

Dirichlet Processes

Emin Orhan
eorhan@bcs.rochester.edu

August 8, 2012

Introduction: A [Dirichlet process](#) (DP) is a distribution over probability distributions. We generally think of distributions as defined over numbers of some sort (real numbers, non-negative integers etc.), so at first it may seem a little exotic to talk about distributions over distributions. If you feel that way at this point, one obvious but very reassuring fact that we would like to point out is that probability theory still applies to these objects, however exotic they may seem initially. So, as we will see shortly, it is quite possible to gain considerable insight into the properties of these objects without having a clear intuition as to what they “look like”.

Suppose that G is a probability distribution over a measurable space Θ (if this is too technical, you can think of G as assigning real numbers between 0 and 1 –probabilities– to subsets of Θ). Now, G is a probability distribution over Θ and a DP is a distribution over all such distributions. A DP is parametrized by a [concentration parameter](#) α and a [base measure](#) or [base distribution](#) H (more on these later). It is barely informative to just say that something is a distribution over something else (compare: a normal distribution is a distribution over real numbers). We would like to know the properties of that distribution. So, what exactly does it mean to say that G is distributed according to a DP with parameters α, H , or more concisely: $G \sim \text{DP}(\alpha, H)$? It means the following:

$$(G(T_1), G(T_2), \dots, G(T_K)) \sim \text{Dirichlet}(\alpha H(T_1), \alpha H(T_2), \dots, \alpha H(T_K)) \quad (1)$$

for any finite partition (T_1, T_2, \dots, T_K) of Θ . Or in English, the probabilities that G assigns to any finite partition of Θ follow a Dirichlet distribution (not to be confused with a Dirichlet process) with parameters $\alpha H(T_1), \alpha H(T_2), \dots, \alpha H(T_K)$. This implicit definition might be too abstract (I, for one, do not claim to get a clear picture of what a DP looks like from this definition), but we will give more constructive (and hopefully more intuitive) characterizations of a DP shortly. For now, as alluded to in the first paragraph, here are some important properties of a DP that you can derive using basic probability theory without having the slightest idea what a DP looks like. These properties follow straightforwardly from Equation 1 and the properties of the Dirichlet distribution and it is very instructive to prove them even if you do not have a solid intuition at the moment as to what a DP looks like:

Mean: The mean of a DP is its base measure: $E[G] = H$ or equivalently $E[G(T)] = H(T)$ for any $T \subset \Theta$. On average, then, distributions drawn from a DP look like H .

Posterior distribution: If $G \sim \text{DP}(\alpha, H)$ and $\theta_1, \dots, \theta_N \sim G$, then the posterior over G is also a DP:

$$G|\theta_1, \dots, \theta_N \sim \text{DP}(\alpha + N, \frac{1}{\alpha + N}(\alpha H(\theta) + \sum_{i=1}^N \delta(\theta = \theta_i))) \quad (2)$$

where $\delta(\theta = \theta_i)$ is a delta function centered at θ_i . In other words, DP is the conjugate prior for arbitrary distributions over a measurable space Θ .

Deriving the following properties requires a little more sophisticated mathematics (see Sudderth (2006) and Görür (2007) and the references cited therein). But they all rely on constructive ideas and there are metaphors or physical analogies associated with each of them, which will hopefully help you build some intuitions about DPs.

Posterior predictive distribution (Pólya urn scheme): What is the posterior predictive distribution of a DP? In other words, if $G \sim \text{DP}(\alpha, H)$ and $\theta_1, \dots, \theta_N \sim G$, what is the posterior predictive distribution for a new item: $p(\theta_{N+1}|\theta_1, \dots, \theta_N) = \int p(\theta_{N+1}|G)p(G|\theta_1, \dots, \theta_N)dG$?

To answer this question, imagine that you generate an infinite sequence $\{\theta_i\}_{i=1}^{\infty}$ (with $\theta_i \in \Theta$) according to the following procedure:

$$\theta_1 \sim H \quad (3)$$

$$\theta_{N+1}|\theta_1, \dots, \theta_N \sim G_N(\theta_{N+1}) = \frac{\alpha H(\theta_{N+1}) + \sum_{i=1}^N \delta(\theta_{N+1} = \theta_i)}{\alpha + N} \quad (4)$$

The physical analogy associated with this construction is as follows. Suppose that you are drawing colored balls from an urn, called urn G (hence the name “urn scheme”). θ_i represents the color of the i -th ball you drew from the urn. For each ball you draw from the urn, you replace that ball and add another ball with the same color to the urn. Note that this induces a “rich gets richer” property on the frequencies of colors inside the urn: as you draw more and more balls with a certain color, it becomes more and more likely to draw a ball with that color at the following iterations. To add diversity, you also occasionally draw a ball from a different urn, H , replace it and add a ball of the same color to the original urn G .

What does this all have to do with the posterior predictive distribution of a DP? It turns out that if you continue the process described in Equations 3-4 *ad infinitum*, G_N converges to a random discrete distribution G which is itself distributed according to $\text{DP}(\alpha, H)$:

$$\lim_{N \rightarrow \infty} G_N \rightarrow G \sim \text{DP}(\alpha, H) \quad (5)$$

Furthermore, the samples $\{\theta_i\}_{i=1}^N$ constitute samples from the random limit distribution G and Equation 4 gives the posterior predictive distribution for a new observation θ_{N+1} : $p(\theta_{N+1}|\theta_1, \dots, \theta_N) = \int p(\theta_{N+1}|G)p(G|\theta_1, \dots, \theta_N)dG$. Thus, this construction gives you the posterior predictive distribution of a DP!

Chinese restaurant process (CRP): The Pólya urn scheme makes it clear that a DP imposes a clustering structure on the observations θ_i : there is a strictly positive probability that two balls drawn from the urn will have the same color, hence the observations, or the balls in the Pólya urn scheme, can be grouped by their colors. CRP makes this clustering structure explicit. More specifically, let us index distinct colors in the Pólya urn scheme by integers. Let c_i denote the color index of the i -th ball drawn from the urn. Note that if two balls i and j have the same color, then $c_i = c_j$. Also note that c_i is different from θ_i . $\theta_i \in \Theta$ is the color of the ball, whereas c_i is the integer index of that color. Suppose that you drew N balls from the urn and so far have encountered K distinct colors. It follows from Equation 4 that:

$$p(c_{N+1}|c_1, \dots, c_N) = \frac{\alpha}{\alpha + N} \delta(c_{N+1} = K + 1) + \sum_{k=1}^K \frac{n_k}{\alpha + N} \delta(c_{N+1} = k) \quad (6)$$

where n_k is the number of balls with color index k (make sure that you understand how Equation 6 follows from Equation 4). So, the color of the next ball will either be the same as one of the existing colors (with probability proportional to the number of balls of that color) or a new color unseen among the first N balls (with probability proportional to α). Therefore, CRP is a straightforward consequence of the Pólya urn scheme. But the statisticians had to come up with an entirely new metaphor for it! According to

this metaphor, as the name suggests, you are to imagine a Chinese restaurant with an infinite number of tables each with infinite seating capacity. When the customer $N + 1$ arrives at the restaurant, she either sits at one of the K occupied tables with probability proportional to the number of customers sitting at that table, n_k , or with probability proportional to α she sits at a new, presently unoccupied table (table $K + 1$). The importance of this process is that it turns out that CRP provides a very useful representation when doing inference in Dirichlet process mixture models (DPMM), i.e. mixture models with DP priors on mixture components (more on DPMMs later).

Stick-breaking construction: We still do not have a clear idea as to what a random draw from a DP looks like. We will have a very clear idea once we learn about the stick-breaking construction. So, suppose that you generate an infinite sequence of “weights” $\{\pi_k\}_{k=1}^{\infty}$ according to the following procedure:

$$\beta_k \sim \text{Beta}(1, \alpha) \quad (7)$$

$$\pi_k = \beta_k \prod_{l=1}^{k-1} (1 - \beta_l) \quad (8)$$

An infinite sequence of weights $\pi = \{\pi_k\}_{k=1}^{\infty}$ thus generated is said to be distributed according to a **GEM (Griffiths-Engen-McCloskey) process** with concentration parameter α ($\pi \sim \text{GEM}(\alpha)$). Now consider the following discrete random probability distribution:

$$G(\theta) = \sum_{k=1}^{\infty} \pi_k \delta(\theta = \zeta_k) \quad \text{where} \quad \zeta_k \sim H \quad (9)$$

It can be shown that $G \sim \text{DP}(\alpha, H)$. Furthermore, all draws from a DP can be expressed as in Equation 9. The physical analogy associated with Equations 7-8 is the successive breaking of a stick of unit length. You first break a random proportion β_1 of the stick. The length of this piece gives you the first weight, π_1 . Then, you break a random proportion β_2 of the *remaining* stick. The length of this second piece gives you the second weight, π_2 and so on. Note that as k gets larger and larger, the stick lengths, or the weights, will tend to get smaller and smaller. The concentration parameter α determines the distribution of the stick lengths. For small α , only the first few sticks will have significant lengths, the remaining sticks having very small lengths. For large α , on the other hand, the stick lengths will tend to be more uniform. This can be seen by noting that $E[\beta_k] = 1/(1 + \alpha)$, hence for small α , the random breaking proportions β_k will tend to be large and the entire length of the stick will be “expended” very rapidly; whereas for large α , the proportions will tend to be smaller and it will take longer to expend the entire length of the stick.

We now know what draws from a DP “look like”: they all look like the infinite discrete distribution in Equation 9. In fact, we can even picture them. Figure 1 shows random draws from DPs with different concentration parameters α and base measures H . The base measure H determines where the “atoms” η_k will tend to be located and, as discussed in the previous paragraph, α controls the weight distribution of the atoms, with smaller α leading to sparser weight distributions.

Dirichlet process mixture models: Where do we use DPs? Are they purely a theoretical curiosity or do they have any practical applications? The main application of DPs is within the context of mixture models. In this context, a DP-distributed discrete random measure is used as a prior over the parameters of mixture components in a mixture model. The resulting model is called a **Dirichlet process mixture model** (DPMM). Let us first describe the DPMM mathematically:

$$G \sim \text{DP}(\alpha, H) \quad (10)$$

$$\theta_i | G \sim G \quad (11)$$

$$x_i | \theta_i \sim F(\theta_i) \quad (12)$$

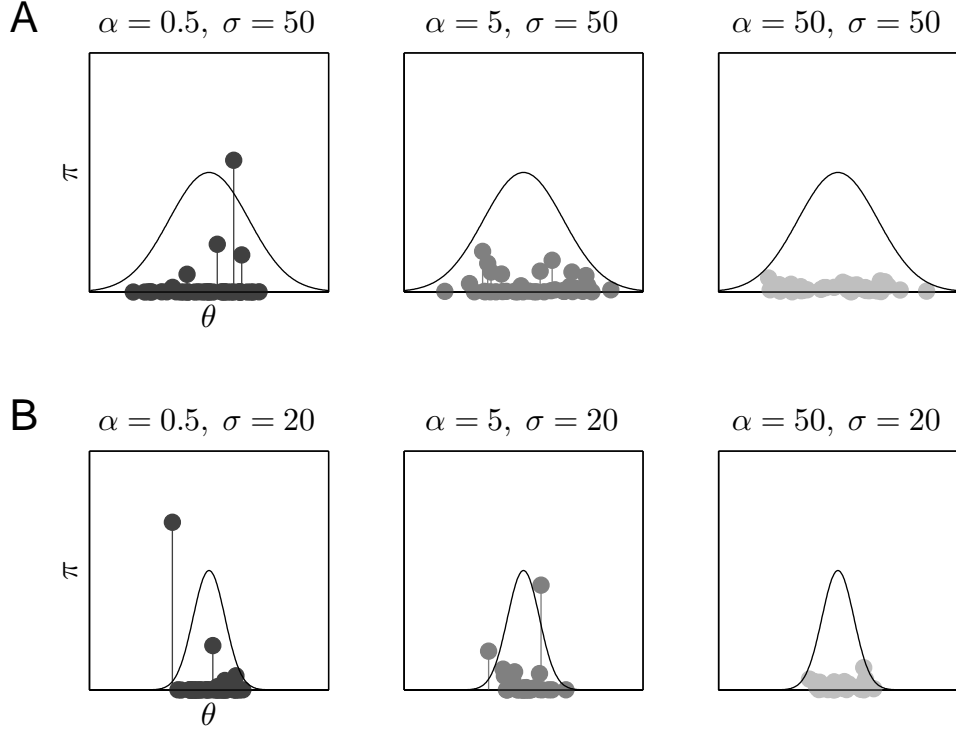


Figure 1: Random draws from stick breaking processes with different parameters. (A) The base measure, H , is a normal distribution with zero mean and standard deviation 50. (B) The base measure, H , is a normal distribution with zero mean and standard deviation 20. The base measure is shown by the solid black lines in each plot. Different columns correspond to different concentration parameters. Note that the collection of stems in each plot constitutes a single random draw, G , from a DP with parameters α and H .

where x_i are the observable variables or data that we want to model. θ_i are the parameters of the mixture component that x_i belongs to and F represents the distribution of mixture components (e.g. Gaussian in a mixture of Gaussians). θ_i can be a single parameter, such as the mean parameter of the Gaussian components in a mixture of Gaussians or a vector of multiple parameters, such as the mean *and* precision of the Gaussian components in a mixture of Gaussians. Note that when two data points x_i and x_j belong to the same component, their component parameters will be identical, i.e. $\theta_i = \theta_j$. Figure 2 shows a graphical representation of the DPMM together with an illustration of the generative process defining the model. In the example show in this figure, $\theta = (\mu, \tau)$ is a vector representing the mean and precision of Gaussian components. We use a uniform base measure over μ and a gamma base measure over τ . H is then simply a product of these two base measures. G is an infinite discrete distribution over such (μ, τ) vectors, as in Equation 9. For each data point i , we draw an atom $\theta_i = (\mu_i, \tau_i)$ from G (which may be identical for different data points due to the discreteness of G) and then generate x_i by a random draw from a Gaussian distribution with mean μ_i and precision τ_i . The DPMM is sometimes called [the infinite mixture model](#). This is due to the fact that the DPMM can be shown to be mathematically equivalent to a finite mixture model when the number of components goes to infinity. It is very instructive to think about exactly what distinguishes a DPMM from a finite mixture model. It is sometimes claimed that the difference between a DPMM and a finite mixture model is that the former computes a whole posterior distribution over the number of components in the dataset, whereas the latter assumes a fixed number of components and hence does not compute a distribution over them. This claim is true only if one uses a non-Bayesian method (e.g. expectation-maximization or EM) for inference in the finite mixture

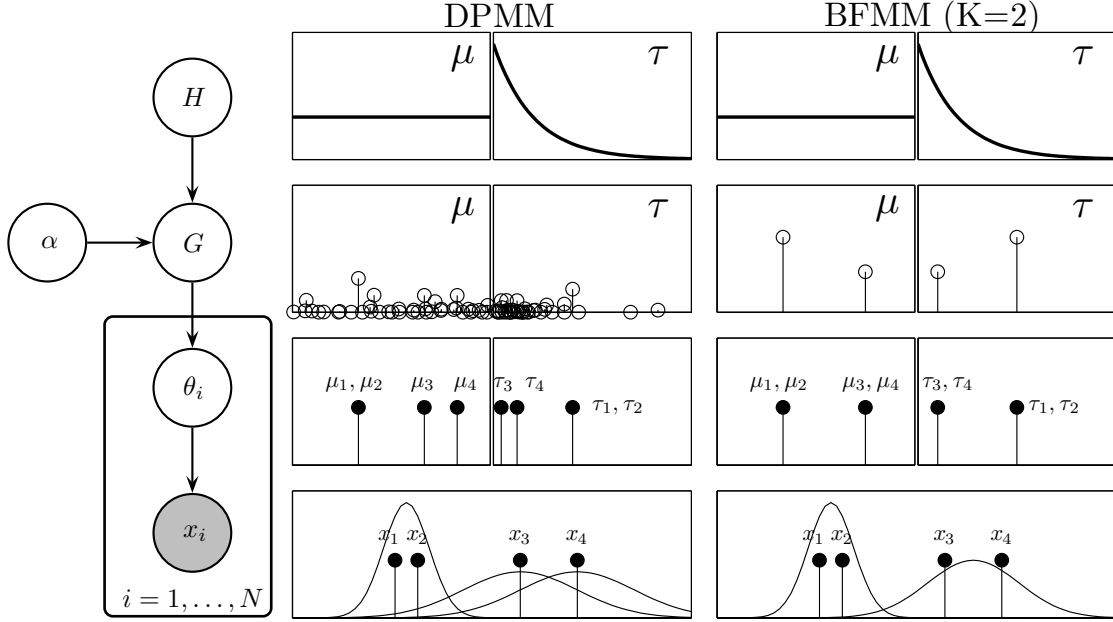


Figure 2: This figure is adapted from Sudderth (2006). Both the DPMM and BFMM have a common structure represented by the graphical model shown on the left. This graphical model uses the plate notation, where the nodes inside the plate are meant to be replicated N times. The shaded node represents the observable variables, i.e. the data points x_i . The other variables are latent or unobservable. The remaining plots illustrate the generative processes defining the two models. Each row illustrates the generation of variables at the corresponding level in the graphical model on the left. The only difference between the two models is in the variable G . For the DPMM, G is a discrete distribution with an infinite number of “atoms”; for the BFMM with K components, it is a discrete distribution with K atoms. In the example shown here, the DPMM uses three components to generate the four data points represented by x_i s, the BFMM uses 2 components. $\theta_i = (\mu_i, \tau_i)$ represents the component parameters, i.e. the mean and precision parameters of Gaussian components, for data point x_i . The distributions at the bottom row illustrate the Gaussian components from which the data points x_i were drawn.

model. However, a fully [Bayesian finite mixture model](#) (BFMM), where inference over all the variables in the model is treated in a Bayesian way, also computes a posterior distribution over the number of components in the given dataset, rather than assuming a single, fixed number of components. The real difference between a DPMM and a BFMM rather lies in the upper bound on the number of components that they use. Whereas the DPMM assumes no upper bound on this number, a BFMM sets a strict, finite upper bound K on the number of components. In terms of the Equations 10-12 describing the DPMM, the only difference between the DPMM and a BFMM comes from the discrete distribution G over the component parameters (compare the middle and the right columns in Figure 2). In the DPMM, G is distributed according to a Dirichlet process with base measure H and concentration parameter α and can be expressed as a weighted sum of an infinite number of discrete atoms (Equation 9). In a finite mixture model with K components, on the other hand, G is a weighted sum of a finite number of atoms only (reflecting the assumption that the data were generated by a finite number of components):

$$G(\theta) = \sum_{k=1}^K \pi_k \delta(\theta = \zeta_k) \quad (13)$$

As in the DPMM, the atoms ζ_k are drawn independently from a base measure H . The weights π , on the

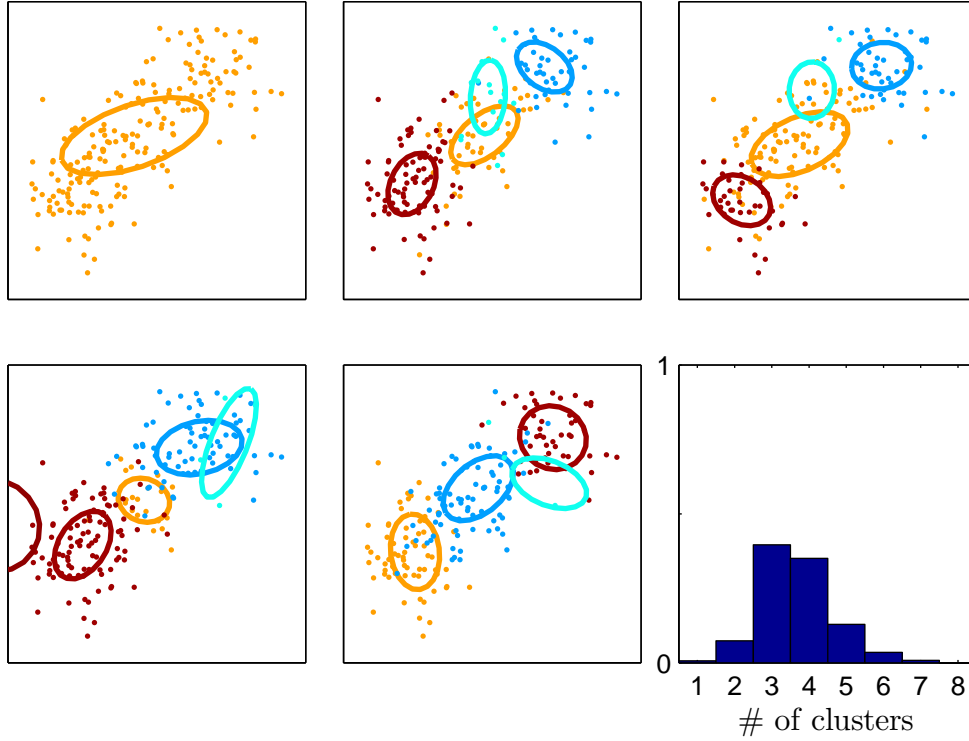


Figure 3: DPMM applied to a two-dimensional dataset. The lower-right plot shows the posterior distribution over the number of components inferred by the DPMM. The remaining plots show the components (represented by the contours) and the color-coded assignment of data points to the components at different iterations of an MCMC sampling algorithm used for inference in the DPMM.

other hand, are drawn from a symmetric Dirichlet distribution with concentration parameters α/K :

$$\pi \sim \text{Dirichlet}(\alpha/K, \dots, \alpha/K) \quad (14)$$

whereas the weights π in the DPMM are distributed, as you will remember, according to a $\text{GEM}(\alpha)$ process.

Figure 3 shows the application of the DPMM to a two-dimensional dataset. Although, we have not addressed the problem of inference for the DPMM in this note, inference is generally performed via Markov chain Monte Carlo (MCMC) sampling algorithms (although there are quite efficient variational inference algorithms for the DPMM as well). For the example shown in Figure 3, we have used an efficient Gibbs sampling algorithm for performing inference (see Sudderth (2006) or Görür (2007) for details). The first five plots in this figure show the components and the color-coded assignment of data points to the components at different iterations of the Gibbs sampling algorithm. The lower-right plot shows the posterior distribution over the number of components.

References

- [1] Görür, D. (2007). Nonparametric Bayesian Discrete Latent Variable Models for Unsupervised Learning. *Doctoral Thesis, Max Planck Institute for Biological Cybernetics*.
- [2] Sudderth, E.B. (2006). Graphical Models for Visual Object Recognition and Tracking. *Doctoral Thesis, MIT*.