An Introduction to Latent Feature Model

Jiguang Li

Center for Applied Artificial Intelligence

Oct 28th, 2021

Motivation

Assume we have N observations $\{x_1, \cdots, x_N\}$, such that each $x_i \in R^d$. In finite mixture model, we assume each observation belongs to a **single** latent class c_i:

- Let the mixture weights be θ and suppose there are K classes, the data likelihood is $P(X|\theta) = \prod_{i=1}^{N} \sum_{k=1}^{K} P(x_i|c_i = k)\theta_k$

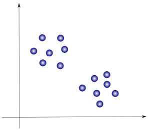

Figure: Two Cluster Mixture

Motivation

Assume we have N observations $\{x_1, \cdots, x_N\}$, such that each $x_i \in R^d$. In finite mixture model, we assume each observation belongs to a **single** latent class c_i:

- Let the mixture weights be θ and suppose there are K classes, the data likelihood is $P(X|\theta) = \prod_{i=1}^{N} \sum_{k=1}^{K} P(x_i|c_i = k)\theta_k$

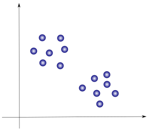

Figure: Two Cluster Mixture

Problem: what if each observation x_i can be generated by more than one latent class?

Finite Feature Model: setup

Suppose there are N objects and K features, and the features are generated independently.

- Each object i has some latent feature values, let $F = [f_1^T, \cdots, f_n^T]^T$

Finite Feature Model: setup

Suppose there are N objects and K features, and the features are generated independently.

- Each object i has some latent feature values, let $F = [f_1^T, \cdots, f_n^T]^T$
- Data is generated from these latent features: $P(X|F)$

Finite Feature Model: setup

Suppose there are N objects and K features, and the features are generated independently.

- Each object i has some latent feature values, let $F = [f_1^T, \cdots, f_n^T]^T$
- Data is generated from these latent features: $P(X|F)$
- let Z be an N by K matrix such that $Z_{ik} = 1$ if object i possess feature k.

Finite Feature Model: setup

Suppose there are N objects and K features, and the features are generated independently.

- Each object i has some latent feature values, let $F = [f_1^T, \cdots, f_n^T]^T$
- Data is generated from these latent features: $P(X|F)$
- let Z be an N by K matrix such that $Z_{ik} = 1$ if object i possess feature k.
- $F = Z \otimes V$, where V is the value of each feature for each object.

Finite Feature Model: Prior for Z

In Bayesian Inference, we are interested in the posterior $P(F|X) \propto P(X|F)P(F)$, where $F = Z \otimes V$.

Finite Feature Model: Prior for Z

In Bayesian Inference, we are interested in the posterior $P(F|X) \propto P(X|F)P(F)$, where $F = Z \otimes V$.
How to specify the prior for Z ?

Finite Feature Model: Prior for Z

In Bayesian Inference, we are interested in the posterior $P(F|X) \propto P(X|F)P(F)$, where $F = Z \otimes V$.
How to specify the prior for Z ?

- Assume each object possess feature k with probability π_k.

Finite Feature Model: Prior for Z

In Bayesian Inference, we are interested in the posterior $P(F|X) \propto P(X|F)P(F)$, where $F = Z \otimes V$.
How to specify the prior for Z ?

- Assume each object possess feature k with probability π_k.
- $P(Z|\pi) = \prod_{k=1}^{K} \prod_{i=1}^{N} P(Z_{ik}|\pi_k) = \prod_{k=1}^{K} \pi_k^{m_k}(1-\pi_k)^{N-m_k}$

Finite Feature Model: Prior for Z

In Bayesian Inference, we are interested in the posterior $P(F|X) \propto P(X|F)P(F)$, where $F = Z \otimes V$.
How to specify the prior for Z ?

- Assume each object possess feature k with probability π_k.
- $P(Z|\pi) = \prod_{k=1}^{K} \prod_{i=1}^{N} P(Z_{ik}|\pi_k) = \prod_{k=1}^{K} \pi_k^{m_k}(1-\pi_k)^{N-m_k}$
- Usually, we assume $\pi_k \sim Beta(r, s)$, such that
  - $P(\pi_k) = \frac{\pi_k^{r-1}(1-\pi_k)^{s-1}}{B(r,s)}$

Finite Feature Model: Prior for Z

In Bayesian Inference, we are interested in the posterior $P(F|X) \propto P(X|F)P(F)$, where $F = Z \otimes V$.
How to specify the prior for Z ?

- Assume each object possess feature k with probability π_k.
- $P(Z|\pi) = \prod_{k=1}^{K} \prod_{i=1}^{N} P(Z_{ik}|\pi_k) = \prod_{k=1}^{K} \pi_k^{m_k}(1-\pi_k)^{N-m_k}$
- Usually, we assume $\pi_k \sim Beta(r, s)$, such that
  - $P(\pi_k) = \frac{\pi_k^{r-1}(1-\pi_k)^{s-1}}{B(r,s)}$
  - $B(r, s) = \int_0^1 \pi_k^{r-1}(1 - \pi_k)^{s-1} d\pi_k = \frac{\Gamma(r)\Gamma(s)}{\Gamma(r+s)}$

Finite Feature Model: Prior for Z

In Bayesian Inference, we are interested in the posterior $P(F|X) \propto P(X|F)P(F)$, where $F = Z \otimes V$.
How to specify the prior for Z ?

- Assume each object possess feature k with probability π_k.
- $P(Z|\pi) = \prod_{k=1}^{K} \prod_{i=1}^{N} P(Z_{ik}|\pi_k) = \prod_{k=1}^{K} \pi_k^{m_k}(1-\pi_k)^{N-m_k}$
- Usually, we assume $\pi_k \sim Beta(r, s)$, such that
  - $P(\pi_k) = \frac{\pi_k^{r-1}(1-\pi_k)^{s-1}}{B(r,s)}$
  - $B(r, s) = \int_0^1 \pi_k^{r-1}(1 - \pi_k)^{s-1} d\pi_k = \frac{\Gamma(r)\Gamma(s)}{\Gamma(r+s)}$
  - $r = \frac{\alpha}{K}, s = 1 \implies B(r, s) = \frac{\Gamma(\frac{\alpha}{K})}{\Gamma(1+\frac{\alpha}{K})} = \frac{K}{\alpha}$
    $(\Gamma(X) = (X - 1)\Gamma(X - 1))$

Finite Feature Model: Distribution of Z

Recall we have defined prior for $\pi_k \sim Beta(\frac{\alpha}{K}, 1)$ and $P(Z|\pi)$:

$$
\begin{aligned}
P(Z) &= \prod_{k=1}^{K} \int (\prod_{i=1}^{N} P(Z_{ik}|\pi_k)) P(\pi_k) d\pi_k \\
&= \prod_{k=1}^{K} \int \prod_{i=1}^{N} \pi_k^{m_k} (1-\pi_k)^{N-m_k} P(\pi_k) d\pi_k \\
&= \prod_{k=1}^{K} \frac{\int \prod_{i=1}^{N} \pi_k^{m_k} (1-\pi_k^{N-mk}) \pi_k^{\alpha/K-1}}{B(\frac{\alpha}{K}, 1)} d\pi_k \qquad (1) \\
&= \prod_{k=1}^{K} \frac{B(m_k + \alpha/K, N - m_k + 1)}{B(\alpha/K, 1)} \\
&= \prod_{k=1}^{K} \frac{\alpha/K \cdot \Gamma(m_k + \alpha/K)\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \alpha/K)}
\end{aligned}
$$

Finite Feature Model: α controls Sparsity

The expectation of the number of non-zero entries in the matrix has an upper bound that is independent of K!

$$\begin{aligned} E[\sum_{i,k} Z_{i,k}] = E[\mathbb{1}^T Z \mathbb{1}] &= KE[\mathbb{1}^T Z_1] \\ &= K \sum_{i=1}^{N} \int_0^1 \pi_k P(\pi_k) d\pi_k \\ &= KN \frac{\alpha/k}{1+\alpha/K} \\ &= \frac{N\alpha}{1+\alpha/K} \leq N\alpha \end{aligned} \tag{2}$$

Infinite Feature Model

Motivation: what if we don't know the number of latent features? Can be define a distribution of binary matrix Z which has N rows but infinite columns?

Infinite Feature Model

Motivation: what if we don't know the number of latent features? Can be define a distribution of binary matrix Z which has N rows but infinite columns?

Naive Approach: recall in the finite case, we have shown

$$P(Z) = \prod_{k=1}^{K} \frac{\alpha/K \cdot \Gamma(m_k + \alpha/K)\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \alpha/K)}$$

What if we let $K \to \infty$?

Equivalence Classes

- lof$(\cdot)$: map binary matrices to left-ordered binary matrices.

Equivalence Classes

- $\text{lof}(\cdot)$: map binary matrices to left-ordered binary matrices.
- $\text{lof}(Z)$ is obtained by reordering the columns of the binary matrix Z from left to right by the magnitude of the binary number expressed by the column:
- $[Z]$: set of binary matrices that are lof-equivalent to Z

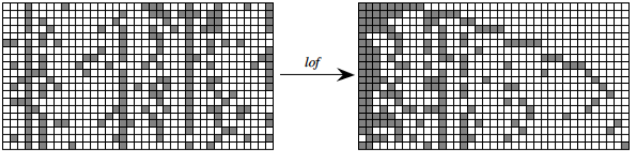

Figure: Visualization of lof operation

Equivalence Classes

- lof$(\cdot)$: map binary matrices to left-ordered binary matrices.
- lof(Z) is obtained by reordering the columns of the binary matrix Z from left to right by the magnitude of the binary number expressed by the column:
- $[Z]$: set of binary matrices that are lof-equivalent to Z

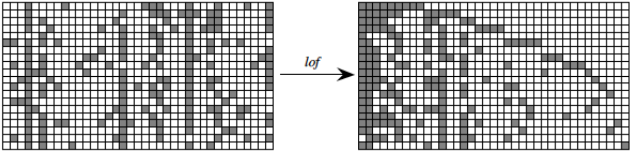

Figure: Visualization of lof operation

More on Equivalence Classes

- The **history** of feature k at object i is $(Z_{1,k}, \cdots, Z_{i-1,k})$

More on Equivalence Classes

- The **history** of feature k at object i is $(Z_{1,k}, \cdots, Z_{i-1,k})$
- k_h: number of features possessing the history h
- K_0: number of features for which $m_k = 0$
- $K_+ = \sum_{h=1}^{2^N-1} k_h$: number of features for which $m_k > 0$, note $K = K_0 + K_+$

More on Equivalence Classes

- The **history** of feature k at object i is $(Z_{1,k}, \cdots, Z_{i-1,k})$
- k_h: number of features possessing the history h
- K_0: number of features for which $m_k = 0$
- $K_+ = \sum_{h=1}^{2^N-1} k_h$: number of features for which $m_k > 0$, note $K = K_0 + K_+$
- Cardinality of $|[Z]|$:

$$|[Z]| = \frac{k!}{\prod_{h=0}^{2^{N-1}} k_n!}$$

Let's make $K \to \infty$

$$
\begin{aligned}
P([Z]) &= \sum_{z \in [Z]} P(Z) \\
&= \frac{K!}{\prod_{h=0}^{2^{N-1}} k_n!} \prod_{k=1}^{K} \frac{\alpha/K \cdot \Gamma(m_k + \alpha/K)\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \alpha/K)}
\end{aligned} \tag{3}
$$

Let's make $K \to \infty$

$$P([Z]) = \sum_{z\in[Z]} P(Z) = \frac{K!}{\prod_{h=0}^{2^{N-1}} k_n!} \prod_{k=1}^{K} \frac{\alpha/K \cdot \Gamma(m_k + \alpha/K)\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \alpha/K)} \tag{3}$$

By 2-page of algebraic manipulations, mathematicians can show

$$\lim_{k\to\infty} P([Z]) = \frac{\alpha^{K_+}}{\prod_{h=1}^{2^N-1} K_h!} e^{-\alpha H_N} \prod_{k=1}^{K^+} \frac{(N - m_k)!(m_k - 1)!}{N!}$$

, where $H_N = \sum_{j=1}^{N} \frac{1}{j}$

Indian Buffet Process (IBP)

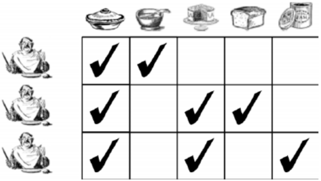

Figure: Indian Buffet Process

Indian Buffet Process (IBP)

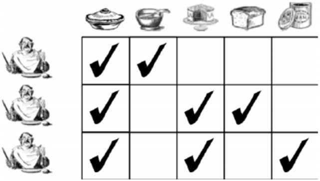

Figure: Indian Buffet Process

- The first customer enters an Indian Buffet with infinitely many dishes.

Indian Buffet Process (IBP)

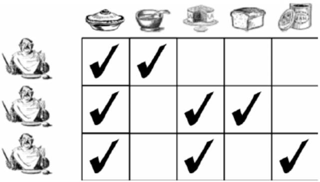

Figure: Indian Buffet Process

- The first customer enters an Indian Buffet with infinitely many dishes.
- The first customer sample the first *Poisson*(α) dishes.

Indian Buffet Process (IBP)

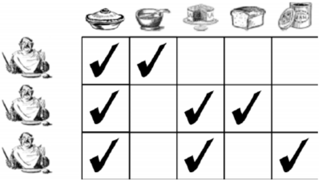

Figure: Indian Buffet Process

- The first customer enters an Indian Buffet with infinitely many dishes.
- The first customer sample the first $Poisson(\alpha)$ dishes.
- The nth customer helps himself to each dish with probability $\frac{m_k}{n}$, where m_k is the number of times dish k has been sampled

Indian Buffet Process (IBP)

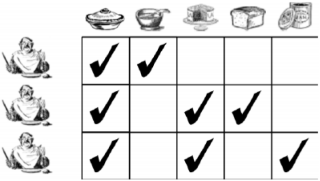

Figure: Indian Buffet Process

- The first customer enters an Indian Buffet with infinitely many dishes.
- The first customer sample the first $Poisson(\alpha)$ dishes.
- The nth customer helps himself to each dish with probability $\frac{m_k}{n}$, where m_k is the number of times dish k has been sampled
- The nth customer tries $poisson(\frac{\alpha}{n})$ dishes.

Research Ideas

Suppose have item response matrix X with size N by K, where N is the number of students, K is number of items, and X_{ik} represent whether student i answer question k correctly.

Research Ideas

Suppose have item response matrix X with size N by K, where N is the number of students, K is number of items, and X_{ik} represent whether student i answer question k correctly.

- **Partially Compensatory MIRT**:
  - Suppose this exam tests M latent abilities, hence students' latent traits $\theta_i \in \mathbb{R}^M$.
  - $\alpha_k \in \mathbb{R}^M$: discrimination term, $d_k \in \mathbb{R}^M$: difficulty term.

Research Ideas

Suppose have item response matrix X with size N by K, where N is the number of students, K is number of items, and X_{ik} represent whether student i answer question k correctly.

- **Partially Compensatory MIRT**:
  - Suppose this exam tests M latent abilities, hence students' latent traits $\theta_i \in \mathbb{R}^M$.
  - $\alpha_k \in \mathbb{R}^M$: discrimination term, $d_k \in \mathbb{R}^M$: difficulty term.
  - $P(X_{ik} = 1) = \prod_{m=1}^{M} \frac{1}{1+\exp\left(-\alpha_k(\theta_i - d_m)\right)}$

Research Ideas

Suppose have item response matrix X with size N by K, where N is the number of students, K is number of items, and X_{ik} represent whether student i answer question k correctly.

- **Partially Compensatory MIRT**:
  - Suppose this exam tests M latent abilities, hence students' latent traits $\theta_i \in \mathbb{R}^M$.
  - $\alpha_k \in \mathbb{R}^M$: discrimination term, $d_k \in \mathbb{R}^M$: difficulty term.
  - $P(X_{ik} = 1) = \prod_{m=1}^{M} \frac{1}{1+\exp(-\alpha_k(\theta_i - d_m))}$
- **Current Approach**: let $c_{km} \sim Beta(2,2)$, $P(X_{ik} = 1) = \prod_{m=1}^{M} (\frac{1}{1+\exp(-\alpha_k(\theta_i - d_m))})^{c_{km}}$

Research Ideas

Suppose have item response matrix X with size N by K, where N is the number of students, K is number of items, and X_{ik} represent whether student i answer question k correctly.

- **Partially Compensatory MIRT**:
  - Suppose this exam tests M latent abilities, hence students' latent traits $\theta_i \in \mathbb{R}^M$.
  - $\alpha_k \in \mathbb{R}^M$: discrimination term, $d_k \in \mathbb{R}^M$: difficulty term.
  - $P(X_{ik} = 1) = \prod_{m=1}^{M} \frac{1}{1+\exp(-\alpha_k(\theta_i - d_m))}$
- **Current Approach**: let $c_{km} \sim Beta(2,2)$, $P(X_{ik} = 1) = \prod_{m=1}^{M} (\frac{1}{1+\exp(-\alpha_k(\theta_i - d_m))})^{c_{km}}$

Research Ideas (cont'd)

We don't know the number of latent abilities, can we apply IDB prior to each column of the item response matrix X?

- **Advantages**:
  - Flexible, no need for cross-validation.

Research Ideas (cont'd)

We don't know the number of latent abilities, can we apply IDB prior to each column of the item response matrix X?

- **Advantages**:
  - Flexible, no need for cross-validation.
  - Allow the number of latent factors to grow over years.

Research Ideas (cont'd)

We don't know the number of latent abilities, can we apply IDB prior to each column of the item response matrix X?

- **Advantages**:
  - Flexible, no need for cross-validation.
  - Allow the number of latent factors to grow over years.
  - Incorporate education expertise? Fix some entries of latent matrix Z?

Research Ideas (cont'd)

We don't know the number of latent abilities, can we apply IDB prior to each column of the item response matrix X?

- **Advantages**:
  - Flexible, no need for cross-validation.
  - Allow the number of latent factors to grow over years.
  - Incorporate education expertise? Fix some entries of latent matrix Z?
  - The learned Z matrix can be treated as a naive representation of the latent structure of knowledges.

Research Ideas (cont'd)

We don't know the number of latent abilities, can we apply IDB prior to each column of the item response matrix X?

- **Advantages**:
  - Flexible, no need for cross-validation.
  - Allow the number of latent factors to grow over years.
  - Incorporate education expertise? Fix some entries of latent matrix Z?
  - The learned Z matrix can be treated as a naive representation of the latent structure of knowledges.
- **Drawbacks**:
  - High-dimensional?

Research Ideas (cont'd)

We don't know the number of latent abilities, can we apply IDB prior to each column of the item response matrix X?

- **Advantages**:
  - Flexible, no need for cross-validation.
  - Allow the number of latent factors to grow over years.
  - Incorporate education expertise? Fix some entries of latent matrix Z?
  - The learned Z matrix can be treated as a naive representation of the latent structure of knowledges.
- **Drawbacks**:
  - High-dimensional?
  - Requires too many items?

Research Ideas (cont'd)

We don't know the number of latent abilities, can we apply IDB prior to each column of the item response matrix X?

- **Advantages**:
  - Flexible, no need for cross-validation.
  - Allow the number of latent factors to grow over years.
  - Incorporate education expertise? Fix some entries of latent matrix Z?
  - The learned Z matrix can be treated as a naive representation of the latent structure of knowledges.
- **Drawbacks**:
  - High-dimensional?
  - Requires too many items?
  - What about Hierarchical structure?

Reference

- Thomas L. Griffiths, Zoubin Ghahramani (2011). The Indian Buffet Process: An Introduction and Review.
- Zoubin Ghahramani, Thomas L. Griffiths, Peter Sollich (2006). Bayesian nonparametric latent feature models