

## Managing Missing Values

### ***I. Introduction to missing values***

In every census or survey, it is always an aim to achieve complete capture of all data. However, large datasets may contain missing data or missing values. Not considering these missing data will affect the interpretation of the generated results. A large amount of missing data can give misleading results. Thus, it is necessary to know how to manage and reduce the amount of missing data. This session will provide procedures on how to check and identify missing data using SQL syntax. This session will also give possible ways to correct, minimize or better yet avoid missing data.

### ***II. What are missing values?***

Missing values occur when a certain case or observation has no data value for a given variable. With missing values, the whole data is regarded as “incomplete”. Data incompleteness can occur in varying extent. Incompleteness due to missing values might be present in one, two or sometimes all variables for some or all observations. In commercial statistical softwares, missing values are indicated by “.” (dot), 99 or blank space. On the other hand, the StatSim, since it is using SQL, assigns NULL as the missing data. NULL is defined as no value in the text field, or either no value or no designated value in a numeric field. For example, in one household the name of the respondent or the highest educational attainment of one member was not encoded hence in StatSim processed data it is indicated as NULL.

### ***III. Types of missing values***

There are two types of missing values.

A. User missing values - data are missing due to not properly field-edited or not properly encoded questionnaire. However, there are possible reasons why the data are missing. By knowing the reason, user missing data can be grouped into valid and invalid cases. For valid cases, reason can be one of the following:

1. Data are missing because the question didn't apply to the respondent. This also applies to skipping patterns in questionnaires.
2. Data are missing because a respondent refused to answer. This is called non-response.

On the other hand, invalid cases of missing values maybe due to the following:

1. The data enumerators might have forgotten to gather information on some questions.
2. The data encoder missed to encode the responses.
3. Data encoder errors.

- B. System missing values- are automatically assigned by the program when no valid value can be produced, such as when an alphabetical character is encountered in the data for a numeric variable. For example, for variable sex valid values are 1 for Male and 2 for Female but sometimes there are encoded values such 0 (zero), a, or A.

For this session, we will focus on how to identify invalid user missing values and minimize or possibly correct these.

#### **IV. Identifying/Checking for missing values using SQL**

In StatSim processed data, all user missing values are denoted by null. Hence, SQL syntax will be used to identify the missing data in order to classify the valid and invalid cases. Identification variables (such as Urbanity, Region, Province, Municipality, Barangay, Purok, HH number and Member number) should all have valid values. The syntaxes below will list the cases with missing values for all identification variables.

##### **Household record**

```
SELECT *
FROM hpq_hh
WHERE prov is NULL OR mun is NULL OR brgy is NULL OR purok is NULL OR hcn is NULL;
```

##### **Member record**

```
SELECT *
FROM hpq_mem
WHERE prov is NULL OR mun is NULL OR brgy is NULL OR purok is NULL OR hcn is NULL OR memno is NULL;
```

It is also crucial to check the missing values for variables used to generate core indicators. The syntaxes below will list the cases with missing data.

##### **Household record**

```
SELECT *
FROM hpq_hh
WHERE roof is NULL OR wall is NULL OR tenur is NULL OR water is NULL OR toil is NULL OR totin is NULL OR fshort is NULL OR hsize is NULL;
```

##### **Member record**

```
SELECT *
FROM hpq_mem
WHERE msname is NULL OR mfname is NULL or age_yr is NULL OR sex is NULL;
```

##### **Malnutrition**

```
SELECT *
FROM hpq_mem
WHERE age_yr<=5 AND mnutind is NULL;
```

**Education**

```

SELECT *
FROM hpq_mem
WHERE (age_yr>=6 AND age_yr<=16) AND educind is NULL;
SELECT *
FROM hpq_mem
WHERE (age_yr>=6 AND age_yr<=16) AND educind =1 AND gradel is NULL;

```

**Employment**

```

SELECT *
FROM hpq_mem
WHERE age_yr>=15 AND jobind is NULL;

```

```

SELECT *
FROM hpq_mem
WHERE age_yr>=15 AND jobind=2 AND fjob is NULL;

```

```

SELECT *
FROM hpq_mem
WHERE age_yr>=15 AND jobind = 2 AND fjob=2 AND (lastlookjob is NULL OR ynotlookjob is NULL OR joppind is
NULL or wtwind is NULL);

```

**Death record**

```

SELECT *
FROM hpq_death
WHERE mdeadage is NULL OR mdeady is NULL;

```

**Crime record**

```

SELECT *
FROM hpq_crime
WHERE ctvicttot is NULL OR ctvictmale is NULL OR ctvictfemale is NULL;

```

**IV. How to reduce missing values**

Now that a list of missing values in the dataset is available using SQL, the next step is to distinguish the user valid and invalid missing data. Take a look at this case:

rtype	urb	regn	prov	mun	brgy	purok	hcn	water	water_o	water_dist	toil	toil_o	tenur	tenur_o	fshort
00	1	17	52	11	002	01	0006	1	NULL	1	1	NULL	1	NULL	NULL
00	1	17	52	11	002	03	0122	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
00	1	17	52	11	002	04	0208	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL
00	1	17	52	11	002	07	0348	1	NULL	1	1	NULL	1	NULL	NULL
00	1	17	52	11	002	05	0652	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL



Household number 6 and 348 have data on *water*, *water\_dist*, *toil* and *tenur* but has a missing data on variable *fshort*. To distinguish if these are non-response or incomplete data encoding, the questionnaire of the said households should be checked. If those are non-response cases then it should be tagged as valid user missing data. On the other hand, if those are cases which the encoder missed to encode the responses then corrections should be made on data encoding.

For households 122, 208 and 652, all given variables are null. Again, the questionnaire should be checked to verify if these are non response or incomplete data encoding. If after checking the encoded data, it was found out that the said households have no data at all then there is a need to verify whether these households:

1. Do not really exist or;
2. Were not in the barangay during the time of survey or;
3. Refused to be interviewed

If data for these households will not be recovered then it is recommended to delete these cases in the encoded file. Below is an example of an encoded questionnaire which has no data after the assessment field.

CBMS Survey Members Record					
Urbanity <i>Lokasyon</i>	Municipality <i>Lungsod/Bayan</i>	Barangay <i>Barangay</i>	Purok <i>Purok</i>	HH ID	208
Interviewer <i>Tanggapayam</i>	Address Line 1 <i>Tirahan</i>		SITIO MALIGAYA, BALATERO, PUERTO		
Respondent <i>Nakapanayam</i>	Address Line 2 <i>Tirahan</i>		GALERA, ORIENTAL MINDORO		
Date <i>Petsa</i>	Time started <i>Nagsimula</i>	Time ended <i>Natapos</i>	Assessment		
4/12/2008	3:29	3:39	SUMAGOT NG MAAYOS		
<b>B. Demography</b> <i>Demograpiya</i>					
Number of Members <i>Bilang ng miyembro</i>					
(1)	(2)	(3)	(4)	(5)	(6)
Line number (Bilang)	Member (Surname, First name)	Relation to the household head (code, other)	Sex (Kasarian)	Birthdate - MM/DD/YYYY (date, age)	Civil status (code, other)
1					
2					
3					