BAYESIAN RECORD LINKAGE

RACHEL ANDERSON

1. Methods

1.1. **Mixture Models.** Mixture models are useful for implementing the Fellegi-Sunter approach. Following Shalizi (ref here):

We say that the distribution f is a mixture of K component distributions $f_1, \ldots, f_K$ if

$$f(x) = \sum_{k=1}^{K} \lambda_k f_k(x; \theta_k)$$

where λ_k are the mixing weights, such that $\lambda_k > 0,\ \sum_k \lambda_k = 1$. This means that the data can generated to the following procedure:

$$\begin{aligned} Z &\sim \text{Multinomial}(\lambda_1, \ldots, \lambda_K) \\ X|Z &\sim f_Z \end{aligned}$$

where Z is a discrete random variable that says which component X is drawn from.

This is useful for record linkage using the FS approach because we assume there are two latent populations corresponding to matches (M) and non-matches (U), that are represented in the population of comparisons with proportions p_M and $p_U = 1 - p_M$ respectively.

Given $(i, j) \in M$ or $(i, j) \in U$, the comparison vector between two files i and j is

$$\gamma(i, j) \sim f_k, \quad k \in \{M, U\}$$

Assuming independent samples, the log likelihood for a generic mixture model for observations $(x_1, \ldots, x_n)$ is:

$$\ell(\theta) = \sum_{i=1}^{n} \log f(x_i, \theta) \tag{1}$$

$$= \sum_{i=1}^{n} \log \sum_{k=1}^{K} \lambda_k f(x_i; \theta_k) \tag{2}$$

The overall parameter vector of the model is thus $\theta = (\lambda_1, \ldots, \lambda_{K-1}, \theta_1, \theta_2, \ldots, \theta_K)$.

In record linkage, $x_i \overset{i.i.d.}{\sim} X$ and $y_j \overset{i.i.d.}{\sim} Y$, so $\gamma(x_i, y_j) \perp\!\!\!\perp \gamma(x_{i'}, y_j)$ for all $j, i' \neq i$, yet, $\gamma(x_i, y_j)$ may not be independent from $\gamma(x_i, y_{j'})$. This gives the likelihood:

$$\ell(\theta) \tag{3}$$

1.2. **Mixture Model Estimation.** As shown in Shalizi (ref), maximizing the likelihood for a mixture model is like doing a weighted likelihood maximization, where the weight of each observation x_i depends on the cluster. This is seen by taking the derivative of (2) with respect to one parameter θ_j,

$$\begin{aligned}\frac{\partial \ell}{\partial \theta_j} &= \sum_{i=1}^{n} \frac{1}{\sum_{k=1}^{K} \lambda_k f(x_i;\theta_k)} \lambda_j \frac{\partial f(x_i;\theta_j)}{\partial \theta_j} \\ &= \sum_{i=1}^{n} \underbrace{\frac{\lambda_j f(x_i;\theta_j)}{\sum_{k=1}^{K} \lambda_k f(x_i;\theta_k)}}_{w_{ij}} \frac{\partial \log f(x_i;\theta_j)}{\partial \theta_j}\end{aligned}$$

Furthermore, the weight has a convenient interpretation:

$$w_{ij} = \frac{\lambda_j f(x_i;\theta_j)}{\sum_{k=1}^{K} \lambda_k f(x_i;\theta_k)} = \frac{P(Z=j, X=x_i)}{P(X=x_i)} = P(Z=j|X=x_i)$$

So if we try to estimate the mixture model, we're doing weighted maximum likelihood, with weights given by the posterior cluster probabilities (which depend on parameters $\lambda_1, \ldots, \lambda_K$ that we are trying to estimate). The EM algorithm makes estimation possible:

2. Datasets

The techniques are tested first using a synthetic data generator developed by Christen and Pudjijono (2009) and Christen and Vatsalan (2013), and then with two real datasets from Enamorado (2018) and Enamorado, Fifield and Imai (2018).

(1) In the first empirical application I merge two datasets on local-level candidates for the 2012 and 2016 municipal elections in Brazil. Each dataset contains more than 450,000 observations with a perfectly-recorded unique identifier, the Brazilian individual taxpayer registration identification number (called the *Cadastro de Pessoas Físicas*). All other common identifiers are manually entered into the database, so they may contain errors.

(2) In the second application, I merge the 2016 American National Election Study(ANES) with a nationwide voter file containing over 160 million voter records.

3. Bipartite Matching Problem

Two sets: $X = \{x_1, \ldots, x_{n1}\}$ and $Y = \{y_1, \ldots, y_{n2}\}$. The goal is to find an assignment of items so that every item in X is matched to exactly one item in Y and no two items share the same match. An assignment corresponds to a permutation π where π is a one-to-one mapping (check?) $\{1, \ldots, n_1\} \to \{1, \ldots, n_2\}$ mapping each item in X to its match in Y. We define $\pi(i) = j$ to denote the index of a match $y_{\pi(i)} = y_j$ for an item x_i, and $\pi^{-1}(j) = i$ to denote the reverse (if it exists).

Uncertainty over assignments expressed as:

$$P(\pi|\theta) = \frac{1}{Z(\theta)} \exp(-E(\pi, \theta))$$