 *if* S5CQ27_2 = 8 *then* YHOUSING = S5CQ27_1 * 1/12 *otherwise* YHOUSING = S5CQ27_1
YSANAT	value of sanitarium/spa/rest home subsidy received monthly through work *if* S5CQ29_2 = 3 *then* YSANAT = S5CQ29_1 * 30 *if* S5CQ29_2 = 4 *then* YSANAT = S5CQ29_1 * 4.3 *if* S5CQ29_2 = 8 *then* YSANAT = S5CQ29_1 * 1/12 *otherwise* YSANAT = S5CQ29_1
SUBSIDY1	value of all subsidies received from work

SUBSIDY1 = SUM (YBONUS, YFOOD, YTRANS, YHOUSING, YSANAT) * DEFLATOR

YSCASH value of salary and in-kind payments received through secondary work

S5DQ11 * DEFLATOR

YSOTHER value of all funds received through other work

S5DQ14 * DEFLATOR

SAL value of all salary

SAL = SUM (YPCASH, YSCASH, YSOTHER)

RENCASH monthly value of cash payments received in the first business for goods and services

if D10022_1 = 3 *then* RENCASH = D10021_1 * 30
if D10022_1 = 4 *then* RENCASH = D10021_1 * 4.3
if D10022_1 = 6 *then* RENCASH = D10021_1/3
if D10022_1 = 7 *then* RENCASH = D10021_1/6
if D10022_1 = 8 *then* RENCASH = D10021_1/12
otherwise RENCASH = D10021_1

RENBARTR monthly value of payments received in the first business in the form of bartered goods of services

if D10022_2 = 3 *then* RENBARTR = D10021_2 * 30
if D10022_2 = 4 *then* RENBARTR = D10021_2 * 4.3
if D10022_2 = 6 *then* RENBARTR = D10021_2/3
if D10022_2 = 7 *then* RENBARTR = D10021_2/6
if D10022_2 = 8 *then* RENBARTR = D10021_2/12
otherwise RENBARTR = D10021_2

RENSLEAS monthly value of payments received in the first business through the sale of business assets

if D10022_3 = 3 *then* RENSLEAS = D10021_3 * 30
if D10022_3 = 4 *then* RENSLEAS = D10021_3 * 4.3
if D10022_3 = 6 *then* RENSLEAS = D10021_3/3
if D10022_3 = 7 *then* RENSLEAS = D10021_3/6
if D10022_3 = 8 *then* RENSLEAS = D10021_3/12
otherwise RENSLEAS = D10021_3

RENRNTAS monthly value of payments received in the first business through the rental of business assets

if D10022_4 = 3 *then* RENRNTAS = D10021_4 * 30
if D10022_4 = 4 *then* RENRNTAS = D10021_4 * 4.3
if D10022_4 = 6 *then* RENRNTAS = D10021_4/3
if D10022_4 = 7 *then* RENRNTAS = D10021_4/6
if D10022_4 = 8 *then* RENRNTAS = D10021_4/12
otherwise RENRNTAS = D10021_4

RENOTHER monthly value of payments received in the first business through other means

if D10022_5 = 3 *then* RENOTHER = D10021_5 * 30
if D10022_5 = 4 *then* RENOTHER = D10021_5 * 4.3
if D10022_5 = 6 *then* RENOTHER = D10021_5/3
if D10022_5 = 7 *then* RENOTHER = D10021_5/6
if D10022_5 = 8 *then* RENOTHER = D10021_5/12
otherwise RENOTHER = D10021_5

RENVNTOT total monthly value of payments received in the first business

SUM (RENCASH, RENBARTR, RENSLEAS, RENRNTAS, RENOTHER) * DEFLATOR

INDEPINC money earned through self-employment or independent entrepreneurship in the last month

SUM (S5CQ10, S5DQ8, S5FQ10, S5HQ7) * DEFLATOR

LANDINC monthly earnings from rental of land to private individuals

SUM (A0909_01 TO A0909_08) * DEFLATOR/12

LEASINC monthly earnings from rental of farming equipment to private individuals

SUM (J0905_01 TO J0905_09) * DEFLATOR/12

REALRENT monthly earnings from rental of real estate assets

SUM (D1209_01 TO D120910_10)/12

RENTIMP monthly earnings from rental of real estate assets and imputed rent from private dwelling ownership

SUM (IMPRENT, REALRENT) * DEFLATOR

RENTINC monthly total rental income

SUM (RENTIMP, LANDINC, LEASINC)

TOTIND total net income

SUM (REVNTOT, INDEPINC, RENTINC, (-EXPNTOT))

PROPERTY total income received from other sources (such as inheritance, gifts etc., not including aid from NGOs)

SUM (B1302_01 TO B1302_15, -(B1302_12)) * DEFLATOR

PENS total income received from pensions (eg: old age, disability, etc.)

SUM (S5JQ1_2, S5JQ2_2, S5JQ3_2) * DEFLATOR

BENF total monthly income received from benefits

S5JQ6_2 * DEFLATOR

UNBENF total monthly payment received from unemployment benefits, the higher value of two reported unemployment benefit income questions

if S5BQ_19 > S5JQ5_2
then UNBENF = S5BQ_19 * DEFLATOR
otherwise UNBENF = S5JQ5_2 * DEFLATOR

SICKNESS total monthly payment received from temporary disability/sickness allowances

S5JQ4_2 * DEFLATOR

SUBSIDY2 total monthly payment received from local authorities (eg: school allowance, scholarships, Chernobyl compensation)

SUM (S3BQ28/12, S3BQ_29/12, S5JQ7_2, S5JQ8_2) * DELFATOR

CHARITY1 total monthly monies received from relatives and friends

SUM (A1304_01 TO A1304_14, S3BQ_30/12) * DELFATOR

CHARITY2 total monthly monies received from other groups

B1302_12 * DELFATOR

Section 4: Labor

Variable *Short Description and Code*

JOBSERCH job search method used by the unemployed, by labor force category

if LFSTAT8 = 2 *and* S5BQ11_1
then JOBSERCH = 1
if LFSTAT8 = 2 *and* S5BQ11_2
then JOBSERCH = 2
if LFSTAT 8= 2 *and* S5BQ11_3
then JOBSERCH = 3
if LFSTAT8 = 2 *and* S5BQ11_4
then JOBSERCH = 4
if LFSTAT8 = 2 *and* S5BQ11_5
then JOBSERCH = 5

LASTPAY number of months since an employee last received payment

if S5CQ16_2 = YY1 *and* LFSTAT = 1
then LASTPAY = (MM1_2 - S5CQ16_1)
if S5CQ16_2 > 0 *and* S5CQ16_2 < YY1_2 *and* LFSTAT = 1
then LASTPAY = 12
if S5FQ17_2 = YY1_2 *and* LFSTAT = 1
then LASTPAY = (MM1_2 - S5FQ17_1)
if S5FQ17_2 > 0 *and* S5FQ17_2 < YY1_2 *and* LFSTAT = 1
then LASTPAY = 12

LFSTAT8 classification of those respondents who are of working age, and who are either employed or unemployed, or not in the labor force

if POP = 1 *and* (LABOR8 = 1 *or* LABOR8 = 2 *or* LABOR8 = 3 *or* LABOR8 = 4 *or* LABOR8 = 5 *or* LABOR8 = 6 *or* LABOR8 = 7 *or* LABOR8 = 8 *or* LABOR8 = 9 *or* LABOR8 = 10 *or* LABOR8 = 11 *or* LABOR8 = 12 *or* LABOR8 = 13)
then LFSTAT8 = 1
if POP = 1 *and* LABOR8 = 14
then LFSTAT8 = 2
if POP = 0 *and* LABOR8 = 15
then LFSTAT8 = 3
otherwise LFSTAT8 = 0

EMPLOYER identification of the type of employer

if LFSTAT8 = 1 *and* (S5CQ_13 = 1 *or* S5FQ_12 = 1)

then EMPLOYER = 1
if LFSTAT8 = 1 *and* (S5CQ_13 = 2 *or* S5FQ_12 = 2)
then EMPLOYER = 2
if LFSTAT8 = 1 *and* (S5CQ_13 = 3 *or* S5FQ_12 = 3)
then EMPLOYER = 3
if LFSTAT8 = 1 *and* (S5CQ_13 = 4 *or* S5FQ_12 = 4)
then EMPLOYER = 4
if LFSTAT8 = 1 *and* (S5CQ_13 = 5 *or* S5FQ_12 = 5)
then EMPLOYER = 5
if LFSTAT8 = 1 *and* (S5CQ_13 = 6 *or* S5FQ_12 = 6)
then EMPLOYER = 6
if LFSTAT8 = 1 *and* (S5CQ_13 = 7 *or* S5FQ_12 = 7)
then EMPLOYER = 7
if LFSTAT8 = 1 *and* (S5CQ_13 = 8 *or* S5FQ_12 = 8)
then EMPLOYER = 8
if LFSTAT8 = 1 *and* (S5CQ_13 = 9 *or* S5FQ_12 = 9)
then EMPLOYER = 9

LABFORC8 — those respondents who are in the labor force—whether employed or unemployed

if LFSTAT8 = 1 *or* LFSTAT8 = 2
then LABFORC8 = 1
otherwise LABFORC8 = 0

LABOR8 — classification of all respondents (whether of working age or not) as employed, unemployed, or not in labor force, with iterations to cover interviewer, keyer or respondent error

if S5AQ3 = 1 *and* (S5CQ2 = 0 *or* S5CQ3 = 0) *and* (S5CQ4 = 1 *or* S5CQ4 = 2 *or* S5CQ4 = 5 *or* S5CQ4 = 6 *or* S5CQ4 = 7)
then LABOR8 = 1
if S5AQ3 = 2 *and* S5AQ4 = 1 *and* (S5CQ2 > 0 *or* S5CQ3 > 0)
then LABOR8 = 2
if S5AQ3 = 2 *and* S5AQ4 = 1 *and* (S5CQ4 = 1 *or* S5CQ4 = 2 *or* S5CQ4 = 5 *or* S5CQ4 = 6 *or* S5CQ4 = 7)
then LABOR8 = 3
if SYSMIS (S5AQ3) *and* (S5CQ2 > 0 *or* S5CQ3 > 0) *and* S5CQ9 = 2
then LABOR8 = 4
if SYSMIS (S5AQ3) *and* (S5CQ2 > 0 *or* S5CQ3 > 0) *and* S5CQ9 = 1
then LABOR8 = 5
if S5AQ3 ≠ 2 *and* (S5CQ2 > 0 *or* S5CQ3 > 0) *and* S5CQ9 = 1
then LABOR8 = 6
if S5AQ3 ≠ 2 *and* (S5CQ2 > 0 *or* S5CQ3 > 0) *and* S5CQ9 = 2
then LABOR8 = 7

if S5AQ6 = 1 *and* (S5CQ2 > 0 *or* S5CQ3 > 0)
then LABOR8 = 8
if S5AQ5 = 1 *and* (S5CQ2 > 0 *or* S5CQ3 > 0)
then LABOR8 = 9
if S5AQ3 = 1 *and* S5CQ9 = 1
then LABOR8 = 10
if S5AQ3 = 1 *and* S5CQ9 = 2
then LABOR8 = 11
if S5AQ3 = 1 *and* (S5CQ2 = 0 *and* S5CQ3 = 0) *and* (S5CQ4 = 1 *or* S5CQ4 = 2 *or* S5CQ4 = 5 *or* S5CQ4 = 6 *or* S5CQ4 = 7)
then LABOR8 = 12
if S5AQ3 = 1 *and* (S5CQ2 > 0 *or* S5CQ3 > 0)
then LABOR8 = 13
if S5AQ3 = 2 *and* S5AQ4 = 2 *and* S5AQ5 = 2 *and* S5AQ6 = 2 *and* S5BQ1 = 12 *and* (S5BQ6 = 1 *or* S5BQ7 = 1 *or* S5BQ7 = 2 *or* S5BQ7 = 14)
then LABOR8 = 14
if S5AQ3 = 2 *and* S5AQ4 = 2 *and* S5AQ5 = 2 *and* S5AQ6 = 2 *and* (S5BQ6 = 2 *and* (S5BQ7 = 3 *or* S5BQ7 = 4 *or* S5BQ7 = 5 *or* S5BQ7 = 6 *or* S5BQ7 = 7 *or* S5BQ7 = 8 *or* S5BQ7 = 9 *or* S5BQ7 = 10 *or* S5BQ7 = 11 *or* S5BQ7 = 12 *or* S5BQ7 = 13)
then LABOR8 = 15

LEFTJOB8 number of months since the respondents have left their job and been interviewed

if S5BQ4_2 = YY1_2 *and* LFSTAT8 = 2
then LEFTJOB8 = SUM (MM1_2, (-S5BQ4_1))
if LFSTAT8 = 2 *and* SUM (YY1_2, -S5BQ4_2) = 1
then LEFTJOB8 = SUM (MM1_2, (12 - S5BQ4_1))
if LFSTAT8 = 2 *and* SUM (YY1_2, (-S5BQ4_2)) > 1
then LEFTJOB8 = 13

LJ8 aggregated categories of LEFTJOB8

if (LEFTJOB8 = 0 *or* LEFTJOB8 = 1 *or* LEFTJOB8 = 2 *or* LEFTJOB8 = 3 *or* LEFTJOB8 = 4)
then LJ8 = 03
if (LEFTJOB8 = 5 *or* LEFTJOB8 = 6 *or* LEFTJOB8 = 7)
then LJ8 = 46
if (LEFTJOB8 = 8 *or* LEFTJOB8 = 9 *or* LEFTJOB8 = 10 *or* LEFTJOB8 = 11 *or* LEFTJOB8 = 12)
then LJ8 = 711
if (LEFTJOB8 > 0)
then LJ8 = 12

OCCUP — those employed respondents categorized according to broad occupational categories

if LFSTAT8 = 1 *and* S5CQ1 = 1 *and* S5CQ12 = 2
then OCCUP = 1
if LFSTAT8 = 1 *and* S5CQ1 = 1 *and* S5CQ12 = 4
then OCCUP = 2
if LFSTAT8 = 1 *and* S5CQ1 = 9 *and* S5CQ12 = 1
then OCCUP = 3
if LFSTAT8 = 1 *and* S5CQ1 = 9 *and* S5CQ12 = 2
then OCCUP = 4
if LFSTAT8 = 1 *and* S5CQ1 = 3 *and* S5CQ12 = 2
then OCCUP = 5
if LFSTAT8 = 1 *and* S5CQ12 = 1
then OCCUP = 6
if LFSTAT8 = 1 *and* S5CQ12 = 2
then OCCUP = 7
if LFSTAT8 = 1 *and* S5FQ1 = 1 *and* S5FQ11 = 2
then OCCUP = 1
if LFSTAT8 = 1 *and* S5FQ1 = 1 *and* S5FQ11 = 4
then OCCUP = 2
if LFSTAT8 = 1 *and* S5FQ1 = 9 *and* S5FQ11 = 1
then OCCUP = 3
if LFSTAT8 = 1 *and* S5FQ1 = 9 *and* S5FQ11 = 2
then OCCUP = 4
if LFSTAT8 = 1 *and* S5FQ1 = 3 *and* S5FQ11 = 2
then OCCUP = 5
if LFSTAT8 = 1 *and* S5FQ11 = 1
then OCCUP = 6
if LFSTAT8 = 1 *and* S5FQ11 = 2
then OCCUP = 7
if LFSTAT8 = 1
then OCCUP = 8

ECSTAT8 — economic status of respondents

if LFSTAT8 = 1 *and* S5CQ9 = 1 OR S5AQ5 = 1 OR S5AQ6 = 1
then ECSTAT8 = 1
if LFSTAT8 = 1 *and* S5CQ9 = 2
then ECSTAT8 = 2
if LFSTAT8 = 1
then ECSTAT8 = 3
if LFSTAT8 = 2
then ECSTAT8 = 4

if LFSTAT8 = 3 *and* POP = 0 *and* S1AQ4 = 1 *and* S1AQ3 > 60
then ECSTAT8 = 5
if LFSTAT8 = 3 *and* POP = 0 *and* S1AQ4 = 2 *and* S1AQ3 > 55
then ECSTAT8 = 5.
if LFSTAT8 = 3 *and* S5BQ7 = 6
then ECSTAT8 = 6
if LFSTAT8 = 3 *and* S5BQ7 = 12
then ECSTAT8 = 7
if LFSTAT8 = 3 *and* S5BQ7 = 7
then ECSTAT8 = 8
if LFSTAT8 = 3 *and* S5BQ7 = 3
then ECSTAT8 = 9
if LFSTAT8 = 3 *and* S5BQ7 = 4 *or* S5BQ7 = 5 *or* S5BQ7 = 9 *or* S5BQ7 = 10 *or* S5BQ7 = 11 *or* S5BQ7 = 13
then ECSTAT8 = 10.
otherwise ECSTAT8 = 0.

Section 5: Agriculture

Variable	*Short Description and Code*
CSALE_01-45	total unadjusted revenue received by crop type per month
	(B0907_01 TO 45 * B0909_01 TO 45) / 12
CROPINC	total revenue received from all crops per month
	SUM(CSALE_01 TO CSALE_45) * DEFLATOR
CSTOR_01-45	total unadjusted value of stored crops by crop type per month
	(B0918_01 TO 45 * B0909_01 TO 45) / 12
CRPSTORE	total value of all stored crops per month
	SUM (CSTOR_01 TO CSTOR_45) * DEFLATOR
EXPSEEDS	monthly total expenditure on seeds and young plants monthly
	SUM (C0901_01 TO C0901_12) / 12
EXPFERT	monthly total expenditure on mineral fertilizer
	SUM (C0905_01 TO C0905_12) / 12
EXPMANUR	monthly total expenditure on organic manure
	SUM (C0909_01 TO C0909_12) / 12
EXPINS	monthly total expenditure herbicide or insecticide
	SUM (C0913_01 TO C0913_12) / 12
EXPTRANS	monthly total expenditure on crop transport
	SUM (C0917_01 TO C0917_12) / 12
EXPPACKG	monthly total expenditure on packaging
	SUM (C0916_01 TO C0916_12) / 12

EXPLABOR	monthly total expenditure on farm labor
	SUM (C0919_01 TO C0919_12) / 12
EXPOTH	monthly total expenditure on other farm expenses
	SUM (C0920_01 TO C0920_12) / 12
EXPCRTAX	monthly total expenditure on land/livestock/VA/agricultural taxes
	SUM (C0921_01 TO C0921_12) / 12
EXPTOCRP	monthly total adjusted expenditures on crops
	SUM (EXPSEEDS, EXPFERT, EXPMANUR, EXPINS, EXPTRANS, EXPPACKG, EXPLABOR, EXPOTH, EXPCRTAX) * DEFLATOR
CGIF_01-45	monthly total unadjusted value of crops given away by crop type
	(B0910_01 TO 45 * B0909_01 TO 45) / 12
CROPGIFT	monthly total value of crops given away
	SUM (CGIF_01 TO CGIF_45) * DEFLATOR
LANRENCR	monthly total expenditure on land rental
	SUM (A0907_01 TO 07) * DEFLATOR / 12
LDIMPRCR	monthly total imputed rent of cropland
	SUM (A0903_01 TO 07) * DEFLATOR / 12
CROPSALE	monthly net value of crops produced
	SUM (CROPINC, CRPSTORE, (-EXPTOCRP), (-LANRENCR))
PSCOW	average income received from the sale of a cow
	SUM (E0907_01/E0906_01) * DEFLATOR
PSPIG	average income received from the sale of a pig
	SUM (E0907_02/E0906_02) * DEFLATOR
PSSHEEP	average income received from the sale of a sheep

SUM (E0907_03/E0906_03) * DEFLATOR

PSGOAT average income received from the sale of a goat

SUM (E0907_04/E0906_04) * DEFLATOR

PSHORSE average income received from the sale of a horse

SUM (E0907_05/E0906_05) * DEFLATOR

PSDONKY average income received from the sale of a donkey or mule (at farm price of existing stock)

SUM (E0907_06/E0906_06) * DEFLATOR

PSCHICN average income received from the sale of a chicken or rooster

SUM (E0907_07/E0906_07) * DEFLATOR

PSTURKY average income received from the sale of a turkey

SUM (E0907_08/E0906_08) * DEFLATOR

PSDUCK average income received from the sale of a duck, goose or other poultry

SUM (E0907_09/E0906_09) * DEFLATOR

PSRABBT average income received from the sale of a rabbit

SUM (E0907_010/E0906_010) * DEFLATOR

PSMINK average income received from the sale of a mink (farm price of existing stock used)

SUM (E0907_011E0906_011) * DEFLATOR

PSOTHFU average income received from the sale of an other fur animal (farm price of existing stock used)

SUM (E0907_012/E0906_012) * DEFLATOR

PSBEES average income received from the sale of a bee (farm price of existing stock used)

SUM (E0907_13/E0906_13) * DEFLATOR

PSOTHAN	average income received from the sale of other animals (farm price of existing stock used) SUM (E0907_14/E0906_14) * DEFLATOR
PBCOW	average price paid for the purchase of a cow SUM (E0909_01/E0908_01) * DEFLATOR
PBPIG	average price paid for the purchase of a pig SUM (E0909_02/E0908_02) * DEFLATOR
PBSHEEP	average price paid for the purchase of a sheep SUM (E0909_03/E0908_03) * DEFLATOR
PBGOAT	average price paid for the purchase of a goat SUM (E0909_04/E0908_04) * DEFLATOR
PBHORSE	average price paid for the purchase of a horse SUM (E0909_05/E0908_05) * DEFLATOR
PBDONKY	average price paid for the purchase of a donkey or mule SUM (E0909_06/E0908_06) * DEFLATOR
PBCHICN	average price paid for the purchase of a chicken or rooster SUM (E0909_07/E0908_07) * DEFLATOR
PBTURKY	average price paid for the purchase of a turkey SUM (E0909_08/E0908_08) * DEFLATOR
PBDUCK	average price paid for the purchase of a duck, goose or other poultry SUM (E0909_09/E0908_09) * DEFLATOR
PBRABBT	average price paid for the purchase of a rabbit SUM (E0909_10/E0908_10) * DEFLATOR
PBMINK	average price paid for the purchase of a mink SUM (E0909_11/E0908_11) * DEFLATOR

PBOTHFU average price paid for the purchase of an other fur animal (farm prices of existing stock used)

SUM (E0909_12/E0908_12) * DEFLATOR

PBBEES average price paid for the purchase of a bee(farm prices of existing stock used)

SUM (E0909_13/E0908_13) * DEFLATOR

PBOTHAN average price paid for the purchase of other animals (farm prices of existing stock used)

SUM (E0909_14/E0908_14) * DEFLATOR

Calculating Media/Mean Values for Animal Sales (per Animal)

PSPIGS MEDIAN (PSPIG ALL)

PSSHEEPS MEAN (PSSHEEP ALL)

PSGOATS MEDIAN (PSGOAT ALL)

PSHORSES MEAN (PSHORSE ALL)

PSDONKYS MEDIAN (PSDONKY ALL)

PSCHICNS MEDIAN (PSCHICN ALL)

PSTURKYS MEDIAN (PSTURKY ALL)

PSDUCKS MEDIAN (PSDUCK ALL)

PSRABBTS MEAN (PSRABBT ALL)

PSMINKS MEDIAN (PSMINK ALL)

PSOTHFUS MEDIAN (PSOTHFU ALL)

PSBEESS MEDIAN (PSBEES ALL)

PSOTHANS MEDIAN (PSOTHAN ALL)

Calculating Media/Mean Values for Animal Sales and Purchases (per Animal)

PBPIGS MEDIAN (PBPIG ALL)

PBSHEEPS MEAN (PBSHEEP ALL)

PBGOATS MEDIAN (PBGOAT ALL)

PBHORSES MEAN (PBHORSE ALL)

PBDONKYS MEDIAN (PBDONKY ALL)

PBCHICNS MEDIAN (PBCHICN ALL)

PBTURKYS MEDIAN (PBTURKY ALL)

PBDUCKS MEDIAN (PBDUCK ALL)

PBRABBTS MEAN (PBRABBT ALL)

PBMINKS MEDIAN (PBMINK ALL)

PBOTHFUS MEDIAN (PBOTHFU ALL)

PBBEESS MEDIAN (PBBEES ALL)

PBOTHANS MEDIAN (PBOTHAN ALL)

Correcting for Outliers in Farm Prices for Sales (PS) ***& Purchases*** (PB)

PSCOWS *if* (PSCOWS > 2760) *then* PSCOWS = 2760
if (PSCOWS <= 1700)
and (OBLAST = 1) *then* PSCOWS = 1950
and (OBLAST = 2) *then* PSCOWS = 2800
and (OBLAST = 3) *then* PSCOWS = 1700
and (OBLAST = 4) *then* PSCOWS = 1950
and (OBLAST = 6) *then* PSCOWS = 2100
and (OBLAST = 7) *then* PSCOWS = 2550
and (OBLAST = 8) *then* PSCOWS = 1850

PSPIGS *if* (PSPIGS > 1500) *then* PSPIGS = 1500
if (PSPIGS <= 1000)
and (OBLAST = 1) *then* PSPIGS = 1500
and (OBLAST = 2) *then* PSPIGS = 1500
and (OBLAST = 3) *then* PSPIGS = 1000
and (OBLAST = 4) *then* PSPIGS = 1500
and (OBLAST = 6) *then* PSPIGS = 1500
and (OBLAST = 7) *then* PSPIGS = 1500
and (OBLAST = 8) *then* PSPIGS = 1500

PSSHEEPS *if* (PSSHEEPS > 549) *then* PSSHEEPS = 549
if (PSSHEEPS <= 390)
and (OBLAST = 1) *then* PSSHEEPS = 459
and (OBLAST = 2) *then* PSSHEEPS = 416
and (OBLAST = 3) *then* PSSHEEPS = 481
and (OBLAST = 4) *then* PSSHEEPS = 464
and (OBLAST = 6) *then* PSSHEEPS = 549
and (OBLAST = 7) *then* PSSHEEPS = 237
and (OBLAST = 8) *then* PSSHEEPS = 391

PSGOATS *if* (PSGOATS > 305) *then* PSGOATS = 305
if (PSGOATS <= 196.25)
and (OBLAST = 1) *then* PSGOATS = 300
and (OBLAST = 2) *then* PSGOATS = 305
and (OBLAST = 3) *then* PSGOATS = 305
and (OBLAST = 4) *then* PSGOATS = 200
and (OBLAST = 6) *then* PSGOATS = 196.25
and (OBLAST = 7) *then* PSGOATS = 300
and (OBLAST = 8) *then* PSGOATS = 300

PSHORSES *if* (PSHORSES > 4755) *then* PSHORSES = 4755
if (PSHORSES <= 1876)
and (OBLAST = 1) *then* PSHORSES = 2988
and (OBLAST = 2) *then* PSHORSES = 3000
and (OBLAST = 3) *then* PSHORSES = 2988
and (OBLAST = 4) *then* PSHORSES = 2250
and (OBLAST = 6) *then* PSHORSES = 1876
and (OBLAST = 7) *then* PSHORSES = 2000
and (OBLAST = 8) *then* PSHORSES = 4755

PSDONKYS *if* (PSDONKYS > 600) *then* PSDONKYS = 600
if (PSDONKYS <= 418)
and (OBLAST = 1) *then* PSDONKYS = 500
and (OBLAST = 2) *then* PSDONKYS = 500
and (OBLAST = 3) *then* PSDONKYS = 418
and (OBLAST = 4) *then* PSDONKYS = 300
and (OBLAST = 6) *then* PSDONKYS = 600
and (OBLAST = 7) *then* PSDONKYS = 300
and (OBLAST = 8) *then* PSDONKYS = 500

PSCHICNS *if* (PSCHICNS >= 50) *then* PSCHICNS = 50
if (PSCHICNS <= 30)
and (OBLAST = 1) *then* PSCHICNS = 30
and (OBLAST = 2) *then* PSCHICNS = 38
and (OBLAST = 3) *then* PSCHICNS = 50

and (OBLAST = 4) *then* PSCHICNS = 30
and (OBLAST = 6) *then* PSCHICNS = 30
and (OBLAST = 7) *then* PSCHICNS = 30
and (OBLAST = 8) *then* PSCHICNS = 30

PSTURKYS *if* (PSTURKYS > 90) *then* PSTURKYS = 90
if (PSTURKYS <= 50)
and (OBLAST = 1) *then* PSTURKYS = 50
and (OBLAST = 2) *then* PSTURKYS = 50
and (OBLAST = 3) *then* PSTURKYS = 90
and (OBLAST = 4) *then* PSTURKYS = 50
and (OBLAST = 6) *then* PSTURKYS = 50
and (OBLAST = 7) *then* PSTURKYS = 75
and (OBLAST = 8) *then* PSTURKYS = 80

PSDUCKS *if* (PSDUCKS >= 35) *then* PSDUCKS = 35
if (PSDUCKS <= 20)
and (OBLAST = 1) *then* PSDUCKS = 35
and (OBLAST = 2) *then* PSDUCKS = 35
and (OBLAST = 3) *then* PSDUCKS = 20
and (OBLAST = 4) *then* PSDUCKS = 35
and (OBLAST = 6) *then* PSDUCKS = 35
and (OBLAST = 7) *then* PSDUCKS = 35
and (OBLAST = 8) *then* PSDUCKS = 35

PSRABBTS *if* (PSRABBTS > 45) *then* PSRABBTS = 45
if (PSRABBTS <= 20)
and (OBLAST = 1) *then* PSRABBTS = 35
and (OBLAST = 2) *then* PSRABBTS = 35
and (OBLAST = 3) *then* PSRABBTS = 20
and (OBLAST = 4) *then* PSRABBTS = 35
and (OBLAST = 6) *then* PSRABBTS = 35
and (OBLAST = 7) *then* PSRABBTS = 35
and (OBLAST = 8) *then* PSRABBTS = 45

PSMINKS *if* (PSMINKS > 50) *then* PSMINKS = 50
if (PSMINKS <= 50)
and (OBLAST = 1) *then* PSMINKS = 50
and (OBLAST = 2) *then* PSMINKS = 50
and (OBLAST = 3) *then* PSMINKS = 50
and (OBLAST = 4) *then* PSMINKS = 50
and (OBLAST = 6) *then* PSMINKS = 50
and (OBLAST = 7) *then* PSMINKS = 50
and (OBLAST = 8) *then* PSMINKS = 50

PSBEESS *if* (PSBEESS >= 250) *then* PSBEESS = 250
if (PSBEESS <= 150)
and (OBLAST = 1) *then* PSBEESS = 200
and (OBLAST = 2) *then* PSBEESS = 250
and (OBLAST = 3) *then* PSBEESS = 200
and (OBLAST = 4) *then* PSBEESS = 250
and (OBLAST = 6) *then* PSBEESS = 150
and (OBLAST = 7) *then* PSBEESS = 200
and (OBLAST = 8) *then* PSBEESS = 200

PSOTHANS PSOTHANS = 104.5

PBCOWS *if* (PBCOWS >= 1550) *then* PBCOWS = 1550
if (PBCOWS <= 650)
and (OBLAST = 1) *then* PBCOWS = 1500
and (OBLAST = 2) *then* PBCOWS = 300
and (OBLAST = 3) *then* PBCOWS = 1550
and (OBLAST = 4) *then* PBCOWS = 1550
and (OBLAST = 6) *then* PBCOWS = 1500
and (OBLAST = 7) *then* PBCOWS = 800
and (OBLAST = 8) *then* PBCOWS = 650

PBPIGS *if* (PBPIGS > 200) *then* PBPIGS = 200
if (PBPIGS <= 163.38)
and (OBLAST = 1) *then* PBPIGS = 163.38
and (OBLAST = 2) *then* PBPIGS = 163.38
and (OBLAST = 3) *then* PBPIGS = 200
and (OBLAST = 4) *then* PBPIGS = 163.38
and (OBLAST = 6) *then* PBPIGS = 163.38
and (OBLAST = 7) *then* PBPIGS = 163.38
and (OBLAST = 8) *then* PBPIGS = 164.17

PBSHEEPS *if* (PBSHEEPS > 525.56) *then* PBSHEEPS = 525.56
if (PBSHEEPS <= 266.67)
and (OBLAST = 1) *then* PBSHEEPS = 431.48
and (OBLAST = 2) *then* PBSHEEPS = 431.48
and (OBLAST = 3) *then* PBSHEEPS = 500
and (OBLAST = 4) *then* PBSHEEPS = 431.48
and (OBLAST = 6) *then* PBSHEEPS = 525.56
and (OBLAST = 7) *then* PBSHEEPS = 431.48
and (OBLAST = 8) *then* PBSHEEPS = 266.67

PBGOATS *if* (PBGOATS > 275) *then* PBGOATS = 275
if (PBGOATS <= 156.75)
and (OBLAST = 1) *then* PBGOATS = 275

and (OBLAST = 2) *then* PBGOATS = 275
and (OBLAST = 3) *then* PBGOATS = 275
and (OBLAST = 4) *then* PBGOATS = 275
and (OBLAST = 6) *then* PBGOATS = 275
and (OBLAST = 7) *then* PBGOATS = 275
and (OBLAST = 8) *then* PBGOATS = 156.75

PBHORSES *if* (PBHORSES > 3750) *then* PBHORSES = 3750
if (PBHORSES <= 1626)
and (OBLAST = 1) *then* PBHORSES = 2472
and (OBLAST = 2) *then* PBHORSES = 2472
and (OBLAST = 3) *then* PBHORSES = 2150
and (OBLAST = 4) *then* PBHORSES = 2471
and (OBLAST = 6) *then* PBHORSES = 1626
and (OBLAST = 7) *then* PBHORSES = 2500
and (OBLAST = 8) *then* PBHORSES = 3750

PBDONKYS *if* (PBDONKYS >= 786.57) *then* PBDONKYS = 786
if (PBDONKYS <= 18)
and (OBLAST = 1) *then* PBDONKYS = 18
and (OBLAST = 2) *then* PBDONKYS = 18
and (OBLAST = 3) *then* PBDONKYS = 18
and (OBLAST = 4) *then* PBDONKYS = 18
and (OBLAST = 6) *then* PBDONKYS = 18
and (OBLAST = 7) *then* PBDONKYS = 18
and (OBLAST = 8) *then* PBDONKYS = 786

PBCHICNS *if* (PBCHICNS >= 21.67) *then* PBCHICNS = 22
if (PBCHICNS < 5)
and (OBLAST = 1) *then* PBCHICNS = 20
and (OBLAST = 2) *then* PBCHICNS = 5
and (OBLAST = 3) *then* PBCHICNS = 21.67
and (OBLAST = 4) *then* PBCHICNS = 20
and (OBLAST = 6) *then* PBCHICNS = 20
and (OBLAST = 7) *then* PBCHICNS = 20
and (OBLAST = 8) *then* PBCHICNS = 5

PBTURKYS *if* (PBTURKYS > 60) *then* PBTURKYS = 60
if (PBTURKYS <= 16.84)
and (OBLAST = 1) *then* PBTURKYS = 60
and (OBLAST = 2) *then* PBTURKYS = 60
and (OBLAST = 3) *then* PBTURKYS = 60
and (OBLAST = 4) *then* PBTURKYS = 60
and (OBLAST = 6) *then* PBTURKYS = 60
and (OBLAST = 7) *then* PBTURKYS = 60

and (OBLAST = 8) *then* PBTURKYS = 16.84

PBDUCKS *if* (PBDUCKS >= 60) *then* PBDUCKS = 60
if (PBDUCKS <= 10.9)
and (OBLAST = 1) *then* PBDUCKS = 27.5
and (OBLAST = 2) *then* PBDUCKS = 27.5
and (OBLAST = 3) *then* PBDUCKS = 60
and (OBLAST = 4) *then* PBDUCKS = 27.5
and (OBLAST = 6) *then* PBDUCKS = 27.5
and (OBLAST = 7) *then* PBDUCKS = 27.5
and (OBLAST = 8) *then* PBDUCKS = 10.9

PBRABBTS *if* (PBRABBTS > 26.27) *then* PBRABBTS = 26.27
if (PBRABBTS <= 22)
and (OBLAST = 1) *then* PBRABBTS = 26.27
and (OBLAST = 2) *then* PBRABBTS = 35
and (OBLAST = 3) *then* PBRABBTS = 26.27
and (OBLAST = 4) *then* PBRABBTS = 26.27
and (OBLAST = 6) *then* PBRABBTS = 26.27
and (OBLAST = 7) *then* PBRABBTS = 26.27
and (OBLAST = 8) *then* PBRABBTS = 22.34

PBMINKS *if* (PBMINKS >= 20) *then* PBMINKS = 20
if (PBMINKS <= 20)
and (OBLAST = 1) *then* PBMINKS = 20
and (OBLAST = 2) *then* PBMINKS = 20
and (OBLAST = 3) *then* PBMINKS = 20
and (OBLAST = 4) *then* PBMINKS = 20
and (OBLAST = 6) *then* PBMINKS = 20
and (OBLAST = 7) *then* PBMINKS = 20
and (OBLAST = 8) *then* PBMINKS = 20

PBBEESS *if* (PBBEESS >= 250) *then* PBBEESS = 250
if (PBBEESS <= 150)
and (OBLAST = 1) *then* PBBEESS = 200
and (OBLAST = 2) *then* PBBEESS = 250
and (OBLAST = 3) *then* PBBEESS = 200
and (OBLAST = 4) *then* PBBEESS = 250
and (OBLAST = 6) *then* PBBEESS = 150
and (OBLAST = 7) *then* PBBEESS = 200
and (OBLAST = 8) *then* PBBEESS = 200

PBOTHANS PBOTHANS = 104.5

Livestock Revenues

CATTLE	{(E0906_01 * PSCOWS) - (E0908_01 * PBCOWS)} / 12
PIGS	{(E0906_02 * PSPIGS) - (E0908_02 * PBPIGS)} / 12
SHEEPS	{(E0906_03 * PS SHEEPS) - (E0908_03 * PBSHEEPS)} / 12
GOATS	{(E0906_04 * PSGOATS) - (E0908_04 * PBGOATS)} / 12
HORSES	{(E0906_05 * PSHORSES) - (E0908_05 * PBHORSES)} / 12
DONKEYS	{(E0906_06 * PSDONKYS) - (E0908_06 * PBDONKYS)} / 12
CHICKENS	{(E0906_07 * PSCHICNS) - (E0908_07 * PBCHICNS)} / 12
TURKEYS	{(E0906_08 * PSTURKYS) - (E0908_08 * PBTURKYS)} / 12
DUCKS	{(E0906_09 * PSDUCKS) - (E0908_09 * PBDUCKS)} / 12
RABBITS	{(E0906_10 * PSRABBTS) - (E0908_10 * PBRABBTS)} / 12
MINK	{(E0906_10 * PSMINKS) - (E0908_10 * PBMINKS)} / 12
OTHERFUR	{(E0906_11 * PSOTHFUS) - (E0908_11 * PBOTHFUS)} / 12
BEES	{(E0906_12 * PSBEES) - (E0908_12 * PBBEES)} / 12
OTHERAN	{(E0906_13 * PSOTHANS) - (E0908_13 * PBOTHANS)} / 12
NETREVAN	total net revenue SUM (CATTLE, PIGS, SHEEPS, GOATS, HORSES, DONKEYS, CHICKENS, TURKEYS, DUCKS, RABBITS, MINK, OTHERFUR, BEES, OTHERAN) * DEFLATOR
EXPANTAX	monthly livestock taxes C0921_02 / 12
EXPREARG	other monthly expenditures incurred in animal rearing SUM (H0902_01 TO H0902_11) / 12
EXPTOANI	total monthly expenditure on animals SUM (EXPREARG, EXPANTAX) * DEFLATOR
AGIF1_01-14	average value of animals that were either lost, stolen, given as a gift or died, per animal type, per year E0914_01 TO E0914_14 * (EO905_01 TO E0905_14 / EO904_01 TO E0904_14)

AGIFT	total value of animals that were either lost, stolen, given as a gift or died per year SUM (AGIF_01 TO AGIF_14) * DEFLATOR
LANRENAN	monthly expenditure on land rental including, pasture, hayfields and barnyards {SUM (A0907_03, A0907_04, A0907_08)} * DEFLATOR / 12
LDIMREAN	monthly imputed expenditure on land rental including, pasture, hayfields and barnyards {SUM (A0903_03, A0903_04, A0903_08)} * DEFLATOR / 12
LIVESTCK	net revenue from livestock trade SUM (NETREVAN, (-EXPTOANI), (-LANRENAN))
REVTRCRP	monthly total received from sale of food products {SUM (D0911_01 TO D0911_16)} * DEFLATOR / 12
REVTRANI	monthly total received from sale of animal products {SUM (F0911_01 TO D0911_26)} * DEFLATOR / 12
EXPTRCRP	monthly total expenditure on the production of food products {SUM (D0912_01 TO D0912_16)} * DEFLATOR / 12
EXPTRANI	monthly total expenditure on the production of animal products {SUM (F0912_01 TO F0912_26)} * DEFLATOR / 12
HARVSALE	monthly net revenue from the production of food products SUM (REVTRCRP, (-EXPTRCRP))
ANPRDSL	monthly net revenue from the production of animal products SUM (REVTRANI), (-EXPTRANI))
I2PRO_01-16	household consumption of home produced items from crop products per crop

{(D0907_01 TO D0907_16) * (D0906_01 TO D0906_16) * (P9D_01 TO P9D_16)}/12

HARVHC — total household consumption of home produced items from crop products

SUM (I2PRO_01 TO I2PRO_16)

I3PRO_01-26 — household consumption of home produced items from animal products per animal

{(F0907_01 TO F0907_26) * (F0906_01 TO F0906_26) * (P9F_01 TO P9F_26)}/12

ANPRODHC — total household consumption of home produced items from animal products

SUM (I3PRO_01 TO I3PRO_26)

LANDRENT — total expenditure in land rental

SUM (LANRENCR, LANRENAN)

HGIF1_01 -16 — value of crop products given as gifts, per crop

{(D0908_01 TO D0908_16) * (P9D_01 TO P9D_16)} / 12

HARVGIFT — total value of crop products given as gifts

SUM (HGIF1_01 TO HGIF1_16)

AGIF1_01 -26 — value of animal products given as gifts, per animal

{(DF908_01 TO F0908_26) * (P9F_01 TO P9F_26)} / 12

ANPRDGT — total value of animal products given as gifts

SUM (AGIF1_01 TO AGIF1_26)

INCFARM — total farm income

SUM (CROPSALE, LIVESTCK, HARVSALE, ANPRDSL, HARVHC, ANPRODHC, HARVGFT, ANPRDGT)

LANDHC — total value of home consumption of crop and animal products

SUM (HARVHC, ANDPROHC)

LANDSALE — total value of sales of crop and animal sales

SUM (HARVSALE, ANPRDSL, CROPSALE, LIVESTCK)

Section 6: Education

Variable	*Short Description and Code*
SEDEXP	household education expenditures in the last 12 months SUM (S3BQ26_1 TO S3BQ2610)
EDEXP	the higher value between total household education expenditure as calculated for SEDEXP and the total reported education expenditure in the questionnaire variable S3BQ2611 *if* SEDEXP < S3BQ2611 *then* EDEXP = S3BQ2611 *otherwise* EDEXP = SEDEXP
SPARCOM	expenditure by parents through school committees on other school expenses (such as repairs, school trips etc.) SUM (S3BQ27_1 TO S3BQ27_6)
PARCOM	the higher value between expenditure by parents through school committees as calculated for SPARCOM and the total reported through household school committee expenditures in the questionnaire variable S3BQ27_7 *if* SPARCOM < S3BQ27_7 *then* PARCOM = S3BQ27_7 *otherwise* PARCOM = SPARCOM
EDUCAT	total household expenditures on education SUM (EDEXP, PARCOM) * DEFLATOR/12
R_PR_A	those respondents who attended primary school *if* S3BQ11 = 0 *then* R_PR_A = 0 *otherwise* R_PR_A = 1
R_NOED	those respondents who did not attend primary school *if* S3BQ11 = 0 *then* R_NOED = 1 *otherwise* R_NOED = 0

R_SC_A those respondents who attended secondary school

if R_PR_A = 1 *and* S3BQ13 = 0
then R_SC_A = 0
otherwise R_SC_A = 1

R_PR_O those respondents who attended primary school only

if R_PR_A = 1 *and* R_SC_A = 0
then R_PR_O = 1
if R_PR_A = 1 *and* R_SC_A = 1
then R_PR_O = 0

R_SC_I those respondents who started but did not complete secondary school

if R_SC_A = 1 *and* S3BQ17 = 1
then R_SC_I = 1
if R_SC_A = 1 *and* S3BQ17 <> 1
then R_SC_I = 0

R_SC_O those respondents who completed primary and secondary school only

if R_SC_A = 1 *and* S3BQ17 = 2
then R_SC_O = 1
if R_SC_A = 1 *and* S3BQ17 <> 2
then R_SC_0 = 0

R_SCAH those respondents who completed primary and secondary school and higher education

if R_SC_A = 1 *and* (S3BQ17 = 3 *or* S3BQ17 = 4 *or* S3BQ17 = 5 *or* S3BQ17 = 6 *or* S3BQ17 = 7)
then R_SCAH = 1

R_SCAO those respondents who completed primary and secondary school and some "other" (not listed) form of higher education

if R_SC_A = 1 *and* S3BQ17 = 8
then R_SCAO = 1

F_NOED those respondents whose father had no education

if S1BQ3 = 17
then F_NOED = 1

F_PR_O — those respondents whose father completed primary education only

if S1BQ3 >= 1 *and* S1BQ3 <= 8
then F_PR_O = 1

F_SC_I — those respondents whose father completed primary education and started but did not complete secondary school

if S1BQ3 = 9
then F_SC_I = 1

F_SC_O — those respondents whose father completed primary education and secondary school only

if S1BQ3 = 10
then F_SC_O = 1

F_SCAH — those respondents whose father completed primary and secondary school and higher education

if S1BQ3 >= 11 *and* S1BQ3 <= 15
then F_ SCAH = 1

F_SCAO — those respondents whose father completed primary and secondary school and some "other" (not listed) form of higher education

if S1BQ3 = 16
then F_ SCAO = 1

M_NOED — those respondents whose mother had no education

if S1BQ8 = 17
then M_NOED = 1

M_PR_O — those respondents whose mother completed primary education only

if S1BQ8 >= 1 *and* S1BQ8 <= 8
then M_PR_O = 1

M_SC_I — those respondents whose mother completed primary education and started but did not complete secondary school

if S1BQ8 = 9
then M_SC_I = 1

M_SC_O — those respondents whose mother completed primary education and secondary school only

if S1BQ8 = 10
then M_SC_O = 1

M_SCAH — those respondents whose mother completed primary and secondary school and higher education

if S1BQ8 >= 11 *and* S1BQ8 <= 15
then M_ SCAH = 1

M_SCAO — those respondents whose mother completed primary and secondary school and some "other" (not listed) form of higher education

if S1BQ8 = 16
then M_ SCAO = 1

R_SC_EN — those respondents who are currently attending regular or special secondary school

if S3BQ_21 = 10 *or* S3BQ_21 = 12
then R_SC_EN = 1

R_PR_EN — those respondents who are currently attending primary school

if S3BQ_21 <= 9
then R_PR_EN = 1

R_TC_EN — those respondents who are currently attending professional-technical school

if S3BQ_21 = 11
then R_TC_EN = 1

R_3Y_EN — those respondents who are currently attending university or undertaking post-graduate studies

if S3BQ_21 = 13 *or* S3BQ_21 = 15
then R_3Y_EN = 1

R_TK_EN — those respondents who are currently attending technical institute

if S3BQ_21 = 14
then R_TK_EN = 1

PRE_NAN those of children of pre-school age who are currently looked after by a nanny

if S3AQ2 = 4
then PRE_NAN = 1

PRE_PVT those of children of pre-school age who are currently attending private pre-school

if S3AQ2 = 3
then PRE_PVT = 1

PRE_PUB those of children of pre-school age who currently attend pre-school at the parent's workplace or at a publicly provided institution

if S3AQ2 = 1 *or* S3AQ2 = 2
then PRE_PUB = 1

NPRE_CLS those children who stopped attending pre-school because it closed down

if S3AQ8 = 1
then NPRE_CLS = 1

NPRE_COS those children who stopped attending pre-school because of financial reasons

if S3AQ8 = 2
then NPRE_COS = 1

NPRE_NWK those children who stopped attending pre-school because parents no longer work where it was provided

if S3AQ8 = 3
then NPRE_ NWK = 1

NPRE_QLY those children who stopped attending pre-school because parents dissatisfied with the quality of care

if S3AQ8 = 4
then NPRE_ QLY = 1

NPRE_PRF those children who stopped attending pre-school because parents prefer to have them elsewhere

if S3AQ8 = 5
then NPRE_ PRF = 1

NPRE_OTH those children who stopped attending pre-school for some other reason

if S3AQ8 = 6
then NPRE_ OTH = 1

CHC_MO those children whose mother cares for them during the day

if S3AQ9 = 1
then CHC_MO = 1

CHC_FA those children whose father cares for them during the day

if S3AQ9 = 2
then CHC_FA = 1

CHC_BP those children who have both parents care for them during the day

if S3AQ9 = 3
then CHC_BP = 1

CHC_OF those children who have other household members care for them during the day

if S3AQ9 = 4
then CHC_OF = 1

CHC_OR those children who have other relatives care for them during the day

if S3AQ9 = 5
then CHC_OR = 1

CHC_SV those children who are cared for by household help during the day

if S3AQ9 = 6
then CHC_SV = 1

CHC_FR those children who are cared for by friends during the day

if S3AQ9 = 7
then CHC_FR = 1

CHC_OTH those children who are cared for by other means during the day

if S3AQ9 = 8
then CHC_OTH = 1

PRE_EXP expenditure on child care in the last month per child

PRE_EXP = S3AQ6

CURR_EN those respondents who are currently enrolled in some form of schooling

if S3BQ_20 = 1
then CURR_EN = 1
if S3BQ_20 = 2
then CURR_EN = 0

INST_PUB those respondents who were last enrolled in a state owned institution

if S3BQ_18 = 1
then INST_PUB = 1

INST_PRV those respondents who were last enrolled in a privately owned institution

if S3BQ_18 = 2
then INST_PRV = 1

INST_REL those respondents who were last enrolled in a religious institution

if S3BQ_18 = 3
then INST_REL = 1

TRS_WLK those who walk to school

if S3BQ_33 = 1
then TRS_WLK = 1

TRS_HRS those who ride a horse to school

if S3BQ_33 = 2
then TRS_ HRS = 1

TRS_PUB those who ride public transport to school

if S3BQ_33 = 3
then TRS_ PUB = 1

TRS_MTB those who ride a motorbike to school

if S3BQ_33 = 4
then TRS_ MTB = 1

TRS_BIC those who ride a bicycle to school

if S3BQ_33 = 5
then TRS_ BIC = 1

TRS_TAX those who ride a taxi to school

if S3BQ_33 = 6
then TRS_ TAX = 1

TRS_CAR those who travel in a private car to school

if S3BQ_33 = 7
then TRS_ CAR = 1

TRS_OTH those who travel to school with other means

if S3BQ_33 = 8
then TRS_ OTH = 1

Creating Travel Time to School Variables

HR_MISS MISSING (S3BQ34_1)

MIN_MISS MISSING (S3BQ34_2)

TIME_SCH *if* HR_MISS = 1 *and* MIN_MISS = 0
then TIME_SCH = S3BQ34_2
if HR_MISS = 0 *and* MIN_MISS = 1
then TIME_SCH = S3BQ34_1 * 60
if HR_MISS = 0 *and* MIN_MISS = 0
then TIME_SCH = SUM ((S3BQ34_1 * 60), S3BQ34_2)

TM0005 travel time to school is 5 minutes or less

if TIME_SCH >= 0 *and* TIME_SCH < 5
then TM0005 = 1

TM0530 travel time to school is between 5 and 30 minutes

if TIME_SCH >= 5 *and* TIME_SCH < 30
then TM0530 = 1

TM30 travel time to school is 30 minutes

if TIME_SCH = 30

then TM30 = 1

TM3060 travel time to school is between 31 and 60 minutes

if TIME_SCH > 30 *and* TIME_SCH < 60
then TM3060 = 1

TM60PL travel time to school is greater than 60 minutes

if TIME_SCH > 60
then TM60PL = 1

NSC_COST those who stopped attending school because of the cost

if S3BQ_23 = 2
then NSC_COST = 1

NSC_EARN those who stopped attending school to earn money to live

if S3BQ_23 = 4
then NSC_ EARN = 1

NSC_FAR those who stopped attending school because it was too far away

if S3BQ_23 = 5
then NSC_FAR = 1

NSC_TCH those who stopped attending school because there was no teacher

if S3BQ_23 = 6
then NSC_ TCH = 1

NSC_SUPP those who stopped attending school because there were no supplies

if S3BQ_23 = 7
then NSC_ SUPP = 1

NSC_SHOE those who stopped attending school because they had no clothing and/or shoes

if S3BQ_23 = 8
then NSC_ SHOE = 1

NSC_ILL those who stopped attending school because of illness

if S3BQ_23 = 9

then NSC_ ILL = 1

NSC_STUD those who stopped attending school because of their dislike for studying

if S3BQ_23 = 10
then NSC_ STUD = 1

NSC_STRT those who never began schooling

if S3BQ_23 = 12
then NSC_ STRT = 1

NSC_OTH those who stopped attending school for other reasons

if S3BQ_23 = 13
then NSC_ OTH = 1

ED_LVL respondents' highest level of education attainment

if S3BQ_17 = 5 *or* S3BQ_17 = 6 *or* S3BQ_17 = 7
then ED_LVL = 4
if S3BQ_17 = 3 *or* S3BQ_17 = 4
then ED_LVL = 3
if S3BQ_17 = 2
then ED_LVL = 2
if S3BQ12 = 4 *and* S3BQ_17 = 9
then ED_LVL = 1
otherwise ED_LVL = 0

EN_PRIM those respondents who are currently enrolled in primary school

if S3BQ_21 >= 1 *and* S3BQ_21 <= 9
then EN_PRIM = 1
otherwise EN_PRIM = 0

EN_SEC those respondents who are currently enrolled in secondary school

if S3BQ_21 = 10 *or* S3BQ_21 = 12
then EN_ SEC = 1
otherwise EN_SEC = 0

EN_PRSEC those respondents who are currently enrolled in either primary or secondary school

if EN_PRIM = 1 *or* EN_ SEC = 1
then EN_PRSEC = 1

otherwise EN_PRSEC = 0

Section 7: Health

Variable	*Short Description, Calculation and Code Used*
HP_PRVD	those who used a private doctor in the last month
	if S4BQ6 = 2 *then* HP_PRVD = 1
HP_PUBD	those who used a public doctor in the last month
	if S4BQ6 = 3 *then* HP_PUBD = 1
HP_NUR	those who used a nurse in the last month
	if S4BQ6 = 4 *then* HP_NUR = 1
HP_FELD	those who used a feldsher in the last month
	if S4BQ6 = 5 *then* HP_FELD = 1
HP_PHAR	those who used a pharmacist in the last month
	if S4BQ6 = 6 *then* HP_PHAR = 1
HP_DENT	those who used a dentist in the last month
	if S4BQ6 = 7 *then* HP_DENT = 1
HP_HEAL	those who used a healer in the last month
	if S4BQ6 = 8 *then* HP_HEAL = 1
HP_PSY	those who used a psychic in the last month
	if S4BQ6 = 9 *then* HP_PSY = 1
HP_PARA	those who used a prarmedic in the last month
	if S4BQ6 = 10 *then* HP_PARA = 1
HP_OTH	those who used another health-care provider in the last month
	if S4BQ6 = 11 *then* HP_OTH = 1

HO_NIGHT length of stay in hospital/facility due of sickness/illness

if S4BQ12 = 1 *then* HO_NIGHT = S4BQ13

NU_NEED those who did not seek medical treatment because it was not needed

if S4BQ6 = 1 *and* S4BQ_31 = 1
then NU_NEED = 1

NU_FAR those who did not seek medical treatment because it was too far away

if S4BQ6 = 1 *and* S4BQ_31 = 2
then NU_FAR = 1

NU_WAIT those who did not seek medical treatment because it entailed a long wait

if S4BQ6 = 1 *and* S4BQ_31 = 3
then NU_WAIT = 1

NU_SERV those who did not seek medical treatment because of poor service

if S4BQ6 = 1 *and* S4BQ_31 = 4
then NU_SERV = 1

NU_EXP those who did not seek medical treatment because of the high cost

if S4BQ6 = 1 *and* S4BQ_31 = 5
then NU_EXP = 1

NU_PERM those who did not seek medical treatment because of no residency permit

if S4BQ6 = 1 *and* S4BQ_31 = 6
then NU_PERM = 1

NU_SELF those who did not seek medical treatment because they self-medicated

if S4BQ6 = 1 *and* S4BQ_31 = 7
then NU_SELF = 1

NU_TIME those who did not seek medical treatment because of lack of time

if S4BQ6 = 1 *and* S4BQ_31 = 8
then NU_TIME = 1

NU_OTH — those who did not seek medical treatment because of other reasons

if S4BQ6 = 1 *and* S4BQ_31 = 9
then NU_OTH = 1

ILL_HART — those respondents who had heart problems lasting longer than 6 months

if S4AQ1 = 1 *and* S4AQ4 = 1
then ILL_HART = 1

ILL_LUNG — those respondents who had lung problems lasting longer than 6 months

if S4AQ1 = 1 *and* S4AQ4 = 2
then ILL_LUNG = 1

ILL_STOM — those respondents who had stomach problems lasting longer than 6 months

if S4AQ1 = 1 *and* S4AQ4 = 3
then ILL_STOM = 1

ILL_ORG — those respondents who had problems with other organs

if S4AQ1 = 1 *and* S4AQ4 = 7
then ILL_ORG = 1

MEDSERV — monthly household expenditure on medicines and medical services

SUM (B1202_11, B1202_12) * DEFLATOR/12

SDISAB — expenditure on the treatment of chronic illness or disability in the last month

SUM (S4AQ5 TO S4AQ7)

DISAB — the higher value between expenditure on the treatment of chronic illness or disability as calculated for SDISAB and the total reported through chronic illness or disability treatment expenditures in the questionnaire variable S4AQ10

if SDISAB < S4AQ10
then DISAB = S4AQ10
otherwise DISAB = SDISAB

SILLINJ — expenditure on the treatment (second opinions and other facilities costs only) of recent illnesses or injury in the last month

SUM (S4BQ_24, S4BQ_27)

ILLINJ — the higher value between expenditure on the treatment (second opinions and other facilities costs only) of recent illnesses or injury as calculated for SILLINJ and the total value of these expenditures reported in the questionnaire variable S4BQ_30 added to the monthly household main expenditures on recent illness and injury

if SILLINJ < S4BQ_30
then ILLINJ = SUM (S4BQ14, S4BQ_30) * DEFLATOR
otherwise ILLINJ = SUM (S4BQ14, SILLINJ) * DEFLATOR

SNODOCT — expenditure on recent illnesses or injury in the last month when no doctor was involved (costs of tests, medicines etc.)

SUM (S4BQ_35, S4BQ_36)

NODOCT — the higher value between expenditure on the treatment recent illnesses or injury as calculated for SNODOCT and the total reported recent illnesses or injury expenditures in the questionnaire variable S4BQ_38

if SNODOCT < S4BQ_38
then NODOCT = S4BQ_38 * DEFLATOR
otherwise NODOCT = SNODOCT * DEFLATOR

SDENTEYE — expenditure on dentist and eye care in the last month

SUM (S4CQ3, S4CQ5)

DENTEYE — the higher value between expenditures on dentist and eye care as calculated for SDENTEYE and the total reported dentist and eye care expenditures in the questionnaire variable S4CQ8

if SDENTEYE < S4CQ8
then DENTEYE = S4CQ8 * DEFLATOR/12
otherwise DENTEYE = SDENTEYE * DEFLATOR/12

HEALTH — total health expenditures

SUM (MEDSERC, DISAB, ILLINJ, NODOCT, DENTEYE)

DRINKYES — those respondents who do drink alcohol

if S4DQ7 = 1
then DRINKYES = 1
if S4DQ7 = 2
then DRINKYES = 0

DRKFREQ the frequency of drinking alcohol

if DRINKYES = 1 *and* S4DQ8 = 6
then DRKFREQ = 1
if DRINKYES = 1 *and* S4DQ8 = 5
then DRKFREQ = 2
if DRINKYES = 1 *and* S4DQ8 = 4
then DRKFREQ = 3
if DRINKYES = 1 *and* S4DQ8 = 3
then DRKFREQ = 4
if DRINKYES = 1 *and* S4DQ8 = 2
then DRKFREQ = 5
if DRINKYES = 1 *and* S4DQ8 = 1
then DRKFREQ = 6

HF_STAY1 length of stay in health facility for any illness or injury (first care-provider/consultation)

if S4BQ12 = 1
then HF_STAY1 = S4BQ13

HF_STAY2 length of stay in health facility for any illness or injury (second care-provider/consultation)

if S4BQ_22 = 1
then HF_STAY2 = S4BQ_23

HF_STAY3 length of stay in health facility for any illness or injury (third care-provider/consultation)

if S4BQ_28 = 1
then HF_STAY3 = S4BQ_29

Calculating Length of Stay Variables

HF1_MISS MISSING (HF_STAY1)

HF2_MISS MISSING (HF_STAY2)

HF3_MISS MISSING (HF_STAY3)

HF_STAY *if* HF1_MISS = 0 *and* HF2_MISS = 1 *and* HF3_MISS = 1
then HF_STAY = HF_STAY1
if HF1_MISS = 1 *and* HF2_MISS = 0 *and* HF3_MISS = 1
then HF_STAY = HF_STAY2
if HF1_MISS = 1 *and* HF2_MISS = 1 *and* HF3_MISS = 0

then HF_STAY = HF_STAY3
if HF1_MISS = 0 *and* HF2_MISS = 0 *and* HF3_MISS = 1
then HF_STAY = SUM (HF_STAY1, HF_STAY2)
if HF1_MISS = 0 *and* HF2_MISS = 1 *and* HF3_MISS = 0
then HF_STAY = SUM (HF_STAY1, HF_STAY3)
if HF1_MISS = 1 *and* HF2_MISS = 0 *and* HF3_MISS = 0
then HF_STAY = SUM (HF_STAY2, HF_STAY3)

Section 8: Women's Issues

Variable *Short Description and Code*

WOMAN defines those respondents who are women

if S1AQ4 = 2 *then* WOMAN =1
if S1AQ4 = 1 *then* WOMAN = 0

SEXAGE defines women between (not including) the ages of 14 and 56

if WOMAN = 1 *and* S1AQ3 >= 15 *and* S1AQ3 <= 55
then SEXAGE = 1
otherwise SEXAGE = 0

PASTACT defines a woman as sexually active (at least at some stage)

if SEXAGE = 1 *and* S8AQ25 = 1
then PASTACT = 1
if SEXAGE = 1 *and* S8AQ25 = 1
then PASTACT = 0

CURRCONT defines a woman as currently using contraception

if PASTACT = 1 *and* S8AQ27 = 1
then CURRCONT = 1
if PASTACT = 1 *and* S8AQ27 = 0
then CURRCONT = 0

PREG defines a woman as having been pregnant at least once in her life

if PASTACT = 1 *and* S8AQ1 = 1
then PREG = 1
if PASTACT = 1 *and* S8AQ1 = 1
then PREG = 0

ABORT defines a woman as having had an abortion at least once in her life

if PASTACT = 1 *and* S8AQ21 = 1 *and* S8AQ22 > 0
then ABORT = 1
if PASTACT = 1
then ABORT = 0

NUMABORT shows the number of abortions had by women who have been pregnant

if PREG = 1 *and* ABORT = 1
then NUMABORT = S8AQ22
if PREG = 1 *and* ABORT = 0
then NUMABORT = 0

CM_ABS women who use abstinence as a method of birth control

if S8AQ29_1 > 0 *or* S8AQ2912 > 0 *or* S8AQ2923 > 0
then CM_ABS = 1

CM_CYC women who observe their cycle as a method of birth control

if S8AQ29_2 > 0 *or* S8AQ2913 > 0 *or* S8AQ2924 > 0
then CM_CYC = 1

CM_INT women who "interrupt the act" as a method of birth control

if S8AQ29_3 > 0 *or* S8AQ2914 > 0 *or* S8AQ2925 > 0
then CM_INT = 1

CM_BATH women who use a hot bath as a method of birth control

if S8AQ29_4 > 0 *or* S8AQ2915 > 0 *or* S8AQ2926 > 0
then CM_BATH = 1

CM_HERB women who use herbs as a method of birth control

if S8AQ29_5 > 0 *or* S8AQ2916 > 0 *or* S8AQ2927 > 0
then CM_ HERB = 1

CM_DCH women who use a douche as a method of birth control

if S8AQ29_6 > 0 *or* S8AQ2917 > 0 *or* S8AQ2928 > 0
then CM_ DCH = 1

CM_COND women who use condoms as a method of birth control

if S8AQ29_7 > 0 *or* S8AQ2918 > 0 *or* S8AQ2929 > 0
then CM_ COND = 1

CM_CAP women who use a cap as a method of birth control

if S8AQ29_8 > 0 *or* S8AQ2919 > 0 *or* S8AQ2930 > 0
then CM_ CAP = 1

CM_PILL women who use the pill as a method of birth control

if S8AQ29_9 > 0 *or* S8AQ2920 > 0 *or* S8AQ31 > 0
then CM_ PILL = 1

CM_IUD women who use an IUD as a method of birth control

if S8AQ2910 > 0 *or* S8AQ2921 > 0 *or* S8AQ32 > 0
then CM_ IUD = 1

CM_INJ women who use injections as a method of birth control

if S8AQ2911 > 0 *or* S8AQ2922 > 0 *or* S8AQ33 > 0
then CM_ INJ = 1

Section 9: Housing

Variable	*Short Description and Code*
OWNHOME	those households that own their home *if* S2AQ5 = 1 *then* OWNHOME = 1
DORM	those households that live in a dormitory *if* S2AQ5 = 7 *or* S2AQ5 = 8 *then* DORM = 1
RENT	those households that rent their dwelling *if* S2BQ13 = 1 *then* RENT = 1
URBAN	those households that live in urban areas *if* RESIDENC = 1 *then* URBAN = 1
RURAL	those households that live in rural areas *if* RESIDENC = 2 *then* RURAL = 1
WATER	those households that have access to a central water pipeline *if* S2BQ23 = 1 *then* WATER = 1
HEAT	those households that have heating *if* S2BQ36 = 1 *then* HEAT = 1
ELECTRIC	those households that have electricity *if* S2BQ33 = 1 *then* ELECTRIC = 1

PHONE — those households that have access to a telephone

if S2BQ39 = 1
then PHONE = 1

BATH — those households that have access to a telephone

if S2BQ29 = 1
then BATH = 1

PUBUT — public utilities expenditure, average per household member

{SUM (S2BQ45, S2BQ46, S2BQ51 TO S2BQ53)} / HHCOMP

HOUSEEXP — house-building/renovations expenditures, average per household member

B1202_13 / HHCOMP

Corrections for Numbers of Floors and Rooms

S2AQ4 — *if* S2AQ4 >= 5
then S2AQ4 = 5

S2AQ6 — *if* S2AQ6 >= 5
then S2AQ6 = 5

S2AQ7 — *if* S2AQ7 >= 5
then S2AQ7 = 5

LIVA_CAP — per capita living area per household

S2AQ9 / HHCOMP = LIVA_CAP

LIVLT5 — *if* LIVA_CAP <= 5
then LIVLT5 = 1

LIVBWT510 — *if* LIVA_CAP > 5 *and* LIVA_CAP <= 10
then LIVBWT510 = 1

LIVB1020 — *if* LIVA_CAP > 10 *and* LIVA_CAP <= 20
then LIVB1020 = 1

LIVGT20 — *if* LIVA_CAP > 20
then LIVGT20 = 1

BUILLT50 — *if* S2AQ10 >= 50

then BUILLT50 = 1

BUIL5060 *if* S2AQ10 > 50 *and* S2AQ10 <= 60
then BUIL5060 = 1

BUIL6070 *if* S2AQ10 > 60 *and* S2AQ10 <= 70
then BUIL6070 = 1

BUIL7080 *if* S2AQ10 > 70 *and* S2AQ10 <= 80
then BUIL7080 = 1

BUIL8090 *if* S2AQ10 > 70 *and* S2AQ10 <= 90
then BUIL8090 = 1

BUILGT90 *if* S2AQ10 > 90
then BUILGT90 = 1

COUPONS *if* S2BQ4_2 = 1
then COUPONS = 1

SOMS *if* S2BQ4_2 = 2
then SOMS = 1

DOLLARS *if* S2BQ4_2 = 3
then DOLLARS = 1

RENTSOM conversion of rent from dollar values to soms

if S2BQ15_2 = 2
then RENTSOM = S2BQ15_1 * EXCHRT
otherwise RENTSOM = S2BQ15_1

RENTMTH conversion of rent to a monthly rate

if S2BQ15_3 = 3
then RENTMTH = RENTSOM * 30
if S2BQ15_3 = 4
then RENTMTH = RENTSOM * 4.3
if S2BQ15_3 = 6
then RENTMTH = RENTSOM/3
if S2BQ15_3 = 7
then RENTMTH = RENTSOM/6
if S2BQ15_3 = 8
then RENTMTH = RENTSOM/12
otherwise RENTMTH = RENTSOM

IMPRENT — the imputed rental value of those households that do not pay rent because they own their homes, converted to monthly rates

if S2BQ12_2 = 3
then IMPRENT = S2B12_1 * 30
if S2BQ12_2 = 4
then IMPRENT = S2B12_1 * 4.3
if S2BQ12_2 = 6
then IMPRENT = S2B12_1/3
if S2BQ12_2 = 7
then IMPRENT = S2B12_1/6
if S2BQ15_3 = 8
then IMPRENT = S2B12_1/12
else IMPRENT = S2B12_1

RENT — total monthly rent payment

SUM (RENTMTH, IMPRENT) * DEFLATOR

Section 10: Migration

Variable	*Short Description and Code*
NON_RES	those respondents who were not born in the area in which they live *if* S6AQ2 = 2 *then* NON_RES = 1
MIG_AGE	age when respondents left their birth place *if* S6AQ7 = 1 *then* MIG_AGE = 1
MIG_FAM	respondents who left their birth place for family reasons *if* S6AQ6 = 1 *and* S6AQ8 = 1 *then* MIG_ FAM = 1
MIG_WRK	respondents who left their birth place because of work *if* S6AQ6 = 1 *and* S6AQ8 = 2 *then* MIG_ WRK = 1
MIG_WRK1	respondents who left their birth place in search of work *if* S6AQ6 = 1 *and* S6AQ8 = 3 *then* MIG_ WRK1 = 1
MIG_STUD	respondents who left their birth place for school or study *if* S6AQ6 = 1 *and* S6AQ8 = 4 *then* MIG_ STUD = 1
MIG_MARR	respondents who left their birth place for marriage *if* S6AQ6 = 1 *and* S6AQ8 = 5 *then* MIG_ MARR = 1
MIG_MIL	respondents who left their birth place for military service *if* S6AQ6 = 1 *and* S6AQ8 = 6 *then* MIG_ MIL = 1
MIG_VIOL	respondents who left their birth place because of the threat of violence

if S6AQ6 = 1 *and* S6AQ8 = 7
then MIG_ VIOL = 1

MIG_ENV respondents who left their birth place because of environmental dangers

if S6AQ6 = 1 *and* S6AQ8 = 8
then MIG_ ENV = 1

MIG_OTH respondents who left their birth place for other reasons

if S6AQ6 = 1 *and* S6AQ8 = 9
then MIG_ OTH = 1

MIG_NUM number of places that respondents have lived for more than 3 months

if S6AQ6 = 1
then MIG_ NUM = S6AQ10

OR_RAI those respondents who moved to their current residence from within the same raion

if S6AQ10 >= 3 *and* S6AQ11 = 1
then OR_RAI = 1

OR_OB those respondents who moved to their current residence from within the same oblast

if S6AQ10 >= 3 *and* S6AQ11 = 2
then OR_ OB = 1

OR_REP those respondents who moved to their current residence from elsewhere within the republic

if S6AQ10 >= 3 *and* S6AQ11 = 3
then OR_ REP = 1

OR_CIS those respondents who moved to their current residence from another CIS country

if S6AQ10 >= 3 *and* S6AQ11 = 4
then OR_ CIS = 1

OR_OTH those respondents who moved to their current residence from another country

if S6AQ10 >= 3 *and* S6AQ11 = 5
then OR_ OTH = 1

MIR_KYRG those migrants of Kyrgyz nationality

if S6AQ6 = 1 *and* S6AQ1 = 1
then MIR_KYRG = 1

MIR_RUSS those migrants of Russian nationality

if S6AQ6 = 1 *and* S6AQ1 = 2
then MIR_RUSS = 1

MIR_TAJ those migrants of Tajik nationality

if S6AQ6 = 1 *and* S6AQ1 = 3
then MIR_ TAJ = 1

MIR_KOR those migrants of Korean nationality

if S6AQ6 = 1 *and* S6AQ1 = 4
then MIR_ KOR = 1

MIR_UZBK those migrants of Uzbek nationality

if S6AQ6 = 1 *and* S6AQ1 = 5
then MIR_ UZBK = 1

MIR_KAZK those migrants of Kazak nationality

if S6AQ6 = 1 *and* S6AQ1 = 6
then MIR_ KAZK = 1

MIR_BELO those migrants of Belo-Russian nationality

if S6AQ6 = 1 *and* S6AQ1 = 7
then MIR_ BELO = 1

MIR_UKRA those migrants of Ukranian nationality

if S6AQ6 = 1 *and* S6AQ1 = 8
then MIR_ UKRA = 1

MIR_TAT those migrants of Tatar nationality

if S6AQ6 = 1 *and* S6AQ1 = 9
then MIR_ TAT = 1

MIR_OTH those migrants of other nationalities

if S6AQ6 = 1 *and* S6AQ1 = 10
then MIR_ OTH = 1