COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

STAT 4224/5224

Bayesian Statistics

Dobrin Marchev

Overview

- Latent class models are used to identify unobservable or latent subgroups within a population. That is, they are unsupervised learning type of models.
- They are also a special case of a multivariate finite mixture models, where the response variables are categorical.
- Each observation *j* is assumed to belong to one of *C* classes, represented by a latent class indicator $Z_j \in \{1, \dots, C\}$.
- The marginal response probabilities are:

$$p(\boldsymbol{y}_j) = \sum_{c=1}^{C} \alpha_c \prod_{i=1}^{I} p(y_{ji} | Z_j = c)$$

where $\boldsymbol{y}_j$ is the response vector $(y_{j1}, \dots y_{jI})$ for individual *j* on a set of *I* items, Z_j is the class the individual belongs to and α_c is the probability of being in class *c*.

Model

- We will assume that $y_{ji} \sim Bernoulli(p_{ci})$, where p_{ci} is the probability that an individual in class c prefers item i.
- The class indicator Z_j can be viewed as a discrete latent variable.
- The model is:

$$\boldsymbol{Y}_j | Z_j = c \sim Bernoulli(\boldsymbol{p}_C)$$
$$Z_j \sim Discrete(\boldsymbol{\alpha})$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_C)$ with $\sum_{c=1}^{C} \alpha_c = 1$ and $\boldsymbol{p}_C$ is the class-specific parameter vector $(p_{c1}, \ldots, p_{cI})$.

Note: Stan does not support direct sampling of discrete parameters. Instead, models that involve discrete parameters can be coded by marginalizing out the discrete parameters.

Stan Code

log_mix

For models with two classes/mixture components, Stan uses log_mix for easy implementation: it takes a probability parameter and two log densities for the two mixture components as input.

For example, to express that y is sampled from a mixture of two Bernoulli distributions with parameters p_1 and p_2 and with a known marginal probability of being in the first class equal to .3, we would write

log_mix(.3, bernoulli_lpmf(y | p_1), bernoulli_lpmf(y | p_2).

Stan Code

log_sum_exp

log_sum_exp is a more general version of log_mix that handles situations where there are more than two mixture components. It takes two arguments *a* and *b* (or more if there are more than two mixture components), and then takes the log of the sum of the exponentials (i.e., log_sum_exp(a,b) = log(exp(a) + exp(b)). It can be viewed as a summation operation on the log scale.).

To express that y is sampled from a mixture of two Bernoulli distributions with parameters p1 and p2 and with a known marginal probability of being in the first class equal to .3, we would write

log_sum_exp(log(.3) + bernoulli_lpmf(y | p_1), log(.7) + bernoulli_lpmf(y | p_2))

Prediction of latent class membership

one common research question with latent class analysis is what class a unit belongs to, that is, we are interested in the posterior probability $P(Z_j = c|y)$

The Bayesian way of predicting class membership does not treat parameter estimates as known and hence takes all the uncertainty regarding parameter estimates into account, as shown below:

$$P(Z_j = c|y) = \int P(Z_j = c|y_j, \theta) P(\theta|y) d\theta$$

$$\approx \frac{1}{S} \sum_{s=1}^{S} P(Z_j = c|y_j, \theta^{(s)})$$