

## Unit 8      Deriving and reclassifying fields

IBM Training



# Deriving and reclassifying fields

IBM SPSS Modeler (v18)

© Copyright IBM Corporation 2016  
Course materials may not be reproduced in whole or in part without the written permission of IBM.

**Unit objectives**

- Use the Control Language for Expression Manipulation (CLEM)
- Derive new fields
- Reclassify field values

Deriving and reclassifying fields

© Copyright IBM Corporation 2016


*Unit objectives*

The focus in this unit is on another task in the data preparation stage: construct the final dataset for modeling by cleansing and enriching your data.

Before reviewing this unit you should be familiar with:

- CRISP-DM
- IBM SPSS Modeler streams, nodes and palettes
- methods to collect initial data
- methods to explore the data

## Methods to create fields



| ID | BDATE      | age | agecat | adult | GENDER | gender_ok |
|----|------------|-----|--------|-------|--------|-----------|
| 1  | 01/24/1940 | 71  | 3      | T     | Fem    | Female    |
| 2  | 05/11/1968 | 43  | 2      | T     | F      | Female    |
| 3  | 09/11/1989 | 22  | 1      | T     | Female | Female    |
| 4  | 10/14/1992 | 19  | 1      | F     | MALE   | Male      |

Deriving and reclassifying fields

© Copyright IBM Corporation 2016

### Methods to create fields

This unit presents two field operation nodes that can be used for cleansing and enriching your data. The Derive node computes new fields; the Reclassify node recodes the values of a categorical field.

This slide shows some examples:

- Based on BIRTHDATE, three new fields are derived: age, age category (1 junior, 2 middle-age, 3 senior) and a field flagging if the person is 21 years or older.
- The GENDER field shows inconsistencies in spelling and is reclassified into a new field with values Female and Male.

IBM SPSS Modeler provides many more field operations nodes to prepare your data for analyses and modeling. Refer to the *Advanced Data Preparation Using IBM SPSS Modeler* course for more information.

## Introduce CLEM

- CLEM = Control Language for Expression Manipulation
- IBM SPSS Modeler's native language to build expressions
- Derive node, Select node use CLEM (amongst others)
- Refer to the online Help for a full presentation of CLEM

Deriving and reclassifying fields

© Copyright IBM Corporation 2016

### *Introduce CLEM*

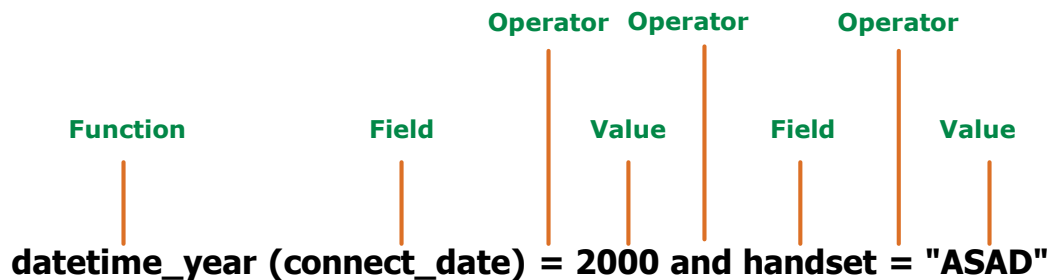
IBM SPSS Modeler implements a powerful language for specifying expressions called Control Language for Expression Manipulation (CLEM). CLEM is used in a number of nodes, among which the Select and the Derive node.

CLEM enables you to:

- specify expressions to assign values to fields, for example,  $\text{tax} = \text{income} * 0.1$
- specify conditions, for example,  $\text{income} < 10000$

Refer to the online Help for a detailed presentation of CLEM. In this unit you are introduced to the basic concepts.

## Identify CLEM expressions



Deriving and reclassifying fields

© Copyright IBM Corporation 2016

### Identify CLEM expressions

This slide shows an example of a CLEM expression, specifying a condition. The condition returns true for a customer if he connected in the year 2000 and has handset ASAD. CLEM expressions are constructed from values, fields, operators and functions. When writing CLEM expressions, take note of the following:

- Field names are case sensitive. For example, fields AGE and Age are not the same.
- When a field name contains a blank or if it is a special field name it needs to be enclosed in single quotes for example: 'INCOME 2012', '\$RC-churn'.
- String values should be within single or double quotes, for example 'male' or "male", 'married' or "married". Occasionally, in specific functions such as locchar, a string value needs to be enclosed in single back quotes ` and `.
- Values smaller than 1 (in absolute value) must be specified with a leading zero, or else IBM SPSS Modeler will issue an error message such as *CLEM error: Illegal token' .' in expression: probability > .5* (the value should have been specified as 0.5).

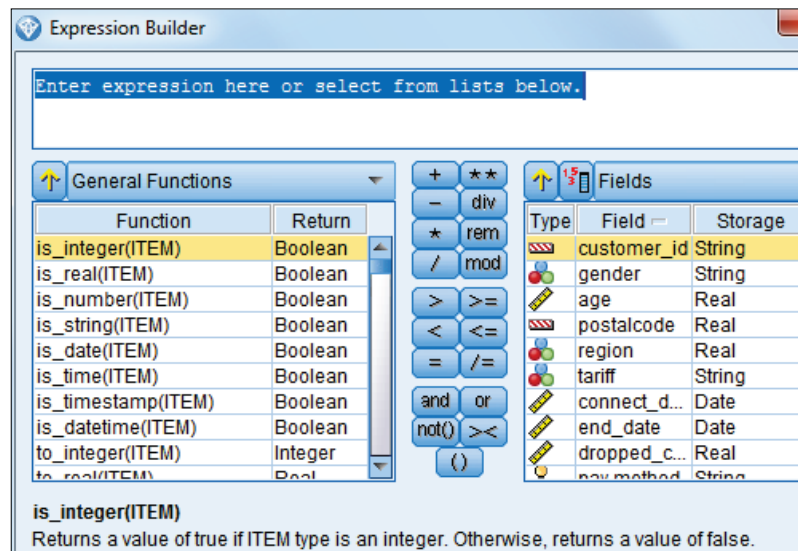
- Operators can be:
  - arithmetic: +, -, \*, /, \*\* (raise to the power)
  - relational: >, <, <=, >=, =, /= (unequal)
  - logical: and, or, not (all in lower case).

To be not dependent on how IBM SPSS Modeler evaluates an expression, it is advised to use parentheses in compound conditions. For example, if you want to refer to men, or those working full time with an income greater than 10000:

(gender= "male") or (job\_status="full time" and income > 10000)

- IBM SPSS Modeler offers many functions. Refer to the *Advanced Data Preparation Using IBM SPSS Modeler (v16)* course for more details or the online Help for a complete overview. Function names are case-sensitive.

## Explore the Expression Builder



Deriving and reclassifying fields

© Copyright IBM Corporation 2016

### Explore the Expression Builder

In nodes such as Select and Derive you can type your CLEM expression, but that is not an efficient option, especially because field names and function names are case-sensitive. Instead of, or in conjunction with typing CLEM expressions, you can use the Expression Builder to create expressions.

You can invoke the Expression Builder by clicking the Launch expression builder button in the Select or Derive dialog box.



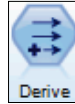
This slide shows the Expression Builder dialog box. Build your expression by selecting and pasting the various elements (fields, functions, values and operators) to the area where the expression must be specified.

Functions are grouped by categories, such as string functions, date and time functions, numeric functions, and logical functions. When you select a function you will have a description of its use at the bottom of the dialog box.

When you need to specify a value of a categorical field IBM SPSS Modeler offers the very user-friendly feature to pick the value from a list of values, provided that the field is instantiated. If the field is not instantiated, its values will not be available.

## Derive fields

- Create new fields
- Use the Derive node (Field Ops)
- Derive types:
  - Formula
  - Flag
  - Nominal
  - Conditional



### Derive fields


The Derive node, located in the Field Ops palette, will add a new field to the dataset. The Derive node does not let you overwrite an existing field. If you want to overwrite a field, use a Filler node.

Using the Derive node you can derive a field of one of the following types:

- **Formula:** An outcome of a formula. For example: a new field TAX derived as:  $TAX = INCOME * 0.20$ .
- **Flag:** A T/F field. For example: a new field ADULT, T when  $AGE \geq 21$ , else F.
- **Nominal:** A categorical field. For example: a new field AGECAT, 1 when  $AGE \leq 35$ , 2 when  $AGE > 35$  and  $AGE \leq 70$ , and 3 when  $AGE > 70$ .
- **Conditional:** An outcome of a formula, but computed conditionally. For example: a new field  $TAX = 0.1 * INCOME$  if  $INCOME \leq 100000$ , and  $TAX = 10000 + 0.2 * (INCOME - 100000)$  if  $INCOME > 100000$ .

For other derive types or for a presentation of the Filler node, refer to the online Help or the *Advanced Data Preparation Using IBM SPSS Modeler* course.



**Note on blanks when deriving fields****Derived fields**


| ID | AGE      | INCOME   | adult    | income_in_1000s |
|----|----------|----------|----------|-----------------|
| 1  | \$null\$ | 21000    | \$null\$ | 21              |
| 2  | 23       | \$null\$ | T        | \$null\$        |
| 3  | 19       | -1       | F        | -0.001          |
| 4  | 999      | 10000    | T        | 10              |

Deriving and reclassifying fields

© Copyright IBM Corporation 2016

*Note on blanks when deriving fields*

IBM SPSS Modeler will treat user-defined blank values as valid values when a field is derived.

This slide shows a few examples of how IBM SPSS Modeler handles blanks. The value 999 is declared as blank value for AGE, and -1 is declared as blank value for INCOME. Based on AGE and INCOME two fields are derived. The first field, adult, equals T when AGE is greater than 20, else it returns F. The second field, income\_in\_1000s, is derived as INCOME/1000. Although 999 is declared as a blank value for AGE, IBM SPSS Modeler will treat the blank as any other value and so adult equals T when AGE equals 999. When INCOME equals -1, the blank value for this field, income\_in\_1000s field is computed as -1/1000, with -0.001 as the result.

When the original values are undefined (\$null\$), the result is also undefined (\$null\$), which is as it should be.

Note: The undefined value (\$null\$) is referred to in IBM SPSS Modeler's user-interface as undef. When you specify undef as value, IBM SPSS Modeler will return \$null\$.

## Reclassify fields

- Recode the values of a categorical field into broader categories.



- Use the Reclassify node (Field Ops).

### *Reclassify fields*

Sometimes you need to recode the values of a categorical field into broader categories. For example, a field that stores a customer's specific job position may be more useful for prediction if it is reclassified into broader job categories.

The Reclassify node, located in the Field Ops palette, enables you to reclassify values of a categorical field.

It should be emphasized that the Reclassify node is to recode categorical fields only. For example, if you want to recode the continuous field AGE into age categories, use the Derive node, not the Reclassify node.

## Check the results

- Preview the data, or run a Table node.
- Use a Matrix node.
- Use an Aggregate node.
- Use a Data Audit node.

Deriving and reclassifying fields

© Copyright IBM Corporation 2016

### *Check the results*

Whenever new fields are created it is advised to check the results. A formula can be specified incorrectly, values for a field can be specified incorrectly, conditions can be specified following human instead of computer logic, blank values can affect the results, and so forth. In many of these situations IBM SPSS Modeler will not issue an error message, so simply not receiving an error does not mean that the new field is correct.

How to check a new field depends on how that field was created:

- For fields created from formulas, simply review the output from a Table node and calculate a few values to check the equation.
- For a categorical fields created from another categorical field, use a Matrix node to cross tabulate the fields.
- For categorical fields created from a continuous field, use the Aggregate node with the new field as key field and request the minimum/maximum of the original field.

Also, running a Data Audit node will show the minimum and maximum and provides a quick check.