Explaining Variational Inference for Dirichlet Process Mixture

Sunsik Kim

1 Dirichlet Process

군집 개수가 정해지지 않은 혼합분포(infinite mixture distribution)에서의 data generating process를 도입할 때 Dirichlet Process를 사용한다.

1.1 Finite Mixture Model

이를 설명하기 전에, 혼합분포 가정 하에서의 data generating process의 대략적인 흐름을 파악하기 위해 군집 수 K가 정해졌다고 하자. n개의 데이터가 K개의 정규분포가 혼합된 분포에서 생성됐다고 하면 밀도함수는 아래와 같이 적을 수 있다.

$$p\left(y_i|\mu_1,\ldots,\mu_K,\sigma_1^2,\ldots,\sigma_K^2,\pi_1,\ldots,\pi_k\right)=\sum_{k=1}^{K}\pi_k\mathcal{N}\left(y_i;\mu_k,\sigma_k^2\right),i=1,\ldots,n$$

이때 i번째 데이터가 속한 군집을 표현하기 위한 n개의 잠재변수 $\{c_i\}_{i=1}^n$를 도입하면 i번째 데이터 y_i의 분포를 $p(y_i|c_i=k)=\mathcal{N}(y_i|\mu_k,\sigma_k^2)$와 같이 적을 수 있다. 여기서 c_i는 모수가 $\{\pi_k\}_{k=1}^K$인 다항분포를 따르고, 여기에 켤레사전분포를 도입하면 $\{\pi_k\}_{k=1}^K$는 Dirichlet(α/K) 분포를 따른다(평평한 사전분포를 도입하기 위해 이와 같은 설정을 사용함).

1.2 Infinite Mixture Model

Dirichlet Process의 시작은 위의 설정에서 K를 한정하지 않는 것(즉, 무한정:$K\to\infty$)이다. 이때 유의해야 할 것은 K가 무한해도 잠재변수 $\{c_i\}_{i=1}^n$는 유한하고, 사실 개별 π_k의 값이 얼마인지보다는 y_i가 무슨 군집에 속하는지가 중요한 정보라는 것이다. 그래서 아래와 같이 $\{\pi_k\}_{k=1}^K$를 모형에서 integrate out 시켜서 $\{c_i\}_{i=1}^n$의 분포가 차원이 무한한 변수에 의존하지 않게 한다(유도 과정은 [2]).

$$\begin{aligned}p\left(c_1,\ldots,c_n|\alpha\right)&=\int p\left(c_1,\ldots,c_n|\pi_1,\ldots,\pi_k\right)\times p\left(\pi_1,\ldots,\pi_k|\alpha\right)d\pi_1\cdots d\pi_k\\&=\frac{\Gamma(\alpha)}{\Gamma(\alpha/K)^k}\int\prod_{k=1}^{K}\pi_k^{n_k+\alpha/K-1}d\pi_k\\&=\frac{\Gamma(\alpha)}{\Gamma(n+\alpha)}\prod_{k=1}^{K}\frac{\Gamma\left(n_k+\alpha/K\right)}{\Gamma(\alpha/K)}\end{aligned}$$

이 과정을 통해 무한차원인 $\{\pi_k\}_{k=1}^K$는 모형에서 사라졌지만 위 표현도 $K\to\infty$인 상황을 다루기 힘든 것은 마찬가지다. 그래서 $\{c_i\}_{i=1}^n$의 결합분포를 한번에 고려하기보다는 chain rule과 같이 c_i의 개별 분포를 하나씩 고려하는 방법을 선택한다. $\mathbf{c}_{-i}=\{c_1,\ldots,c_{i-1}\}$라 하고 $n_{-i,k}$를 $y_1,\ldots,y_{i-1}$ 중 k번째 군집에 할당된 데이터의 개수라 했을 때, [2]을 통해 아래와 같은 사실을 확인할 수 있다.

$$p\left(c_i=k|\boldsymbol{c}_{-i},\alpha\right)=\frac{n_{-i,k}+\alpha/K}{i-1+\alpha}\tag{1}$$

그러면 $K\to\infty\Rightarrow\alpha/K\to0$이 되어 i번째 데이터가 k번째 군집에 할당될 확률을 비로소 유한한 값으로 얻을 수 있게 된다.

이와 같은 설정에서 10개의 데이터를 획득했고 이 데이터는 3, 7개씩 두 군집(각각 $k = 1, 2$로 표현)에 할당되었다고 하자. 그러면 11번째 데이터 y_{11}가 군집 1, 2중 하나에 할당될 확률은 $\frac{10}{10+\alpha}$와 같이 구할 수 있다(즉, 기존 데이터가 이미 할당된 군집에 새로운 데이터가 할당될 확률이 $\frac{10}{10+\alpha}$). 따라서 아직 데이터가 할당되지 않은 어떤 군집에 11번째 데이터가 할당될 확률을 $\frac{\alpha}{10+\alpha}$와 같이 계산할 수 있다는 것을 확인할 수 있다.

보다 일반적으로 표현하면 $n + 1$번째 데이터 y_{n+1}에는 아래의 사건들 중 하나가 각각 아래와 같은 확률로 발생한다:

$$\begin{cases} n_{-n+1,k} > 0\text{인 군집 } k\text{에 할당됨} & \dfrac{n_{-n+1,k}}{n+\alpha} \\ \text{새로운 군집을 생성하고 그 군집에 할당됨} & \dfrac{\alpha}{n+\alpha} \end{cases} \tag{2}$$

(2)에서 자명하게 확인할 수 있는 것은

- n에 비해 α가 크면 클수록 새로운 군집이 생성될 확률이 높다.
- y_{n+1}이 기존 군집에 할당된다 하더라도 $n_{-n+1,k}$가 큰 군집에 할당될 확률이 높다.

1.3 Dirichlet Process(DP)

지금까지의 내용의 핵심인 (2)는 군집의 개수를 특정하지 않고 군집을 생성하면서 데이터를 군집에 할당하는 방법을 제시한다. 이러한 (2)를 핵심 동력으로 설정하여 혼합분포를 도입하는 data generating process가 DP다. DP는 두 가지 분포의 혼합분포에서 데이터가 생성되었다는 설정을 기저에 두는데, 이 설정은 Polya Urn scheme하의 표현을 통해 직관적으로 설명 가능하다.

미지의 분포 G를 따르는 확률변수 $\eta : \Omega \to \Theta$의 support를 domain으로 갖는 확률분포 G_0(base distsribution)이 있다고 하자. 그리고 흰 공이 무수히 담긴 항아리1, G_0에서 추출한 임의표본들의 값이 적힌 공들이 무수히 담긴 항아리2, 빈 항아리3이 있다고 하자. DP(α, G_0)로부터 값의 분포를 생성하는 과정은 아래와 같다.

1. (초기화) 항아리 1, 2에서 공을 하나씩 뽑아 2에서 뽑은 공에 적혀있는 값을 1에서 뽑은 흰 공에 적음. 항아리 2에서 뽑은 공은 다시 항아리 2에, 값을 적은 흰 공은 항아리 3에 넣음.
2. for i in range(1, n):
   (a) 항아리 1에서 흰 공을 뽑음.
   (b) $\dfrac{\alpha}{i+\alpha}$의 확률로 항아리 2에서, $\dfrac{i}{i+\alpha}$의 확률로 항아리 3에서 공을 뽑음.
   (c) 위에서 뽑은 공에 적혀있는 값을 흰 공에 적고 값을 적은 흰 공은 항아리 3으로, 뽑은 공은 원래 있던 항아리에 도로 집어넣음.

이 과정을 거쳐 항아리 3에 들어있는 n개의 공들에 쓰여 있는 값들의 분포를 DP(α, G_0)으로부터 생성한 분포라고 한다. 이런 점에서 DP는 확률분포에 대한 분포(measure on measure)라고 한다. 무엇보다 **DP에서 생성한 분포는 이산형 분포**임을 알 수 있는데, 항아리 2에서 뽑은 공들에 적혀있던 값들이 중복되어 항아리 3 속의 n개의 공들에 적혀있을 것이기 때문이다.

또한 이 장 첫 부분에서 DP는 두 가지 분포의 혼합분포에서 데이터가 생성되었다는 설정을 기저에 두고 있다고 했는데, 그 두 가지 분포란 항아리 2, 3을 지칭한다. $\delta_{\eta_i} : \Theta \to \{0, 1\}$이 $\eta_i \in A \Rightarrow \delta_{\eta_i}(A) = 1$를 만족하는 함수라고 한다면 위의 2-(b)에서 설명된 추출을 아래와 같은 혼합분포에서의 추출이라고 표현할 수 있다.

$$p(\eta_{n+1}|\eta_1, \ldots, \eta_n) = \frac{\alpha}{\alpha+n} G_0(\eta_{n+1}) + \frac{n}{\alpha+n} \left(\frac{1}{n} \sum_{i=1}^{n} \delta_{\eta_i}(\eta_{n+1}) \right) \tag{3}$$

2 Dirichlet Process Mixture

[1]의 notation을 적용하면, Dirichlet Process Mixture는 지수족 분포중 하나를 따르는 데이터 x_n의 모수 η_n이 DP로부터 추출된 이산형 분포를 따르게 하여 x_n이 해당 지수족 분포의 혼합분포를 따르게 하는 모형이다. 즉, 모형의 계층구조를 아래와 같이 적을 수 있다.

$$
\begin{aligned}
G \mid \{\alpha, G_0\} &\sim \mathrm{DP}(\alpha, G_0) \\
\eta_n \mid G &\sim G \\
X_n \mid \eta_n &\sim p(x_n \mid \eta_n)
\end{aligned}
$$

여기서 $p(x_n \mid \eta_n)$은 지수족 분포다. η_n은 특정 군집의 모수 값이고 이 값이 이산형 분포 G에서 추출되기 때문에 η_n이 가질 수 있는 값의 가지수는 유한하다. 다시 말하면 X_n은 지수족 분포인 $p(x_n \mid \cdot)$의 finite mixture distribution을 따르게 된다는 것이다. 구체적인 추정 방법을 살펴보기 전에 [1]에서 도입한 DP의 다른 characterization인 stick-breaking construction을 이해해야 한다.

2.1 Stick-breaking construction

사실 (3)을 이미 뽑힌 값들이 형성한 분포와 아직 뽑히지 않은 값들이 형성한 분포의 선형결합으로 볼 수 있다. 즉, 원소별로 분포를 구성하는 비중이 다를 뿐, (3)을 Θ내 모든 원소들이 구성하는 분포의 선형결합으로 표현할 수 있다는 것이다. 이때 Θ가 무한집합이라 Θ내 모든 원소들의 선형결합을 표현하기 위해서는 모든 항의 합이 1인 수열이 필요하다. 이 수열을 구성하는 로직을 stick-breaking construction이라고 한다.

$V_i \in (0,1),\ \forall i$라 하자. 처음엔 길이가 1인 막대기의 v_1만큼을 떼어낸다. 그렇게 얻은 막대기의 길이 $\pi_1(v_1)$는 v_1일 것이다. 다음으론 남은 막대기 $(1-v_1)$의 v_2만큼을 떼어낸다. 그렇게 얻은 막대기의 길이 $\pi_2(v_1, v_2)$는 $(1-v_1)v_2$일 것이다. 한번만 더 해보면 남은 막대기 $(1-v_1)(1-v_2)$의 v_3만큼을 떼어낸다. 그렇게 얻은 막대기의 길이 $\pi_3(v_1, v_2, v_3)$는 $(1-v_1)(1-v_2)v_3$일 것이다. 이런 식으로 계속해서 막대기를 떼어내면 총합이 1인 수열을 만들 수 있게 된다.

정리하면, $\mathrm{Beta}(1,\alpha)$에서 $\{V_i\}_{i\in\mathbb{N}}$를 계속 생성한 후 아래와 같이 $\{\pi_i\}_{i\in\mathbb{N}}$을 계산했을 때 stick-breaking construction은 $\sum_{i=1}^{\infty}\pi_i = 1$임을 암시한다. 따라서 이는 G_0를 따르는 확률변수들의 수열 $\{\eta_i\}_{i\in\mathbb{N}}$로 정의한 atomic function $\{\delta_{\eta_i}\}_{i\in\mathbb{N}}$들에 의한 mixture distribution의 계수로 사용될 수 있다. 그러므로 DP에서 추출한 확률분포 G를 아래와 같은 무한급수로 표현할 수 있음을 알 수 있다.

$$
\pi_i(\mathbf{v}) = v_i \prod_{j=1}^{i-1}(1-v_j)\,, \quad G = \sum_{i=1}^{\infty}\pi_i(\mathbf{v})\delta_{\eta_i} \tag{4}
$$

2.2 Dirichlet Process Mixture(DPM)

2.2.1 Data generating process

G에서 x_i의 모수를 추출한다는 것은 $\pi_i(\mathbf{v})$의 확률로 δ_{η_i}에서 모수를 추출한다는 것인데, δ_{η_i}는 η_i에서만 정의된 함수이므로 여기서 어떤 값을 생성하면 항상 η_i일 것이다. 이는 다시 말하면 π_i의 확률로 모수가 η_i라는 것이므로 모수가 $\pi_1, \pi_2, \ldots$인 다항분포에서 지시변수를 뽑아 그 지시변수가 가리키는 군집의 모수를 x_i의 모수로 삼는 것과 동일하다. 따라서 stick breaking construction 하에서 DPM의 data generating process는 최종적으로 아래와 같다.

1. Draw $V_i \mid \alpha \sim \mathrm{Beta}(1,\alpha)$ and $\eta_i \mid G_0 \sim G_0,\ i = 1, 2, \ldots$
2. For the nth data point:
   (a) Draw $Z_n \mid \mathbf{v} \sim \mathrm{Mult}(\pi_1(\mathbf{v}), \pi_2(\mathbf{v}), \ldots)$
   (b) Draw $X_n \mid z_n \sim p(x_n \mid \eta_{z_n})$

2.2.2 Model specification

[1]의 표현에 친숙해지기 위해 초모수 $\theta = (\alpha, \lambda)$가 주어졌을 때의 주변가능도를 분해해보면 아래와 같다 (여기서 α는 DP의 concentration parameter고 λ는 데이터의 분포에서 사용되는 초모수임. α에 Gamma prior를 부여해서 확률변수 처리하는 것도 당연히 가능).

$$\begin{aligned}
\ln p(\mathbf{X}|\alpha,\lambda) &= \int q(\mathbf{W}) \ln p(\mathbf{X}|\alpha,\lambda) \mathrm{d}\mathbf{W} \\
&= \int q(\mathbf{W}) \ln \frac{p(\mathbf{X}|\alpha,\lambda) p(\mathbf{W}|\mathbf{X},\alpha,\lambda) q(\mathbf{W})}{p(\mathbf{W}|\mathbf{X},\alpha,\lambda) q(\mathbf{W})} \mathrm{d}\mathbf{W} \\
&= \int q(\mathbf{W}) \ln \frac{p(\mathbf{X},\mathbf{W}|\alpha,\lambda) q(\mathbf{W})}{p(\mathbf{W}|\mathbf{X},\alpha,\lambda) q(\mathbf{W})} \mathrm{d}\mathbf{W} \\
&= \int q(\mathbf{W}) \ln \frac{p(\mathbf{X},\mathbf{W}|\alpha,\lambda)}{q(\mathbf{W})} \mathrm{d}\mathbf{W} + \int q(\mathbf{W}) \ln \frac{q(\mathbf{W})}{p(\mathbf{W}|\mathbf{X},\alpha,\lambda)} \mathrm{d}\mathbf{W} \\
&\geq \int q(\mathbf{W}) \ln \frac{p(\mathbf{X},\mathbf{W}|\alpha,\lambda)}{q(\mathbf{W})} \mathrm{d}\mathbf{W} \left(\triangleq \mathrm{ELBO}[q(\mathbf{W})] \right)
\end{aligned}$$

여기서 잠재변수들의 집합 $\mathbf{W}$은 아래와 같이 구성되어 있다.

$$\mathbf{W} \quad \begin{cases} \mathbf{V} & : \text{막대 길이 생성자들의 집합. 각 원소는 } \mathrm{Beta}(1,\alpha)\text{에서 생성됨.} \\ \boldsymbol{\eta} & : \text{막대 하나하나에 대응되는 군집의 모수가 담긴 집합. 각 원소는 } G_0\text{에서 생성됨.} \\ \mathbf{Z} & : \text{막대 길이들이 모수인 다항분포에서 추출한 군집 할당자들의 집합.} \end{cases}$$

이에 따라, N개의 데이터가 있을 때 ELBO는 아래와 같이 분해된다.

$$\begin{aligned}
\mathrm{ELBO}[q(\mathbf{W})] &= \mathrm{E}_q[\ln p(\mathbf{X},\mathbf{W}|\alpha,\lambda)] - \mathrm{E}_q[\ln q(\mathbf{W})] \\
&= \mathrm{E}_q[\ln p(\mathbf{X},\mathbf{V},\boldsymbol{\eta},\mathbf{Z}|\alpha,\lambda)] - \mathrm{E}_q[\ln q(\mathbf{V},\boldsymbol{\eta},\mathbf{Z})] \\
&= \mathrm{E}_q[\ln p(\mathbf{V}|\alpha)] + \mathrm{E}_q[\ln p(\boldsymbol{\eta}|\lambda)] + \mathrm{E}_q[\ln p(\mathbf{Z}|\mathbf{V})] + \mathrm{E}_q[\ln p(\mathbf{X}|\mathbf{Z})] - \mathrm{E}_q[\ln q(\mathbf{V},\boldsymbol{\eta},\mathbf{Z})] \\
&= \mathrm{E}_q[\ln p(\mathbf{V}|\alpha)] + \mathrm{E}_q[\ln p(\boldsymbol{\eta}|\lambda)] + \sum_{n=1}^{N} \left(\mathrm{E}_q[\ln p(Z_n|\mathbf{V})] + \mathrm{E}_q[\ln p(x_n|Z_n)] \right) - \mathrm{E}_q[\ln q(\mathbf{V},\boldsymbol{\eta},\mathbf{Z})]
\end{aligned}$$

일반적으로 n개의 데이터에 대한 모수가 k개인 다항분포는 아래와 같이 적을 수 있다.

$$\frac{n!}{x_1! \cdots x_k!} \pi_1^{x_1} \cdots \pi_k^{x_k}$$

여기서 $n = 1$이라면 이 분포는 $\pi_1^{x_1} \cdots \pi_k^{x_k}$로 정리됨을 확인할 수 있다. 따라서 $q(Z_n = i) = \pi_i$과 같이 쓸 수 있고, 그렇기 때문에 $n = 1$인 경우엔 다항분포를 지시변수에 관해 표현할 수 있다.

데이터를 하나씩 고려한 다항분포를 지시함수에 대해 표현하는 $p(Z_n|\mathbf{V})$의 경우도 이와 마찬가지고, 특히 π_i가 막대 길이 생성자 $\{V_j\}_{j=1}^{i}$에 의해 표현되므로 $p(Z_n|\mathbf{V})$를 $\mathbf{V}$에 관해 표현할 수 있다. $Z_n = k$일 때 (4)에 나타난 막대 길이 생성 패턴을 보면 k보다 작은 인덱스를 갖는($\mathbf{1}[Z_n > i]$) 막대 길이 생성자는 1에서 그만큼을 뺀 값이 곱해지고, 정확히 k인 생성자는 그대로, k보다 큰 생성자는 곱해지지 않는다. 이를 종합해보면 $\prod_{i=1}^{\infty}(1-V_i)^{\mathbf{1}[k>i]} V_i^{\mathbf{1}[k=i]} V_i^{0*\mathbf{1}[k<i]}$와 같다. 따라서 아래와 같이 적는다.

$$p(Z_n|\mathbf{V}) = \prod_{i=1}^{\infty} (1-V_i)^{\mathbf{1}[Z_n>i]} V_i^{\mathbf{1}[Z_n=i]}$$

2.2.3 Truncated stick-breaking representation

현재까지의 설정에서는 막대 길이 생성자(v)들의 개수가 무한해서 막대(π)의 개수도 무한하다. 이는 모수의 개수가 무한한 다항분포를 의미하는데, 여기서 지시변수 z를 추출하는 것은 불가능하다. 따라서 적어도 근사분포 q에서는 막대의 개수를 일정 T로 제한하자는 생각을 하게 되고, 이것이 truncated stick-breaking representation 이다.

$q(v_T = 1) = 1$로 두면 막대가 생성되는 원리에 의해 π_{T+1}부터는 계속 0이 되어 막대가 T개만 생성되게 된다. 그러면 지시변수 Z_n이 $T+1$번째 막대부터는 가리킬 수가 없으므로 $q(z_n > T) = 0$이 된다. 이렇게 함으로써 $\mathrm{E}_q[\ln p(Z_n|\mathbf{V})]$가 아래와 같이 정리된다.

$$
\begin{aligned}
\mathrm{E}_q[\ln p(Z_n|\mathbf{V})] &= \mathrm{E}_q\left[\ln\left(\prod_{i=1}^{\infty}(1-V_i)^{\mathbf{1}[Z_n>i]}V_i^{\mathbf{1}[Z_n=i]}\right)\right] \\
&= \sum_{i=1}^{\infty}\left\{q(z_n>i)\,\mathrm{E}_q[\ln(1-V_i)] + q(z_n=i)\,\mathrm{E}_q[\ln V_i]\right\} \\
&= \sum_{i=1}^{T}\left\{q(z_n>i)\,\mathrm{E}_q[\ln(1-V_i)] + q(z_n=i)\,\mathrm{E}_q[\ln V_i]\right\} \, (\because q(z_n>T)=0)
\end{aligned}
$$

References

[1] David M. Blei and Michael I. Jordan. Variational inference for dirichlet process mixtures. *Bayesian Anal.*, 1(1):121–143, 03 2006.

[2] Yuelin Li, Elizabeth Schofield, and Mithat Gönen. A tutorial on dirichlet process mixture modeling. *Journal of Mathematical Psychology*, 91:128 – 144, 2019.