The Korean Journal of Applied Statistics (2015) DOI: http://dx.doi.org/10.5351/KJAS.2015.28.2.251
28(2), 251–267

Introduction to the Indian Buffet Process: Theory and Applications

Youngseon Lee[a] · Kyoungjae Lee[a,1] · Kwangmin Lee[a] · Jaeyong Lee[a] · Jinwook Seo[b]

[a]Department of Statistics, Seoul National University
[b]Department of Computer Science and Engineering, Seoul National University

(Received March 16, 2015; Revised March 30, 2015; Accepted March 30, 2015)

Abstract

The Indian Buffet Process is a stochastic process on equivalence classes of binary matrices having finite rows and infinite columns. The Indian Buffet Process can be imposed as the prior distribution on the binary matrix in an infinite feature model. We describe the derivation of the Indian buffet process from a finite feature model, and briefly explain the relation between the Indian buffet process and the beta process. Using a Gaussian linear model, we describe three algorithms: Gibbs sampling algorithm, Stick-breaking algorithm and variational method, with application for finding features in image data. We also illustrate the use of the Indian Buffet Process in various type of analysis such as dyadic data analysis, network data analysis and independent component analysis.

Keywords: Indian buffet process, latent feature model, Gaussian linear model, Gibbs sampling, stick-breaking sampling, variational method.

1. 서론

2000년대 초반부터 일단의 컴퓨터 공학자들은 비모수 베이지안 모형이 기계학습(machine learning)분야에 사용될 수 있다는 사실에 주목하였고, 디리크레 프로세스(Dirichlet process; Ferguson, 1973), 피트만-요 프로세스(Pitman-Yor process; Pitman과 Yor, 1997), 종추출모형(species sampling model; Pitman, 1996) 등을 이용한 혼합모형을 문서 데이터 마이닝(text mining), 이미지 데이터 마이닝(image mining) 등의 문제에 적용하였다. 이들의 연구는 기계학습 연구자들과 비모수 베이지안 연구자들을 연결시켜주는 역할을 하였다. 즉, 비모수 베이지안 모형이 기계학습 연구자들에게 소개되었고, 기계학습 연구자들이 주로 다루는 응용 문제인 문서 자료와 이미지 자료의 분석과 관련된 문제들이 베이지안 연구자들에게 소개되는 효과를 가져 오게 되었다.

그들은 기존에 존재하는 비모수 베이지안 모형을 새로운 접하게 된 문제들에 적용하는 것에 그치지 않고, 문제에서 요구되는 것에 따라 다양한 베이지안 모형과 방법론을 개발하였다. 대표적인 모형이 바로 인도부페 프로세스(Indian Buffet Process; Griffiths와 Gharahmani, 2006)이다. 인도부페 프로세스는

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 2011-0030811).
[1]Corresponding author: Department of Statistics, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 151-747, Korea. E-mail: leekjstat@gmail.com

2006년에 Griffiths와 Ghahramani에 의해 처음 제안된 통계모형이다. 디리크레 프로세스나 종추출모형에 기반한 잠재변수모형은 한 개의 관측치와 한 개의 특성치가 대응되는데 반해, 인도부페 프로세스를 이용한 모형은 한 개의 관측치에 복수의 특성치가 대응될 수 있다는 특징을 가지게 된다. 인도부페 프로세스로 대표되는 잠재특성모형에 대한 이론 및 응용 연구는 현재 베이지안 연구자들 사이에 가장 뜨거운 주제 중 하나이다.

잠재변수를 이용한 모형화는, 베이지안 모형에서의 핵심적인 기법 중 하나이다. 디리크레 프로세스 혼합모형(Dirichlet process mixture model)은 잠재변수의 분포를 디리크레 프로세스에서 추출된 랜덤 확률분포로 모형화하는 방법이고, 디리크레 프로세스 혼합모형에서 확장된 종추출 혼합 모형(species sampling mixture model)은 잠재변수의 분포를 종추출모형에서 추출한 랜덤 확률분포로 모형화한 것이다. 디리크레 프로세스 혼합모형과 종추출 혼합모형은 주로 군집화의 목적으로 사용되는데, 이러한 방법들은 군집의 개수를 사전에 고정할 필요가 없으며 모형을 적합하는 과정에서 군집의 개수가 자동으로 추정된다는 장점이 있다. 이 때, 자료의 군집화는 관측치 한 개 마다 하나의 잠재변수(latent variable) 혹은 잠재특성(latent feature)을 대응시켜, 잠재변수의 값이 동일한 관측치들은 동일한 군집에 속하도록 함으로써 이루어진다. 디리크레 프로세스 혼합모형은, 통계학뿐만 아니라 다양한 분야에 적용되어 엄청난 성공을 거두게 되었고, 따라서 비모수 베이지안 모형의 대표적인 모형으로 자리 잡았다.

하지만, 이러한 방식의 모형화는 관측치끼리 공통된 특성을 가질 수도 있고 서로 다른 특성을 가질 수도 있는 자료에는 사용될 수 없다. 예를 들면, 한 장의 사진은 사람, 나무, 고양이 등 복수의 특성을 포함할 수 있고, 또한 이 특성은 여러 장의 사진이 공통으로 가질 수 있는 특성이 되기도 한다. 잠재특성모형(latent feature model)이란, 이와 같이 한 개의 관측치가 여러 개의 특성을 가질 수 있고, 각 특성이 복수의 관측치의 공통된 특징이 될 수 있는 모형을 통칭한다.

인도부페 프로세스는 잠재특성에 대한 모형화를 위해 만들어진 프로세스이다. 디리크레 프로세스와 마찬가지로 인도부페 프로세스를 이용한 모형에서는 특성의 개수가 모형을 적합하는 과정에서 자연스럽게 추정된다. 인도부페 프로세스가 무한개의 특성을 가질 수 있다는 성질 때문에, 인도부페프로세스는 잠재특성모형 외에도 다양한 형태의 모형에 적용될 수 있으며 특히 복잡한 구조를 가진 자료에 적합한 모형을 만드는데 이용될 수 있다. 인도부페 프로세스를 통해 실제 자료에 의해 나타나는 특성의 개수를 자연스럽게 추정할 수 있다는 점, 그리고 이를 이용한 모형이 복잡한 구조의 자료를 설명하기에 적절하다는 사실 때문에, 인도부페 프로세스에 관한 연구는 2000년 후반부터 베이지안 연구자들의 연구 주제 중 큰 축을 담당하게 된다.

인도부페 프로세스에 관한 이론적인 결과 중 하나는 Thibaux와 Jordan이 인도부페 프로세스와 베타 프로세스의 연관성을 밝힌 것으로, 인도부페 프로세스로 생성된 교환가능한 특성의 디 피네트 측도(de Finetti measure)가 베타 프로세스가 된다는 사실이다 (Thibaux와 Jordan, 2007). 인도부페 프로세스를 이용한 모형의 추론을 위해 마코프체인 몬테카를로(Markov chain Monte Carlo; MCMC) 알고리즘을 이용할 때 계산 시간이 길어 현실적으로 모형 적합이 어려운 경우들이 있는데, 베타프로세스와 인도부페 프로세스와의 관련성이 밝혀지면서 이러한 문제들을 어느 정도 해결할 수 있는 새로운 마코프체인 몬테카를로 알고리즘들이 등장하게 된다. 현재까지 제안된 인도부페 프로세스의 대표적인 마코프체인 몬테카를로 알고리즘은, 인도부페 프로세스의 막대자르기(stick-breaking) 성질을 이용한 알고리즘 (Teh 등, 2007), 포아송 프로세스(Poisson process)의 성질을 이용한 알고리즘 (Paisley 등, 2012) 등이 있다.

그러나 확률적 근사에 기반한 마코프체인 몬테카를로 알고리즘들은 이러한 개선에도 불구하고, 여전히 방대한 양의 자료에 적용하기 어렵다는 문제가 존재했다. 이를 해결하기 위해 마코프체인 몬테카를로 알고리즘의 대안적 방법인 변분 방법(variational method)이 등장하게 된다. 자료의 차원이 클 때, 인

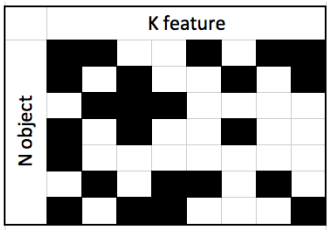

Figure 2.1. An example of the binary feature matrix Z.

도부페 프로세스를 이용한 모형의 추론에 변분방법을 이용하는 것이 마코프체인 몬테카를로 방법보다 더 효과적일 수 있다는 사실이 연구를 통해 밝혀졌다 (Doshi-Velez 등, 2008). 이와 더불어 컴퓨터의 병렬계산을 이용한 분산처리를 통해, 근사적으로 인도부페 프로세스를 이용한 모형을 추론하려는 노력도 함께 있어 왔다 (Doshi-Velez 등, 2009).

인도부페 프로세스와 베타프로세스와 관계는, 인도부페 프로세스의 확장에도 큰 영향을 주었다. 인도부페 프로세스의 확장은 두 방향으로 진행되고 있는데, 첫째는 인도부페 프로세스를 두 개 이상의 모수를 가지도록 확장하는 것이고 (Teh와 Gorur, 2009; Griffiths와 Gharahmani 2011) 둘째는, 중국식당 프로세스(Chinese restaurant process)와 비슷한 방법으로, 공변량과의 종속성을 부여하거나 계층성(hierachy)를 고려하여 어떤 특수한 구조를 갖는 인도부페 프로세스의 형태로 확장하는 것이다. 물론 이 둘을 융합한 확장도 생각할 수 있다. 즉, 두 개 이상의 모수를 가지면서 특수한 구조를 갖는 인도부페 프로세스에 대해 고려하는 것을 말한다. 특수한 구조를 갖는 인도부페 프로세스의 확장으로는 공변량이 존재할 때 공변량 간의 유사성을 바탕으로 비슷한 특성을 공유하게 만드는 종속 인도부페 프로세스(dependent IBP; dIBP; Williamson 등, 2010), 베타프로세스의 성질을 바탕으로 하여 계층적으로 공변량 종속성 가지게 하는 종속 계층 베타프로세스(hierarchical Beta process, dHBP; Zhou 등, 2011), 커널 베타프로세스(kernel Beta process; Ren 등, 2011) 등이 있다.

이 논문에서는 인도부페 프로세스에 대해 소개하고자 한다. 2장에서는 인도부페 프로세스의 이론, 3장에서는 인도부페 프로세스를 이용한 베이지안 모형의 계산 방법들을 소개한다. 4장에서는 모의 자료와 실제 자료에 적용한 예들을 보여주고, 5장에서는 실제 인도부페 프로세스가 이용되고 있는 응용분야에 대해 언급한다.

2. 인도부페 프로세스 이론

2.1. 인도부페 프로세스의 유도

잠재특성모형은 다음과 같이 구성된다. 만약 D차원으로 표현 가능한 N개의 관측치가 있다고 하면, 이 관측치는 $N \times D$차원의 $\boldsymbol{X}$의 행렬로 나타낼 수 있다. 잠재특성모형에서 k번째 잠재특성을 $\boldsymbol{f_k} = (f_{k1}, f_{k2}, \ldots, f_{kD})^T$의 벡터로 표현할 때, K개의 잠재특성은 행렬, $\boldsymbol{F} = [\boldsymbol{f}_1 \boldsymbol{f}_2 \cdots \boldsymbol{f}_K]^T$로 나타낼 수 있다. i번째 관측치인 $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{iD})^T$는 잠재특성으로부터 영향을 받는다고 가정한다. 즉, $p(\boldsymbol{X}|\boldsymbol{F})$의 형태로 관측치를 모형화 할 수 있는 경우, 이를 잠재특성모형이라고 부른다.

일반적으로 $\boldsymbol{F}$ 행렬은 다시 두 개의 요소로 나뉠 수 있다. 하나는 이진행렬인 $\boldsymbol{Z}$이고, 다른 하나는 각 잠재특성의 가중치를 나타내는 행렬인 $\boldsymbol{V}$이다. 이때, $\boldsymbol{F} = \boldsymbol{Z} \otimes \boldsymbol{V}$로 나타낼 수 있다. 여기서 $\otimes$는 두 행렬의 원소별 곱을 의미한다. $\boldsymbol{Z}$의 (i,k) 원소인 z_{ik}는 0 또는 1의 값을 가지며 이는 i번째 관측치가 k번째 특성을 포함하고 있는지의 여부를 나타낸다. $\boldsymbol{Z}$ 행렬을 간단한 그림으로 나타내면 Figure 2.1과 같다.

인도부페 프로세스는 이러한 이진행렬 $\boldsymbol{Z}$에 가정할 수 있는 모형 중 하나이다. 인도부페 프로세스를 이용하여 $\boldsymbol{Z}$를 모형화 하는 것의 장점은, 군집 구조를 모형화 하는 중국식당 프로세스와 마찬가지로, 잠재적인 특성의 개수를 무한한 것으로 가정하며 따라서 특성의 개수를 자료로부터 자연스럽게 추론할 수 있다는 것이다.

무한개의 특성을 가질 수 있는 특성모형을 무한특성모형(infinite feature model)이라고 부른다. 무한특성모형은 유한특성모형으로부터 유도할 수 있다. 유한특성모형이란 고정된 K개의 특성을 가진 이진행렬에 관한 모형으로, 다음과 같이 나타낼 수 있다.

$$\begin{aligned} \mu_k &\overset{iid}{\sim} \text{Beta}\left(\frac{\alpha}{K}, 1\right) \quad (k = 1, \ldots, K), \\ z_{ik} \mid \mu_k &\overset{ind}{\sim} \text{Bernoulli}(\mu_k) \quad (i = 1, \ldots, n). \end{aligned} \tag{2.1}$$

위 모형으로부터 계산된 이진행렬 $\boldsymbol{Z}$의 주변확률은 다음과 같다.

$$\begin{aligned} P(\boldsymbol{Z}) &= \prod_{k=1}^{K} \int \left(\prod_{i=1}^{N} p(z_{ik}|\mu_k) \right) p(\mu_k) d\mu_k \\ &= \prod_{k=1}^{K} \frac{B\left(m_k + \frac{\alpha}{K}, N - m_k + 1\right)}{B\left(\frac{\alpha}{K}, 1\right)} \\ &= \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma\left(m_k + \frac{\alpha}{K}\right)\Gamma(N - m_k + 1)}{\Gamma\left(N + 1 + \frac{\alpha}{K}\right)}. \end{aligned} \tag{2.2}$$

$K \to \infty$일 때의 식 (2.1)의 극한분포를 찾기 위해서는 이진행렬의 동등클래스(equivalence class)를 정의할 필요가 있다. 동등클래스는 이진행렬에 대한 lof$(\cdot)$ 함수를 이용해서 정의할 수 있으며, 임의의 이진행렬은 이 함수를 통해 왼쪽정렬(left-ordered) 이진행렬로 변환된다. 왼쪽정렬 이진행렬이란, 열에 의해 표현되는 이진숫자의 크기에 따라 왼쪽에서부터 오른쪽으로 차례로 정렬한 행렬을 뜻한다. 동일한 왼쪽정렬 이진행렬을 갖는 이진행렬들이 동등클래스에 속하며, 이를 $[\boldsymbol{Z}]$로 표기한다. $[\boldsymbol{Z}]$의 분포는 다음과 같이 나타낼 수 있다.

$$\begin{aligned} P([\boldsymbol{Z}]) &= \sum_{\boldsymbol{Z} \in [\boldsymbol{Z}]} P(\boldsymbol{Z}) \\ &= \frac{K!}{\prod_{h=0}^{2^N-1} K_h!} \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma\left(m_k + \frac{\alpha}{K}\right)\Gamma(N - m_k + 1)}{\Gamma\left(N + 1 + \frac{\alpha}{K}\right)}. \end{aligned} \tag{2.3}$$

위 식에서 m_k는 k번째 특성을 가지고 있는 관측치의 개수를 의미하며, K_h란 동일한 히스토리를 갖는 열의 개수를 뜻한다. 히스토리는 N개만큼의 이진수들을 갖는, 즉, 길이가 N인 이진수열의 경우들을 의미한다. 따라서 히스토리의 경우의 수는 모든 값이 0인 경우를 제외하면, $2^N - 1$개가 된다. 동일한 히스토리를 가진 열끼리는 순서를 바꿔도 동일한 이진행렬을 구성하게 되므로, 동일한 히스토리의 개수를 세어 왼쪽정렬 이진행렬의 확률을 계산하여야 하고 총 K개의 열이 있는 경우에는 결과적으로 식 (2.3)와 같은 식으로 정리할 수 있다. 위 동등이진행렬의 확률을 정리하여 K를 극한으로 보내면,

$$P([\boldsymbol{Z}]) = \frac{\alpha^{K_+}}{\prod_{h=0}^{2^N-1} K_h} \exp\left\{-\alpha \sum_{j=1}^{N} \frac{1}{j}\right\} \prod_{k=1}^{K^+} \frac{(N - m_k)!(m_K - 1)!}{N!} \tag{2.4}$$

으로 나타낼 수 있다. 여기서 K_+는 전체 관측치가 갖는 총 특성의 개수를 뜻한다.

동일한 왼쪽정렬 이진행렬을 갖는 동등클래스에 대한 확률인 식 (2.4)는 어떠한 확률과정으로부터 정의될 수 있는데, 이 확률과정을 바로 인도부페 프로세스라고 부른다. 인도부페 프로세스는 무한개의 요리가 있는 인도부페에서 차례로 들어온 손님이 요리를 선택하는 과정으로 설명할 수 있다. 이진행렬의 행에 해당하는 관측치를 손님, 열에 해당하는 특성을 요리로 각각 간주한다. 첫 번째 들어온 손님은 Poisson(α)로부터 생성된 개수만큼의 요리를 왼쪽부터 차례로 선택한다. i번째 손님은 앞선 손님이 선택한 요리들을 m_k/i의 확률로 선택하고, 아무도 선택하지 않은 요리를 Poisson(α/i)의 개수만큼 선택한다. m_k란 k번째 요리를 선택한 손님의 수이다. 이러한 과정들을 계속해 나가면 무한개의 특성을 가진 이진행렬를 생성할 수 있다. 이 프로세스를 통해 생성된 이진행렬 $\boldsymbol{Z}$는 다음의 확률을 갖는다.

$$P(\boldsymbol{Z}) = \frac{\alpha^{K_+}}{\prod_{i=1}^{N} K_1^{(i)}!} \exp\left\{-\alpha \sum_{j=1}^{N} \frac{1}{j}\right\} \prod_{k=1}^{K_+} \frac{(N-m_k)!(m_k-1)!}{N!}.$$

위 식에서 $K_1^{(i)}$란 행기준으로 i번째 행에서 처음 나타난 특성의 개수를 뜻한다. 이진행렬 $\boldsymbol{Z}$가 위와 같은 확률과정을 따를 때, $\boldsymbol{Z} \sim \text{IBP}(\alpha)$로 표기한다.

인도부페 프로세스로부터 생성되는 이진행렬에 대한 확률을 동일한 왼쪽정렬 이진행렬을 갖는 동등클래스 대한 확률로 변환하면,

$$P([\boldsymbol{Z}]) = \frac{\alpha^{K_+}}{\prod_{h=0}^{2^N-1} K_h} \exp\left\{-\alpha \sum_{j=1}^{N} \frac{1}{j}\right\} \prod_{k=1}^{K^+} \frac{(N-m_k)!(m_K-1)!}{N!}$$

와 같고, 이는 유한특성모형으로부터 유도된 무한특성모형의 동등클래스의 확률과 동일하다.

그러나 위에서 정의된 확률과정으로부터 생성된 이진행렬로부터 정의되는 특성은 교환가능하지 않다. 따라서 $\boldsymbol{Z} \sim \text{IBP}(\alpha)$를 가정했을 때의 모형의 추론을 위해서는, 왼쪽정렬된 형태의 이진행렬을 이용해야한다. 왼쪽정렬 이진행렬은 행에 대해 교환가능하며, 따라서 마코프체인 몬테카를로 알고리즘을 통해서 표집하는 행을 마치 마지막 행인 것처럼 생각할 수 있게 된다.

2.2. 베타프로세스와의 관련성

교환가능한 확률변수열 $(Z_1, \ldots, Z_n)$이 Q라는 분포를 따른다고 가정하면, 디 피네트의 정리(de Finetti theorem)에 따라

$$Z_1, \ldots, Z_n \mid P \overset{iid}{\sim} P$$

를 만족하는 측도 P가 항상 유일하게 존재한다. 즉, 디 피네트 정리란 교환가능한 확률변수열을 조건부 독립으로 만드는 측도의 존재성에 대한 정리이다. 이는 다시 표현하면

$$\mathbb{P}(Z_1, \ldots, Z_n) = \int \prod_{i=1}^{n} P(Z_i) \mathbb{P}(dP) \tag{2.5}$$

와 같이 쓸 수 있다. 여기서 $\mathbb{P}$는 해당 랜덤원소(random element)의 측도를 나타낸다.

중국레스토랑 프로세스의 경우 식 (2.5)를 만족하는 디 피네트 측도가 디리크레 프로세스임이 알려져 있다. 인도부페 프로세스를 따르는 왼쪽정렬 이진행렬은 교환가능하기 때문에 디 피네트 측도가 존재한다는 사실은 보장이 된다. 그러나 그 정확한 모양은 알려지지 않았다. 많은 베이지안 연구자들이 중국레스토랑 프로세스와 인도부페 프로세스의 이론적인 대칭성을 원했기 때문에 인도부페 프로세스에서도 식 (2.5)에 해당하는 측도를 찾는 것이 큰 관심사였고, 이를 처음 밝힌 것은 Thibaux와 Jordan이었다.

그들은 베타프로세스(Beta process)와 베르누이프로세스(Bernoulli process)의 관련성을 이용하여 베타프로세스가 인도부페 프로세스로부터 생성된 확률변수열에 대한 디 피네트 측도가 된다는 것을 보였다. 즉, B가 베타프로세스를 따르고 $Z_1, \ldots, Z_n \mid B$가 서로 독립이면서 B를 기저측도(base measure)로 가지는 베르누이프로세스를 따른다고 할 때, $Z_1, \ldots, Z_n$의 주변분포가 인도부페 프로세스가 된다는 것을 밝혔다 (Thibaux와 Jordan, 2007).

3. 계산

인도부페 프로세스를 가정한 모형의 추론을 위한 알고리즘은, 모형의 종류에 따라 그 방법이 매우 다양하다. 본 논문에서는 가우시안 선형모형(Gaussian linear model)에 대한 추론 알고리즘을 소개한다.

인도부페 프로세스를 이진행렬에 대한 분포로 가정했을 때의 가우시안 선형모형은 다음과 같다.

$$\begin{aligned} \boldsymbol{x}_i|\boldsymbol{z}_i, \boldsymbol{A}, \sigma_X &\sim N_D\left(\boldsymbol{A}^T\boldsymbol{z}_i, \sigma_X^2\boldsymbol{I}\right) \quad (i = 1, \ldots, N) \\ \boldsymbol{a}_k|\sigma_A &\sim N_D\left(\boldsymbol{0}, \sigma_A^2\boldsymbol{I}\right) \quad (k = 1, \ldots, K) \\ \boldsymbol{Z}|\alpha &\sim \mathrm{IBP}(\alpha) \end{aligned}$$

위 모형에서 $\boldsymbol{Z}$는 각 관측치가 특성을 포함하는지 여부에 대한 이진행렬이며, $\boldsymbol{z}_i$는 이진행렬의 i번째 행을 뜻한다. $\boldsymbol{a}_k$들은 특성으로 이해할 수 있으며, $\boldsymbol{A} = [\boldsymbol{a}_1 \cdots \boldsymbol{a}_K]^T$이고 이를 특성행렬이라고 부른다. $\boldsymbol{x}_i$는 D차원의 벡터로 i번째 관측치를 뜻하며, $\boldsymbol{X} = [\boldsymbol{x}_1 \cdots \boldsymbol{x}_n]^T$이다. σ_X^2은 $\boldsymbol{X}$에 대한 오차의 크기, σ_A^2은 $\boldsymbol{A}$에 대한 오차의 크기를 뜻하며, α는 인도부페 프로세스의 모수이다.

본 논문에서는 가우시안 선형모형의 추론을 위한 알고리즘으로, 사후분포로부터 모수를 표집하여 추론하는 마코프체인 몬테카를로 방법을 이용한 알고리즘과 사후분포의 근사분포를 최적화하는 추정값을 직접 찾는 변분방법을 이용한 알고리즘을 소개한다. 마코프체인 몬테카를로 방법을 이용한 알고리즘 중, 인도부페 프로세스에서 생성된 왼쪽정렬 이진행렬의 교환 가능한 성질에 기반한 것을 깁스표집(Gibbs sampling) 알고리즘, 그리고 인도부페 프로세스의 막대자르기 표현에 기반한 알고리즘인 막대자르기 알고리즘이라고 명명한다.

3.1. 깁스표집 알고리즘

깁스표집 알고리즘을 이용하기 위해서는, 먼저 추론하고자 하는 모수에 대한 사후분포를 구하여야 한다. 편의를 위해 확률변수 Y가 주어졌을 때 확률변수 X의 조건부 분포를 간략하게 $[X|Y]$로 표현하자. 이 표현을 이용하면 특성행렬 $\boldsymbol{A}$와 이진행렬 $\boldsymbol{Z}$를 추론의 대상이라고 할 때, $\boldsymbol{A}$와 $\boldsymbol{Z}$의 사후분포를 $[\boldsymbol{A}, \boldsymbol{Z}|\boldsymbol{X}, \sigma_X, \sigma_A, \alpha]$로 나타낼 수 있다. 여기서 $\sigma_X, \sigma_A, \alpha$는 고정된 모수라고 가정하면, 사후분포는 $[\boldsymbol{A}, \boldsymbol{Z}|\boldsymbol{X}]$로 표현할 수 있다.

사후분포로부터 $\boldsymbol{A}$와 $\boldsymbol{Z}$를 표집하는 방법은 두 가지가 있는데, 첫째는 $\boldsymbol{A}, \boldsymbol{Z}$는 각각의 조건부 사후분포인 $[\boldsymbol{A}|\boldsymbol{Z}, \boldsymbol{X}]$와 $[\boldsymbol{Z}|\boldsymbol{A}, \boldsymbol{X}]$에서 표집하는 것이고, 둘째는 $[\boldsymbol{Z}|\boldsymbol{X}]$에서 $\boldsymbol{Z}$를 표집하고 $[\boldsymbol{A}|\boldsymbol{Z}, \boldsymbol{X}]$로부터 $\boldsymbol{A}$를 차례로 표집하는 것이다. 첫 번째 방법을 비붕괴깁스표집(uncollapsed Gibbs sampler), 두 번째 방법을 붕괴깁스표집(collapsed Gibbs sampler)이라고 부른다. 구체적인 표집방법은 다음에 설명한다.

3.1.1. 비붕괴깁스표집 알고리즘은 기존에 나타난 특성에 대해 이진행렬의 각 원소를 표집하는 과정과 새로운 특성을 추가하는 과정, 그리고 $\boldsymbol{A}$를 표집하는 과정으로 나눌 수 있다. 사후분포로부터 이진행렬 $\boldsymbol{Z}$를 표집 할 때는, 각 관측치에 대한 반복마다 기존에 존재하는 특성들을 포함여부를 결정하는 과

정과 새로운 특성을 얼마나 추가할지를 결정하는 과정이 필요하다. i번째 관측치가 기존에 존재하는 특성을 포함할지에 대한 여부는

$$p(z_{ik}=1|\boldsymbol{z_{-ik}},\boldsymbol{A},\boldsymbol{X}) \propto \frac{m_{-ik}}{N-1}p(\boldsymbol{X}|\boldsymbol{Z},\boldsymbol{A}),$$
$$p(z_{ik}=0|\boldsymbol{z_{-ik}},\boldsymbol{A},\boldsymbol{X}) \propto \left(1-\frac{m_{-ik}}{N-1}\right)p(\boldsymbol{X}|\boldsymbol{Z},\boldsymbol{A}) \tag{3.1}$$

를 기반으로 한다. 이 식에서 m_{-ik}는, 이진행렬의 k번째 열에서 i번째 원소를 제외한 나머지 원소 중 1의 값을 갖는 원소의 개수를 뜻한다.

새로운 특성은

$$p\left(k_{new}|\boldsymbol{X}\right) \propto \text{Poisson}\left(k_{new};\frac{\alpha}{N}\right)p\left(\boldsymbol{X}|\boldsymbol{Z_{new}},\boldsymbol{A_{new}}\right) \tag{3.2}$$

의 확률에 근거하여 표집한다. 이 식에서 $\boldsymbol{Z}_{new},\boldsymbol{A}_{new}$는 새로운 특성의 개수만큼 새로 사전분포로부터 생성된 이진행렬과 특성행렬을 뜻한다.

$\boldsymbol{A}$를 표집하기 위해서는 평균이 $\boldsymbol{\mu}_A$, 분산이 $\boldsymbol{\Sigma}_A$인 행렬 정규분포를 이용한다 (Doshi-Velez와 Ghahramani, 2009):

$$\boldsymbol{\mu}_A = \left(\boldsymbol{Z^T Z}+\frac{\sigma_X^2}{\sigma_A^2}\boldsymbol{I}\right)^{-1}\boldsymbol{Z^T X},$$
$$\boldsymbol{\Sigma}_A = \sigma_X^2\left(\boldsymbol{Z^T Z}+\frac{\sigma_X^2}{\sigma_A^2}\boldsymbol{I}\right)^{-1}.$$

3.1.2. 붕괴깁스표집 붕괴깁스표집방법은, 비붕괴깁스표집방법에서 자료와 $\boldsymbol{A}$가 주어진 상황에서 $\boldsymbol{Z}$를 표집하는 것과 달리, 자료만 주어진 상황에서 $\boldsymbol{Z}$를 표집하고 그 후 $\boldsymbol{A}$를 표집하는 방법이다. 붕괴깁스표집 알고리즘은, 비붕괴깁스표집 알고리즘에서 이진행렬을 표집하는 방법인 식 (3.1), (3.2)에서 $p(\boldsymbol{X}|\boldsymbol{Z},\boldsymbol{A})$대신 $p(\boldsymbol{X}|\boldsymbol{Z})$를 대입한 것과 동일하다. $\boldsymbol{A}$를 표집하는 방법은 비붕괴깁스표집에서와 같다.

3.2. 막대자르기 알고리즘

인도부페 프로세스를 식 (2.1) 같은 유한특성모형에서 $K\to\infty$으로 확장한 무한특성모형으로 생각할 수 있다는 것은 이미 밝혔다. 하지만 특성의 개수 K가 무한으로 확장되는 상황에서 μ_k를 사후분포로부터 표집하는 것이 사실상 불가능하기 때문에, Teh 등 (2007)은 이러한 문제를 해결하며 무한특성모형의 μ_k들을 표현하기 위하여 다음의 사실을 이용하였다: $\mu_1,\ldots,\mu_K \overset{iid}{\sim} \text{Beta}(\alpha/K,1)$이라고 할 때, 이들의 순서통계량인 $\mu_{(1)}>\cdots>\mu_{(K)}$의 분포는

$$\nu_k \overset{iid}{\sim} \text{Beta}(\alpha,1) \quad (k=1,\ldots,K),$$
$$\mu_{(k)} = \nu_k\cdot\mu_{(k-1)} = \prod_{l=1}^{k}\nu_l \tag{3.3}$$

와 같다. 이 사실을 이용하면, 식 (3.3)의 표현을 사용하여 베타분포를 따르는 확률변수들인 ν_k를 특성의 개수만큼 표집하고 이를 바탕으로 각 특성의 출현 확률인 μ_k의 순서통계량을 구할 수 있게 된다. 이것을 인도부페 프로세스의 막대자르기 표현이라고 한다.

막대자르기 표현에서도 사후분포 추론을 위해서는 특성의 개수를 유한개로 제한해야 하는데, 이것을 임의로 절단하는 것은 오차를 포함하는 근사를 사용하게 된다는 의미가 된다. 또한 절단의 정도가 자연스

럽게 결정되는 것이 아니라 사용자가 선택을 해야 하기 때문에, 모형선택의 문제가 발생하게 된다. 이는 보조변수 s를 도입하는 슬라이스 샘플러(slice sampler)를 사용함으로써 해결된다 (Teh 등, 2007).

보조변수를 생각할 때, 어떻게 절단 정도가 결정이 되는지를 살펴보자. 먼저 보조변수 s를 아래와 같이 두고

$$s \mid \boldsymbol{Z}, \boldsymbol{\mu}_{(1:\infty)} \sim \text{Uniform}[0, \mu^*], \quad \mu^* := \min\left\{1, \min_{k:\exists i, z_{ik}=1} \mu_{(k)}\right\}, \tag{3.4}$$

이를 이용하면 행렬 $\boldsymbol{Z}$의 조건부분포를

$$p\left(\boldsymbol{Z} \mid \boldsymbol{X}, s, \boldsymbol{\mu}_{(1:\infty)}\right) \propto p\left(\boldsymbol{Z} \mid \boldsymbol{X}, \boldsymbol{\mu}_{(1:\infty)}\right) \frac{1}{\mu^*} I\left(0 \le s \le \mu^*\right)$$

와 같이 구할 수 있다. 즉, 보조변수 s가 절단기준이 되어 전체 특성 중 s보다 큰 특성들만 남고 나머지 특성에 해당하는 $\boldsymbol{Z}$의 열은 0이 되어 고려대상에서 제외되게 된다. $s < \mu_{(k)}$를 만족하는 가장 큰 인덱스를 K_+라 하고, 그 인덱스보다 작은 특성들을 활성화 된 특성들이라고 부른다.

보조변수 s를 포함하여 추론을 진행하는 다음 일련의 과정을 막대자르기 알고리즘이라고 부른다. 보조변수 s에 대한 표집은 식 (3.4)를 이용하여 균일분포에서 진행한다. 새로운 s를 뽑은 후 $k = K_+ + 1, K_+ + 2, \ldots$에 대하여

$$p\left(\mu_{(k)} \mid \mu_{(k-1)}, \boldsymbol{z}_{:,k} = 0\right) \propto \exp\left(\alpha \sum_{i=1}^{n} \frac{1}{i}(1-\mu_{(k)})^i\right) \mu_{(k)}^{\alpha-1} \left(1-\mu_{(k)}\right)^n I(0 \le \mu_{(k)} \le \mu_{(k-1)}) \tag{3.5}$$

에서 $\mu_{(k)} \le s$를 만족할 때 까지 발생시킨 뒤, 이 중 $\mu_{(k)} > s$를 만족하는 특성들을 활성화 된 특성에 추가하여 K_+를 갱신한다. 이 때, 새롭게 활성화 된 특성들에 대한 $\boldsymbol{z}_{:,k} = (z_{1k}, \ldots, z_{nk})^T$는 0으로, $\boldsymbol{a}_k = (a_{k1}, \ldots, a_{kd})^T$는 사전분포에서 뽑아놓는다. 식 (3.5)의 분포가 $\log \mu_{(k)}$에 대하여 로그 오목(log concave) 성질을 가지므로, 적응기각표집(adaptive rejection sampling; ARS)을 사용한다.

이진행렬 $\boldsymbol{Z}$에 대한 추론을 할 때에는 다음의 사후분포

$$\begin{aligned} p(z_{ik} = 1 \mid \boldsymbol{z}_{-ik}, \boldsymbol{A}, \boldsymbol{X}) &\propto \frac{\mu_{(k)}}{\mu^*} p\left(\boldsymbol{x}_i \mid \boldsymbol{z}_{i,-k}, z_{ik} = 1, \boldsymbol{A}\right), \\ p(z_{ik} = 0 \mid \boldsymbol{z}_{-ik}, \boldsymbol{A}, \boldsymbol{X}) &\propto \left(1 - \frac{\mu_{(k)}}{\mu^*}\right) p\left(\boldsymbol{x}_i \mid \boldsymbol{z}_{i,-k}, z_{ik} = 0, \boldsymbol{A}\right) \end{aligned}$$

에서 $k \le K_+$인 특성들을 갱신한다. 위 식에서 $\boldsymbol{z}_{i,-k}$는 이진행렬의 i번째 열에서 k번째 원소를 제외한 것을 뜻한다.

순서를 매긴 특성의 출현 확률인 $\mu_{(k)}$, $k = 1, \ldots, K^+ - 1$의 사후분포는

$$p\left(\mu_{(k)} \mid \mu_{(k-1)}, \mu_{(k+1)}, \boldsymbol{Z}\right) \propto \mu_{(k)}^{m_k - 1}(1-\mu_{(k)})^{n-m_k} I\left(\mu_{(k+1)} \le \mu_{(k)} \le \mu_{(k-1)}\right) \tag{3.6}$$

과 같이 주어지고, $\mu_{(K_+)}$의 사후분포는 식 (3.5)가 된다. 여기서 $m_k = \sum_{i=1}^n z_{ik}$이다. 식 (3.5)와 (3.6)은 각각 $\log \mu_{(k)}$와 $\mu_{(k)}$에 대하여 로그 오목 성질을 만족하므로 적응기각표집 방법을 이용해서 표집한다. 특성행렬 $\boldsymbol{A}$는 깁스표집 알고리즘과 동일한 다변량 정규분포로부터 표집한다.

3.3. 변분 방법

사후분포를 직접 계산하기 어려운 경우, 다루기 쉬운 분포들의 집합인 $\mathcal{Q}$를 생각하고, 그 중에서 목표로 하는 사후분포와 쿨백-라이블러 발산(Kullback-Leibler divergence)이 가장 작은 분포를 찾음으로써 사

후분포에 대해 추론한다. 만일 $\mathcal{Q}$가 모든 분포의 집합이 된다면, 사후분포 그 자체가 자신과의 쿨백-라이블러 발산이 0으로 가장 작기 때문에 정확한 사후분포를 구할 수 있으나 실제로 이를 구하는 것은 불가능하기 때문에 최대한 유사한 분포 찾아 근사적으로 추론하는 것을 목표로 한다.

인도부페 프로세스를 이진행렬의 사전분포로 이용한 가우시안 선형모형에서 추정해야하는 모수는 $\boldsymbol{\mu}_{(1:\infty)}, \boldsymbol{Z}, \boldsymbol{A}$이다. 고정된 값을 가지는 초모수(hyperparameter)를 $\boldsymbol{\theta} = (\sigma_X, \sigma_A, \alpha)$라고 나타내고 추정해야 할 모수를 $\boldsymbol{W} = (\boldsymbol{\mu}_{(1:\infty)}, \boldsymbol{Z}, \boldsymbol{A})$로 나타낼 때, 사후분포의 로그가능도함수를 정리하면 다음과 같다.

$$\log p(\boldsymbol{W}|\boldsymbol{X}, \boldsymbol{\theta}) = \log p(\boldsymbol{W}, \boldsymbol{X}|\boldsymbol{\theta}) - \log p(\boldsymbol{X}|\boldsymbol{\theta}). \tag{3.7}$$

이 때, 일반적으로 $p(\boldsymbol{X}|\boldsymbol{\theta})$의 형태를 알기 어렵기 때문에, 변분 방법을 통해 $p(\boldsymbol{W}|\boldsymbol{X}, \theta)$를 근사적으로 추론하게 된다. 즉, 쉽게 다룰 수 있는 분포의 집합에서 쿨백-라이블러 발산인, $D(q(\boldsymbol{W})||p(\boldsymbol{W}|\boldsymbol{X}, \boldsymbol{\theta}))$를 최소로 하는 $q \in \mathcal{Q}$를 찾아 이를 사후분포에 대한 근사로 사용한다. 그러나 쿨백-라이블러 발산을 직접 최소화하는 것이 어렵기 때문에, $p(\boldsymbol{X}|\boldsymbol{\theta})$의 하한을 q를 통해 표현하고 그것을 최대화하는 방법으로 쿨백-라이블러 발산을 최소화 하는 q를 찾게 된다. 이것은 아래 식에 의해 정당화 된다.

$$p(\boldsymbol{X}|\boldsymbol{\theta}) = E_q[\log(p(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\theta})] + H(q) + D(q||p) \geq E_q[\log(p(\boldsymbol{X}, \boldsymbol{W}|\boldsymbol{\theta})] + H(q). \tag{3.8}$$

변분 방법을 이용한 추론 알고리즘은 인도부페 프로세스를 유한특성모형으로 근사한 후 이와 쿨백-라이블러 발산이 최소인 q를 찾는 방법과, 인도부페 프로세스를 막대자르기 표현으로 생각하고 이와 쿨백-라이블러 발산이 최소가 되게하는 q를 찾는 방법이 있다. 이를 각각 유한변분방법(finite variational mehtod)와 무한변분방법(inifinite variational method)이라고 부른다.

3.3.1. 유한변분방법 인도부페 프로세스를 유한특성모형으로 나타낸 것은 식 (2.1)와 같다. 가우시안 선형모형에서 고려하는 분포의 집합 $\mathcal{Q}$는 아래와 같이 $\boldsymbol{\tau}, \boldsymbol{\phi}, \boldsymbol{\Phi}, \boldsymbol{\nu}$에 의해 각 모수에 대해 독립적으로 정의할 수 있다.

$$\begin{aligned} q_{\tau_k}(\mu_k) &= \text{Beta}(\tau_{k1}, \tau_{k2}), \\ q_{\phi_k}(\boldsymbol{a}_k) &= N_D(\boldsymbol{\phi}_k, \boldsymbol{\Phi}_k), \\ q_{\nu_{nk}}(z_{nk}) &= \text{Bernoulli}(\nu_{nk}). \end{aligned}$$

따라서 $q(\boldsymbol{W}) = q_\tau(\boldsymbol{\mu}) q_\phi(\boldsymbol{A}) q_\nu(\boldsymbol{Z})$로 쓸 수 있다. p_K를 인도부페 프로세스를 유한특성모형으로 나타낸 가우시안 선형모형이라고 하면, 위와 같은 분포 가정 하에서 $D(q||p_K)$를 최소로 하는 $\boldsymbol{\tau}, \boldsymbol{\phi}, \boldsymbol{\Phi}, \boldsymbol{\nu}$를 찾는 것이 추론의 목적이 된다. 이를 위해서는 $\log p_K(\boldsymbol{X}|\boldsymbol{\theta})$의 하한을 최대화하는 값들을 찾아야 한다. 하한을 최대로 하는 값들은 수치적 최적화방법을 통해 $\boldsymbol{\tau}, \boldsymbol{\phi}, \boldsymbol{\Phi}, \boldsymbol{\nu}$를 갱신하는 방법으로 구한다. 갱신을 위한 식은 아래와 같다 (Doshi-Velez 등, 2008).

$$\boldsymbol{\Phi}_k = \left(\frac{1}{\sigma_A^2} + \frac{\sum_{n=1}^{N} \nu_{nk}}{\sigma_X^2} \right)^{-1} \boldsymbol{I},$$

$$\boldsymbol{\phi}_k = \left[\frac{1}{\sigma_X^2} \sum_{n=1}^{N} \nu_{nk} \left(\boldsymbol{X}_{n\cdot} - \left(\sum_{l:l \neq k} \nu_{nl} \boldsymbol{\phi}_l \right) \right) \right] \left(\frac{1}{\sigma_A^2} + \frac{\sum_{n=1}^{N} \nu_{nk}}{\sigma_X^2} \right)^{-1},$$

$$\nu_{nk} = \frac{1}{1+e^{-\mathcal{N}}},$$
$$\tau_{k1} = \frac{\alpha}{K}\sum_{n=1}^{N}\nu_{nk},$$
$$\tau_{k2} = N+1-\sum_{n=1}^{N}\nu_{nk}.$$

위 식에서 $\mathcal{N}$은 $\psi(\tau_{k1})-\psi(\tau_{k2})-1/(2\sigma_X^2)\left(\mathrm{tr}(\boldsymbol{\Phi}_k)+\boldsymbol{\phi}_k\boldsymbol{\phi}_k^T\right)+1/(\sigma_X^2)\,\boldsymbol{\phi}_k(\boldsymbol{X}_{n\cdot}^T-(\sum_{l:l\neq k}\nu_{nl}\boldsymbol{\phi}_l^T))$를 나타낸다.

3.3.2. 무한변분방법 무한변분방법에서는 인도부페 프로세스의 막대자르기 표현을 이용한 모형인 식 (3.3)을 사용한다. 무한변분방법이라는 이름을 붙이기는 하였지만, 사실상 이 방법에서는 무한개의 막대를 생각하지 않고 유한개의 K까지 절단한 막대를 이용하여 근사한 모형을 이용하게 된다.

분포의 집합인 $\mathcal{Q}$는 유한변분모형에서와 동일하게 정의하고 p_K를 인도부페 프로세스를 절단된 막대자르기 표현으로 나타낸 가우시안 선형모형이라고 하면, 추론의 목적은 이러한 가정하에서 쿨백-라이블러 발산인 $D(q||p_K)$을 최소로 하는 $\boldsymbol{\tau},\boldsymbol{\phi},\boldsymbol{\Phi},\boldsymbol{\nu}$를 찾는 것이 된다. 이를 위해서는 유한변분모형에서와 같이 $\log p_K(\boldsymbol{X}|\theta)$의 하한을 최소로 하는 $\boldsymbol{\tau},\boldsymbol{\phi},\boldsymbol{\Phi},\boldsymbol{\nu}$를 찾아야 하며, 역시 최적화 방법을 통해 반복적으로 갱신 하는 방법을 이용한다. 갱신을 위한 식은 아래와 같다.

$$\boldsymbol{\Phi}_k = \left(\frac{1}{\sigma_A^2}+\frac{\sum_{n=1}^{N}\nu_{nk}}{\sigma_X^2}\right)^{-1}\boldsymbol{I},$$
$$\boldsymbol{\phi}_k = \left[\frac{1}{\sigma_X^2}\sum_{n=1}^{N}\nu_{nk}\left(\boldsymbol{X}_{n\cdot}-\left(\sum_{l:l\neq k}\nu_{nl}\boldsymbol{\phi}_l\right)\right)\right]\left(\frac{1}{\sigma_A^2}+\frac{\sum_{n=1}^{N}\nu_{nk}}{\sigma_X^2}\right)^{-1},$$
$$\nu_{nk} = \frac{1}{1+e^{-\mathcal{N}}},$$
$$\tau_{k1} = \alpha+\sum_{m=k}^{K}\sum_{n=1}^{N}\nu_{nm}+\sum_{m=k+1}^{K}\left(N-\sum_{n=1}^{N}\nu_{nm}\right)\left(\sum_{i=k+1}^{m}q_{mi}\right),$$
$$\tau_{k2} = 1+\sum_{m=k}^{K}\left(N-\sum_{n=1}^{N}\nu_{nm}\right)q_{mk}.$$

위 식에서 $\mathcal{N}=\sum_{i=1}^{k}(\psi(\tau_{i1})-\psi(\tau_{i1}+\tau_{i2}))-E_v\left[\log(1-\prod_{m=1}^{k}v_m)\right]-(1/(2\sigma_X^2))(\mathrm{tr}(\boldsymbol{\Phi}_k)+\boldsymbol{\phi}_k\boldsymbol{\phi}_k^T)+(1/\sigma_X^2)\boldsymbol{\phi}_k(\boldsymbol{X}_{n\cdot}^T-(\sum_{l:l\neq k}\nu_{nl}\boldsymbol{\phi}_l^T))$이다. 위 갱신식을 위해서는 $E_v[\log(1-\prod_{m=1}^{k}v_m)]$를 계산해야 하는데, 이 계산 역시 하한으로 근사하여 계산한다. 즉,

$$E_v\left[\log(1-\prod_{m=1}^{k}v_m)\right] \geq \left(\sum_{m=1}^{k}q_{km}\psi(\tau_{m2})\right)+\left(\sum_{m=1}^{k-1}\left(\sum_{n=m+1}^{k}q_{kn}\right)\psi(\tau_{m1})\right)$$
$$-\left(\sum_{m=1}^{k}\left(\sum_{n=m}^{k}q_{kn}\right)\psi(\tau_{m1}+\tau_{m2})\right)-\sum_{m=1}^{k}q_{km}\log q_{km}$$

이며, 여기서 $q_{ki}\propto\exp(\psi(\tau_{i2})+\sum_{m=1}^{i-1}\psi(\tau_{m1})-\sum_{m=1}^{i}\psi(\tau_{m1}+\tau_{m2}))$이다.

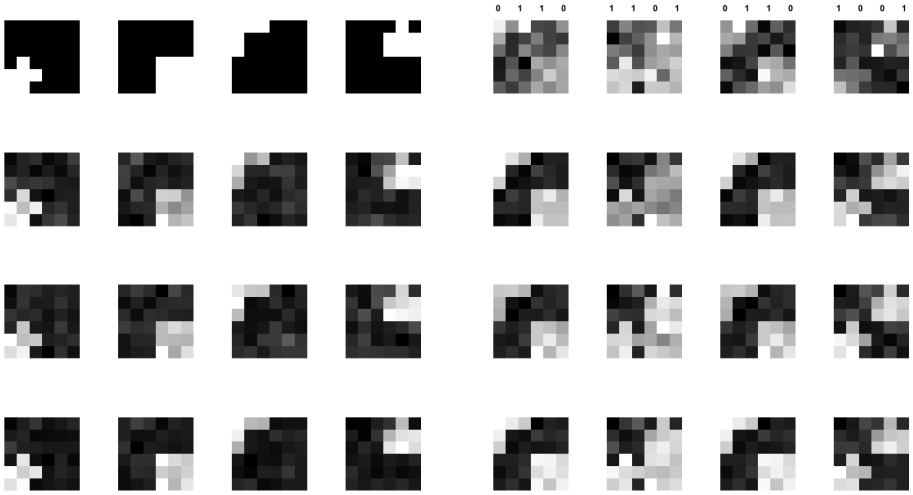

Figure 4.1. The four true latent features (top of the left column). Estimated latent features using the algorithm 1–3 (from the second to fourth row of the left column). Four sample images from the data set and the corresponding binary vectors (top of the right column). Reconstructed four sample images using the algorithm 1–3 (from the second to fourth row of the right column).

4. 분석

4.1. 시뮬레이션 분석

무한개의 잠재특성에 관한 모형은 많은 분야에서 적용가능하다. 특히 이미지 자료를 표현하는 특성, 예를 들면 각 이미지에 포함되어 있는 물체를 찾는 문제 등에 유용하게 적용될 수 있다.

간단한 그림찾기 문제에서 인도부페 프로세스 모형을 이용한 분석을 생각해 보자. Figure 4.1의 왼쪽 첫 번째 행에 표현된 4개의 서로 다른 종류의 그림들이 선형결합되고, 노이즈가 추가되어 얻어진 자료가 주어졌다고 하자. 이 때, 실제 인도부페 프로세스를 이용한 모형이 이 4종류의 특성을 얼마나 잘 찾아내며 이 특성들의 조합으로 실제 자료가 얼마나 잘 복원되는가를 확인하려고 한다. 시뮬레이션을 통해 4개의 이미지특성의 선형결합으로 이루어진 100개의 6×6픽셀 그림을 얻었다. 주어진 자료는 100×36의 행렬 형태로 나타낼 수 있다. 이러한 문제에 대한 추론에 적합한 모형은 가우시안 선형모형이며 이진행렬에 대한 모형으로 인도부페 프로세스를 이용하였다. 이 예제를 앞서 소개된 3가지 방법의 알고리즘(알고리즘1: 비붕괴깁스표집, 알고리즘2: 막대자르기 알고리즘, 알고리즘3: 변분방법)을 이용하여 추론을 시행했을 때의 결과는 Figure 4.1에서 확인할 수 있다.

이 중 비붕괴깁스표집을 이용했을 때, 반복에 따라 변화하는 이미지 특성의 정보를 특정 반복수 에서의 시계열 그림의 형태로 나타낸 결과는 Figure 4.2와 같다. 시계열 그림을 통해, 표집이 반복되면서 실제 이미지 자료를 생성한 이미지 특성을 찾아가는 것을 확인할 수 있다.

이를 좀 더 실제적인 고차원 자료에 적용하여 보자. 다양한 표정을 가진 서로 다른 4명의 실제 얼굴을 바탕으로 하여 노이즈가 추가된 100개의 자료를 생성하였다. 얼굴이미지는 128×128픽셀로 이루어져 있으며 따라서 각각의 관측치는 16384차원의 자료로 생각할 수 있다. 전체 자료는 100×16384의 행렬 형태로 나타낼 수 있으며, 이는 자료의 차원이 관측치의 개수보다 큰 고차원 자료이다. 고차원자료에서의 인도부페 프로세스의 적용은 표집시간이 오래걸린다는 문제가 있다. 이러한 문제를 해결하기 위해

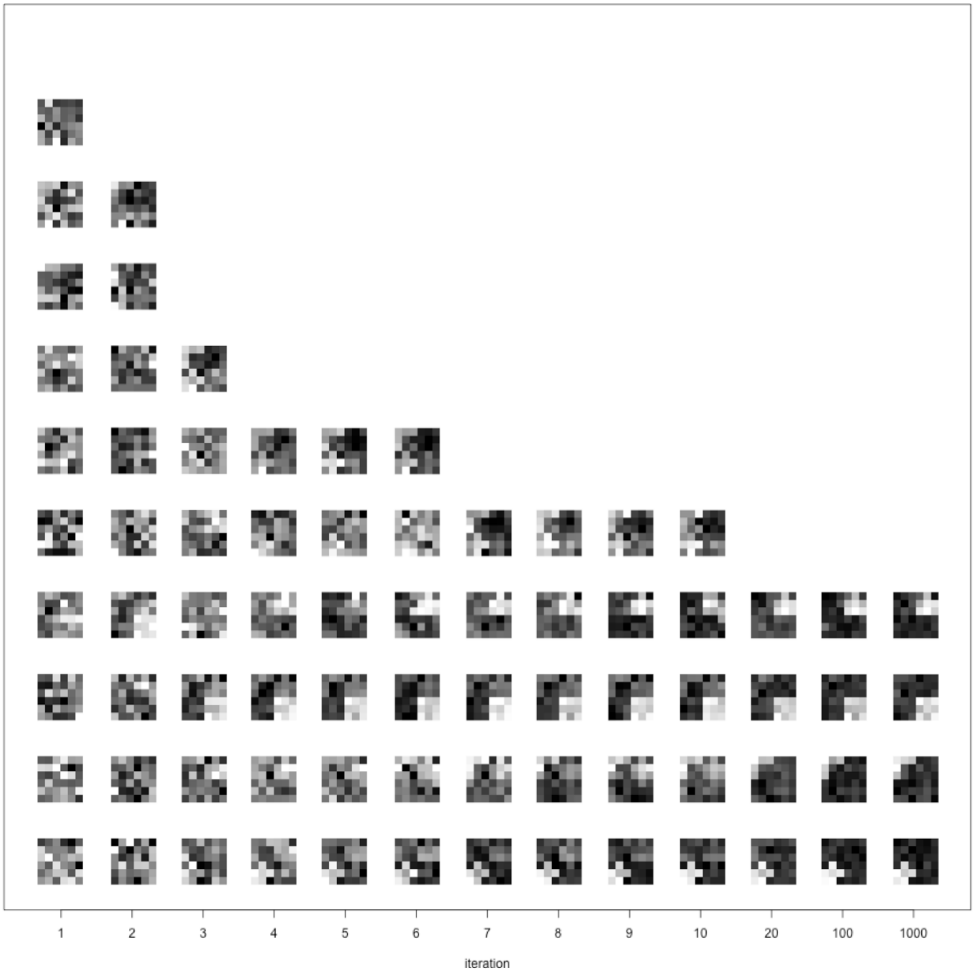

Figure 4.2. Time series plot of latent features at 1∼10, 20, 100, 1000th iteration using the algorithm 1.

서 고차원의 자료를 주성분분석(Principal Components Analysis; PCA)을 이용하여 10개의 차원을 가진 자료로 축소하였고 축소된 자료에 시뮬레이션 자료와 동일한 가우시안 선형모형을 적용하였다. 3가지 알고리즘을 이용한 추론 결과는 Figure 4.3에서 확인할 수 있다.

첫 번째 알고리즘을 이용해서 찾은 특성은 5개로 나타났지만, 그 중 하나의 특성의 경우 그 특성을 포함하고 있는 관측치의 개수가 5퍼센트 이하였기 때문에 제외하였다. 두 번째 알고리즘을 이용한 결과는 정확히 4개의 얼굴을 이미지특성으로 찾아냈다. 인도부페 프로세스를 이용해 찾은 이미지 특성을 조합하여 복원한 4개의 자료는 노이즈가 있는 원래의 자료에 비해 훨씬 깨끗하고 정확한 이미지를 보여준다.

주성분분석을 이용해 찾은 요인을 이용하여 복원한 자료도 인도부페 프로세스를 이용한 결과와 거의 비슷한 결과를 준다는 것을 확인할 수 있다. 그러나 주성분분석을 통해 찾은 요인들은 개별적인 얼굴을 이미지 특성으로 정확하게 판별하지 못하는 양상을 보인다. 이 결과는, 만약 어떠한 자료로부터 자료를 구성하는 특성을 판별하고자하는 목적을 가지고 있는 경우에는 인도부페 프로세스를 이용한 추론이 좀 더

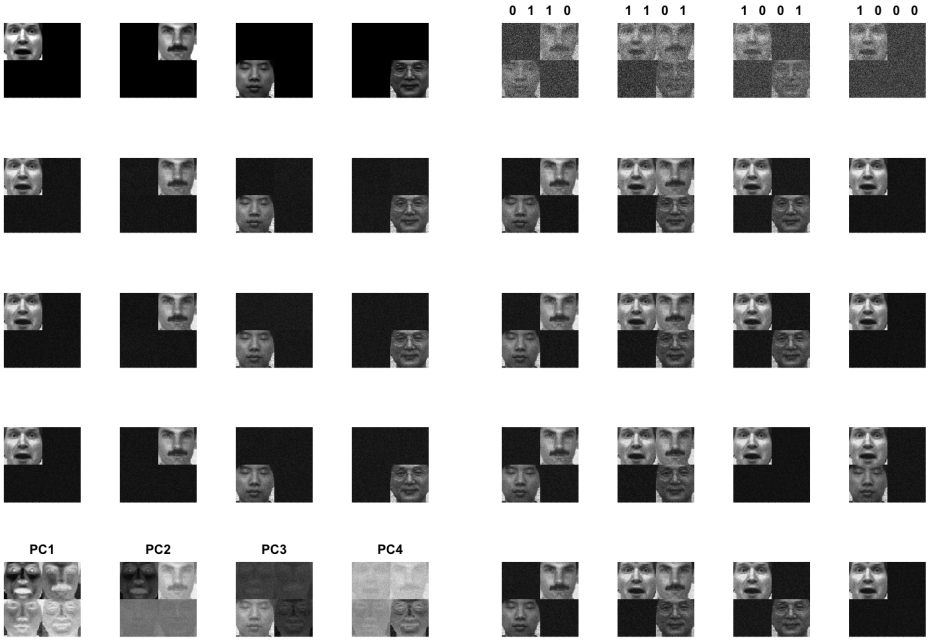

Figure 4.3. The four true latent features (top of the left column). Estimated latent features using the algorithm 1–3 (from the second to fourth row of the left column). Four principal components (bottom of the left column). Four sample images from the data set and the corresponding binary vectors (top of the right column). Reconstructed four sample images using the algorithm 1–3 and PCA(from the second to fifth row of the right column).

직관적일 수 있다는 것을 보여준다.

두 가지 예제의 결과는 인도부페 프로세스를 이용한 모형이 자료에 내재된 이미지특성을 거의 완벽하게 찾아내고 있음을 확인시켜준다. 또한 찾아낸 이미지특성을 통해 노이즈가 섞인 이미지로부터 선명한 이미지를 복원할 수 있음을 보여준다. 따라서 인도부페 프로세스를 이용한 모형은 여러 개의 이미지 자료로부터 자동차, 사람 등의 특정한 이미지 형태를 구분해 내거나, 노이즈가 있는 이미지로부터 선명한 이미지를 복원하는 문제등에 유용하게 적용될 수 있다는 것을 알 수 있다.

5. 응용

이미지 분석 외에도 인도부페 프로세스를 적용할 수 있는 분야는 매우 다양한다. 생물학, 의학 등의 다양한 분야에서 관측할 수 있는 쌍(dyadic) 자료분석, 소셜네트워크서비스와 사회학 등에서 이용되는 네트워크 자료분석, 그리고 시그널자료에 대해 적용할 수 있는 독립성분분석 등이 대표적인 예이다.

5.1. 쌍자료분석

쌍자료는 행렬로 표현할 수 있으며, 쌍자료에 대한 대부분의 모형화는 행렬의 분해를 통해 이루어진다. 이러한 자료는 영화-관객 평점 자료, 마이크로어레이(microarry array)자료 등이 대표적이다. 마이크로어레이자료 중 유전자발현자료(gene expression data)는 유전자(gene)과 샘플(sample)이라는 두 개의 영역에서 유전자발현레벨(gene expression level)을 관측한 것을 뜻한다.

이러한 자료에 대한 분석으로 주로 이용되는 방법은 이중클러스터링(bi-clustring)이다. 이는 행과 열을 그룹화하는 방법으로 혼합모형의 일종이다. 이 모형에서는 행에서의 하나의 특성, 그리고 열에서의 하나의 특성에만 관측치가 속할 수 있다고 가정한다. 즉, 관측치가 하나 이상의 그룹에 포함될 수 없다는 가정이 필요하다.

그러나 관측치가 하나의 그룹에만 포함될 수 있다는 가정은 너무 제한적이다. 예를 들어 유전자 발현레벨은, 유전자 영역에서 알지 못하는 특성에 의해 영향을 받을 수 있는데 특정 유전자에 영향을 미치는 특성으로써 여러 개의 패스웨이(pathway) 등 고려할 수 있기 때문이다. 샘플 영역 또한 알지 못하는 여러 특성에 의해 영향을 받을 수 있다. 만약 샘플을 특정부위에서의 조직이라고 할 때, 각 부위는 여러 가지 요인들에 의해서 서로 관련성이 발생할 수 있기 때문이다.

Meeds 등 (2006)에서는 쌍자료들이, 각 행이나 열은 한 개 이상의 숨겨진 특성들과의 관계로 표현이 된다고 생각하고 따라서 자료인 $\boldsymbol{X}$는 $\boldsymbol{UWV}^T$로 분해될 수 있다고 가정한다. 여기서 $\boldsymbol{U}, \boldsymbol{V}$는 이진행렬이고 $\boldsymbol{W}$는 가중치행렬이다. 자료를 이렇게 분해하는 것을 이진행렬분해(binary matrix factorization)이라고 부른다. 이들은 분해된 두 이진행렬인 $\boldsymbol{U}, \boldsymbol{V}$에 인도부페 프로세스 모형을 가정한 비모수 모형을 제안했고, 디지트(digit)자료와 유전자발현자료 등의 예제들을 통해서 이러한 모형을 이용한 쌍자료의 추론이 효과적임을 보였다.

5.2. 네트워크 자료분석

네트워크 자료란, 각 네트워크에 참여한 참여자들과 그들간에 연결고리가 있는지 여부가 주어져 있는 자료이다. 소셜네트워크 자료를 예를 들어보면, 네트워크 참여자는 소셜 네크워크 서비스 가입자들이고 그들간의 연결고리는 사용자 간에 친구관계에 대한 것이라 볼 수 있다.

총 참여자가 N명이라 한다면 네트워크 자료는 $N \times N$ 행렬에 i번째 참여자와 j번째 참여자간에 연결여부에 따라 행렬의 (i, j)원소의 값이 1또는 0이 되는 자료형태이다. 네트워크 자료의 생성 바탕에는 각 연결고리가 연결될지에 대한 확률 값이 내재되며 그 확률에 따라 네트워크 자료가 발현되었다고 볼 수 있다. 각 연결고리에 대한 확률은 참여자의 속성이 서로 잘 맞는지에 따라 다른 값을 가지게 될 것이라 가정한다. 예를 들어 참여자1은 축구취미 속성을 가지고 있고, 참여자2는 농구취미 속성을 가지고 있고, 참여자3은 서예취미 속성을 가지고 있을때, 참여자1과 참여자2 간에 친구관계가 될 확률은 높을 것이지만, 참여자1과 참여자3 간에 친구관계에 대한 확률은 낮을 것이다.

연결고리에 대한 확률을 위해서는 각 참여자들의 특성여부를 나타내는 이진행렬이 필요하고, 각 특성간에 확률에 어떤 영향을 줄지에 대한 계수행렬도 필요하다. $\boldsymbol{\mu}$는 $N \times N$ 행렬로서 각 연결고리에 대한 확률이고, $\boldsymbol{Z}$는 $N \times K$ 행렬로서 모든 참여자의 특성여부를 나타내는 이진행렬이며, $\boldsymbol{W}$는 $K \times K$ 행렬로서 각 특성 간에 연결에 영향을 끼치는 계수에 대한 행렬이다. 이를 일반화 선형모형으로 나타내기 위한 연결함수는 다음과 같다 (Foulds, 2014).

$$\boldsymbol{\mu} = g^{-1}(\eta), \quad \eta = \boldsymbol{ZWZ^T}.$$

위 모형에서 $\boldsymbol{Z}$의 분포에 인도부페 프로세스를 가정할 수 있다. 이러한 모형은, 소셜네트워크서비스에서 친구 추천의 프로세스에 적용할 수 있다. 즉, 두 참여자 간에 연결되려는 속성이 강하지만 연결되어 있지 않은 경우에 해당 참여자를 친구 추천 목록 띄우는 방식으로 이용할 수 있다.

5.3. 독립성분분석

독립성분분석(Independent component analysis; ICA)은 관측된 자료가 서로 독립인 은닉요인(hidden

source)들의 선형결합으로 표현되었다고 가정하는 모형이다. 각각의 관측치가 D차원이라고 할 때, n개의 관측치 $\boldsymbol{Y} = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)^T$를 다음과 같은 형태로 표현할 수 있다:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{G} + \boldsymbol{\epsilon},$$

여기서 $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$, $\boldsymbol{x}_i \in \mathbb{R}^K$는 서로 독립인 은닉요인들이고, $\boldsymbol{G} = (\boldsymbol{g}_1, \ldots, \boldsymbol{g}_k)^T$는 은닉요인들의 선형결합으로 자료를 표현하기 위한 계수 행렬이다. 일반적으로 $\boldsymbol{\epsilon} = (\boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_n)^T$, $\boldsymbol{\epsilon}_i \overset{iid}{\sim} N(0, \sigma_\epsilon^2 \boldsymbol{I}_D)$를 가정하며, 은닉요인 $\boldsymbol{x}_i$에는 다양한 분포를 가정할 수 있다.

위와 같은 독립성분모형에서 은닉요인들의 차원 K의 선택에 유연함을 주면서 베이지안 방식의 분석을 진행하기 위해 인도부페 프로세스를 적용할 수 있다. $\boldsymbol{X} = \boldsymbol{Z} \otimes \boldsymbol{V}$으로 표현하면, $\boldsymbol{Z}$가 인도부페 프로세스를 따르고 $\boldsymbol{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)^T$는 임의의 분포를 따른다고 생각할 수 있다. 이 때, 인도부페 프로세스의 특징으로 인해 은닉요인의 개수 K는 제한되지 않고 자료의 설명에 실제로 사용되는 활성화 된 요인들의 개수는 확률적으로 유한하게 정해지게 된다. 이진행렬 $\boldsymbol{Z}$의 성분 z_{ik}는 i번째 관측치의 설명에 k번째 은닉요인이 사용되는지의 여부를 말해주고, 행렬 $\boldsymbol{V}$의 성분 v_{ik}는 그 때 k번째 은닉요인에 곱해지는 계수로 해석된다.

Knowles와 Ghahramani (2007)는 이러한 모형을 유전자발현 자료를 분석하는데 적용하였다. 그들은 172개의 유전자($n = 172$)와 17개의 조직($D = 17$)에 대하여 유전자발현수준을 나타낸 자료를 이용하였다. 인도부페 프로세스로 이루어진 이진행렬 $\boldsymbol{Z}$는 표현되지 않는 유전자를 선택하는 역할, 행렬 $\boldsymbol{V}$는 활성화 된 유전자의 발현 정도를 나타내는 역할로 해석될 수 있다. 이 외에도 인도부페 프로세스로 이루어진 독립성분분석은 제한되지 않은 요인들의 결합형태로 표현된 자료를 분석하는 분야에 다양하게 응용될 수 있다.

6. 결론

본 논문에서는 인도부페 프로세스의 이론과 그 응용에 대해서 소개하였다. 인도부페 프로세스는 복잡한 자료 구조를 모형화 할 수 있다는 점에서 많은 분야에서 관심을 가지고 있는 비모수 베이지안 모형이다. 또한 인도부페 프로세스는 디리크레 프로세스와 마찬가지로 특성의 개수가 추론을 통해 자연스럽게 추정되므로, 모형선택의 문제를 해결하며 따라서 모형에 유연성을 부여할 수 있다는 장점을 가진다. 논문에서 소개한 가우시안 선형모형과 그 알고리즘은 인도부페 프로세스 적용할 수 있는 모형의 일부분일 뿐이며, 다양한 문제 상황에 따라 인도부페 프로세스를 이용한 모형을 고려하고 그에 적합한 알고리즘을 개발할 수 있다. 따라서 인도부페 프로세스를 이용한 모형은 많은 가능성을 가지고 있으며 그 연구 범위 또한 무궁무진하다고 본다. 본 논문이 인도부페 프로세스를 처음 접하는 국내 연구자에게 작은 도움이 되길 바라며, 이러한 기회를 통해 앞으로 인도부페 프로세스에 대한 연구가 국내 연구자들에 의해 지속되기를 바란다.

References

Doshi-Velez, F., Miller, K. T., Gael, J. V. and Teh, Y. W. (2008). Variational inference for the Indian buffet process.

Doshi-Velez, F., Knowles, D., Mohamed, S. and Ghahramamni, Z. (2009). Large Scale Nonparametric Bayesian inference: Data Parallelisation in the Indian Buffet Process, *Advances in Neural Information Processing Systems*.

Doshi-Velez, F. and Ghahramani, Z. (2009). Accelerated sampling for the Indian buffet process, *Proceedings of the 26th Annual International Conference on Machine Learning. ACM*.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems, *The Annals of Statistics*, 209–230.

Foulds, J. R. (2014). Latent Variable Modeling for Networks and Text: Algorithms, Models and Evaluation Techniques Ph.D., Thesis, Department of Computer Science, University of California, Irvine.

Griffiths, T. and Ghahramani, Z. (2006). Infinite latent feature models and the Indian buffet process, *In Advances in Neural Information Processing Systems*, **18**, 475–482.

Griffiths, T. and Ghahramani, Z. (2011). The indian buffet process: An introduction and review, *The Journal of Machine Learning Research*, **12**, 1185–1224.

Knowles, D. and Ghahramani, Z. (2007). Infinite sparse factor analysis and infinite independent components analysis, *Independent Component Analysis and Signal Separation*, 381–388.

Meeds, E., Ghahramani, Z., Neal, R. and Roweis, S. (2006). Modeling dyadic data with binary latent factors, *Advances in Neural Information Processing Systems*, 977–984.

Ren, L., Wang, Y., Carin, L. and Dunson, D. (2011). The kernel beta process, *Advances in Neural Information Processing Systems*, 963–971.

Paisley, J. W., Blei, D. M. and Jordan, M. I. (2012). Stick-breaking beta processes and the Poisson process, *International Conference on Artificial Intelligence and Statistics*.

Pitman, J. (1996). Random discrete distributions invariant under size-biased permutation, *Advances in Applied Probability*, 525–539.

Pitman, J. and Yor, M. (1997). The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator, *The Annals of Probability*, 855–900.

Teh, Y. W., Gorur, D. and Ghahramani, Z. (2007). Stick-breaking construction for the Indian buffet process, *International Conference on Artificial Intelligence and Statistics*.

Teh, Y. W. and Gorur, D. (2009). Indian buffet processes with power-law behavior, In *Advances in Neural Information Processing Systems*, 1838–1846.

Ten, L., Wang, Y., Dunson, D. and Carin, L. (2011). The kernel beta process, *Advances in Neural Information Processing Systems*, 963–971.

Thibaux, R. and Jordan, M. I. (2007). Hierarchical beta processes and the Indian buffet process, *International Conference on Artificial Intelligence and Statistics*.

Williamson, S., Orbanz, P. and Ghahramani, Z. (2010). Dependent Indian buffet processes, *International Conference on Artificial Intelligence and Statistics*, 924–931.

Zhou, M., Yang, H., Sapiro, G., Dunson, D. and Carin, L. (2011). Dependent hierarchical beta process for image interpolation and denoising, *International Conference on Artificial Intelligence and Statistics*, 883–891.

인도부페 프로세스의 소개: 이론과 응용

이영선[a] · 이경재[a,1] · 이광민[a] · 이재용[a] · 서진욱[b]

[a]서울대학교 통계학과, [b]서울대학교 컴퓨터공학부

(2015년 3월 16일 접수, 2015년 3월 30일 수정, 2015년 3월 30일 채택)

요 약

인도부페 프로세스는 유한개의 행과 무한개의 열로 이루어진 이진행렬의 분포와 관련된 확률과정이다. 무한특성모형을 유한개의 행과 무한개의 열로 이루어진 이진행렬을 이용해서 표현할 때, 이진행렬에 대한 사전분포로써 인도부페 프로세스가 이용될 수 있다. 본 논문에서는 인도부페 프로세스를 유한특성모형과 연관지어서 유도하는 방법을 소개하고, 베타프로세스와의 관련성을 간략히 설명한다. 실제 모형의 추론에 인도부페 프로세스가 이용되는 예제를 살펴보기 위해서 가우시안 선형모형에 인도부페 프로세스를 적용한 모형화 방법을 언급하고, 깁스표집 알고리즘, 막대자르기 알고리즘, 변분방법을 이용한 추론방법을 설명한다. 그리고 이 세 가지 알고리즘을 이용하여 이미지 자료를 분석하는데 적용해본다. 나아가 쌍자료 분석, 네트워크 분석, 독립성분 분석에서 인도부페 프로세스가 어떻게 이용될 수 있는지도 알아본다.

주요용어: 인도부페 프로세스, 잠재특성모형, 가우시안 선형모형, 깁스표집, 막대자르기 표현, 변분방법

이 논문은 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2011-0030811).

[1]교신저자: (151-747) 서울시 관악구 관악로 1, 서울대학교 통계학과. E-mail: leekjstat@gmail.com