ACC DAR Adapting to Climate Change in Coastal Dar es Salaam

Technical Workshop on Data Analysis

WORKING ON SPSS USING THE SYNTAX

Dar es Salaam, 28 September 2011

Centro Interuniversitario di Ricerca Per lo Sviluppo sostenibile - CIRPS

Sapienza Università di Roma

ARDHI UNIVERSITY TANZANIA UTAFITI ELIMU USHAURI

Pietro DEMURTAS

IRPPS – National Research Council

p.demurtas@irpps.cnr.it

What is the syntax of SPSS ?

A syntax file is a simple text file in which you write the commands you want for run the statistical analysis.

- When you open a new syntax file, first of all you see a white page (as with any word-processing program) and you must write the commands you need

Benefits from the use of SPSS syntax

- A **diary** of operations carried out during one or more sessions

- The ability to **replicate the procedures** you have done on the same matrix or other matrices

- A considerable **saving of time**, especially when you have to repeat many times the same operations on different variables. It's necessary to clarify that if you work through menus and dialog boxes you must proceed one step at a time. If, for example, you have to recode the values of ten variables that have the same mode of recoding, you must replicate ten times the same command, each time specifying a different variable. As we will see in practice, with a syntax file is possible to recode the values of ten variables with a single command.

To create a syntax file:

An **experienced user** can open a new syntax file from "File" menu and write the list of commands that will be applied to an array of data

(in the composition of the program you should follow the syntax rules and you will have to memorize the name of variables, of commands and sub-commands that allow to do the analysis on data)

For inexperienced users the easiest way to create a syntax is as follows**:**

Open a dialog box and set the selections you need. At this point, instead of directly running with the "OK" button, selecting the "PASTE“ button, SPSS will transcribe the syntax on a file that is opened automatically by the program.

Example of sintax file:

FREQUENCIES *[name of the procedure required]*
VARIABLES= d1_1 d1_2 d1_3 d1_4 d1_5 d1_6 . *[names of variables subject to the procedure]*
/STATISTICS= MEANS *[this sub-command requires the calculation of the MEANS for each variable]*
/HISTOGRAM *[this sub-command requires the histograms for each variable]*
/ORDER= ANALYSIS . *[this sub-command asks that the variables are processed in the order in which they are in the matrix]*

- NOTICE THAT:
- All rows that contain sub-commands begin with the symbol (/)
- From the second row, commands are preceded from a space
- At the end of the last row the symbol (.) represents the end of the command

General rules of syntax in SPSS

- The keywords and specifications must be separated by an equals sign
- Each command must begin on a new line and term with a period (.)
- Most of the sub-commands must be separated by the symbol (/). Generally, the symbol (/) can be omitted if it precedes the first sub-command of a command
- Variable names must be written in full
- The labels of variables and procedures must be written within the symbols (') and the text should not be divided into two lines
- Each row of syntax can not be longer than 80 characters
- The decimal must be preceded by a period (.) not by a comma (,).
- Variable names that end with a period (.) may cause errors in commands. So it's better avoid the use of period (.) in the label of a variable

DEFINING, MANIPULATING AND TRANSFORMING OF DATA

Before analyzing the data
you need to **prepare the data**
and this implies management and
manipulation operations of variables

- labeling variables
- specify values for missing data
- recode the categories of some variables
- constructing new variables from the variables in this data matrix

Labeling variables

This operation is done automatically in the transcription of data from LimeSurvey to SPSS but can be useful when we want to correct the labels.

In this case we label the variable dom44:

```
VARIABLE LABELS
dom44 'gender of respondents'.
 VALUE LABELS
dom44 1 'male' 2 'female'.
EXECUTE.
```

In the same syntax file we can labeling other variables (for example Unità ‘city where the research was done’)

Specify values for missing data

Often, in interviews with questionnaire, some data are missing (for example, if a person did not answer to the question).

Before the data analysis we treat missing data in this way:

```
MISSING VALUE
gender (9).
EXECUTE
```

Generally, the variable “gender” has two values (1 male and 2 female). In this case, for missing data we can use the values 9 or 0.

Obviously, 99 is the best value for the missing data in the variable "age", because it is possible that there are interviewed that have 9 years old

Recoding of variables

This treatment is necessary when a variable have unbalanced values, like this:

Class of Age	Frequency	Valid Percent	Cumulative Percent
20 -29	15	3,1	3,1
30 - 39	20	4,2	7,3
40- 49	143	29,7	37,0
50-59	303	63,0	100,0
TOTAL	481	100	

If we work with one variable, an unbalanced distribution is not a problem, while when we cross this variable with an other, it is possible that we have some cells empty. For this reason it is better to recode the first two classes in one.

- The procedure for recoding this categorical variable is:

RECODE age (1 thru 2 =1) (else = copie old values) INTO age_recod.
EXECUTE.

… and the distribution of new variable “age_recod” is this

Class of Age	Frequency	Valid Percent	Cumulative Percent
20-39	35	7,3	7,3
40- 49	143	29,7	37
50-59	303	63,0	100
TOTAL	481	100	

… a clarification!

We have three types of variables:

- **Categorical variables**
- **Ordinal variables**
- **Cardinal variables**

What is the difference between categorical, ordinal and cardinal variables?

Categorical (or nominal) variables

- A categorical variable (sometimes called a nominal variable) is one that has two or more categories, but there is no intrinsic ordering to the categories. For example, gender is a categorical variable having two categories (male and female) and there is no intrinsic ordering to the categories. Hair color is also a categorical variable having a number of categories (blonde, brown, brunette, red, etc.) and again, there is no agreed way to order these from highest to lowest. A purely categorical variable is one that simply allows you to assign categories but you cannot clearly order the variables. If the variable has a clear ordering, then that variable would be an ordinal variable.

Ordinal variables

- An ordinal variable is similar to a categorical variable. The difference is that in the ordinal variable there is a clear ordering of the values. For example, suppose you have a variable, economic status, with three categories (low, medium and high). In addition to being able to classify people into these three categories, you can order the categories as low, medium and high. Now consider a variable like educational experience (with values such as elementary school graduate, high school graduate some college and college graduate). These also can be ordered as elementary school, high school, some college, and college graduate. Even though we can order these from lowest to highest, the spacing between the values may not be the same across the levels of the variables. Say we assign scores 1, 2, 3 and 4 to these four levels of educational experience and we compare the difference in education between categories one and two with the difference in educational experience between categories two and three, or the difference between categories three and four. The difference between categories one and two (elementary and high school) is probably much bigger than the difference between categories two and three (high school and some college). In this example, we can order the people in level of educational experience but the size of the difference between categories is inconsistent (because the spacing between categories one and two is bigger than categories two and three). If these categories were equally spaced, then the variable would be an interval variable

Cardinal variables

we can distinguish two types of cardinal variables

- An ***Interval* variable** is a measurement where the difference between two values is meaningful. The difference between a temperature of 100 degrees and 90 degrees is the same difference as between 90 degrees and 80 degrees.

- A ***Ratio variable*** has all the properties of an interval variable, and also has a clear definition of 0.0. When the variable equals 0.0, there is none of that variable. Variables like height, weight, enzyme activity are ratio variables. Temperature, expressed in °F or °C, is not a ratio variable. A temperature of 0.0 on either of those scales does not mean 'no temperature'. However, temperature in degrees Kelvin in a ratio variable, as 0.0 degrees Kelvin really does mean 'no temperature!

Ordinal versus Cardinal Variables

- Cardinal variable: the difference between the values j and the value j +1 is the same as the difference between the values k and k + 1. Some common cardinal variables: wages, population.
- Ordinal variable: j +1 is bigger than j; but the difference between j + 1 and j is not necessarily the difference between k and k+ 1.

Does it matter for data analysis?

- The concepts are mostly pretty obvious, but putting names on different kinds of variables can help prevent mistakes like taking the average of a group of zip (postal) codes.

We can compute:	Nominal	Ordinal	Interval	Ratio
frequency distribution.	Yes	Yes	Yes	Yes
median and percentiles.	No	Yes	Yes	Yes
add or subtract.	No	No	Yes	Yes
mean, standard deviation, standard error of the mean.	No	No	Yes	Yes
ratio, or coefficient of variation.	No	No	No	Yes

Characteristic values of position

Measures of central tendency:

- Mode = value which corresponds to the highest frequency (nominal variables)
- Median= Md = middle value of an ordered distribution if N is odd: Md = (N +1) / 2,

  if N is even there are two middle cases, those who occupy the positions N / 2 and (N / 2 +1) (ordinal variables)
- Arithmetic mean = sum of all values of x divided by the number of cases (cardinal variables)

…as we have seen the recoding of variables is useful for all types of variables

Suppose that you want to reduce the variable "age" into a new variable “classes of age " (0-10, 11-20, 21-30, etc.):

the reduction of a cardinal variable in a categorical variable, of course, implies a loss of information, but this allows you to analyze it together with other categorical variables, which usually are the most of the variables in sociological research!

Command to recode the variable "age"

```
RECODE
  age (18 thru 22=1) (23 thru 27=2) (28 thru Highest=3) INTO C_age.
VARIABLES LABELS C_age 'classes of age'.
EXECUTE.
```

First row: *name of the procedure for variable recoding*

Second row: *inside the brackets are the old values of the variable "age", and after the equal sign (=) indicates the new value will be inserted into a new variable called "C_age"*

…

- The new variable C_age will be included in the first blank column of the data matrix and, once saved, will remain in the data matrix.
- Recoding can be done in the same variable or in a different variable. It's always better to create a new variable, because it can be useful to have the original variable as well as its recoded version.
- If you want to recoding in the same variable, simply omit from the syntax the command INTO followed by the name of the new variable

When you have to recode several variables that have the same pattern of recoding is possible to synthesize all operations in one command:

```
RECODE
  d1_1 TO d1_7
  (1 thru 2 =1) (3=2) (4 thru 5 =3) INTO R1_1 TO R1_7.
EXECUTE.
```

COMPUTE a new variable

The COMPUTE command is used to calculate the values of a new variable using arithmetic , statistics or logic transformations of other variables in the data matrix. It's a very important function, used for example when you want to build an additive index (which will be discussed later)

For example, if we have a variable "year of birth" and we need to determine the age of the people, we will use the **COMPUTE** command:

COMPUTE age = 2011 – yearbirth.

VARIABLE LABELS age ‘age’.

EXECUTE.

First row: keyword of the command, followed by the name the new variable which is the result of the expression after the equal sign (=) 2011 - yearbirth

In this way, for each subject, it will be calculated the difference between the year 2011 and his year of birth and the result it will wrote in the new variable "age", which will be placed in the first empty column on the data matrix

Calculate the Unemployment Rate

If you want to compare the number of unemployed in two different cities (city-A and city-B) it's necessary to take into account the population in the two cities. In fact, thirty thousand unemployed people in City-A do not have the same meaning of thirty thousand unemployed people in City-B.

To assess the impact of the unemployed in the two cities is therefore necessary to relate the number of unemployed to active population (number of people who are not too young or too old to work) and then multiply this result by one hundred

The program to calculate the unemployment rate is:

```
COMPUTE r_un = (n_un/active)*100.
VARIABLE LABELS  r_un ‘unemployement rate’.
EXECUTE.
```

*r_un is the new variable “unenployment rate’

** “n_un” is the number of unemployed (30.000)

*** “active” is the active population

(if this value is higer in City-B than in City-A the rate of unemployment in City-A will be higher than in City-B)

To analyse a sub-group of cases (individuals)

If we want to do an analysis in a subset of respondents males, we must set the following program command:

```
FILTER OFF
USE ALL.
SELECT IF (sex=1).
EXECUTE.
```

- The command in the two first rows deletes previous filters on the data matrix
- in the third row, the keyword "SELECT IF" asks to select all cases (individuals) that have the code 1 (male) on the variable SEX

- Warning!

This version of program selects the cases that have code 1, but deletes all other cases with a different code.

- If we want to select some cases without losing the others from the data matrix, we must use this program:

```
USE ALL.
COMPUTE filter_$=(sex = 1).
VARIABLE LABEL filter_$ ‘Sex = 1 (FILTER)'.
VALUE LABELS filter_$ 0 'Not Selected' 1 'Selected'.
FORMAT filter_$ (f1.0).
FILTER BY filter_$.
EXECUTE.
```

- The last program creates a new variable called '$ filter_', whose values are 1 for selected cases and 0 for unselected cases.

- The unselected cases are marked on matrix with a bar, but not deleted from the data file.

- Deleting the variable '$ filter_' all cases will be available again

DATA ANALYSIS

Compare means

Frequencies

Multiple responses

Cross tabs