Python, Cadastral Geocoding, and EMR Data

Amy Hughes*[12]
Sandi L. Pruitt†[23]
Tammy Leonard†[14]

* Presenter
†Supervising Author
[1]University of Texas Southwestern Department of Clinical Sciences
[2]University of Texas at Dallas School of Economics, Political, and Policy Sciences
[3]Harold C. Simmons Comprehensive Cancer Center
[4]University of Dallas Department of Economics

Contents

- Introduction
- Problem statement
- A Python & SAS Solution
- Results
- Next Steps

Introduction: Research Interests

- Population-Based Research Optimizing Screening Through Personalized Regimens (PROSPR)
- Cancer screening behaviors
  - Outreach RCTs
  - Observational studies
- Neighborhood effects
- Residential mobility

Introduction: Electronic Medical Record (EMR)

- EMR – an electronic copy of a paper medical chart
- HITECH Act (2009) – government mandate for EMRs and EMR systems
- Number of health benefits
- Offers new, more comprehensive data for observational research

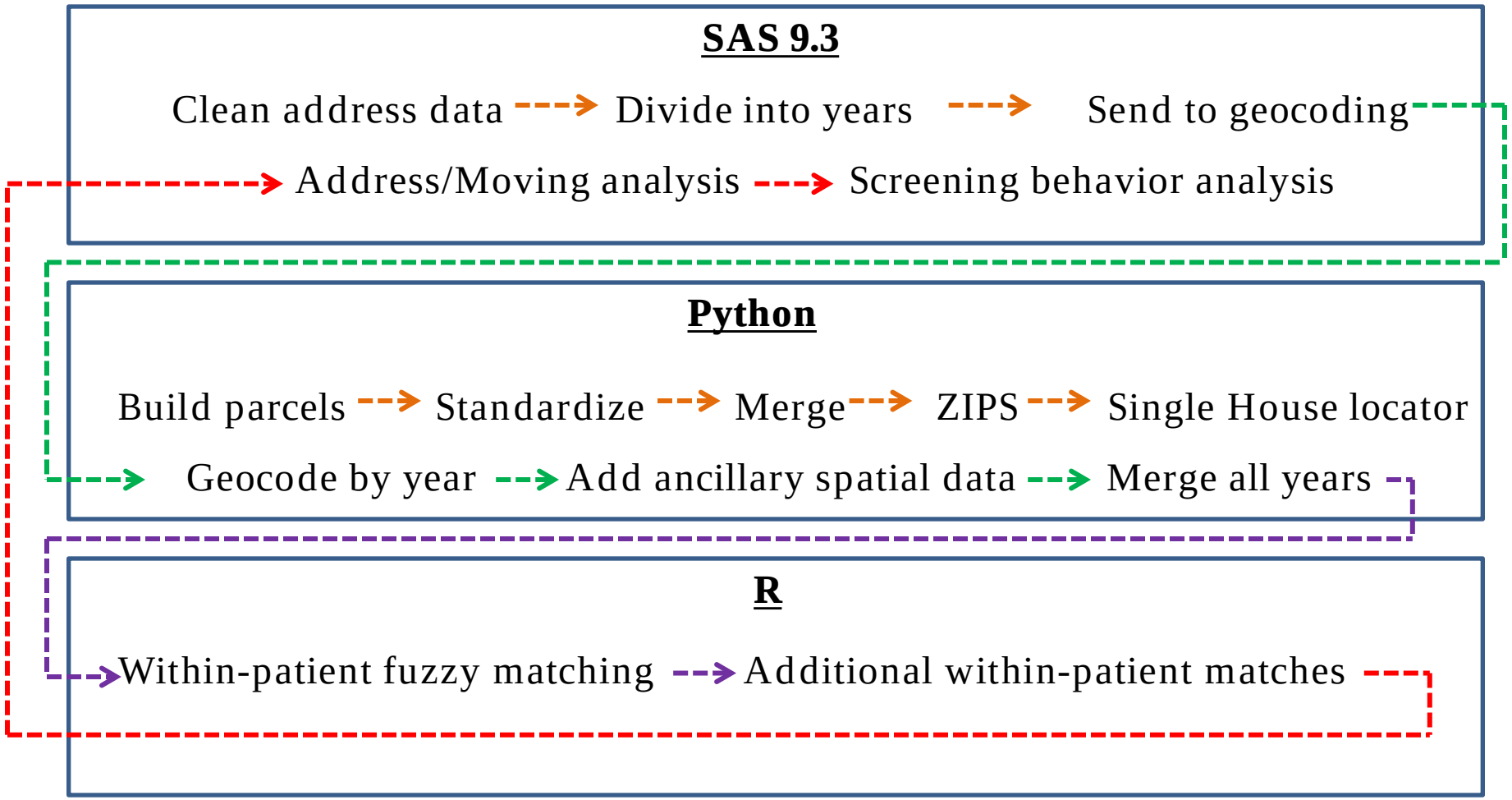
Introduction: PROSPR Data

- EMRs from county safety-net hospital during 2005-2012
- CRC screening exclusion restrictions
- Data enters the EMR when:
  - Patient visits any clinic within the hospital
  - A test or procedure is ordered
  - A diagnosis is given
- Records can be viewed as
  - An address history
  - A medical history

Our Problem

- This particular dataset:
  - Address cleaning of 275,000+ records
  - Cadastral geocoding across 9 counties and 7 years
  - Addition of ancillary data to medical record
- Future datasets:
  - Data instability
  - Interchangeability of data preparation method
  - Quick turn-around

Our Solution: Workflow



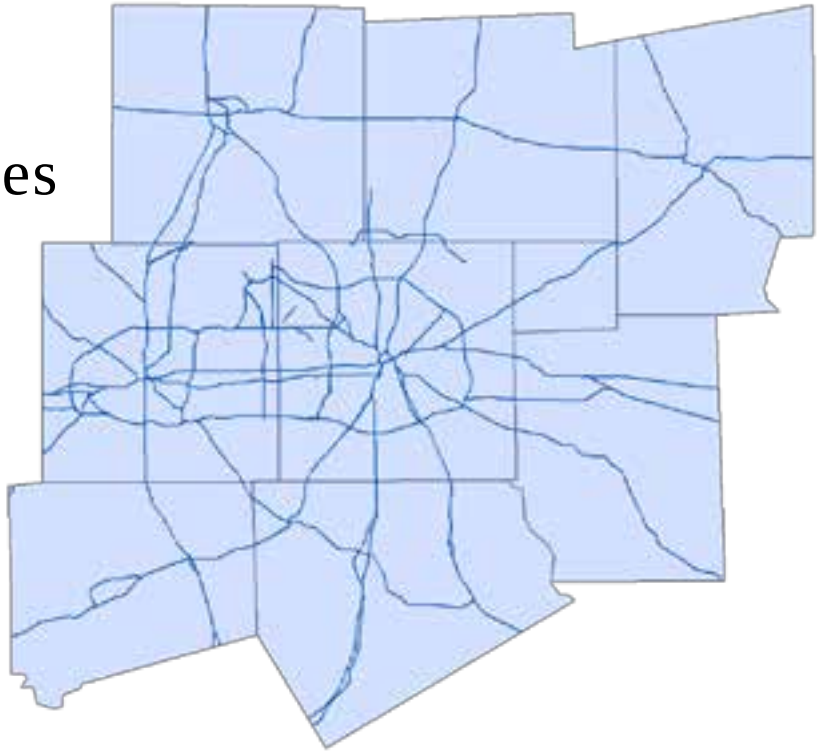
Our Solution: SAS I

1. Address validation:
   - Flag addresses that do not contain both a street number and a street name
   - Flag addresses corresponding to P.O. Boxes
   - Detect addresses spread across two fields and combine into one
2. Temporal filtering
   - Group addresses based on date stamp for cadastral geocoding

Our Solution: Python I

- Problem: patient addresses span multiple counties

3. Assemble parcel files
4. Standardize parcel addresses
5. Merge to cover entire metro area
6. Spatially merge in ZIP codes for cadastral zone geocoding



Our Solution: Python II

7. Cadastral zone geocoding
   - Patient address files (by year) are geocoded using parcels from the following year
   - Use ZIP codes as zones
     - ZIP codes are used to score matches, but not to limit the search for matches
     - Allows for correct disambiguation of the same street address in different counties
8. Merge results from all years

Our Solution: R

- Problem: inconsistent data entry
  - The same address for a patient is recorded with different spellings across clinic visits

9. Fuzzy matching within-patient
   - Unmatched addresses are compared to matched addresses **for the same patient** using Levenshtein string distance
   - If the distances is small enough, we "manually" match the address to circumvent the typo

Our Solution: SAS II

10. Unique address analysis
    - Use distance between geocoded points to determine unique addresses across years
    - Use within-group analysis in SAS to calculate unique address summary stats for each patient
11. Screening behavior analysis

Our Solution: Implementation

- Currently not a geoprocessing tool
- Scripts are run within SAS analysis file
  - Execution of Python and R scripts via batch to Windows command line
  - Entire process requires the click of a single button for reproducibility

Results

- Can quickly geocode many addresses to the parcel level across 9 counties and 7 years
- Creation of input-driven code enables:
  - Use of the same process for future datasets
  - Analysis of data at the cadastral level from multiple EMR sources

Results

- Benefits:
  - Account for changing structure of the urban area
  - Use parcel attributes in subsequent screening behavior analyses
  - Define our own “neighborhoods” for analyses
  - Reproducibility
  - Ease of implementation
  - Speed

Results

- Drawbacks:
  - Use of third-party software can be expensive
  - Loss of data due to more rigorous geocoding processes
    - Using dual-range locator, can geocode ~92% of all addresses (includes “messy” data)
    - Using single-house locator and fuzzy matching across years, can geocode ~76% of data (so far)

Next Steps

- Development of geoprocessing tool and/or toolbox
- Increase ability to correctly geocode difficult addresses
- Customization of zone geocoding behavior in single house locator through XML
- Eliminate the use of R by conducting fuzzy matching in SAS

Acknowledgments:

- Project funded by
  - CPRIT grant #PP100039
  - NCI grant #5U54CA163308
- Special thanks to Bennett Grinsfelder and Erica Cuate at UT Southwestern
- Data provided through PROSPR at UTSW
- ESRI: http://www.esri.com
- SAS: www.sas.com
- R: www.r-project.org