Simultaneous Edit-Imputation for Categorical Microdata

Daniel Manrique–Vallier
Department of Statistics, Indiana University

Jerome P. Reiter
Department of Statistical Science, Duke University

2013 FCSM Research Conference
November 6th, 2013

Research supported by NSF grant SES-11-31897.

The problem

Inconsistent Datasets

- Many individual level multivariate datasets, e.g. surveys, have consistency requirements specifying combinations of responses that are not allowed.
- In real-life, however, datasets often include errors.
  - When the errors end up in a violation of a consistency rule, we can detect the error.
  - When the error doesn't result in a consistency rule violation, the error is not detectable.

The problem

Inconsistent Datasets

- Many individual level multivariate datasets, e.g. surveys, have consistency requirements specifying combinations of responses that are not allowed.
- In real-life, however, datasets often include errors.
  - When the errors end up in a violation of a consistency rule, we can detect the error.
  - When the error doesn't result in a consistency rule violation, the error is not detectable.

We Want

1. Detect and locate errors (even if they don't result in the violation of a consistency rule.)
2. Impute consistent values, respecting the distribution the data, and reflecting the uncertainty associated with the procedure.

Conceptualizing the Problem

- Data consists of vectors $\mathbf{Y}_i = (Y_{i1}, ..., Y_{iJ})$, $i = 1, ..., n$ (e.g. *recorded* responses to J survey questions)
- Each of the J components take values from a finite set $Y_{ij} \in \{1, 2, ..., L_j\}$.
- Entries in $\mathbf{Y}_i$ might be inconsistent. Then $\mathbf{Y}_i \in \mathcal{C} = \prod_{j=1}^{J} \{1, ..., L_j\}$.
- Consistency rules are a collection of $S \subsetneq \mathcal{C}$ that specify which values of $\mathbf{Y}_i$ shouldn't be present in the dataset.
- Connections to structural zeros in contingency tables.

A Generative Perspective

- The observed response $\mathbf{Y}_i$ is a contaminated version of a "true" underlying response, $\mathbf{X}_i$.
- $\mathbf{Y}_i$ is observed. $\mathbf{X}_i$ is unobserved.
- $\Pr(\mathbf{Y}_i \in S) > 0$. $\Pr(\mathbf{X}_i \in S) = 0$.
- We assume a generation process for $\mathbf{X}_i$

$$\mathbf{X}_i \stackrel{iid}{\sim} F,$$

  which doesn't allow for inconsistent values. $\mathbf{X}_i \in \mathcal{C} \setminus S$.
- $\mathbf{Y}_i$s come from an "error process"

$$\mathbf{Y}_i | \mathbf{X}_i \sim E(\mathbf{X}_i).$$

  which allows for inconsistent values. $\mathbf{Y}_i \in \mathcal{C}$.

A Generative Perspective

- The observed response $\mathbf{Y}_i$ is a contaminated version of a "true" underlying response, $\mathbf{X}_i$.
- $\mathbf{Y}_i$ is observed. $\mathbf{X}_i$ is unobserved.
- $\Pr(\mathbf{Y}_i \in S) > 0$. $\Pr(\mathbf{X}_i \in S) = 0$.
- We assume a generation process for $\mathbf{X}_i$

$$\mathbf{X}_i \overset{iid}{\sim} F,$$

which doesn't allow for inconsistent values. $\mathbf{X}_i \in \mathcal{C} \setminus S$.
- $\mathbf{Y}_i$s come from an "error process"

$$\mathbf{Y}_i | \mathbf{X}_i \sim E(\mathbf{X}_i).$$

which allows for inconsistent values. $\mathbf{Y}_i \in \mathcal{C}$.

Our objective is to estimate F.

Error models

- Given true data, the error process determines what we observe.
- We differentiate two components:
  1. **Location model:** Which items are in error?
  2. **Substitution model:** Given that there's an error at the (i, j) location, how does Y_{ij} is generated from X_{ij}?

Error models

- Given true data, the error process determines what we observe.
- We differentiate two components:
  1. **Location model:** Which items are in error?
  2. **Substitution model:** Given that there's an error at the (i, j) location, how does Y_{ij} is generated from X_{ij}?
- Let $E_{ij} = 1$ if there's an error at the (i, j) location, and 0 otherwise. We define the *error mask* $\mathbf{E}_i = (E_{i1}, ..., E_{iJ}) \in \{0, 1\}^J$.

Error models

- Given true data, the error process determines what we observe.
- We differentiate two components:
  1. **Location model:** Which items are in error?
  2. **Substitution model:** Given that there's an error at the (i, j) location, how does Y_{ij} is generated from X_{ij}?
- Let $E_{ij} = 1$ if there's an error at the (i, j) location, and 0 otherwise. We define the *error mask* $\mathbf{E}_i = (E_{i1}, ..., E_{iJ}) \in \{0, 1\}^J$.
- The **location model** is the distribution of $\mathbf{E}_i$.
- The **substitution model** is the conditional distribution of $\mathbf{Y}_i$ given $\mathbf{E}_i$ and $\mathbf{X}_i$
- (This separation allows to specify a priori which values we *know* are correct or incorrect.)

Specifying the Error Model

Location: Independent Errors Model

$$E_{ij}|\epsilon_j \overset{indep}{\sim} \text{Bernoulli}(\epsilon_j)$$
$$\epsilon_j \overset{iid}{\sim} \text{Beta}(a_\epsilon, b_\epsilon)$$

- Error locations are independent.
- Each item has its own error rate, ϵ_j.
- Other specifications possible.

Specifying the Error Model

Location: Independent Errors Model

$$E_{ij}|\epsilon_j \overset{indep}{\sim} \text{Bernoulli}(\epsilon_j)$$
$$\epsilon_j \overset{iid}{\sim} \text{Beta}(a_\epsilon, b_\epsilon)$$

- Error locations are independent.
- Each item has its own error rate, ϵ_j.
- Other specifications possible.

Substitution: Uniform Substitution Model

$$Y_{ij}|X_{ij}, E_{ij} \sim \begin{cases} \delta_{X_{ij}} & \text{if } E_{ij} = 0 \\ \text{Uniform}\left(\{1, ..., L_j\} \setminus \{X_{ij}\}\right) & \text{if } E_{ij} = 1 \end{cases}$$

"True Responses" Distribution

$$\mathbf{X}_i \sim F$$

- In principle it can be any distribution over $\mathcal{C} \setminus S$.
- In practice we need a flexible enough specification, able to capture the nuances of the multivariate structure.
- Challenges:
  - Sparsity (very high-dimensional tables with many zero-counts).
  - Model selection. We want high prediction power.
  - Handling of structural zeros!

We use the Nonparametric Truncated Latent Class Model from Manrique-Vallier and Reiter, 2013 (JCGS, to appear)

Non Parametric Truncated Latent Class Models

Truncated mixtures of discrete distributions:

$$\mathbf{x}_i|\boldsymbol{\lambda},\boldsymbol{\pi} \sim 1\{\mathbf{x}_i \notin S\} \sum_{k=1}^{\infty} \pi_k \prod_{j=1}^{J} \lambda_{jk(x_{ij})}$$

with $\boldsymbol{\pi} = (\pi_1, \pi_2, ...) \sim DP(\alpha)$, $\lambda_{jk} \overset{iid}{\sim} Dirichlet(\mathbf{1}_K)$, and $\alpha \sim Gamma(a_\alpha, b_\alpha)$.

- Very flexible models.
- Method by Manrique-Vallier and Reiter (2013) to obtain posterior parameter samples subject to truncated (to $\mathcal{C} \setminus S$) data support.
- Several advantages: Automatic overfitting control. Computationally tractable. High tolerance to sparsity. Capacity to handle large collections of structural zeros.

Test Application - Data Based Simulation

$J = 10$ variables from 5% public use microdata from 2000 U.S. census (NY)

Variable	Levels (L_j)	Variable	Levels (L_j)
Ownership of dwelling	3	Mortgage status	4
Age	9	Sex	2
Marital status	6	Race	5
Education	11	Employment	4
Work disability	3	Veteran Status	3

- Take $N = 953,076$ as a population. Compute statistics.
- Sub-sample $n = 1,000$, introduce errors, fix them, and try to estimate population quantities back.

Notes:

- Resulting contingency table has $2,566,080$ cells.
- $|S| = 2,317,030$ possible inconsistent responses. Originally specified as 60 pair-wise rules (e.g. veteran toddlers).
- Original data without inconsistencies.

Test Application - Introducing Errors

Contaminate the data using independent errors and uniform substitution,

$$Y_{ij}|X_{ij}, E_{ij} \sim \begin{cases} \delta_{X_{ij}} & \text{if } E_{ij} = 0 \\ \text{Uniform}\left(\{1, ..., L_j\} \setminus \{X_{ij}\}\right) & \text{if } E_{ij} = 1 \end{cases}$$

$$E_{ij} \overset{iid}{\sim} \text{Bernoulli}(\varepsilon)$$

- Try with different error rates $\varepsilon = 0.1, 0.3, 0.5$.
- Pretend that we only observe $\mathbf{Y}$.

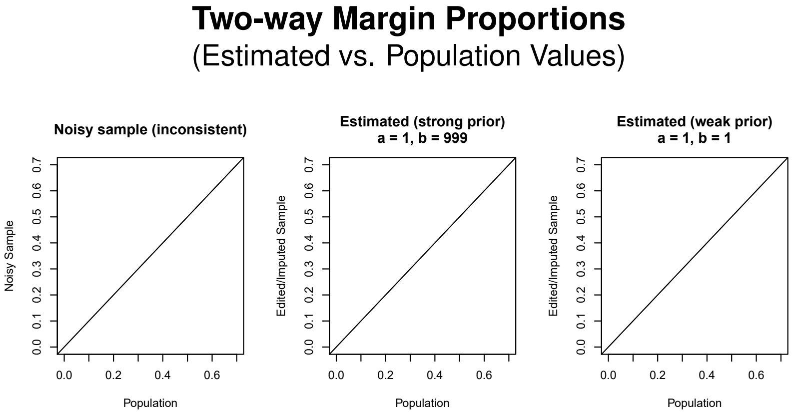
Prior Specification for Error Model

- We use the independent errors / uniform substitution model.
- Need to specify prior distribution for item error rates:

$$\epsilon_j \sim Beta(a_\epsilon, b_\epsilon)$$

- The method will always detect and correct *detectable* errors.
- The prior specification determines how much we trust what we observe:
  - a_ϵ / b_ϵ = Prior expected rate of error.
  - Large $a_\epsilon + b_\epsilon$ (relative to sample size) puts more weight on our beliefs than on the data.
  - Small $a_\epsilon + b_\epsilon$ puts more weight on data.
- For variables that we don't want to ever alter, we set $E_{ij} = 0$ a priori. This forces $Y_{ij} = X_{ij}$. (can have unintended consequences, though)

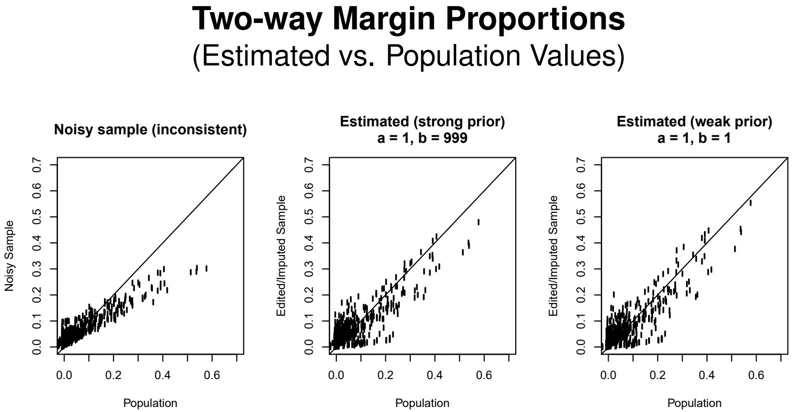
Results (1)- Two-Way margins (ε 0.1)



Simulation Parameters:

- $\varepsilon = 0.1, n = 1,000$
- Rows with errors = 626. Detectable errors = 306

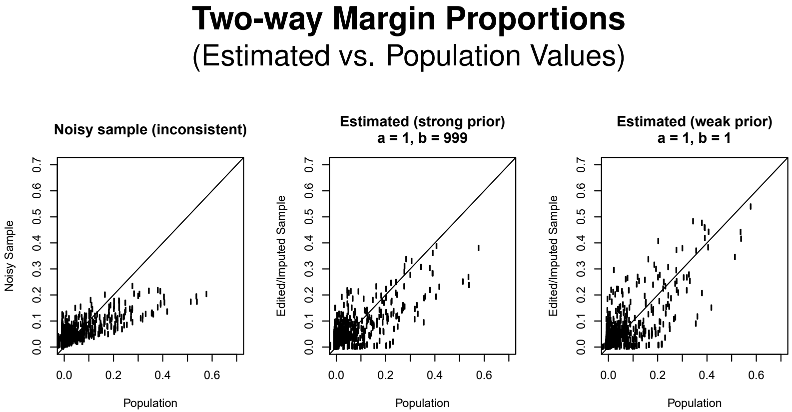
Results (2)- Two-Way margins (ε 0.3)



Simulation Parameters:

- $\varepsilon = 0.3, n = 1,000$
- Rows with errors = 980. Detectable errors = 685

Results (3)- Two-Way margins (ε 0.5)



Simulation Parameters:

- $\varepsilon = 0.5, n = 1,000$
- Rows with errors = 999. Detectable errors = 833

Concluding Remarks

- Full Bayesian model-based approach to edit-imputation.
- Integrates data generation with measurement error.
- Automatic over-fitting protection.
- Edit and imputation based on joint distribution. Respects data distribution.
- Does not require full analysis of consistency rules. Guaranteed to generate consistent imputations.
- Computationally feasible, but can be demanding in tough problems. (runtime example = 1.6 min)
- Prior specification matters:
  - Strong prior w/low error rate.
  - Weak prior.
- Open issue: Which values do we really want to change? (prior for ϵ_j and which E_{ij} set to 0 a priori)

The End

(Thanks!)

For details about truncated latent structure models:

http://mypage.iu.edu/~dmanriqu/papers/lcm_zeros.pdf

For multiple imputation see:

http://mypage.iu.edu/~dmanriqu/papers/LCM_Zeros_Imputation.pdf