C2 Metadata

SDTL, SPSS, R, AND APPS
Overview

- SDTL
- SPSS
- R
- Apps to detect and view transforms
SDTL

Structured Data Transform Language
SDTL

- Structured Data Transform Language
- A model to describe things people do to transform data
SDTL: Assign a variable label

- **Stata**
  - `label variable price "the price of the car in 1978"

- **SPSS**
  - `VARIABLE LABELS price "the price of the car in 1978"

- **SDTL**
  - Command: "setVariableLabel"
  - Variable: "price"
  - Label: "the price of the car in 1978"
Recode Values (Stata)

**recode** -- Recode categorical variables

```stata
recode var1 var2 (1 2 = 2) (3/max = 5)
recode a b (1 = 2), prefix("modified_")
recode x y (1 = 2), generate(modified_x modified_y)
recode x y (1 = 2 "Labelfor2") (3 = 5 "Labelfor5")
recode total (0/140=0 F) (141/180=1 D) (181/210=2 C) (211/234=3 B) (235/300=4 A), gen(grade)
recode a (1 . .a 5/6 = 7) (nonmissing = 8) (missing = 9) (* = 2)
```
RECODE var1 var2 var3 (-1=7) (-2=8).

RECODE AGE (MISSING=9) (18 THRU 110=1) (0 THRU 18=0) INTO VOTER.
Recode in SDTL

- Command: “recode”
- recodedVariables: “age”
- Rules [0]:
  - fromValue: -1 to: 7
- Rules [1]:
  - fromValue: -2 to: 8
SDTL Development

- Using COGS, an open source production framework to generate JSON schema, XML schema, and rich documentation
  
- http://c2metadata.gitlab.io/sdtl-docs/
SDTL Documentation

Recode

Describes recoding values in one or more variables according to a specified mapping.

Contents

- Recode
  - Item Type Hierarchy
  - Relationships
  - Properties
    - RecodedVariables
    - RecodedVariableRange
    - Rules

Item Type Hierarchy

- TransformBase
  - Recode
SDTL Development

- Still in active development
- gitlab.com/c2metadata/sdtl-cogs
SPSS
Detecting Transforms in SPSS Syntax

1. Lexer
2. Parser
3. Tree walker
4. Standard format

SPSS syntax → Token stream → Abstract syntax tree (AST) → SDTL
Detecting Transforms in SPSS Syntax

- Inputs: SPSS source code
- Output: SDTL JSON
R Package for Transforming Data

- R is a more sophisticated, less domain specific language than SPSS and others
- Instead of detecting transforms: produce an R package to perform transforms
- Wrap commonly used libraries like hmisc and dplyr
- Perform transforms easily. Output SDTL as a byproduct

- Work begins later this year
Apps to Detect Transforms
SPSS: Command line

C:\svn\spss-sdtl-converter\src\C2Metadata_SpssToSdtl.Cli (master)
A dotnet run -- c:\svn\spss-samples-public\basic\recode.sps -o d:\out\recode.json
Recognized command count: 4
SDTL Reader

- Desktop application for Windows, macOS, and Linux
- Open SPSS syntax files (*.sps)
- Open SDTL JSON files
Summary:

- M3_PHONE_STEP 1_GENERAL CLEANING
- M3_PHONE_STEP 2_INITIAL FREQS
- M3_PHONE_STEP 3_RECODE MISSING VALUES
- M3_PHONE_STEP 4_CHECKING VALUE RANGES
- M3_PHONE_STEP 5_CHECK SKIP PATTERNS
- M3_PHONE_STEP 6_CREATING NEW VARIABLES FOR
- M3_PHONE_STEP 7_RELABEL VARIABLE LABELS
- M3_PHONE_STEP 8_DEFINE MISSING VALUES
- M3_PHONE_STEP 9_UPDATE VALUE LABELS
- M3_PHONE_STEP 10_RENAME UWSC VARIABLE NAMES
- M3_PHONE_STEP 11_MERGE ROSTER WITH PHONE
- M3_PHONE_STEP 12_GENERAL CLEANING
- M3_PHONE_STEP 13_SCALES AND CONSTRUCTED VARS
- M3_PHONE_STEP 14_FORMATS
- M3_PHONE_STEP 15_ADD OCC-IND CODES

Summary:

<table>
<thead>
<tr>
<th>Total</th>
<th>401</th>
</tr>
</thead>
<tbody>
<tr>
<td>recode</td>
<td>132</td>
</tr>
<tr>
<td>execute</td>
<td>117</td>
</tr>
<tr>
<td>compute</td>
<td>53</td>
</tr>
<tr>
<td>rename</td>
<td>28</td>
</tr>
<tr>
<td>load</td>
<td>17</td>
</tr>
<tr>
<td>title</td>
<td>15</td>
</tr>
<tr>
<td>subtitle</td>
<td>15</td>
</tr>
<tr>
<td>save</td>
<td>15</td>
</tr>
<tr>
<td>invalid</td>
<td>9</td>
</tr>
</tbody>
</table>
Recoded Variables: A49 -> A49

Recode Rules:
-1 -> 97
-2 -> 98
-3 -> 96

Source File: C:svn\spss-samples-private\midus\M3\M3_PHONE_STEP_3_RECODE_MISSING_VALUES.RECODE

Source:
RECODE A49
(-1=97) (-2=98) (-3=96).
Web Service and Docker Container

• Inputs: SPSS Syntax
• Outputs: SDTL JSON
• Install and run:

$ docker pull registry.gitlab.com/c2metadata/sdtl-reader:latest
$ docker run -d -p 8080:80 registry.gitlab.com/c2metadata/sdtl-reader

$ curl -X POST "http://localhost:8080/api/spsstosdtl" -d "COMPUTE x = 1"
Follow Along

gitlab.com/c2metadata