Updating a Generalized Imputation System to Include a Quadratic Program

Laura Bechtel

Nicole Czaplicki

Thong Nguyen

United States Census Bureau

U.S. Department of Commerce
Economics and Statistics Administration
U.S. CENSUS BUREAU
census.gov

Background

- Standard Economic Processing System (StEPS)
  - Generalized software based entirely in SAS used for the various stages of a survey like data entry, editing, imputation, and estimation.
  - SAS is no longer supporting the AF screens that StEPS uses, so the next generation of generalized software is under development, StEPS II.
- General Imputation Subcommittee
  - Our focus is imputation and testing the new generalized software's imputation capabilities.
  - We found a problem!
    - Erroneous solutions could arise when resolving an out of balance complex with negative data.

Balance Complex

- $y = \sum_{i=1}^{D} x_i$
  - D is the number of details items, x_i's
  - y is the total item

  - Nested balance complexes
    - Detail item that is a total in another complex should be held constant

Many Ways to Resolve a Balance Complex

- Replace total with the sum of the details
- Place residual ($y - \sum_{i=1}^{D} x_i$) in one detail item
- Modify each detail by a “little” bit so that the sum of the details is equal to the total item, *y*

Raking

- **Originally developed for non-negative data**
- **Modified for negative data**:

$$x_i' = \left(1 + sign(x_i)\frac{y - \sum x_i}{\sum abs(x_i)}\right) x_i$$

  - x_i' is the imputed (perturbed) value for item i
  - x_i is the original detail value for item i
  - y is the total
- **Implementation issues**
  - No straightforward way to hold an item constant
  - When x_i' <0 and item i cannot be negative, a modification needed to be made….

Fictitious Motivating Example

Order of x_i's	Y	x_1	x_2 (non-negative)	x_3
Input	**-200**	**-12**	**59**	**-17**
w/o additional requirements	***-200***	***-43***	***-95***	***-62***
x_1, x_2, x_3	**-200**	**-43**	**0**	**-157**
x_2, x_1, x_3	**-200**	**-138**	**0**	**-62**
x_1, x_3, x_2	**-200**	**-43**	**0**	**-62**

Researching a Solution

- The Quarterly Financial Report (QFR)
  - A sample survey used to produce estimates of
    - Financial statements and ratios
    - Two principal economic indicators
  - Some data items can be negative
  - Nested balance complexes

- Two Objectives for the Solution
  - Analyst Correction (AC)
    - Currently no automated procedures
    - Corrections according to detail item reliability
  - Raking Imputed Data
    - All detail items being adjusted assumed to have equal reliability

Quadratic Program (QP)

Equivalent to raking when $c_i = \frac{1}{|x_i|}$

$$f(\mathbf{x}') = \sum_{i=1}^{D} c_i (x_i - x_i')^2$$

- Minimize $f(\mathbf{x}')$ subject to:
  - $y = \sum_i x_i'$
  - Nonnegative items must have solution ≥ 0
  - Input zero values should not be perturbed

Fictitious Motivating Example – QP Results

Costs	Y	x_1	x_2 (non-negative)	x_3
Input	-200	-12	59	-17
$c_1 = c_2 = c_3 = 1$	-200	-97.5	0	-102.5
$c_1 = 10$, $c_2 = 25$ $c_3 = 75$	-200	-162.88	0	-37.12
$c_i = \frac{1}{\lvert x_i \rvert}$	-200	-99.7	0	-100.3

Implementation

- Two-Phased
  - Survey-specific code based on research code
    - This helped make it a priority for StEPS II
  - Added to generalized software (StEPS II)

- Requirements Gathering
  - User interface vs backend SAS code
  - Largely based on research code

- Research Code
  - SAS PROC NLP

Requirements Gathering

- Team
  - Project Managers
  - Subject Matter Experts (Analysts)
  - Methodologists (Mathematical Statisticians)
  - User Interface Programmers
  - SAS Programmers

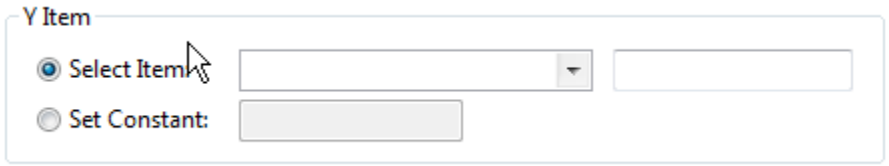
- Initial requirements developed for backend SAS code
  - Used to guide user interface requirements

Interface Requirements

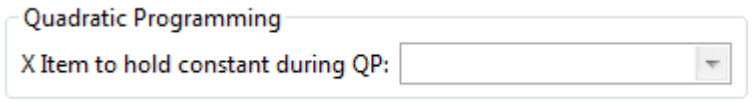
- Led by project managers not involved in the research
- Initial interpretation of backend SAS code requirements sometimes needed to be corrected
- Finding appropriate wording for parameter definition was difficult
- Sometimes exact directions on implementation needed to be provided
- Having the User Interface Programmer available during discussions was key to the timeliness of the project

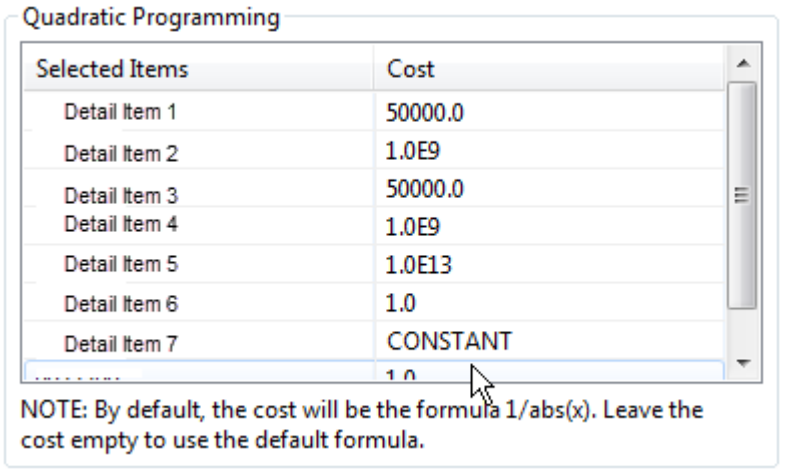
Examples of Interface Issues

- Numerical Constant



- Constant Detail Item





Backend SAS Code Requirements

- Very complicated to explain a quadratic program
- Led by the researchers
- Researchers knew the research code AND the generalized imputation code very well, which helped bridge the gap
- Statistical background of backend SAS programmer was key
- Having the backend SAS programmer continually in communication was/is imperative to the success of the project
- Some issues were directly related to using PROC NLP

PROC NLP

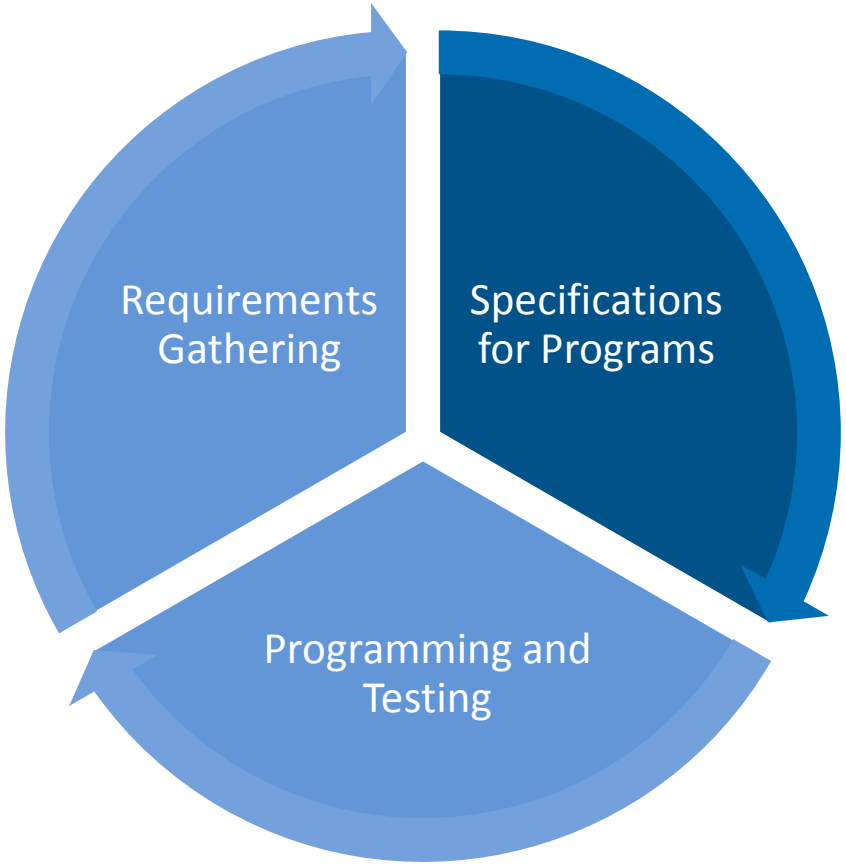
- We briefly considered PROC OPTMODEL, but it requires more complicated input
- Input parameters
  - Research – ALL input parameters for ALL cases had macro variables
  - Generalized Code – backend SAS programmer cleaned this up
- PROC NLP has ERRORs in the log that are not really errors
  - When default algorithm does not converge and an alternative algorithm is used to resolve the balance complex, an error is output to the log
  - Problem – Generalized Code bombs if there are errors in the log
  - Solution – output PROC NLP log elsewhere
- Generalized Imputation processes all cases in one data step, but PROC NLP processes one case at a time
  - CALL EXECUTE
  - Update the output differently

The Database

- Adding the generalized capability to StEPS resulted in an additional 100 (new) columns in the database for EVERY survey
- Changes to the database are met with resistance
- This is a challenge when adding a new method to a generalized system

It is a Cyclic Process and Communication is Key

We had a real customer, a real problem and a real need. This helped us to advocate for the implementation when needed.



Vital issues came to light and were resolved when the key team members were in the same room or on the same phone speaking to one another.

We Are Still Working!

- New error was found two weeks ago!

- Another round of researchers troubleshooting, identifying the solution for the program managers and discussing the solution with the SAS programmer.

Thank You

- Laura.Bechtel@census.gov
- Nicole.Czaplicki@census.gov
- Thong.Minh.Nguyen@census.gov