NSF-CBMS Regional Conference Series
in Probability and Statistics
Volume 9

NONPARAMETRIC BAYESIAN INFERENCE

Peter Müller and Abel Rodriguez

Institute of Mathematical Statistics
Beachwood, Ohio

American Statistical Assocation
Alexandria, Virginia

Conference Board of the Mathematical Sciences

Regional Conference Series
in Probability and Statistics

Supported by the
National Science Foundation

The production of the *NSF-CBMS Regional Conference Series in Probability and Statistics* is managed by the Institute of Mathematical Statstics: Patrick Kelly, IMS Production Editor; and Elyse Gustafson, IMS Executive Director.

Library of Congress Control Number: 2012941332

International Standard Book Number 978-0-940600-82-9

Printed in the United States of America

Contents

Preface

These notes arose out of a short course at UC Santa Cruz in summer 2010. Like the course, the notes provide an overview of some popular Bayesian nonparametric (BNP) probability models. The discussion follows a logical development of many commonly used nonparametric Bayesian models as generalizations of the Dirichlet process (DP) in different directions, including Pólya tree (PT) models, species sampling models (SSM), dependent DP (DDP) models and product partition models (PPM). The selection of topics is subjective, simply driven by what the authors are familiar with. As a result, some useful and elegant classes of models such as normalized random measures with random increments (NRMIs) are reviewed only briefly.

We focus on BNP models for random probability measures, keeping for example a discussion of Gaussian process priors to only a brief review in the introductory chapter. Also, we put the emphasis on developing models, rather than a discussion of BNP data analysis for important statistical inference problems. However, some data analysis is introduced by way of short examples and in an introductory chapter. Inference for BNP models often requires computation-intensive implementations. Keeping the focus on models, we decided against a discussion of computational algorithms at much length. The only exception are posterior simulation schemes for Dirichlet process (DP) and DP mixture models. Finally, we do not discuss asymptotic results. These are important and non-trivial. Excellent recent reviews appear in the monograph by Ghosh and Ramamoorthi (2003) and a review paper by Ghoshal (2010).

The notes start with an overview of discussed models, and a bit more, in Chapter 1. This overview serves the purpose of clarifying the relationships of the many models that are introduced later. We hope that the introduction will put the proliferation of BNP models in some perspective. But it also creates some repetition when some of the material that is already included in this initial overview is re-introduced later, reflecting also the nature of this manuscript as lecture notes. Figure 1.2 can serve as a one-figure overview of the rest of the notes. Chapter 2 motivates the upcoming long list of models by discussing some typical applications of BNP in data analysis. Then, Chapters 3 through 8 introduce some of the most popular BNP in more detail.

Finally, a word about notation. We generically use $p(\cdot)$ to indicate a probability model. The arguments clarify which model is meant. We use specific names for probability models only when the probabilty model itself is a random variable. For example $p(G)$ refers to a BNP model for the random probability measure G. We use boldface to distinguish vectors from scalars only when needed, for example $\mathbf{x} = (x_1, \ldots, x_n)$, but usually do not use bold face when no confusion arises. Sometimes we use (x_i) to indicate a vector $(x_i,\ i = 1, \ldots, n)$, when the range of the indices is clear from the context. Finally, we use notation like $\mathsf{N}(x \mid \mu, \sigma)$ to indicate a normal distributed random variable x with moments (μ, σ). By a slight abuse of notation we use $\mathsf{N}(x \mid \mu, \sigma)$ also for the corresponding p.d.f.

Chapter 1

Introduction

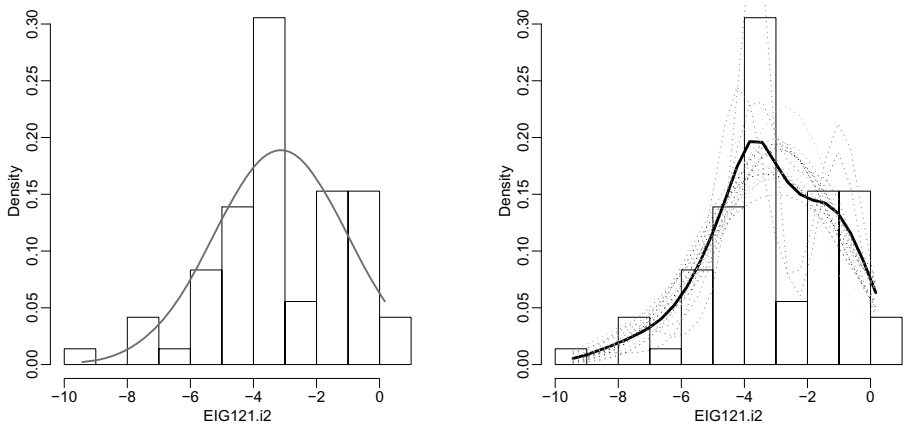
1.1. Definition

All models are wrong, but some are useful. Many statisticians know and appreciate G.E.P. Box's comment on statistical modeling (Box, 1979). Often the choice of the final inference model is a compromise of an accurate representation of the experimental conditions, a preference for parsimony and the need for a practicable implementation. However, these competing goals are not always honestly spelled out, and the resulting uncertainties are not fully described.

Over the last 20 years a powerful inference approach that allows us to mitigate some of these limitations has become increasingly popular. Bayesian nonparametric (BNP) inference allows us to acknowledge uncertainy about an assumed model while maintaining a practically feasible inference approach. We could take this feature as a pragmatic characterization of BNP as flexible prior probability models that generalize traditional models by allowing for positive prior probability for a very wide range of alternative models, while centering the prior around a parsimonious traditional model. A more formal definition of BNP is as probability models on infinite dimensional parameter spaces, such as functional spaces.

Example 1 (Density estimation) *Consider a simple random sample $y_i \sim F$ i.i.d., $i = 1, \ldots, n$, from some unknown distribution F. Bayesian inference requires that the model be completed with a prior for the unknown F in the sampling model. One could proceed by restricting F to a normal location family, $F = \mathsf{N}(\theta, 1)$. The model F is indexed by a finite dimensional parameter vector θ and the model is completed with a prior probability model for the finite dimensional θ. We are back to parametric Bayesian inference. Figure 1.1a shows the resulting inference conditional on an assumed random sample $\boldsymbol{y}$. Naturally, inference about the unknown F is restricted to the assumed normal location family and does not allow for multimodality or skewness. In contrast, a BNP model would proceed with a prior probability model $p(F)$ for the unknown distribution. Figure 1.1b contrasts the parametric inference with the flexible BNP inference under a Dirichlet process mixture prior.*

In Example 1 the infinite dimensional random quantity is an unknown distribution. Alternatively, the infinite dimensional quantity might be the unknown mean function $f(\cdot)$ in a regression problem, a response surface, a spectral density, or perhaps an autoregressive mean function in a nonparametric time series model. In the rest of these notes we will mostly focus on problems where the infinite dimensional quantity is an unknown probability measure $F(\cdot)$, as in example 1. The reason for this focus is simply tradition; most BNP models in the recent literature consider random probability measures.



(a) Parametric model (b) Nonparametric model

FIG 1.1. *Example 1. Inference on the unknown distribution F under a parametric model and nonparametric model. The histogram are the observed data $y_i \sim F$.*

1.2. BNP Models for Random Probability Measures

Figure 1.2 summarizes in a stylized diagram the relationships between some of the most popular BNP models for random probability measures (RPM). The diagram highlights the central role of the popular Dirichlet process (DP) model, which arises as a special case of several other BNP models. The diagram serves as a short outline of these notes. After a brief introductory definition of the models in the rest of this Introduction we will in the following chapters discuss some of the models in more detail.

1.2.1. Species Sampling Models

Species sampling models (SSMs) define an RPM $p(G)$ indirectly, by considering a predictive rule for $x_{n+1} \mid x_1, \ldots, x_n$ in a random sample $x_i \sim G$. For a discrete probability measure G, random sampling includes a positive probability for ties among the x_i. We use groups of tied sequence elements x_i to define clusters. These clusters will play a prominent role in the upcoming discussion. It is helpful to introduce some related notation. Let k_n denote the number of unique values ("species") among $(x_1, \ldots, x_n)$, let $x_j^\star$, $j = 1, \ldots, k_n$, denote the unique values, and let n_{nj} denote the number of x_i equal to the j-th unique value $x_j^\star$. Finally, $\boldsymbol{n} = \{n_{n1}, \ldots, n_{nk_n}\}$ characterizes the cluster sizes of the partition created by the ties. We drop the subindex $_n$ when the sample size n is understood from the context.

Definition 1 (Pitman, 1996) *An exchangeable sequence of r.v.'s $x_1, x_2, \ldots$ is a* species sampling sequence *(SSS) if $x_1 \sim G_0$ where G_0 is a non-atomic measure and*

$$x_{n+1} \mid x_1, \ldots, x_n \sim \sum_{j=1}^{k_n} p_j(\boldsymbol{n}_n)\delta_{x_j^\star} + p_{k+1}(\boldsymbol{n}_n)G_0, \tag{1.1}$$

where $p_j(\boldsymbol{n}_n) \geq 0$ and $\sum_{j=1}^{k_n+1} p_j(\boldsymbol{n}_n) = 1$.

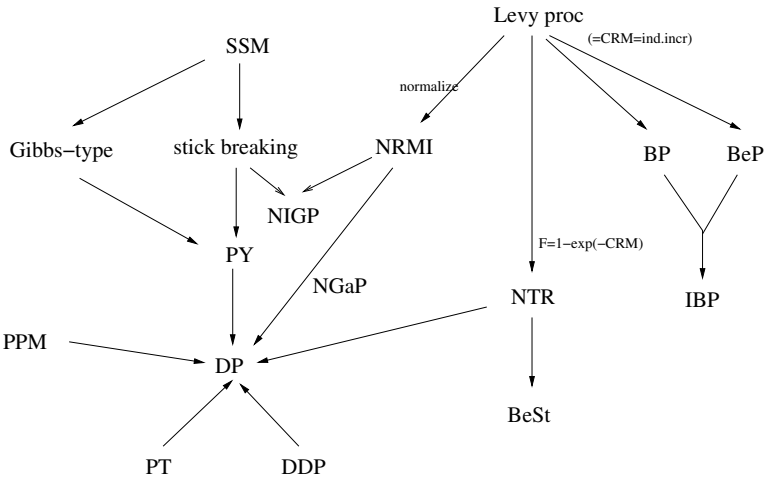


FIG 1.2. *Popular BNP models. An arrow from model A to B indicates that B is a special case (or variation) of A. Details of the models are discussed in the text. The graph includes species sampling models (SSM), the Pitman-Yor process (PY), the Dirichlet process (DP), the product partition model (PPM), Pólya trees (PT), dependent DP (DDP), normalized random measures with independent increments (NRMI), the normalized gamma process (NGaP), the normalized inverse Gaussian process (NIGP), neutral to the right processes (NTR), the Beta-Stacey process (BeSt), the beta process (BP), the Bernoulli process (BeP) and the Indian buffet process (IBP). The annotations are not exhaustive. For example, the NIGP is not the only NRMI that defines a SSM.*

The sequence of weights $\{p_j(\boldsymbol{n})\}$ is known as predictive probability function (PPF). Any SSS can be characterized by the PPF $\{p_j(\boldsymbol{n})\}$ and G_0. The opposite is not true. The critical property is the exchangeability of the sequence. Not every family of weights with $\sum_{j=1}^{k} p_j(\cdot) = 1$ and $p_j(\cdot) \geq 0$ characterizes an SSS because for an arbitrary choice of $\{p_j(\cdot)\}$ the implied sequence x_i might not be exchangeable.

At this moment the reader might wonder how the SSS defines a prior probability model for an unknown probability measure. The SSS defines a random probability measures as the de Finetti measure in the corresponding representation of the exchangeable sequence as a hierarchical model.

Theorem 1.2.1 (Pitman, 1996) *(x_i) is an SSS if and only if $x_i \sim G$, i.i.d., for*

$$G(\cdot) = \sum_{h=1}^{\infty} p_h \delta_{\widetilde{x}_h}(\cdot) + RG_0, \tag{1.2}$$

for some sequence of positive random variables (p_h) and R such that $1 - R = \sum_{i=h}^{\infty} p_h \leq 1$, $(\widetilde{x}_h)$ is a random sample from G_0, and $\{p_h\}$ and $\{\widetilde{x}_h\}$ are mutually independent.

In other words, a constructive definition of SSMs is possible as a discrete RPM with point masses at i.i.d. locations $\widetilde{x}_h$ and an arbitrary probability model for the weights p_h, subject only to $\sum p_h \leq 1$. Unless otherwise stated we will always assume $R = 0$. In contrast to the alternative definition through a PPF, any choice of G_0 and (p_h) will do. Exchangeability of the sequence x_i is already ensured by construction.

The Pitman-Yor (PY) process (Ishwaran and James, 2001; Pitman, 1995; Pitman and Yor, 1997) is a SSM with PPF

$$p_j(\boldsymbol{n}) \propto \begin{cases} (b+ka) & j = k+1 \\ n_{nj} - a & j = 1, \ldots, k, \end{cases} \tag{1.3}$$

for $0 \leq a < 1$ and $b > 0$. It is also known as the two-parameter Poisson Dirichlet process. Exploiting the construction of a SSM as a discrete RPM we can characterize the PY by a base measure G_0 and the law for the weights p_h in (1.2). The distribution of the weights p_h for the PY process can be described by a sequence of independent Beta random variables

$$p_h = v_h \prod_{l<h}(1-v_l), \qquad v_h \sim \mathsf{Beta}(1-a, b+ha), \tag{1.4}$$

for $h = 1, 2, \ldots$. We write $\mathsf{PY}(a, b, G_0)$ for a PY random probability measure with the joint distribution of the weights indexed by (a, b) and the locations of the point masses generated as i.i.d. sample from a base measure G_0.

Ishwaran and James (2001) refer to (1.4) as a stick breaking construction. The name arises from picturing (1.4) as repeated breaking of a stick of initial length 1. The first weight $p_1 = v_1$ is a beta random fraction of the stick, p_2 is a beta random fraction of the remaining stick of length $(1-p_1)$, etc. Sethurman (1994) introduced the construction for the special case of $a = 0$, which defines a Dirichlet process (DP), $\mathsf{DP}(b, G_0)$. Here we encounter for the first time the DP model. The characterization of the DP as a SSM is one of its many alternative defining properties. One of the reasons for the wide use of the DP prior is the simple form of the implied PPF (1.1). Assume $x_i \sim G$, i.i.d. and $G \sim \mathsf{DP}(M, G_0)$. Then

$$x_{n+1} \mid x_1, \ldots, x_n \sim \begin{cases} \delta_{x_j^\star} & \text{with prob } \propto n_{nj} \\ G_0 & \text{with prob} \propto M, \end{cases} \tag{1.5}$$

i.e., (1.3) with $a = 0$ and $b = M$.

The simple form of (1.5) greatly simplifies posterior simulations when the BNP model is used for statistical inference. Indeed, exchangeability of the x_i implies that the same rule applies for the complete conditional probability $p(x_i \mid x_\ell, \ell \neq i)$. We will come back to the DP several times in the following review before we discuss it in more detail in Chapter 3.

Gnedin and Pitman (2006) and Lijoi *et al.* (2007b) define another special case of SSMs. They consider Gibbs type priors that are characterized by a PPF

$$p_j(\boldsymbol{n}) \propto \begin{cases} V_{n+1,k} \frac{n_{nj}-\sigma}{n} & j = 1, \ldots, k \\ V_{n+1,k+1} & j = k+1, \end{cases}$$

with $\{V_{n,k},\ k \leq n\}$ a sequence of coefficients with $V_{1,1} = 1$ and subject to $V_{n,k} = V_{n+1,k+1} + (n - k\sigma)V_{n+1,k}$, and $0 \leq \sigma < 1$. The model defines a variation of PY priors. Conditional on $k_{n+1} = k_n$ the conditional PPF remains the same as under a PY prior. Only the probability of a new species, i.e., $p(k_{n+1} = k_n + 1 \mid \boldsymbol{n}_n)$, changes.

1.2.2. Stick Breaking Prior

One of the characteristics of the SSM construction is the unlimited flexibility in defining the joint distribution of the weights p_h. Ishwaran and James (2001) exploit

this flexibility to propose stick breaking priors for RPMs by generalizing the beta distribution of the fractions v_h in the construction of the PY process.

They propose two generalizations. First, they allow the number of non-zero weights to be finite,

$$G(\cdot) = \sum_{h=1}^{H} p_h \delta_{\widetilde{x}_h}(\cdot)$$

for $H \leq \infty$. Second, the beta prior for the fractions v_h is replaced by $v_h \sim \mathsf{Beta}(a_h, b_h)$, independently, $h = 1, \ldots, H-1$. For $H < \infty$ we add $v_H = 1.0$ to ensure $\sum p_h = 1.0$. The locations of the point masses remain unchanged as $\widetilde{x}_h \sim G_0$, i.i.d.

Naturally the DP remains a special case, with $a_h = 0$, $b_h = b$ and $H = \infty$. Ishwaran and James (2001) propose the model $a_h = 0, b_h = b$ and $H < \infty$ as a natural simplification of the DP prior. We refer to it as the finite DP, $\mathsf{DP}_H(b, G_0)$. An alternative version of the truncated DP prior is the ϵ-DP of Muliere and Tardella (1998), see §3.4.

1.2.3. Product Partition Models

While not strictly a prior for a random probability measure, we include the product partition model (PPM) in this review because of the close connection with popular BNP models for random probability measures. We have already seen how the random clustering that is defined by the ties in an i.i.d. sample from a discrete distribution G can be useful to characterize a discrete random probability measures $G \sim p(G)$. In many applications of BNP models the investigators are not primarily interested in the random probability measure G itself, but rather focus on the induced clustering. It is therefore useful to consider probability models for random cluster arrangements.

We need a minimum of notation. Let $S = \{1, 2, \ldots, n\}$ index a set of experimental units. A partition or cluster arrangement of S is a family of subsets $\rho_n = \{S_1, \ldots, S_k\}$ with $\bigcup S_j = S$ and $S_{j_1} \cap S_{j_2} = \emptyset$ for $j_1 \neq j_2$. When ρ_n is treated as a random quantity we have a random partition $p(\rho_n)$. For example, any discrete probability model G implies a random partition $p(\rho_n)$ by grouping random samples $x_i \sim G$, $i = 1, \ldots, n$, by unique values, as $S_j = \{i : \ x_i = x_j^\star\}$. Here $x_j^\star$ denotes the j-th unique value. The $x_j^\star$ are indexed by order of appearance. The random partition $p(\rho_n)$ is determined by the probability masses in G. The same remains true when G is an unknown discrete random probability measure with prior $p(G)$, but this is not the only interesting class of random partition models $p(\rho_n)$.

Hartigan (1990) introduces the product partition models (PPM). A random partition $p(\rho_n)$ is called a product partition model if it can be written as a product

$$p(\rho_n) = \prod_{j=1}^{k} c(S_j)$$

of factors that depend on S_j only, $j = 1, \ldots, k$. Let y_i denote an outcome for the i-th experimental unit, let $y_j^\star = \{y_i;\ i \in S_j\}$ denote the outcomes arranged by clusters and let $\boldsymbol{y} = (y_1, \ldots, y_n)$ denote the entire data. The PPM combines the prior $p(\rho_n)$ with a sampling model $p(\boldsymbol{y} \mid \rho_n)$ that factors similarly and assumes

exchangeability within each cluster

$$p(\boldsymbol{y} \mid \rho_n) = \prod_{j=1}^{k} p(y_j^\star)$$

for an exchangeable model $p(y_j^\star)$.

Again we run into the DP model as a special case. The random partition induced by the ties in a random sample $x_i \sim G$ with DP prior $G \sim \mathsf{DP}(M, G_0)$ forms a PPM with

$$p(\rho_n) \propto \prod_{j=1}^{k_n} M\,\Gamma(n_{nj}).$$

Recall that $n_{nj} = |S_j|$ is the size of the j-th cluster.

1.2.4. Pólya Trees

Essentially, the Pólya tree (PT) model defines a RPM G as a random histogram. The bins are created by nested partitions of the desired sample space B. The random probabilities for each bin are products of (independent) conditional probabilities of each layer of the nested partition sequence.

Figure 1.3 illustrates the construction. The bins B_ϵ are indexed by binary se-

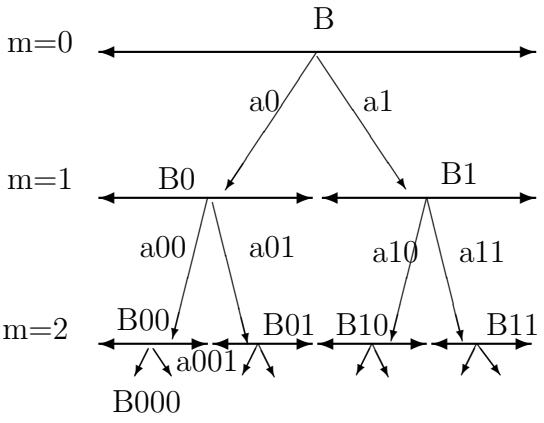


FIG 1.3. *PT prior for an RPM G. At level m of the nested partition sequence the sample space B is partitioned into $\{B_\epsilon\}$ indexed by binary sequences $\epsilon = \epsilon_1 \cdots \epsilon_m$ and defined by repeated splits into $B_\epsilon = B_{\epsilon 0} \cup B_{\epsilon 1}$.*

quences $\epsilon = e_1 \cdots e_m$ with $e_j \in \{0, 1\}$. The bins are created by nested partitions of the desired sample space B into $B = B_0 \cup B_1$, $B_0 = B_{00} \cup B_{01}$, etc. The random probabilities $G(B_\epsilon)$ are defined by the (independent) conditional probabilities $G(B_{e_1 \cdots e_j 0} \mid B_{e_1 \cdots e_j})$. Let $\epsilon = e_1 \cdots e_{j-1}$ and let $Y_{\epsilon 0} = G(B_{\epsilon 0 j} \mid B_\epsilon)$. The PT prior characterizes $p(G)$ as a prior probability model for all $Y_{\epsilon 0}$. It defines $p(G)$ by assuming

$$Y_{\epsilon 0} \sim \mathsf{Beta}(\alpha_{\epsilon 0}, \alpha_{\epsilon 1})$$

independently across ϵ and $Y_{\epsilon 1} = 1 - Y_{\epsilon 0}$. In short, the definition of an RPM is reduced to independent beta priors for the conditional probabilities in the nested

partition sequence. The PT is indexed by the partition sequence $\mathcal{B} = \{B_\epsilon\}$ and the set of beta parameters, $\mathcal{A} = \{\alpha_\epsilon\}$. We write $G \sim \mathsf{PT}(\mathcal{A}, \mathcal{B})$.

Again the DP arises as a special case when $\alpha_\epsilon = \alpha_{\epsilon 0} + \alpha_{\epsilon 1}$. For example, $\alpha_{\epsilon_1 \cdots \epsilon_m} = c2^{-m}$ implies a $\mathsf{DP}(c, G_0)$ with G_0 defined as the distribution with dyadic quantiles given by B_ϵ. Thus $G_0(B_0) = 0.5$, $G_0(B_{01} = 0.25)$, etc.

1.2.5. DDP

Many applications call for more than one random probability measure G. For example, the generic regression problem of predicting an outcome y conditional on a covariate x could be described as inference for the conditional distributions $G_x(\cdot) = p(y_i \mid x_i = x)$ for $x \in X$. When $p(y_i \mid x_i = x)$ is indexed by finitely many parameters we are back to parametric, possibly non-linear regression. However, when the investigator is unwilling or unable to restrict $p(y_i \mid x_i)$ to a parametric family, then the problem becomes one of inference for a family of random probability measures $\mathcal{G} = \{G_x,\ x \in X\}$, indexed by the covariates x. We thus need a BNP prior $p(G_x;\ x \in X)$ for the entire family. In the application to nonparametric regression as well as many other applications it is natural to require that G_x be dependent across x. Surely we would not expect G_x to change substantially for minor changes of x.

MacEachern (1999) introduced the class of dependent DP (DDP) priors to construct such prior models $p(\mathcal{G})$. In particular, the marginal distribution $p(G_x)$ remains a DP prior, $G_x \sim \mathsf{DP}(c, G_{0x})$. But the model allows for the desired dependence. Recall that the DP prior is a special case of a SSM. A DP random measure can therefore be written as an infinite discrete probabilty measure with independent locations for the point masses

$$G_x(\cdot) = \sum_h p_h \delta_{\widetilde{x}_{hx}}(\cdot). \tag{1.6}$$

We will discuss the DDP prior in more detail in Chapter 5. One important feature is the independence of the point mass locations $\widetilde{x}_{hx}$ across h. The DDP construction leaves the independence across h untouched, but adds dependence for $\widetilde{x}_{hx}$ across x to induce the desired dependence of G_x across x. The notation in (1.6) implies the use of common weights p_h across all G_x. This variation of DDP models is known as the common weight DDP. In general, the weights could have an additional $_x$ index, defining $G_x(\cdot) = \sum p_{hx} \delta_{\widetilde{x}_{hx}}(\cdot)$.

1.2.6. Completely Random Measures

A rich variety of BNP models are based on completely random measures (CRM) (Kingman, 1993). A random measure μ is a CRM when $\mu(B_1)$ and $\mu(B_2)$ are independent for any two non-overlapping measurable sets B_1, B_2 of some space X. The independence property implies in particular that μ can not be a probability measure, lest the restriction to total mass 1.0 induces dependence between $\mu(B_1)$ and $\mu(B_2)$.

As a consequence of the desired independence a CRM must always be discrete, i.e., it can be written as a sum of point masses. An alternative construction of CRMs that will turn out to be useful in the upcoming discussion is based on a

Poisson process. Let $N(\cdot)$ denote a Poisson process on $X \times R^+$ with intensity $\nu(\cdot)$. Then

$$\mu(A) \equiv \int_A \int_{R^+} sN(dx, ds)$$

for measurable $A \subset X$. In words, each point (x, s) of the Poisson process in $X \times R^+$ defines a locations x and weight s for a point mass of μ. The intensity $\nu(\cdot)$ is known as Levy intensity, which also features in the Levy-Khintchine representation for μ. For $X = R$, $\mu_x \equiv \mu((-\infty, x])$ is also known as increasing additive process or independent increments process.

CRMs are useful tools to define BNP priors for random probability measures. The simplest construction is to normalize a CMR to define

$$G \equiv \mu/\mu(X).$$

Regazzini *et al.* (2003) introduce such random probability measures as normalized random measures with independent increments (NRMI).

Here we run again into the DP prior. The original discussion of the DP in Ferguson (1973) discusses as an alternative defining property of the DP the construction as an NRMI, using a normalized version of a gamma process. The definition of the DP as a normalized gamma process immediately implies another useful characterization. Let $\boldsymbol{w} \sim \mathsf{Dir}(a_1, \ldots, a_k)$ denote a Dirichlet distribution for a random vector of weights $\boldsymbol{w}$. Recall that a Dirichlet random vector can be generated by normalized gamma random variables, as $w_i = x_i/(\sum_j x_j)$ for $x_i \sim \mathsf{Gamma}(\alpha_i, \theta)$, i.i.d. Let $\{A_1, \ldots, A_k\}$ denote a partition of the sample space. The nature of the DP as a normalized gamma process implies $(G(A_1), \ldots, G(A_k)) \sim \mathsf{Dir}(a_1, \ldots, a_k)$ with $a_j = \alpha G_0(A_j)$.

There is at least one other NRMI model that allows a similarly simple characterization. Lijoi *et al.* (2005) introduce the normalized inverse Gaussian process (NIGP) as an NRMI. Alternatively, the NIGP can be defined by the following property. For any partition $(A_1, \ldots, A_k)$ of the sample space,

$$(G(A_1), \ldots, G(A_k)) \sim \mathsf{NIG}(MG_0(A_1), \ldots, MG_0(A_k)),$$

where $\mathsf{NIG}(a_1, \ldots, a_k)$ denotes a normalized inverse Gaussian distribution. The NIG distribution is a parametric probability model for a (finite) vector of weights that add up to 1. The definition starts with the inverse Gaussian distribution, $\mathsf{IG}(\alpha, \gamma)$ with p.d.f.

$$p(x) \propto x^{-3/2} e^{-\frac{1}{2}\left(\frac{\alpha^2}{x} + \gamma^2 x\right) + \gamma\alpha},$$

where $x \geq 0$ and $\alpha > 0$. Now, let $x_j \sim \mathsf{IG}(a_j, 1)$, $j = 1, \ldots, k$, denote k independent inverse Gaussian random variables. The NIG is the distribution of the normalized values $w_j = x_j/\sum x_\ell$. We say $\boldsymbol{w} \equiv (w_1, \ldots, w_k) \sim \mathsf{NIG}(a_1, \ldots, a_k)$. Despite the name, the inverse Gaussian has no obvious relation with the normal distribution.

1.2.7. NTR Priors

Normalization is not the only mechanism to construct nonparametric Bayes models from CRMs. Another popular class of models based on CRMs are neutral to the right (NTR) priors. NTR priors are nonparametric priors for random distributions on the real line. Typical applications are to modeling event time distributions in survival analysis.

The defining property of NTR models are independent normalized increments. An RPM G is NTR if the normalized increments

$$G(\,(t_{i-1}, t_i]\,)/G(\,(t_{i-1}, \infty)\,),$$

$i = 1, \ldots, M$, are independent for any $t_0 < t_1 \ldots < t_M$. Docksum (1974, Theorem 3.1) shows that G is NTR if and only if its distribution function can be written as $1-\exp(-\mu(\infty, t])$ for some CRM μ on the real line with $\lim_{t\to\infty} \mu(0, t] = \infty$ almost surely. The DP is again a special case; Doksum (1974) shows that the DP is NTR and gives the specific CRM μ that defines the DP as NTR prior.

1.2.8. Indian Buffet Process

The Indian buffet process (IBP) defines a random binary matrix $\boldsymbol{Z}$, not (naturally) a random probability measure. The name is explained by the analogy to a large buffet in an Indian restaurant. Customers arrive at the buffet to select dishes. Let $Z_{ik} \in \{0, 1\}$ denote an indicator for the i-th customer selecting the k-th dish. The first customer selects a number k_1 of dishes. We index the dishes in the sequence of first selection. Thus the first customer, $i = 1$, selects dishes $k = 1, \ldots, k_1$. This defines $Z_{ik} = 1$, $i = 1$ and $k = 1, \ldots, k_1$. Let $K_i = \sum_{j\le i} k_j$ denote the number of distinct dishes selected by the first i customers. The next, $(i + 1)$-st customer selects or does not select some of the previously selected dishes, defining $Z_{i+1,k} \in \{0, 1\}$, $k = 1, \ldots, K_i$. In addition to dishes selected by previous customers the next customer selects a number k_{i+1} of new dishes $k = K_i + 1, \ldots, K_i + k_{i+1}$, defining $Z_{i+1,k} = 1$. We set $Z_{j,k} = 0$ for earlier customers, $j \le i$.

Let $\boldsymbol{Z}_i = [Z_{jk};\ j = 1, \ldots, i,\ k = 1, \ldots, K_i]$ denote the selections of the first i customers. The random matrix is defined by specifying the probability $p(Z_{i+1,k} = 1 \mid \boldsymbol{Z}_i)$ of the next customer choosing already earlier selected dishes and the distribution of the number of new dishes, $p(k_{i+1})$. Let $m_{-(i+1),k}$ denote the number of customers before $(i + 1)$ selecting dish k. We assume

$$p(Z_{i+1,k} = 1 \mid \boldsymbol{Z}_i) = \frac{m_{-(i+1),k}}{i+1}$$

and $k_{i+1} \sim \mathsf{Poi}\{\alpha/(i + 1)\}$. For n customers the process defines the random binary $(n \times K_n)$ matrix $\boldsymbol{Z}_n$, with a random number of columns K_n, $K_n \sim \mathsf{Poi}\,(\alpha \sum_{i=1}^{n} 1/i)$. In an alternative construction Thibaux and Jordan (2007) show that the IBP can be constructed by a Beta process and a Bernoulli process.

The IBP is useful as a prior model for (possibly overlapping) random subsets. Interpret Z_{ik} as an indicator for experimental unit i being in the $k-$subset. Then $S_k = \{i :\ Z_{ik} = 1\}$ denotes the k-th subset. We see the parallel to the random partition that is defined by, for example, the PPM. While the PPM defines a prior for a random partition $p(S_k;\ k = 1, \ldots, K)$ with $S_{k_1} \cap S_{k_2} = \emptyset$ and $\cup_k S_k = \{1, \ldots, n\}$, the IBP defines a prior $p(S_k;\ k = 1, \ldots, K)$ for a family of subsets with possibly overlapping subsets, and with $\cup_k S_k \subseteq \{1, \ldots, n\}$. In an application the experimental units and subsets could be, for example, proteins and different molecular pathways. A protein i could be part of multiple pathways, i.e., $i \in S_{k_1} \cap S_{k_2}$, and some proteins might not be in any pathway of interest, i.e., $\cup_k S_k \neq \{1, \ldots, n\}$.

1.3. BNP Models for Random Functions

The models that were introduced in §1.2 are priors $p(G)$ for RPMs. Recall the earlier definition of BNP as probability models on infinite dimensonsal spaces. Besides random distributions, another large number of BNP models defines priors $p(f)$ for random functions f. The main motivating applications is to priors for non-linear regression mean functions $f(\cdot)$.

1.3.1. Gaussian Process

We first discuss Gaussian processes as nonparametric priors $p(f)$ for a function $f(\cdot)$ on $\mathbb{X}$. Consider any finite collections of $n \geq 1$ points $x_1, \ldots, x_n \in \mathcal{X}$, and let $\boldsymbol{f} = (f(x_1), \ldots, f(x_n))$. A stochastic process $\{f(x) : x \in \mathbb{X} \subset \mathbb{R}^d\}$ is said to follow a Gaussian process with mean function $g(x)$ and symmetric covariance function $\gamma(x, x')$, denoted $f \sim \mathsf{GP}\{g(x), \gamma(x, x')\}$, if

$$\begin{pmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{pmatrix} \sim \mathsf{N}\left(\begin{pmatrix} g(x_1) \\ g(x_2) \\ \vdots \\ g(x_n) \end{pmatrix}, \begin{bmatrix} \gamma(x_1, x_1) & \gamma(x_1, x_2) & \cdots & \gamma(x_1, x_n) \\ \gamma(x_2, x_1) & \gamma(x_2, x_2) & \cdots & \gamma(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(x_n, x_1) & \gamma(x_n, x_2) & \cdots & \gamma(x_n, x_n) \end{bmatrix} \right) \tag{1.7}$$

or, more succinctly, $\boldsymbol{f} \sim \mathsf{N}(\boldsymbol{g}, \boldsymbol{\Gamma})$. In our notation we distinguish the random function $f(\cdot)$ versus the finite-dimensional vector $\boldsymbol{f}$. The collection of finite-dimensional distributions described above defines a proper stochastic process since it satisfies Kolmogorov's consistency conditions. Indeed, for any collection of measurable sets $A_1, \ldots, A_n$ the joint distribution $\nu_{(x_1,\ldots,x_n)}$ for $(f(x_1), \ldots, f(x_n))$ satisfies

$$\nu_{(x_1,\ldots,x_n)}(A_1, \ldots, A_n) = \nu_{(x_{\pi_1},\ldots,x_{\pi_n})}(A_{\pi_1}, \ldots, A_{\pi_n})$$

for any permutation $\pi_1, \ldots, \pi_n$ of the integers $\{1, \ldots, n\}$ and

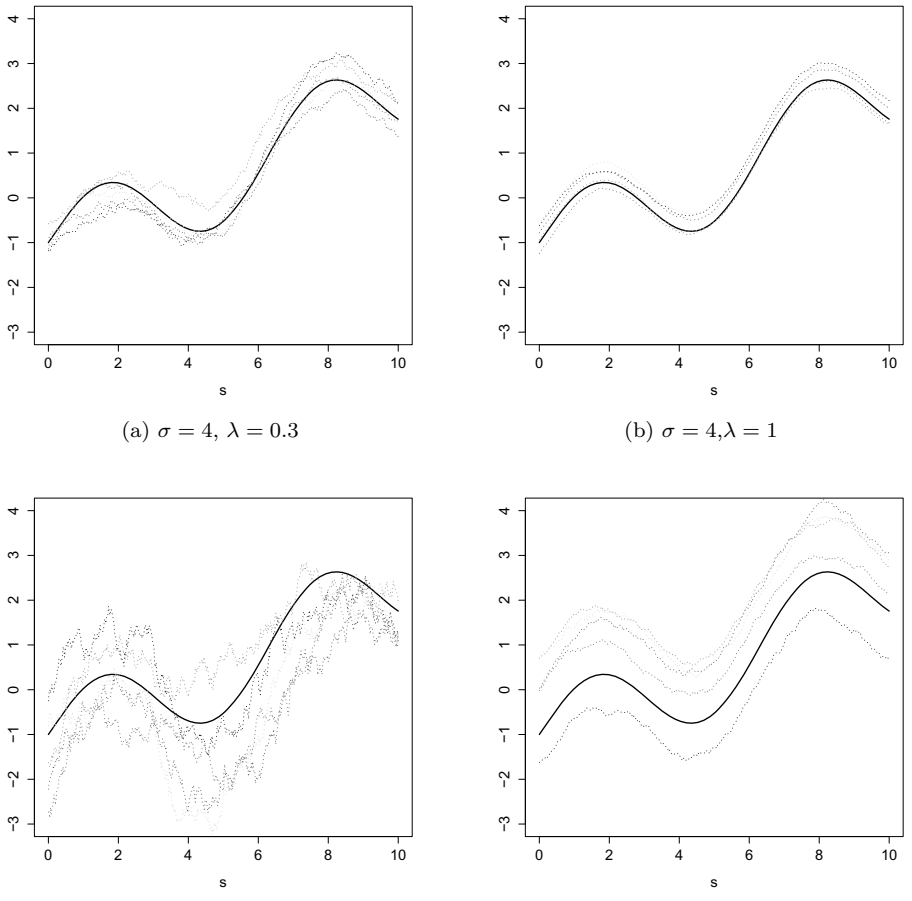
$$\nu_{(x_1,\ldots,x_{n-1},x_n)}(A_1, \ldots, A_{n-1}, \mathbb{R}) = \nu_{(x_1,\ldots,x_{n-1})}(A_1, \ldots, A_{n-1}).$$

Example 2 (Realizations from a Gaussian process) *Consider a Gaussian process on $\mathcal{X} = [0, 10]$ with mean function $g(x) = \sin(x) - \cos(x/4) + 0.15x$ and covariance function $\gamma(x, x') = \sigma^2 \exp\{-|x - x'|/\lambda\}$. For any collection of points, $x_1, \ldots, x_n \in \mathcal{X}$, we can obtain a realization from the stochastic process on these locations by sampling from the multivariate normal distribution (1.7). Figure 1.4 shows realizations from the process on a fine regular grid on $\mathcal{X}$, for different values of σ and λ. The simulations illustrate the effect of these two parameters on the realizations of the process; the range parameter λ controls the local variability, while σ controls the global variability in the realizations.*

In addition to controlling how close the realizations from the process are to the mean function, the covariance function also controls other important properties such as smoothness. For example, the exponential covariance function that we used in example 2 implies that realizations are almost surely not differentiable anywhere, hence the jagged look of the curves. A more detailed discussion of these issues can be found, for example, in Banerjee and Gelfand (2002).

One of the appealing features of the Gaussian process is its tractability. Given the values $\boldsymbol{f}_o = (f(x_1), f(x_2), \ldots, f(x_n))'$, predictions for the value of the function at new levels of the covariates

$$\boldsymbol{f}_p = (f(x_{n+1}), \ldots, f(x_{n+m}))'$$



(a) $\sigma = 4$, $\lambda = 0.3$ (b) $\sigma = 4$,$\lambda = 1$

(c) $\sigma = 100$, $\lambda = 0.3$ (d) $\sigma = 100$, $\lambda = 1$

FIG 1.4. *Realizations from a Gaussian process with mean function* $g(x) = \sin(x) - \cos(x/4) + 0.15x$ *and exponential covariance function* $\gamma(x, x') = \sigma^2 \exp\{-|x - x'|/\lambda\}$ *over a regular grid of 500 points on the interval* $[0, 10]$. *Each panel corresponds to a different combination of the parameters* σ *and* λ. *Within each panel, the solid line shows the mean function* g *and the dotted lines show five different realizations generated under the corresponding values of* λ *and* σ.

can be obtained by noting that the joint distribution for $\boldsymbol{f} = (\boldsymbol{f}_o', \boldsymbol{f}_p')'$

$$\begin{pmatrix} \boldsymbol{f}_o \\ \boldsymbol{f}_p \end{pmatrix} \sim \mathsf{N}\left(\begin{pmatrix} \boldsymbol{g}_o \\ \boldsymbol{g}_p \end{pmatrix}, \begin{bmatrix} \boldsymbol{\Gamma}_{oo} & \boldsymbol{\Gamma}_{op} \\ \boldsymbol{\Gamma}_{po} & \boldsymbol{\Gamma}_{pp} \end{bmatrix} \right),$$

where $\boldsymbol{g}_o$ and $\boldsymbol{g}_p$ denote the marginal means of $\boldsymbol{f}_o$ and $\boldsymbol{f}_p$, $\boldsymbol{\Gamma}_{oo}$, $\boldsymbol{\Gamma}_{pp}$ denote their marginal variance matrices, and $\boldsymbol{\Gamma}_{op} = \boldsymbol{\Gamma}_{po}'$ denotes the cross-covariance matrix. From this, standard results for the normal distribution yield

$$\boldsymbol{f}_p \mid \boldsymbol{f}_o \sim \mathsf{N}\left\{ \boldsymbol{g}_p + \boldsymbol{\Gamma}_{po}\boldsymbol{\Gamma}_{oo}^{-1}(\boldsymbol{f}_o - \boldsymbol{g}_o), \boldsymbol{\Gamma}_{pp} - \boldsymbol{\Gamma}_{po}\boldsymbol{\Gamma}_{oo}^{-1}\boldsymbol{\Gamma}_{op} \right\}.$$

In this way, the predictive distribution $p(\boldsymbol{f}_p \mid \boldsymbol{f}_0)$ is obtained by marginalizing with respect to the random function $f(\cdot)$. In other words, inference can proceed without

the need to record the infinite dimensional $f(\cdot)$. This is critical for practical implementation, computation and extrapolation. The predictive distribution implies that the optimal predictor for $\boldsymbol{f}_p$ under squared-error loss is simply

$$\widehat{\boldsymbol{f}}_p = \boldsymbol{g}_p + \boldsymbol{\Gamma}_{po}\boldsymbol{\Gamma}_{oo}^{-1}(\boldsymbol{f}_o - \boldsymbol{g}_o) \tag{1.8}$$

If $\boldsymbol{f}_o$ is observed, then equation (1.8) can be used to estimate $\boldsymbol{f}_p$ for any value of the covariate x. In that case, $\widehat{f}(x_i) = f(x_i)$ for the covariate values $x_1, \ldots, x_n$ where the function was observed and (1.8) acts as an interpolator at $x_{n+1}, \ldots, x_{n+m}$. However, in practice we often observe $\boldsymbol{f}_o$ only indirectly through some noisy observations $\mathbf{y}_o = (y_1, \ldots, y_n)$ with $E(y_i) = f(x_i)$. If we assume normal residuals we get a hierarchical model

$$\mathbf{y}_o \mid \boldsymbol{f}_o \sim \mathsf{N}(\boldsymbol{f}_o, \tau^2\mathbf{I}), \qquad \boldsymbol{f} \sim \mathsf{N}(\boldsymbol{g}_o, \boldsymbol{\Gamma}_{oo}), \tag{1.9}$$

which implies $\mathbf{y}_o \sim \mathsf{N}(\boldsymbol{g}_o, \boldsymbol{\Gamma}_{oo} + \tau^2\mathbf{I})$. Then, a posteriori,

$$\boldsymbol{f}_o \mid \mathbf{y}_o \sim \mathsf{N}\left\{\left(\boldsymbol{\Gamma}_{oo}^{-1} + \frac{1}{\tau^2}\mathbf{I}\right)^{-1}\left(\boldsymbol{\Gamma}_{oo}^{-1}\boldsymbol{g}_o + \frac{1}{\tau^2}\mathbf{y}_o\right), \left(\boldsymbol{\Gamma}_{oo}^{-1} + \frac{1}{\tau^2}\mathbf{I}\right)^{-1}\right\},$$

and the optimal predictor under squared error loss for $\boldsymbol{f}_p$ is given by

$$\widehat{\boldsymbol{f}}_p = \boldsymbol{g}_p + \boldsymbol{\Gamma}_{po}\boldsymbol{\Gamma}_{oo}^{-1}\left\{\left(\boldsymbol{\Gamma}_{oo}^{-1} + \frac{1}{\tau^2}\mathbf{I}\right)^{-1}\left(\boldsymbol{\Gamma}_{oo}^{-1}\boldsymbol{g}_o + \frac{1}{\tau^2}\mathbf{y}_o\right) - \boldsymbol{g}_o\right\}. \tag{1.10}$$

In this case, the predictor acts as a smoother rather than an interpolator, with the value of τ^2 controlling the level of smoothing. In particular, note that if $\tau^2 \to 0$ then (1.10) converges to (1.8).

An excellent reference on Gaussian process models for regression is Rasmussen and Williams (2006). For applications of Gaussian processes in spatial statistics, see Cressie (1993) and Banerjee *et al.* (2004).

1.3.2. Models Based on Basis Representations

An alternative strategy to create rich models for random functions is to consider representations of the unknown function f in terms of a basis system. This approach reduces the problem of modeling a random function to that of modeling the coefficients associated with the bases.

In the sequel, let $f \in \mathcal{F}$, where $\mathcal{F}$ represents an appropriate function space, and let $(\phi_l(x))$ be a system of basis functions spanning $\mathcal{F}$, implying

$$f(x) = \sum_{\ell=1}^{\infty} \beta_\ell \phi_\ell(x). \tag{1.11}$$

Nonparametric Bayesian inference on f can now be carried out by introducing a prior distribution for coefficients $\boldsymbol{\beta}$. For any $x \in \mathcal{X}$, the optimal estimator under squared error loss is

$$\widehat{f}(x) = \sum_{\ell=1}^{\infty} \mathsf{E}\left(\beta_i \mid \mathbf{y}\right) \phi_\ell(x).$$

A popular example of this approach is wavelet regression. See, for example, Müller and Vidakovic (1999) and Vidakovic (1998).

Let $\mathcal{F} = L^2(\mathbb{R})$ denote the space of square-integrable functions on $\mathbb{R}$. A basis for $\mathcal{F}$ is obtained by translations and dyadic dilations of a mother wavelet $\psi(x) \in L^2(\mathbb{R})$, so that

$$\psi_{j\ell}(x) = 2^{j/2}\psi(2^j x - \ell), \qquad \ell \in \mathbb{Z}, j = 0, \ldots, 2^j - 1,$$

and $(\psi_{j\ell})$ forms an orthonormal basis for L^2. The fact that shifted and scaled versions of $\psi(\cdot)$ form an orthonormal basis is a defining characteristic of a wavelet function. Transformation and dilation of an arbitrary function would not necessarily define an orthonormal basis. Hence, any function $f \in L^2$ can be written as

$$f(x) = \sum_j \sum_\ell \beta_{j\ell} 2^{j/2} \psi(2^j x - \ell). \tag{1.12}$$

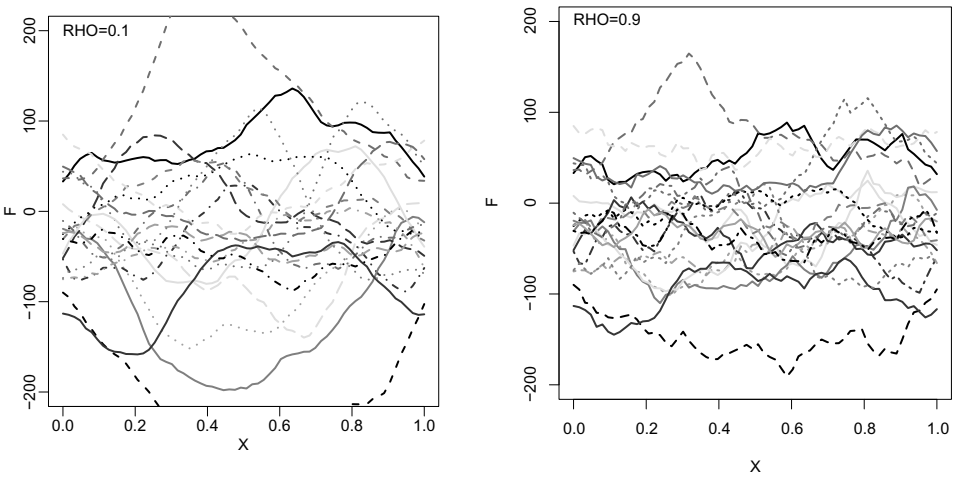
The representation of a function with respect to a wavelet basis can be thought of as a localized version of a Fourier transform. The localization is provided by the index ℓ, while the index j is the level of detail explained by the basis function, with larger values of j corresponding to basis functions explaining higher frequency properties of the function.

Since in practice just a finite number of the coefficients $(\beta_{j\ell})$ can be estimated from a finite sample of size n, the estimation problem is often regularized by assuming that coefficients at the higher levels of details are zero, say for $j \geq \lfloor \log_2 n \rfloor$. In addition, variable selection priors (such as zero inflated Gaussians or double exponential priors) can be used to further reduce the number of coefficients to be estimated.

An important practical feature of the wavelet bases is the existence of a superfast algorithm to carry out the transformation from $\boldsymbol{f}$ to $\boldsymbol{\beta}$ and the reconstruction from $\boldsymbol{\beta}$ to $\boldsymbol{f}$ when $\boldsymbol{f}$ is evaluated over a regular grid. The algorithm is known as the pyramid scheme and allows easy implementation of nonparametric regression if the data are observed on a regular grid. In the absence of a regular grid computation becomes challenging and the practical advantage of wavelet bases fades. An excellent introduction to wavelet appears in Vidakovic (1999).

Example 3 (Wavelet prior for a periodic function.) *Figure 1.5 shows prior simulations for a random function $f \sim p(f)$ with $p(f)$ defined as a prior on wavelet coefficients. The function is defined on $[0, 1]$ and is constrained to $f(0) = f(1)$. The prior is a multivariate normal dependent prior on the wavelet coefficients $\beta_{j\ell}$ in (1.12). We start with a regular grid $\{\ell/n;\ \ell = 0, \ldots, n = 2^J\}$ on $[0, 1]$, and define a multivariate normal prior on $d_\ell = f(\ell/n) - f[(\ell-1)/n]$, $\ell = 1, \ldots, n$. Working with the differences makes it easy to impose the constraint $f(0) = f(1)$. A multivariate normal on $\boldsymbol{d} = (d_1, \ldots, d_n)$ is defined with mean 0 and $\mathsf{Cor}(d_k, d_\ell) = \exp(-\rho|k-\ell|)$. By the pyramid scheme this implies a multivariate normal prior on all $\beta_{j\ell}$. See Berger* et al. *(2012) for details. Modeling in the wavelet domain allows us to later add prior information about the spikiness of the function, formalized by selecting wavelet coefficients $\beta_{j\ell}$ with prior probability $p(\beta_{j\ell} = 0) = 1 - \alpha^j$ (see §2.2.2).*

Models based on basis representations are particularly attractive because model fitting can often be carried out using tools for linear regression. This requires that the basis system is fixed in advance, and that the number of basis functions that are used in the representation is finite. As we discussed above for the case of wavelet bases, this last requirement is often satisfied by truncating the basis system and introducing regularization priors on the remaining coefficients.



(a) $\rho = 0.1$ (b) $\rho = 0.9$

FIG 1.5. *Prior simulation of random curves $f \sim p(f)$ using a dependent prior on the wavelet coefficients in (1.12), subject to $f(0) = f(1)$, with high (panel a) and low (panel b) prior correlation.*

1.3.3. Basis Representation and Gaussian Process Priors

There is a close connection between models based on basis representations and those based on Gaussian process priors. The Kaurhunen-Loève representation theorem (Karhunen, 1947; Lòeve, 1978) states that if f follows a Gaussian process prior with mean $g(x) = 0$ and covariance function $\gamma(x, x')$, then it admits a representation of the form (1.11), where each β_ℓ is independently distributed as a normal random variable and the functions $\{\phi_\ell(x)\}$ are the eigenfunction of the covariance function $\gamma(x, x')$, i.e., they satisfy the integral equations

$$\lambda_k \phi_k^*(x) = \int \gamma(x, x')\phi_k^*(x')dx', \qquad \int \phi_k^*(x)\phi_\ell^*(x)dx = \begin{cases} 1 & k = \ell \\ 0 & k \neq \ell. \end{cases} \tag{1.13}$$

Similar results can be obtained for Gaussian processes with more general mean function $g(x)$ by expanding $g(x)$ in terms of the orthonormal basis functions (ϕ_k). Hence, when we fit a Gaussian process model to data we are implicitly estimating a model that uses an infinite-dimensional basis representation where the basis functions satisfy the constrains in (1.13) and where each β_ℓ is given an independent Gaussian prior.

Chapter 2

Data Analysis

2.1. Density Estimation and Survival Analysis

The most straightforward application of BNP priors for statistical inference is in density estimation problems. Consider the generic density estimation problem, with data y_i, $i = 1, \ldots, n$, that is believed to be generated as an i.i.d. sample from some unknown distribution G. A BNP model can be used as a prior for G to complete the model

$$y_i \mid G \sim G \qquad\qquad G \sim p(G). \tag{2.1}$$

We could use, for example, a DP prior to specify $p(G)$ as $G \sim \mathsf{DP}(M, G_0)$, or a PT prior $G \sim \mathsf{PT}(\Pi, \mathcal{A})$. Many BNP models are conjugate under i.i.d. sampling. In other words, $p(G \mid y_1, \ldots, y_n)$ is in the same family as the prior, with updated parameters. This is true, for example, for the DP prior or the PT prior.

A limitation of many popular BNP models $p(G)$ for random probability measures is the discrete nature of G. This is the case, for example for the DP prior. A simple fix is the use of mixture models, convoluting the discrete RPM with a continuous kernel $f(x;\ \mu)$, e.g. $\mathsf{N}(x;\ \mu, 1)$,

$$y_i \mid F \sim F(y_i),\ F(y) = \int f(y;\ \theta)\ dG(\theta) \text{ and } G \sim p(G)$$

Such models are known as DP mixtures (DPM) etc. The model is illustrated in Figure 2.1. The point masses are the discrete probability measure G. Each point mass is smeared out with a kernel $f(x;\ \mu)$. The convolution of G and the kernel creates the continuous probability measure F. Posterior inference usually proceeds in an equivalent hierarchical model with latent variables $\theta_i \sim G$, $i = 1, \ldots, n$. The mixture is rewritten as

$$y_i \sim f(y_i;\ \theta_i) \qquad\qquad \theta_i \mid G \sim G \qquad\qquad G \sim p(G). \tag{2.2}$$

Posterior inference is still almost conjugate. If $p(G)$ was conjugate under i.i.d. sampling, then the complete conditional posterior $p(G \mid \theta_1, \ldots, \theta_n)$ for G given the imputed latent variables θ_i remains in the same family. And conditional on G the latent variables θ_i are usually easy to impute. We will discuss detail strategies in §3.3.

A special case of density estimation arises in survival analysis as density estimation with event time data, usually involving censoring. Survival analysis is a very traditional application of BNP in the early literature. Some BNP models for random probability measures remain conjugate even under (right) censoring. For example, a PT prior can be specified such that the posterior process for the unknown distribution remains a PT, even in the presence of censoring.

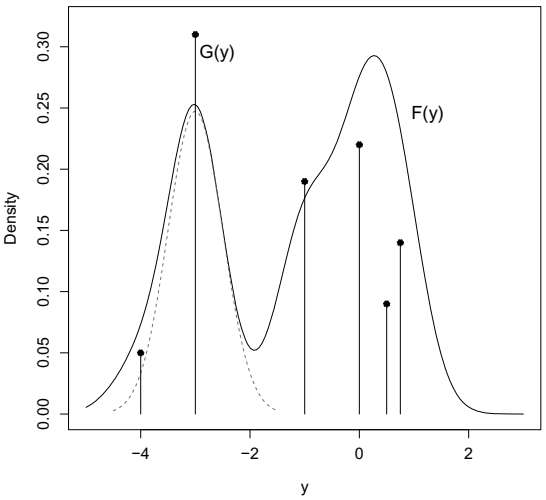


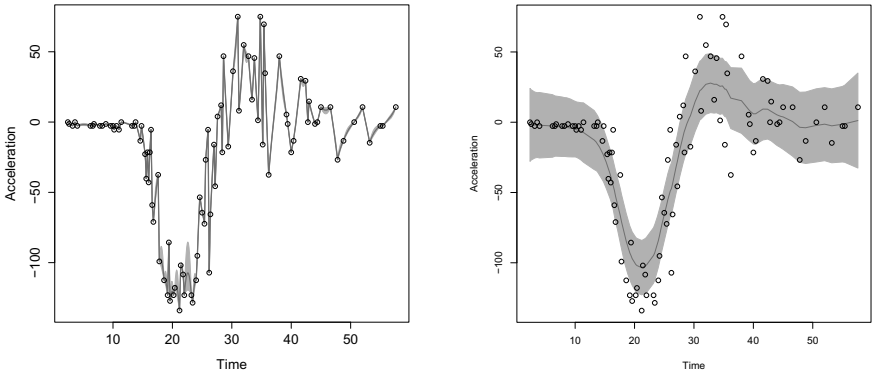
FIG 2.1. *Dirchlet Process mixture prior. The discrete random probability measure G is convoluted with a smooth kernel to create a continuous distribution F.*

2.2. Regression

Consider a generic regression problem with dependent variable y_i, covariates x_i, $i = 1, \ldots, n$, and an assumed model $y_i = f(x_i) + \epsilon_i$ with $\epsilon_i \sim p_\epsilon(\epsilon_i)$. As long as both, the regression function $f(\cdot)$ and the residual distribution $p_\epsilon(\cdot)$, are indexed by finitely many parameters, inference reduces to a traditional parametric regression problem. The problem becomes a non-parametric regression when the investigator wants to relax the parametric assumptions of either of the two model elements. This characterization of non-parametric regression allows for three cases.

2.2.1. Non-Parametric Residuals

The model can be generalized by going non-parametric on the residual distribution, assuming $\epsilon_i \sim G$ and a non-parametric prior $p(G)$, while keeping the regression mean function parametric as $f_\theta(\cdot)$ for a finite dimensional parameter vector θ. We refer to this case as a non-parametric error model. Essentially this becomes density estimation for the residual error. Of course the residuals ϵ_i are not usually observable. Hence, the problem reduces to one of density estimation conditional on assumed values for the parameters θ. A typical implementation using Markov chain Monte Carlo posterior simulation would include a transition probability that updates the currently imputed RPM G conditional on currently imputed values of θ. Conditional on θ the problem of updating inference on G reduces to density estimation for the residuals $\epsilon_i = y_i - f_\theta(x_i)$. And vice versa, conditional on imputing G, updating θ reduces to a regression problem with a known residual distribution.



(a) Interpolating Gaussian process (b) Smoothing Gaussian process

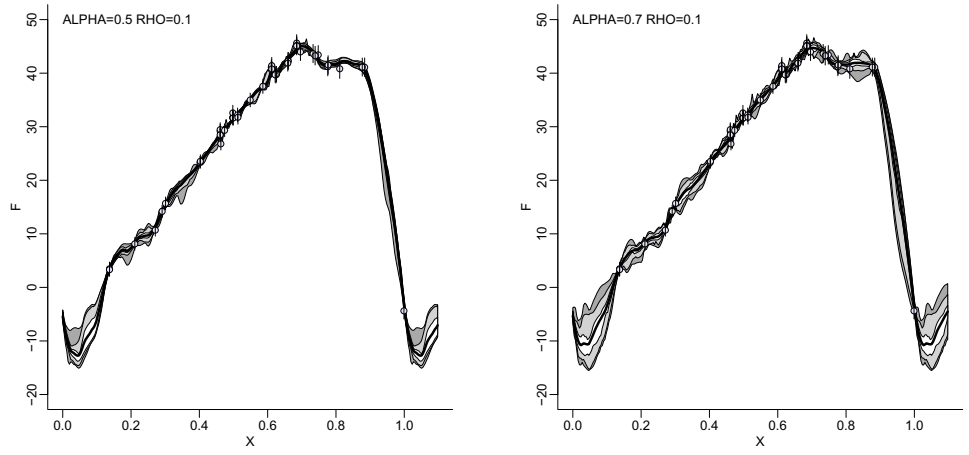
FIG 2.2. *An example of nonlinear regression using Gaussian processes. Points correspond to the observed data, the solid line corresponds to the posterior mean, and the grey bands (in panel (b)) are 95% credible intervals. The left panel corresponds to a model where* $\tau^2 = 0$*, so that the model acts as an interpolator. The model on the right panel allows for* $\tau^2 > 0$*, so that the predictor arising from the model behaves as a smoother.*

2.2.2. Non-Parametric Mean Function

Alternatively one could relax the parametric assumption on the mean function and complete the model with a non-parametric prior $f(\cdot) \sim p(f)$. We refer to this as a non-parametric regression mean function. As discussed in §1.3, popular choices for $p(f)$ are Gaussian process priors or priors based on basis expansions, such as wavelet based priors or neural network models.

Example 4 (Nonparametric mean function with GP prior) *We illustrate the use of Gaussian process models in nonparametric regression using a widely studied dataset originally analyzed by Silverman (1985). The data are the measurements of head acceleration in a simulated motorcycle accident used to test crash helmets. The regression function is clearly non-linear, and even a piecewise linear function would have a difficulty fitting this data. The left panel of Figure 2.2 presents the regression function obtained from an interpolation model, corresponding to* $\tau^2 = 0$ *in* (1.9)*. The right panel shows the fit obtained from a smoothing mode* $\tau^2 > 0$*. In both cases, the parameters* τ^2*,* σ^2 *and* λ *where learned from the data using a Markov chain Monte Carlo algorithm.*

Example 5 (Nonparametric regression using wavelets) *Barnes* et al. *(2003) consider data from cepheid stars, i.e., pulsating stars. Figure 2.3 plots observed radial velocities* y_i *against phase* x_i*, together with a non-linear regression estimate* $f(\cdot)$ *based on an BNP model. The model used a basis expansion for the unknown phase-velocity curve. The basis is a wavelet basis. We discussed the prior for this example before, in Example 3. Figure 1.5 shows draws from the prior* $f \sim p(f)$*. We now add a prior to select wavelet coefficients, with* $p(\beta_{j\ell} = 0) = 1 - \alpha^{j+1}$*. Smaller* α *imposes more prior shrinkage and reduces the prior probability for high frequency features. Conditional on the selected wavelet coefficients we continue to use the dependent prior introduced before. Figure 2.3 shows inference under* $\alpha = 0.5$ *and* $\alpha = 0.7$*.*



(a) $\alpha = 0.5$ (b) $\alpha = 0.7$

FIG 2.3. *Phase-velocity curve $f(x)$ for cepheid stars. The figures show the posterior estimated phase-velocity curve $E(f \mid data)$ (thick central line), and pointwise central HPD 50% (light grey) and 95% (dark grey) intervals for $f(x)$. The circles shows the data points. Inference is under an BNP model $p(f)$ using a basis expansion of f with wavelets and $p(\beta_{j\ell} = 0) = 1 - \alpha^{j=1}$. Recall from Example 3 that ρ defines the level of prior dependence.*

2.2.3. Fully Non-Parametric Regression

Finally, one could go non-parametric on both assumptions. We refer to this as a fully nonparametric regression. The sampling model becomes $p(y_i \mid x_i) = G_x$, with a prior on the family of conditional RPMs, $p(G_x,\ x \in X)$. Many commonly used BNP priors for $\mathcal{G} = \{G_x\}$ are variations of dependent DP priors.

Example 6 (Fully nonparametric regression) *Klein and Moeschberger (1997, chapter 1.11) show data from a clinical trial. The data are survival times for patients with tongue cancers. The study investigated the effect of aneuploidy (abnormal number of chromosomes) of the tumor cells. Let $G_x(\cdot)$ denote the distribution of survival times for patients with aneuploid ($x = 1$) and (normal) diploid ($x = 0$) tumor cells. Figure 2.4 shows the Kaplan-Meier estimator of G_x, $x \in \{0, 1\}$, together with an BNP estimate. The BNP estimate is under a DDP prior on $\{G_x,\ x = 0, 1\}$.*

2.3. Mixed Effects Models

BNP priors are often used for model features that are important for appropriate modeling of the observed data, but that are not of interest in themselves. A typical example are random effects distributions in mixed effects models. Random effects are a convenient and common approach to represent the dependence structure in the observed data. Sometimes random effects also have a meaningful interpretation as a property specific to sampling units. For example, when the experimental units are patients in a clinical study, then patient-specific random effects represent the heterogeneity of the patient population, which needs to be accounted for.

In many analyses the distributional assumptions for such random effects distributions are driven entirely by technical convenience and simplicity, using for example a multivariate normal distribution. However, there is often no good scientific reason

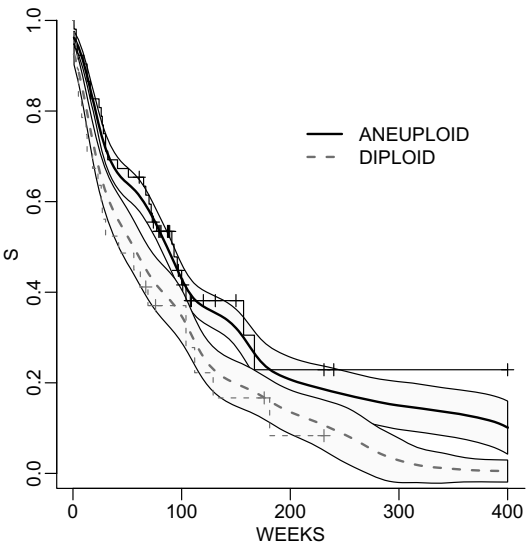


FIG 2.4. *Survival times for tongue cancer patients. The figure shows a Kaplan-Meier estimate (step function) and an NP Bayes estimate (smooth curves) for the survival functions G_x for patients with anaploid ($x = 1$) and diploid ($x = 0$) tumors. The bands around the BNP estimates show pointwise $\pm$ 1.0 posterior standard deviation bounds. The BNP estimate is based on a DDP prior for $\{G_x,\ x = 0, 1\}$.*

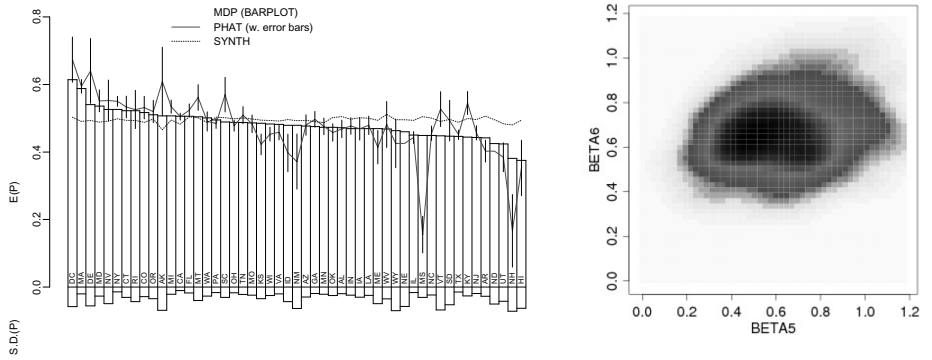
to assume a particular parametric form. Quite to the contrary, patient populations are known to be highly heterogeneous, including outliers, subpopulations and other features that are inconsistent with a multivariate normal model.

This is where BNP priors come in. Let z_i denote a generic random effect specific to the i-th experimental unit. When an investigator wants to avoid a strict parametric assumptions, he or she could instead use $z_i \sim G$ with a BNP prior $G \sim p(G)$. The types of priors used for $p(G)$ are again similar to the density estimation problem, with the difference that in a mixed effects model the random effects z_i are only latent.

Example 7 (Semiparametric mixed effects model) *Malec and Müller (2008) consider a mixed effects model for mammography utilization in the U.S. The data are mammography usage by county and demographic group. The model includes a regression on some county level covariates and county-specific random effects z_i. The random effects are 6-dimensional. Figure 2.5a shows posterior estimated rates of mammography by state. Figure 2.5b shows the estimated random effects distribution.*

2.4. Clustering and Classification

Some statistical inference problems involve a partition of a population of experimental units into clusters. For example, hospitals might be clustered into more homogeneous subgroups, disease subtypes might be grouped by comparable prognosis, states could be grouped by comparable patterns of use of preventive care, etc. Probability models for random partitions can be used to define appropriate



(a) Estimated mammography utilization by state

(b) Estimated r.e. distribution $E(G \mid data)$

FIG 2.5. *Mammography utilization. Estimated rate of mammography utilization by state (a). The histogram shows the estimated use under the semi-parametric Bayesian model. The solid line (with many spikes) shows for comparison the observed sample averages. The dotted line shows an estimate known as synthetic estimate. States are ordered by estimated mammography utilization under the BNP model. Panel (b) shows a bivariate marginal of the estimated random effects distribution $E(G \mid data)$ for county-specific random effects z_i (BETA in the plot).*

inference models for these applications (recall our discussion of product partition models from §1.2.3).

Example 8 (Clustering of morphological data.) *We consider data from Lubischew (1962) who reports measurements on five external characteristics (lengths etc.) of male insects of three species of leaf beetles. We use the 5-dimensional data set, ignoring the species labels. Figure 2.6 shows model-based clustering and classification for this 5-dimensional data set. The plotting symbols show the measurements (showing two of the five dimensions).*

We fit model (2.2) with $\theta_i = (\mu_i, \Sigma_i)$ and $f(x;\, \mu_i, \Sigma_i) = \mathsf{N}(x;\, \mu_i, \Sigma_i)$. The discrete nature of $G \sim \mathsf{DP}$ implies a positive probability of ties among the θ_i. Let $\theta_j^\star$ denote the unique values. The model implies a prior probability model on a partition of the beetles into clusters $S_j = \{i :\ \theta_i = \theta_j^\star\}$ defined by these ties. We will come back to this model several times in the upcoming discussion. The implied prior on the partition of the experimental units, beetles in this case, is known as the Pólya urn. Figure 2.6 shows clusters S_j by different plotting symbols, together with the posterior predictive distribution for a future beetle (contours). In these contours we can recognize the cluster specific $(\mu_j^\star, \Sigma_j^\star)$ as the location $\mu_j^\star$ and orientation $\Sigma_j^\star$ of three ellipses.

2.5. Computation

The flexible nature of BNP inference comes at a price. Implementation of posterior inference for some models can be a bit more involved than similar parametric models. However, actual use of BNP models for data analysis is usually less complicated than what it might seem at first glance. One reason is that inference usually proceeds in a reduced model, after marginalizing with respect to the infinite dimensional quantitity. For example, in a density estimation problem (2.2)

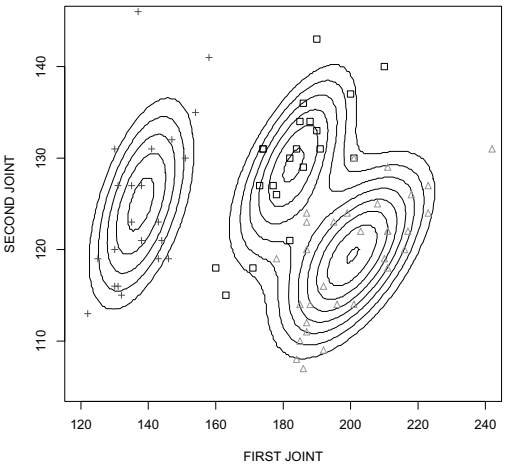


FIG 2.6. *Clustering of beetles by 5 morphological measurements (only two are shown). The plotting symbols show a partition of the beetles into three clusters. The contours show the posterior predictive distribution for a future beetle. We can recognize cluster-specific locations* $\mu_j^\star$ *and covariance matrices* $\Sigma_j^\star$, $j = 1, \ldots, 3$.

with a DP prior, $G \sim \mathsf{DP}(\cdot)$, the marginal model $p(\boldsymbol{\theta}, \boldsymbol{y})$ of all latent variables $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)$ and all data $\boldsymbol{y} = (y_1, \ldots, y_n)$ is available in closed form. This allows for relatively straightforward computation for posterior predictive inference and many other relevant inference summaries.

In Appendix A we show actual implementations in R for inference under DP mixture and PT priors.

Another important feature that makes the use of BNP models practically feasible is the availability of public domain programs. One popular program is the R package `DPpackage` (Jara *et al.*, 2011) that implements inference for PT priors, DP models, Bernstein polynomials, dependent DP models and many variations of these models. The program can be downloaded from `http://www.mat.puc.cl/~ajara/Softwares.html`. The *BNPDensity* package (Barrios *et al.*, 2011) implements density estimation using semi-parametric mixtures with a non-parametric normalized generalized gamma (NGG) prior on the mixing measure (James *et al.*, 2009). Another R package that implements inference for some BNP models is `BayesM` (Rossi *et al.*, 2005).

Chapter 3

Dirichlet Process

3.1. The Dirichlet Process Prior

3.1.1. Definition

The Dirichlet process (DP) is arguably the most popular BNP model for random probability measures (RPM), and plays a central role in the literature on RPMs, appearing as a special case of a number of other more general models (recall our discussion in Chapter 1). Hence, the DP can be characterized in a number of different ways.

The original definition of the DP is due to Ferguson (1973), who considered a probability space $(\Theta, \mathcal{A}, G)$ and an arbitrary partition $\{A_1, \ldots, A_k\}$ of Θ. A random distribution G is said to follow a Dirichlet process prior with baseline probability measure G_0 and mass parameter M, denoted $G \sim \mathsf{DP}(M, G_0)$, if

$$(G(A_1), \ldots, G(A_1)) \sim \mathsf{Dir}(MG_0(A_1), \ldots, MG_0(A_k)). \tag{3.1}$$

This collection of finite dimensional distributions implies a well defined infinite dimensional model $p(G)$ because they satisfy Kolmogorov's consistency conditions; proving this fact is one of the main focuses of Ferguson's original paper.

An alternative definition of the DP, known as the "stick-breaking" construction, is provided in Sethurman (1994). Let $\delta_\theta(\cdot)$ denote a point mass at θ. An RPM

$$G(\cdot) = \sum_{h=1}^{\infty} w_h \delta_{\widetilde{\theta}_h}(\cdot) \tag{3.2}$$

has a $\mathsf{DP}(M, G_0)$ prior if $(\widetilde{\theta}_h)$ are i.i.d. samples from G_0 and $w_h = v_h \prod_{k<h}\{1-v_k\}$ with $v_h \sim \mathsf{Beta}(1, M)$, i.i.d. This constructive definition of the DP is extremely useful for extending the model to more complex problems (see for example Chapter 5) and to highlight important properties of the model. Implicit in (3.2) is the fact that G is discrete, even if G_0 is a continuous distribution.

Recall that the DP also induces a species sampling model. In particular, let $\theta_1, \theta_2, \ldots$ be an i.i.d. sequence such that $\theta_i \mid G \sim G$ where $G \sim \mathsf{DP}(M, G_0)$. Since G is almost surely discrete, there will be ties among the θ_is; let k_n be the number of unique values among $\{\theta_1, \ldots, \theta_n\}$, let $\{\theta_1^\star, \ldots, \theta_{k_n}^\star\}$ be these unique values and let n_{nj} be the number of draws among $\{\theta_1, \ldots, \theta_n\}$ that are equal to $\theta_j^\star$. Blackwell and MacQueen (1973) showed that the joint distribution of the θ_is can be characterized in terms of the predictive probability function

$$p(\theta_{n+1} \mid \theta_n, \ldots, \theta_1) \propto \sum_{j=1}^{k_n} n_{nj}\delta_{\theta_j^\star} + MG_0, \tag{3.3}$$

that is, a new θ_i is identical to a previously observed $\theta_j^\star$ with probability proportional to n_{nj} (i.e., how many times that value has been observed) or a new value sampled from the baseline measure with probability proportional to the total mass parameter M. The predictive distribution (3.3) is exactly in the format of (1.1). After integrating G, the observations are exchangeable and have identical marginal distribution G_0, but are not independent.

The allocation process associated with the predictive distribution in (3.3) is also known as the Pólya urn. Consider an urn that initially has M black balls and one colored ball (whose "color" is randomly selected according to G_0). We sequentially draw balls from the urn; if a colored ball is drawn then we returned it to the urn along with another ball of the same color, if a black ball is drawn, we returned it to the urn along with a ball of a new color randomly selected according to G_0. Another metaphor, the Chinese restaurant process (CRP), is popular in the machine learning community and essentially describes the same model.

For another characterizing property, recall that the DP can be characterized as an NRMI. In particular, let μ be a standard Gamma process on Θ with intensity function $\lambda(\cdot) = MG_0(\cdot)$, i.e., $\mu(A) \sim \mathsf{Gamma}(MG_0(A), 1)$ for any $A \subset \Theta$. Then $G(\cdot) \equiv \mu(\cdot)/\mu(\Theta) \sim \mathsf{DP}(M, G_0)$. This follows from (3.1) and the construction of Dirichlet random variables as normalized Gamma random variates (see, for example Robert and Casella, 2005).

Finally, recall that the DP can be characterized as a PPM with cohesion function $c(S_j) = M(n_j - 1)$, as a special case of the PT with $\alpha_\epsilon = \alpha_{\epsilon 0} + \alpha_{\epsilon 1}$, and as a NTR process.

3.1.2. Properties

Since the Dirichlet process places a distribution on the random measure G, the quantity $G(A)$ for any $A \subset \Theta$ is a random variable. From Ferguson's definition we have $G(A) \sim \mathsf{Beta}\{MG_0(A), M(1 - G_0(A))\}$. Hence

$$\mathsf{E}\{G(A)\} = G_0(A), \qquad \mathsf{Var}\{G(A)\} = \frac{G_0(A)\{1 - G_0(A)\}}{M + 1}.$$

This means that we can interpret G_0 as the expected shape of the random distribution G, while M controls the variability of the realizations around G_0.

To further clarify this interpretation of the parameters of the DP, we plot in Figure 3.1 realizations from DPs with standard normal baseline measure and different values of M. The random distributions G are discrete with probability one. We therefore use the c.d.f. to display the random distributions. Larger values of M reduce the variability of the realizations of the process, and for small values of M a small number of weights concentrate most of the probability mass, i.e., a few large steps dominate the cdf. Indeed, a priori, the size of the weights decreases geometrically,

$$\mathsf{E}(w_h) = \frac{1}{M + 1}\left(\frac{M}{M + 1}\right)^{h-1}.$$

A particularly appealing property of the Dirichlet process is its conjugacy under i.i.d. sampling. If $\theta_1, \ldots, \theta_n$ is an i.i.d. sample with $\theta_i \mid G \sim G$ and $G \sim \mathsf{DP}(M, G_0)$ then

$$G \mid \theta_1, \ldots, \theta_n \sim \mathsf{DP}\left(M + n, \frac{MG_0 + \sum_{i=1}^n \delta_{\theta_i}}{M + n}\right). \tag{3.4}$$

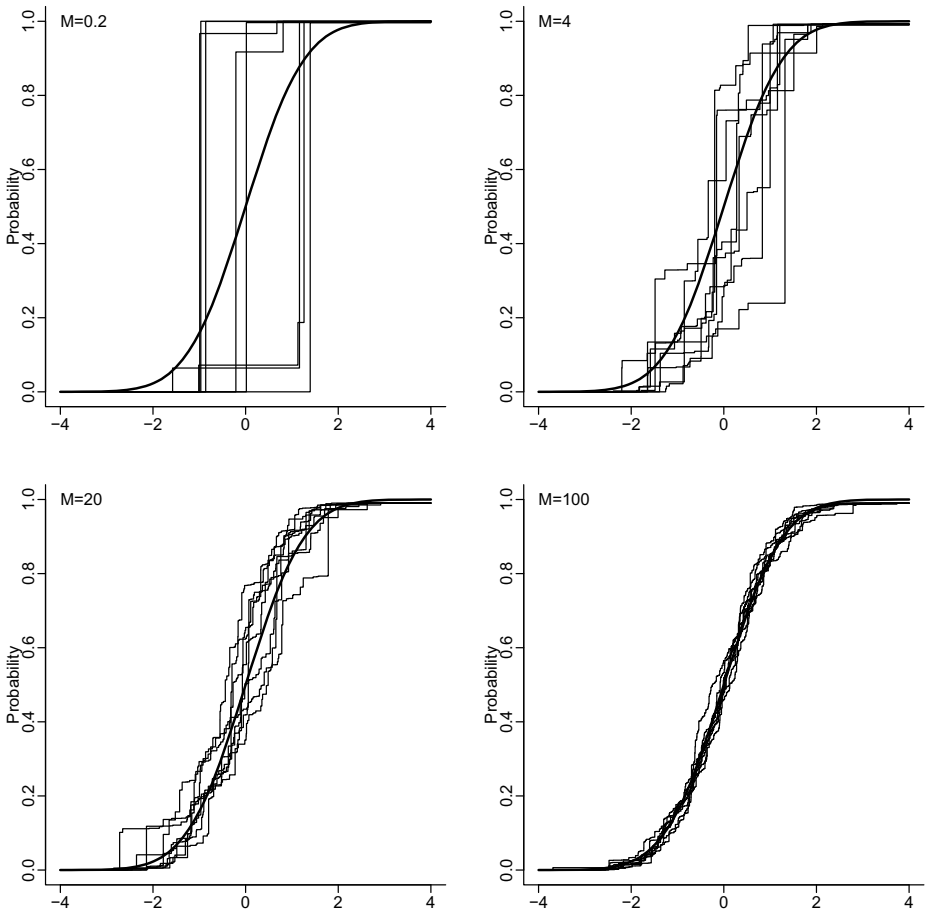


FIG 3.1. *Random distributions generated from a Dirichlet process prior with varying precision parameters M. In all cases, the baseline measure corresponds to a standard normal distribution (thick black curve). Each box contains 8 independent realizations (grey curves) with a common value for M. Note how M controls not only the variability of the realizations around G_0, but also the relative size of the jumps.*

The posterior mean,

$$\mathsf{E}(G \mid \theta_1, \ldots, \theta_n) = \frac{MG_0 + \sum_{i=1}^n \delta_{\theta_i}}{M+n},$$

can be interpreted a weighted average between the baseline measure G_0 and the empirical distribution $\frac{1}{n}\sum_{i=1}^n \delta_{\theta_i}$. In addition, since the empirical cdf is a consistent estimator if the θ_is are indeed i.i.d. from some true distribution G_T, it is easy to show from (3.4) that, as $n \to \infty$, we have $G(A) \mid \theta_1, \ldots, \theta_n \xrightarrow{P} G_T(A)$ for any measurable set A.

Example 9 (DP Nonparametric density estimation) *We carry out a simulation study with $\theta_i \sim G$, $i = 1, \ldots, n$, independently. We generate two datasets, with $n = 8$ and $n = 50$ observations, respectively, from the true model $G = \mathsf{N}(2, 4)$.*

In both cases, we pretend that G is unknown and carry out density estimation under a BNP prior, $G \sim \mathsf{DP}(M, G_0)$ with $G_0 = \mathsf{N}(0, 1)$ and total mass parameter $M = 5$.

Figure (3.2) shows the simulation truth, the empirical distribution, and the posterior mean $E(G \mid \boldsymbol{\theta})$ under the Dirichlet process prior. We see how the posterior mean is a weighted average of the prior mean and the empirical distribution of the observed data. Note that the posterior distribution converges relatively quickly to the empirical cdf as the sample size grows.

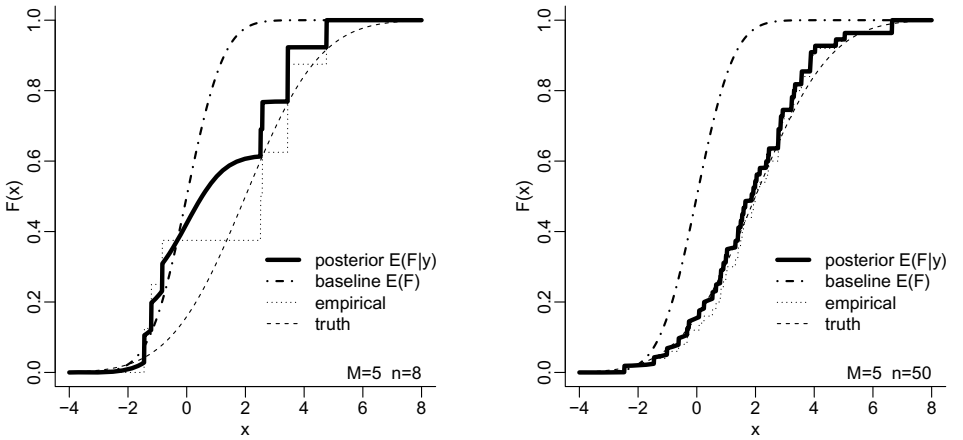


FIG 3.2. *An example of nonparametric density estimation using Dirichlet process priors. Two independent samples with sizes $n = 8$ and $n = 50$ where generated from a normal $\mathsf{N}(2, 4)$ distribution (whose c.d.f. is shown as the dashed black line "truth"). In both cases, the prior precision parameter is $M = 5$, while the baseline measure is a standard normal distribution (dashed dotted, "baseline"). The empirical CDF (dotted step function) and posterior mean (thick black line) are also shown.*

Finally, we discuss some properties of a random sample from the DP that follow from the Pólya urn representation of the process. As mentioned in Chapter 1, the predictive probability function in (3.3) implies a probability model for any partition of the experimental units into clusters $S_j = \{i : \theta_i = \theta_j^\star\}$, i.e., into clusters defined by the ties among the draws θ_i. Recall that we used $\boldsymbol{n} = (n_1, \ldots, n_k)$ for the cluster sizes for a partition of n experimental units into clusters S_j, $j = 1, \ldots, k$. The probability model for $(k, \boldsymbol{n})$ implied by (3.3) can easily be determined as

$$p(k, n_1, \ldots, n_k) = \frac{\Gamma(M)}{\Gamma(M+n)} M^k \prod_{j=1}^{k} \Gamma(n_j). \tag{3.5}$$

The model $p(k, \boldsymbol{n})$ for a random partition is known as the exchangeable product partition function (EPPF). From this EPPF we can obtain the probability mass function for the number of unique values k_n (Antoniak, 1974),

$$p(k_n) = S_{n,k} n! M^{k_n} \frac{\Gamma(M)}{\Gamma(M+n)}, \tag{3.6}$$

where $S_{n,k}$ is the unsigned Stirling number of the first kind. Using a conditional

expectation argument we find

$$\mathsf{E}(k) = \sum_{i=1}^{n} \frac{M}{M+i-1} \approx M \log\left(\frac{M+n}{M}\right)$$

for large n. Another consequence of (3.6) is that the partitions favored by the DP are very uneven, i.e., the DP favors partitions with a small number of large clusters and a large number of smallish ones. This feature of the model is often inappropriate in applications, which has motivated many of the generalizations that we discuss in later chapters.

3.2. DP Mixtures

The discrete nature of the DP random measures is awkward when the unknown distribution is known to be continuous. Even worse, for some hierarchical models the Dirichlet process prior can lead to inconsistent estimators if the true distribution is continuous (for examples, see Diaconis and Freedman, 1986a,b). One way to mitigate this limitation of the DP is to add to the discrete distribution G a convolution with a continous kernel. This is similar in spirit to kernel density estimators, where the empirical distribution is smoothed by convoluting it with an appropriate kernel.

Let $y_1, y_2, \ldots$ be an i.i.d. sample with unknown distribution F. A Dirichlet process mixture prior (DPM) on F posits that

$$y_i \sim F(y_i) = \int p(y_i \mid \theta) G(\mathrm{d}\theta), \qquad G \sim \mathsf{DP}(M, G_0), \tag{3.7}$$

where $p(y_i \mid \theta)$ is a parametric distribution (often referred to as the kernel of the mixture), which is indexed by a finite dimensional parameter θ. For example, in a DP location mixture of normals we have

$$y_i \mid G \sim \int \mathsf{N}(y_i \mid \mu, \sigma^2) G(\mathrm{d}\mu), \qquad G \sim \mathsf{DP}(M, G_0).$$

Figure 2.1 illustrates a DP mixture of normal model.

The model in (3.9) can be represented in a number of alternative ways. Exploiting the stick-breaking construction of the Dirichlet process we can write

$$y_i \mid (w_h), (\widetilde{\theta}_h) \sim \underbrace{\sum_{h=1}^{\infty} w_h p(y_i \mid \widetilde{\theta}_h)}_{F(y_i)}, \tag{3.8}$$

where

$$\widetilde{\theta}_h \sim G_0, \qquad w_h = v_h \prod_{k<h} \{1 - v_k\}, \qquad v_h \sim \mathsf{Beta}(1, M).$$

This representation highlights the nature of the DP mixture model as a discrete mixture. DP mixtures are countable mixtures with an infinite number of components and a specific prior on the weights and the component-specific parameters. Working with an infinite number of components is particularly appealing because it ensures that, for appropriate choices of the kernel $p(y_i \mid \theta)$, the DPM model has

support on a large classes of distributions. For example, Lo (1984) showed that a DP location-scale mixture of normals,

$$y_i \mid G \sim \int \mathsf{N}(y_i \mid \mu, \sigma^2) G(\mathrm{d}\mu, \mathrm{d}\sigma^2), \qquad G \sim \mathsf{DP}(M, G_0),$$

has full support on the space of absolutely continuous distributions. Similarly, a mixture of uniform distributions

$$y_i \sim \int \mathsf{Uni}(y_i \mid -\theta, \theta) G(\mathrm{d}\theta), \qquad G \sim \mathsf{DP}(M, G_0),$$

where $\mathsf{Uni}(x \mid a, b)$ indicates a random variable x with a uniform distribution on $[a, b]$, has full support on the space of all unimodal symmetric distributions.

Another consequence of (3.8) is that the DPM induces clustering among the observations, with M controlling the a priori expected number clusters in the sample. In particular, note that if $M \to 0$, the model reduces to a single component mixture where all observations are i.i.d. from $p(y \mid \theta)$ and $\theta \sim G_0$, i.e., a fully parametric model. On the other hand for $M \to \infty$ each observation is assigned its own singleton cluster and we have $y_i \sim \int p(y_i \mid \theta) G_0(\mathrm{d}\theta)$, i.i.d. Nothing is unknown about the sampling model for y_i.

An alternative representation for (3.9) introduces latent random effects (θ_i) to replace the mixture by a hierarchical model

$$y_i \mid \theta_i \sim p(y_i \mid \theta_i), \qquad \theta_i \mid G \sim G, \qquad G \sim \mathsf{DP}(M, G_0). \tag{3.9}$$

The hierarchical model (3.9) also highlights the nature of clusters generated by ties among the θ_i that arise under sampling from the discrete probability measure G.

Or, integrating out the random measure G,

$$y_i \mid \theta_i \sim p(y_i \mid \theta_i), \qquad (\theta_1, \ldots, \theta_n) \sim p(\theta_1, \ldots, \theta_n),$$

where the joint distribution $p(\theta_1, \ldots, \theta_n)$ is implicitly defined by the sequence of predictive distributions in (3.3). As before, denote by $(\theta_j^\star)$ the unique values among $\theta_1, \ldots, \theta_n$ and introduce indicator variables (s_i) such that $\theta_i = \theta_{s_i}^\star$. Then we can further rewrite the model as

$$y_i \mid s_i, (\theta_j^\star) \sim p(y_i \mid \theta_{s_i}^\star), \quad \theta_j^\star \sim G_0, \quad p(s_1, \ldots, s_n) = \frac{\Gamma(M)}{\Gamma(M+n)} M^k \prod_{j=1}^{k} \Gamma(n_j),$$

where k is the number of distinct values among $s_1, \ldots, s_n$ and $n_j = \sum_i I(s_i = j)$ is the number of s_is that are equal to j. By creating the implied clusters the DPM places a prior distribution on all possible partitions of the data into at most n groups. This is precisely the probability model stated in (3.5).

The last two representations marginalize with respect to the infinite dimensional G. Hence, they are particularly useful for the development of computational tools for the DP (see §3.3.1). Finally, we note that although the mixture in (3.8) has infinitely many terms, for any finite sample size n, at most n distinct $\widetilde{\theta}$ are sampled as $\theta_j^\star$.

3.3. Posterior Simulation for DP Mixture Models

One of the attractive features of the Dirichlet process mixture model is that a number of simulation-based algorithms are available for posterior inference. In this

section we review some of the most commonly used algorithms. Many can be extended to other nonparametric models with just minor modifications. Throughout this section we assume the model

$$y_i \mid \theta_i \sim p(y_i \mid \theta_i), \qquad \theta_i \mid G \sim G(\theta_i), \qquad G \sim \mathsf{DP}(M, G_0). \tag{3.10}$$

That is, a DP mixture model with kernel $p(y_i \mid \theta_i)$ and unknown mixing measure G which follows a Dirichlet process prior.

3.3.1. Collapsed Gibbs Samplers

Conjugate models

Collapsed Gibbs samplers exploit the representation of the DP as a SSM that was discussed in §1.2.1 and §3.1. The first version of this algorithm was developed in Escobar (1988), well before Gibbs samplers were widely used in the statistics literature. Recall the notation from §3.1.1 with $\theta_j^\star$, $j = 1, \ldots, k_{n-1}$ denoting the unique values among $\{\theta_1, \ldots, \theta_{n-1}\}$, $n_{n-1,j}$ denoting the number of θ_i equal to $\theta_j^\star$, and (s_i) denoting the cluster membership indicators with $s_i = j$ if $\theta_i = \theta_j^\star$. Then

$$\theta_n \mid \theta_{n-1}, \ldots, \theta_1 \sim \sum_{j=1}^{k_{n-1}} \frac{n_{n-1,j}}{M+n-1}\delta_{\theta_j^\star} + \frac{M}{M+n-1}G_0.$$

Since sequences generated by a species sampling model are exchangeable, this expression gives us the form of the full conditional prior distribution for any θ_i given $\boldsymbol{\theta}_{-i} = (\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_n)$. To see this, just permute the order of the observations so that θ_i becomes the last observation in the sequence. Multiplying by the likelihood $p(y_i \mid \theta_i)$ we find the full conditional posterior distribution for θ_i

$$\begin{aligned}\theta_i \mid \boldsymbol{\theta}_{-i}, \mathbf{y} &\propto \sum_{j=1}^{k^-} n_j^- p(y_i \mid \theta_j^{\star -})\delta_{\theta_j^{\star -}} + M\, p(y_i \mid \theta_i)G_0(\theta_i) \\ &= \sum_{j=1}^{k^-} \left\{n_j^- p(y_i \mid \theta_j^{\star -})\right\}\delta_{\theta_j^{\star -}} + \left\{M \int p(y_i \mid \theta_i) dG_0(\theta_i)\right\} p(\theta_i \mid y_i, G_0),\end{aligned} \tag{3.11}$$

where the superscript $^-$ represents the appropriate quantity with θ_i excluded from the sample. In the last term, $p(\theta_i \mid y_i, G_0) = p(y_i \mid \theta_i)dG_0(\theta_i)/\int p(y_i \mid \theta_i)dG_0(\theta_i)$ is the posterior on θ_i in a singleton cluster, and $\int p(y_i \mid \theta_i)dG_0(\theta_i)$ is the (prior) marginal distribution for y_i under G_0.

The previous results lead to a Gibbs sampler for (θ_i) that proceeds by iteratively sampling each θ_i, which is either equal to one of the unique $\theta_j^\star$s with probability proportional to $n_j^- p(y_i \mid \theta_j^\star)$, or sampled from the posterior distribution based solely on y_i with probability $M \int p(y_i \mid \theta_i)dG_0(\theta_i)$.

The described algorithm tends to mix very slowly when the mixture components are well separated. A faster mixing Markov chain is achieved by including an additional transition probability. Noting that sampling θ_i implies a new value for s_i too, the complete conditional posterior probability (3.11) can be characterized as $p(\theta_i, s_i \mid \mathbf{s}^-, \boldsymbol{\theta}^{\star -}, \mathbf{y})$. A more efficient sampler proceeds by first sampling the indicators from $p(s_i \mid \mathbf{s}^-, \mathbf{y})$ sequentially, and then sampling each $\theta_j^\star$ from $p(\theta_j^\star \mid \mathbf{y}, \mathbf{s})$.

To find $p(s_i \mid \mathbf{s}^-, \mathbf{y})$ note first that (3.11) can be written as a hierarchical model with

$$p(s_i = j \mid \mathbf{s}^-, \boldsymbol{\theta}^{\star -}, \mathbf{y}) \propto \begin{cases} n_j^- p(y_i \mid \theta_j^{\star -}) & j = 1, \ldots, k^- \\ M \int p(y_i \mid \theta_i) dG_0(\theta_i) & j = k^- + 1 \end{cases}$$

and

$$p(\theta_i \mid s_i = j, \mathbf{s}^-, \boldsymbol{\theta}^{\star -}, \mathbf{y}) = \begin{cases} \delta_{\theta_j^{\star -}} & j = 1, \ldots, k^- \\ p(\theta_i \mid y_i, G_0) & j = k^- + 1. \end{cases} \tag{3.12}$$

Marginalizing w.r.t. θ_i, but still conditioning on $\boldsymbol{\theta}^{\star -}$, we simply drop the last line. Let $\boldsymbol{y}_j^{\star -} = (y_\ell;\ s_\ell = j \text{ and } \ell \neq i)$ denote the observations in the j-th cluster without y_i. Finally, we remove $\theta_j^{\star -}$ from the conditioning set by integrating with respect to $p(\theta_j^{\star -} \mid \mathbf{s}^-, \mathbf{y}) = p(\theta_j^{\star -} \mid \boldsymbol{y}_j^{\star -})$ and get:

$$p(s_i = j \mid \mathbf{s}^-, \mathbf{y}) \propto \begin{cases} n_j^- \int p(y_i \mid \theta_j^{\star -}) dp(\theta_j^{\star -} \mid \boldsymbol{y}_j^{\star -}) & j \leq k^- \\ M \int p(y_i \mid \theta_i) dG(\theta_i) & j = k^- + 1. \end{cases} \tag{3.13}$$

The full conditional posterior for $\theta_j^\star$ is proportional to

$$p(\theta_j^\star \mid \mathbf{s}, \mathbf{y}) \propto G_0(\theta_j^\star) \prod_{\{i : s_i = j\}} p(y_i \mid \theta_j^\star). \tag{3.14}$$

When $G_0(\theta)$ is conjugate to $p(y_i \mid \theta)$, all of $\int p(y_i \mid \theta_j^{\star -}) dp(\theta_j^{\star -} \mid \boldsymbol{y}_j^{\star -})$, $\int p(y_i \mid \theta_i) dG_0$, and $p(\theta_j^\star \mid \mathbf{s}, \mathbf{y})$ are usually available in closed form and implementation of the algorithm is straightforward.

Example 10 (DPM with Gaussian kernels) *Consider a location mixture of Gaussian kernels, with $p(y_i \mid \theta_i) = \mathsf{N}(\theta_i, \sigma^2)$, and a conjugate baseline measure $G_0 = \mathsf{N}(m, B)$. In that case,*

$$\int p(y_i \mid \theta_i) dG_0(\theta_i) = \mathsf{N}(y_i \mid m, B + \sigma^2),$$

while

$$\int p(y_i \mid \theta_j^{\star -}) dp(\theta_j^{\star -} \mid \boldsymbol{y}_j^{\star -}) = \mathsf{N}(y_i \mid m_j^-, V_j^- + \sigma^2),$$

with $1/V_j^- = 1/B + n_j^-/\sigma^2$ and $m_j^- = V_j^-(m/B + 1/\sigma^2 \sum_{h \in S_j^-} y_h)$. Here $S_j^- = \{h \neq i : \ s_h = j\}$. Also, $p(\theta_j^\star \mid \mathbf{y}) = \mathsf{N}(m_j, V_j^2)$ with the expressions for m_j and V_j being the formulas for m_j^- and V_j^-, but now without excluding the i-th observation.

Non-conjugate models

When $p(y \mid \theta)$ and $G_0(\theta)$ are not conjugate, the integral $\int p(y_i \mid \theta_i) dG_0(\theta_i)$ is often analytically intractable. In that case the collapsed Gibbs samplers require that the integral be approximated numerically, making the implementation inefficient, particularly in high dimensional problems. To overcome this issue, a number of alternative collapsed samplers have been devised to accommodate non-conjugate models; a common feature of most of these methods is that they replace the predictive $\int p(y_i \mid \theta_i) dG_0(\theta_i)$ by $p(y_i \mid \theta_{k^-+1}^\star)$, where $\theta_{k^-+1}^\star$ is a random draw from G_0.

This can be justified on the basis of on an auxiliary probability model that includes the values of the parameters $\theta^\star_{k^-+1}, \ldots, \theta^\star_n$ for hypothetical empty clusters.

We start by describing the "no-gaps" algorithm introduced by MacEachern and Müller (1998). The name derives from the fact that the description of the algorithm relies on the no gaps convention, i.e., occupied clusters are consecutively labeled from 1 to k. As before, the algorithm proceeds by first sampling the indicators (s_i) conditional on $(\theta^\star_j)$, and then sampling the component-specific $(\theta^\star_j)$ conditional on (s_i).

To sample s_i we consider two cases. If in the currently imputed state $n_{s_i} > 1$ then we sample s_i from

$$p(s_i = j \mid \theta^\star_1, \ldots, \theta^\star_n, \mathbf{y}) \propto \begin{cases} n_j^- p(y_i \mid \theta^\star_j) & j = 1, \ldots, k^- \\ \frac{M}{k^-+1} p(y_i \mid \theta^\star_j) & j = k^- + 1. \end{cases} \tag{3.15}$$

On the other hand, if $n_{s_i} = 1$, i.e., y_i is currently forming a singleton cluster on its own, then with probability $(k^- - 1)/k^-$ we leave s_i unchanged, and with probability $1/k^-$ we resample s_i according to (3.15). See MacEachern and Müller (1998) for a justification.

Given the indicators (s_i), the component specific parameters $\theta^\star_1, \ldots \theta^\star_n$ are conditionally independent and can be sampled from the full conditional in (3.14). Since the model is not conjugate, this might require Metropolis-Hastings steps to sample $p(\theta^\star_j \mid \mathbf{s}, \mathbf{y})$. Cluster-specific $\theta^\star_{k+1}, \ldots,$ for hypothetical future clusters are sampled from G_0. However in actual implementation, when G_0 is a distribution for which a direct sampler is available, then we do not need to store the values $\theta^\star_{k+1}, \ldots, \theta^\star_n$, as they can be generated when and as needed to evaluate (3.15). However, one detail in the described MCMC is the following implication. Updating s_i in (3.15) might create new empty clusters when s_i is moved from a current singleton cluster, say $s_i = j_0$, to another existing cluster, $s_i = j \neq j_0$. The currently imputed $\theta^\star_{j_0}$ for the now empty cluster j_0 remains unchanged. In particular, it is not replaced by a draw from G_0.

The "no gaps" algorithm is easy to implement, but mixes slowly due to the reduced probability of opening new clusters. More general algorithms were proposed by Neal (2000), who noted that the joint posterior of the any set of parameters (s_i) and $(\theta^\star_j)$ can be evaluated as

$$p((s_i), (\theta^\star_j) \mid \mathbf{y}) \propto p(s_1, \ldots, s_n) \prod_{j=1}^{k} G_0(\theta^\star_j) \prod_{i=1}^{n} p(y_i \mid \theta^\star_{s_i}), \tag{3.16}$$

where $p(s_1, \ldots, s_n) \propto M^{k-1} \prod_{j=1}^{k} (n_j - 1)!$ (recall equation (3.5)). In principle, this joint distribution can be combined with any reversible proposal to develop Metropolis-Hastings transition probabilities for DP mixture models. As one example, consider making proposals $\widetilde{\theta}_i$ for θ_i (and thus implicitely for s_i) by a draw from the prior conditional:

$$p(\widetilde{\theta}_i) \propto \sum_j n_j^- \delta_{\theta^\star_j -}(\theta_i) + M\, G_0(\theta_i).$$

The acceptance probability is

$$\pi = \min \left\{ 1, \frac{p(y_i \mid \theta_{\widetilde{s}_i})}{p(y_i \mid \theta_{s_i})} \right\}.$$

Then, as in the "no gaps" algorithm, all the of the component parameters $\theta_j^\star$s can be resampled according to (3.14). Other variations of this approach are described in Neal (2000).

Random baseline measures

The DP mixture in (3.10) is often extended by assigning prior distributions for M or for hyperparameters ϕ that index G_0. Since M implicitly controls the number of clusters, a prior on M allows us to introduce prior uncertainty about the distribution of the number of clusters k. Similarly, a prior on G_0 allows us to reflect uncertainty on aspects of the distribution such as the "closeness" or the "size" of the clusters.

Consider a baseline measure that is indexed with hyperparameters ϕ, $G_0(\theta \mid \phi)$, and augment the model with a hyperprior $p(\phi)$ on ϕ. Note that, from the definition of the Dirichlet process, the values of $\theta_1^\star, \ldots, \theta_k^\star$ are independent draws rom G_0. Hence, the full conditional posterior for ϕ is simply

$$p(\phi \mid \ldots) \propto p(\phi) \prod_{j=1}^{k} G_0(\theta_j^\star \mid \phi).$$

When $G_0(\theta \mid \phi)$ and $p(\phi)$ are chosen as a conjugate pair, this posterior reduces to a well known distributions.

Example 11 (DPM with Gaussian kernels, continued) *Consider again the location mixture of Gaussian kernels from Example 10. The base measure G_0 is indexed by $\phi = (m, B)$. It is natural to extend the model with conditionally conjugate priors $p(m) = \mathsf{N}(m_0, D)$ and $p(B) = \mathsf{IGamma}(a, b)$, where $\mathsf{IGamma}(a, b)$ denotes the inverse Gamma distribution with shape parameter a and mean $b/(a-1)$ for $a > 1$. We can interpret m as representing the center of mass for the cluster locations, while B represents the average distance between cluster centers. The full conditional distributions associated with m reduce to*

$$m \mid \ldots \sim \mathsf{N}(m_1, D_1),$$

with $D_1^{-1} = 1/D + k/B$ and $m_1 = D_1(m_0/D + 1/B \sum_{j=1}^{k} \theta_j^\star)$. On the other hand, the full conditional posterior distribution for B is simply

$$B \mid \ldots \sim \mathsf{IGamma}(a_1, b_1),$$

with $a_1 = a + k/2$ and $b_1 = b + \sum_{j=1}^{k} (\theta_j^\star - m)^2/2$.

On the other hand, to estimate the precision parameter M we can use (3.6),

$$p(k \mid M) \propto M^k \frac{\Gamma(M)}{\Gamma(M+n)} = M^k \frac{(M+n)}{M\Gamma(n)} \int_0^1 \eta^M (1-\eta)^{n-1} d\eta.$$

The last equality exploits the normalizing constant of a $\mathsf{Be}(M+1, n)$ beta distribution. Therefore, we can devise a sampler for M by first introducing a latent variable η such that

$$\eta \mid M, k, \ldots \sim \mathsf{Beta}(M+1, n).$$

If, a priori, $M \sim \mathsf{Gamma}(c, d)$, then also

$$M \mid \eta, k, \ldots \sim \frac{c+k-1}{c+k-1+n(d-\log\{\eta\})}\mathsf{Gamma}(c+k, d-\log\{\eta\}) + \frac{n(d-\log\{\eta\})}{c+k-1+n(d-\log\{\eta\})}\mathsf{Gamma}(c+k-1, d-\log(\eta)).$$

This clever auxiliary variable sampler was first introduced in Escobar and West (1995).

3.3.2. Slice Samplers

Slice samplers for DP mixture models were introduced in Walker (2007). Unlike collapsed samplers, slice samplers do not marginalize over G, but use the stick-breaking representation of the process. The discrete nature of $G \sim \mathsf{DP}(M, G_0)$ allows us to write the DPM model as

$$p(y_i \mid (w_h), (\widetilde{\theta}_h)) = \int p(y_i \mid \theta_i) dG(\theta_i) = \sum_{h=1}^{\infty} w_h p(y_i \mid \widetilde{\theta}_h). \tag{3.17}$$

This expression is computationally intractable because of the infinite sum. However, a clever model augmentation with latent variables $u_1, \ldots, u_n$, $0 \le u_i \le 1$, reduces (3.17) to a finite sum. Consider the augmented model

$$p(y_i, u_i \mid (w_h), (\widetilde{\theta}_h)) = \sum_{h=1}^{\infty} I(u_i < w_h) p(y_i \mid \widetilde{\theta}_h), \tag{3.18}$$

where $I(A)$ denotes the indicator function on the set A. Integating w.r.t. u_i reduces the model again to (3.17), as desired. The important trick is that we have $w_h > u_i$ only for a finite number of weights. Hence, conditioning on the latent variables (u_i) has the effect of transforming the infinite mixture into a finite mixture with a fixed number $N_u = \sum_h I(u_i < w_h)$ of components. We agument the model a second time with latent indicators $r_i \in \{1, 2, \ldots\}$ to

$$p(y_i, u_i, r_i \mid (w_h), (\widetilde{\theta}_h)) = I(u_i < w_{r_i}) p(y_i \mid \widetilde{\theta}_{r_i}). \tag{3.19}$$

Marginalizing w.r.t. r_i we immediately get the sum in (3.18), while integrating over r_i and u_i yields (3.17), as desired. The joint distribution of the data, the latent indicators r and u in the extended model is

$$p(\mathbf{y}, \mathbf{u}, \mathbf{r} \mid (w_h), (\widetilde{\theta}_h)) = \prod_{i=1}^{n} I(u_i < w_{r_i}) p(y_i \mid \widetilde{\theta}_{r_i}). \tag{3.20}$$

Note that the indicators r_i in (3.20) are different from the indicators s_i in (3.2). The latter are cluster membership indicators that match θ_i with the unique values $\theta_j^\star$. The earlier are indicators that match θ_i with the point masses $\widetilde{\theta}_h$ in (3.2). However, the two are related because the $\theta_j^\star$ are a sample of $\widetilde{\theta}_h$. With another set of indicators, $t_j = h$ when $\theta_j^\star = \widetilde{\theta}_h$ we would have $r_i = t_{s_i}$.

Working with (3.20) allows for simple updates for all model parameters. In particular, the weights can be updated through the stick-breaking ratios by sampling

$$v_h \mid \ldots, \cancel{(u_i)} \sim \mathsf{Beta}\left(1 + n_h, M + \sum_{k>h} n_k\right),$$

where $n_h = \sum_{i=1}^{n} I(r_i = h)$ is the number of observations such that $r_i = h$. Similarly, the atoms $\widetilde{\theta}_h$ for the occupied components are sampled from

$$p(\widetilde{\theta}_h \mid \ldots) \propto G_0(\widetilde{\theta}_h) \prod_{\{i:r_i=h\}} p(y_i \mid \widetilde{\theta}_h),$$

while the atoms associated with empty components, i.e., $n_h = 0$, can be sampled, on demand, directly from G_0.

Finally, the latent variables u_i are, a posteriori, uniformly distributed

$$u_i \mid \ldots \sim \mathsf{Uni}[0, w_{r_i}],$$

and the indicators are updated from the full conditional

$$\mathsf{Pr}(r_i = h \mid \ldots) \propto I(w_h > u_i) p(y_i \mid \widetilde{\theta}_h).$$

Only a finite number of components satisfy the constrain $w_h > u_i$. Therefore, the normalizing constant for this last full conditional distribution can be computed in closed form. Let $H_i(u_i) = \{h : \ w_h > u_i\}$. Then

$$p(r_i = h \mid \ldots) = \begin{cases} \frac{p(y_i|\widetilde{\theta}_h)}{\sum_{\{k \geq 1 : w_k > u_i\}} p(y_i|\widetilde{\theta}_k)} & h \in H_i(u_i) \\ 0 & \text{otherwise.} \end{cases}$$

3.3.3. Retrospective Samplers

Retrospective samplers for Dirichlet process mixtures were developed by Roberts and Papaspiliopoulos (2008). Like the slice sampler, the retrospective sampler is based on an explicit representation of the mixing distribution G; to avoid the problem of storing an infinite number of weights and atoms, a Metropolis Hastings step is used to allocate observations to components, while the parameters associated with empty components are sampled retrospectively as they become necessary. The same algorithm is developed in Nieto-Barajas *et al.* (2012).

To formalize the idea of the retrospective sampler, consider simulating observation from the prior model, i.e., simulating an i.i.d. sequence $\theta_1, \ldots, \theta_n$ where $\theta_i \mid G \sim G$ where $G \sim \mathsf{DP}(M, G_0)$. This can be done directly by exploiting the species sampling representation of the process in (3.3). Alternatively, we can first simulate the distribution G using the stick-breaking construction in (3.2), and then sample θ_i conditional on G. Under the second approach we can avoid the difficulties associated with having a countably infinite number of components by utilizing the following scheme.

1. Simulate $w_1 = v_1 \sim \mathsf{Beta}(1, M)$ and $\widetilde{\theta}_1 \sim G_0$, and set $H = 1$, $i = 1$ and $w_0 = 0$.
2. For $i = 1, \ldots, n$
   (a) Simulate $U_i \sim \mathsf{Uni}[0, 1]$.
   (b) If for some $k \leq H$ we have $\sum_{h=0}^{k-1} w_h < U_i \leq \sum_{h=0}^{k-1} w_h + w_k$, then set $\theta_i = \widetilde{\theta}_k$.
   (c) If $U_i > \sum_{h=0}^{H} w_h$, then simulate $v_{H+1} \sim \mathsf{Beta}(1, M)$ and $\widetilde{\theta}_{H+1} \sim G_0$, and set $w_{H+1} = v_{H+1} \prod_{k<H+1}\{1 - v_H\}$ and $H = H + 1$. Go back to step (b).

In words, we generate the weights w_h and point masses only as and when needed. That is all!

We proceed now to describe the posterior sampler for the DPM. As in §3.3.2, we introduce indicator variables (r_i) such that $\theta_i = \widetilde{\theta}_{r_i}$ and define $n_h = \sum_{i=1}^{n} I(r_i = h)$ as the number of observations assigned to component h.

As with the slice sampler, the full conditionals for the v_hs and the $\widetilde{\theta}_h$ are independent and given by

$$v_h \mid \ldots \sim \mathsf{Beta}\left(1 + n_h, M + \sum_{k>h} n_k\right) \tag{3.21}$$

and

$$p(\widetilde{\theta}_h \mid \ldots) \propto G_0(\widetilde{\theta}_h) \prod_{\{i:r_i=h\}} p(y_i \mid \widetilde{\theta}_h). \tag{3.22}$$

On the other hand, the full conditional distribution for the indicator variables is given by

$$\mathsf{Pr}(r_i = h \mid \ldots) \propto w_h p(y_i \mid \widetilde{\theta}_h), \qquad h = 1, 2, \ldots$$

Since computation of the normalizing constant involves a sum over an uncountable number of terms, directly sampling from this distribution is not feasible. To avoid this issue, Roberts and Papaspiliopoulos (2008) describe a Metropolis Hastings algorithm for $\mathbf{s} = (s_1, \ldots, s_n)$. More specifically, for every $i = 1, \ldots, n$ a proposal $\widetilde{\mathbf{s}} = (\widetilde{s}_1, \ldots, \widetilde{s}_n)$ is created by setting $\widetilde{s}_j = s_j$ for $j \neq i$ and generating $\widetilde{s}_i$ according to the proposal distribution

$$p(\widetilde{s}_i = h) \propto \begin{cases} w_h p(y_i \mid \widetilde{\theta}_h) & h \leq \max_i\{r_i\} \\ M_i(\mathbf{s}) w_h & h > \max_i\{r_i\}, \end{cases}$$

where $M_i(\mathbf{s})$ is a user-selected parameter. The corresponding normalizing constant is given in this case by

$$c_i(\mathbf{s}) = \sum_{h=1}^{\max\{r_i\}} w_h p(y_i \mid \widetilde{\theta}_h) + M_i(\mathbf{s}) \left(1 - \sum_{h=1}^{\max\{r_i\}} w_h\right).$$

The proposed value is accepted with probability

$$\alpha_i(\mathbf{s}, \widetilde{\mathbf{s}}) = \begin{cases} 1 & \widetilde{s}_i \leq \max\{r_i\} \text{ and } \max_i\{\widetilde{s}_i\} = \max_i\{r_i\} \\ \min\left\{1, \frac{c_i(\mathbf{s}) M_i(\widetilde{\mathbf{s}})}{c_i(\widetilde{\mathbf{s}}) p(y_i \mid \widetilde{\theta}_{r_i})}\right\} & \widetilde{s}_i \leq \max_i\{r_i\} \text{ and } \max_i\{\widetilde{s}_i\} < \max_i\{r_i\} \\ \min\left\{1, \frac{c_i(\mathbf{s}) p(y_i \mid \widetilde{\theta}_{\widetilde{s}_i})}{c_i(\widetilde{\mathbf{s}}) M_i(\mathbf{s})}\right\} & \widetilde{s}_i > \max_i\{r_i\}. \end{cases}$$

If an observation is allocated to a new component (i.e., a proposal $\widetilde{s}_i > \max\{r_i\}$ is accepted), then the necessary values for the v_hs and $\widetilde{\theta}_h$s are sample retrospectively from (3.21) and (3.22). The constant $M_i(\mathbf{s})$ is selected so that

$$M_i(\mathbf{s}) = \max_{h \leq \max_i\{r_i\}} \left\{p(y_i \mid \widetilde{\theta}_h)\right\}$$

in order to generate a sampler that is geometrically ergodic.

3.3.4. Other Computational Approaches

We have but scratched the surface of possible Markov chain Monte Carlo methods for inference under the Dirichlet process mixture model. For example, Dahl (2003), Jain and Neal (2004) and Jain and Neal (2007) propose split-merge collapsed samplers that provide mechanisms to make global moves in the space of partitions that are induced by the DP prior. Alternatively, MacEachern *et al.* (1999) and Carvalho *et al.* (2010) consider sequential Monte Carlo approaches that are particularly useful for problems where observations are collected sequentially and it is necessary to update the model after each observation is received. Finally, Blei and Jordan (2006) propose a variational algorithm that is computationally efficient for large sample sizes.

3.4. The Finite DP

The Dirichlet process mixture model potentially allows for an infinite number of clusters as $n \to \infty$. However, for any finite sample size n the number k of occupied components cannot be greater than n, and is typically much smaller than that. This suggests that instead of dealing with a countably infinite number of components, we could work with mixtures with a large but finite number of components. This should simplify computation while retaining most of the theoretical advantages of the Dirichlet process.

The first approach we discuss to construct truncated versions of the Dirichlet process is the $\epsilon-$DP of Muliere and Tardella (1998). For any $\epsilon \in (0,1)$, a random distribution G^ϵ is said to follow an ϵ-Dirichlet process if it admits a representation of the form

$$G^\epsilon(\cdot) = \sum_{h=1}^{H_\epsilon} w_h \delta_{\widetilde{\theta}_h}(\cdot) + \left\{1 - \sum_{h=1}^{H_\epsilon} w_h\right\} \delta_{\widetilde{\theta}_{H_\epsilon+1}}(\cdot),$$

where $(\widetilde{\theta}_h)$ is a collection of i.i.d. draws from the baseline measure G_0, $w_h = v_h \prod_{k<h}(1-v_k)$, where (v_h) is another i.i.d. sequence such that $v_h \sim \mathsf{Beta}(1, M)$, and $H_\epsilon = \inf\{m \in \mathbb{N} : \sum_{h=1}^m w_h \geq 1-\epsilon\}$.

The definition of the ϵ-Dirichlet process is analogous to that of the (regular) Dirichlet process, but the sum stops after a random number of draws, $H_\epsilon \sim \mathsf{Poi}(-M\log\epsilon)$, and the remaining mass (which, by construction, must be no larger than ϵ) is assigned to the last atom. By bounding the probability associated with this last atom, the definition ensures that the total variation distance between the finite and the infinite versions of the process is no larger than ϵ. Indeed, let $(\widetilde{\theta}_h)$ and (v_h) be two i.i.d. sequences such that $\widetilde{\theta}_h \sim G_0$ and $v_h \sim \mathsf{Beta}(1, M)$, and define

$$G(\cdot) = \sum_{h=1}^{\infty} w_h \delta_{\widetilde{\theta}_h}, \qquad G^\epsilon(\cdot) = \sum_{h=1}^{H_\epsilon} w_h \delta_{\widetilde{\theta}_h}(\cdot) + \left\{1 - \sum_{h=1}^{H_\epsilon} w_h\right\} \delta_{\widetilde{\theta}_{H_\epsilon+1}}(\cdot),$$

where $H_\epsilon = \inf\{m \in \mathbb{N} : \sum_{h=1}^m w_h \geq 1-\epsilon\}$. Then

$$\sup_B \{|G(B) - G^\epsilon(B)|\} \leq \epsilon.$$

Naturally, as $\epsilon \to 0$, draws from the ϵ-DP converge (in total variation norm) to the draws from a regular DP. Hence, the class of ϵ-DPs is dense, in the sense that it

is rich enough to approximate arbitrarily well any distribution on the underlying probability space.

An alternative definition of a truncated Dirichlet process was introduced in Ishwaran and James (2001, 2002). Rather than using a random stopping rule for the number of point masses that ensures a bound on the total variation norm, they argue instead for using a fixed number of atoms H and study the behavior of the random distribution as the number of atoms grows. In particular, Ishwaran and James (2001) show that the marginal distributions for a sample $(x_1, \ldots, x_n)$ under $G = \sum_{h=1}^{\infty} w_h \delta_{\widetilde{\theta}_h}$ and $G^H = \sum_{h=1}^{H} w_h \delta_{\widetilde{\theta}_h}$ are almost indistinguishable and the L_1 distance between these marginal distribution is bounded by $4ne^{-(H-1)/M}$. This results suggest that, as long as M is not very large, small H already obtain a good approximation, even if n is large.

Hierarchical models based on finite DPs have some important computational advantages over models based on infinite DPs. For example, since the number of atoms is finite, techniques for posterior inference on finite mixture models can be employed to perform estimation on finite DP mixtures. Indeed, the main motivation of Ishwaran and James (2001) and Ishwaran and James (2002) is to develop alternative computational algorithms for stick-breaking priors, which they call blocked Gibbs samplers.

We introduce latent indicators $r_1, \ldots, r_n$ such that $\theta_i = \widetilde{\theta}_{r_i}$. The joint distribution is then given by

$$p\{(r_i), (\widetilde{\theta}_h), (v_h) \mid \mathbf{y}\} \propto \prod_{i=1}^{n} p(y_i \mid \widetilde{\theta}_{r_i}) \prod_{i=1}^{n} p\{r_i \mid (v_h)\} \prod_{h=1}^{H} dG_0(\widetilde{\theta}_h) \prod_{h=1}^{H-1} p(v_h \mid M).$$

From this we find

$$p(\widetilde{\theta}_h \mid \ldots) \propto G_0(\widetilde{\theta}_h) \prod_{\{i:r_i=h\}} p(y_i \mid \widetilde{\theta}_h). \tag{3.23}$$

Similarly, the stick-breaking weights (v_h) are conditionally independent with

$$v_h \mid \ldots \sim \mathsf{Beta}\left(1 + n_h, M + \sum_{k>h} n_k\right), \tag{3.24}$$

where $n_h = \sum_{i=1}^{n} I(r_i = h)$ is the number of observations such that $r_i = h$. Finally, the posterior full conditional distribution for r_i is

$$\mathsf{Pr}(r_i = h \mid \ldots) \propto w_h p(y_i \mid \widetilde{\theta}_h), \tag{3.25}$$

where $w_h = v_h \prod_{k<h}(1 - v_h)$ for $h < H$ and $w_H = \prod_{k \leq H}(1 - v_h)$.

Notice the similarities between this algorithm and the slice sampler we described in §3.3.2. The structure of both algorithms is almost identical, with the main distinction being that in the blocked Gibbs sampler the number of components H is predetermined before running the algorithm, while in the slice sampler the number of components being used for computation (which is typically larger than the number of occupied components) is determined dynamically as part of the algorithm.

Another advantage of working with a truncated version of the Dirichlet process is that computation for functionals of the random distribution G is greatly simplified because G can be explicitly evaluated. For example, the predictive distribution for

a new observation y_{n+1} can be easily evaluated by noting that, under the truncated model,

$$p(y_{n+1} \mid y_1, \ldots, y_n) = \mathsf{E}\left\{ \sum_{h=1}^{H} w_h p(y_{n+1} \mid \widetilde{\theta}_h) \mid y_1, \ldots, y_n \right\}.$$

Other functionals of G^H can be easily computed in the same way (see also §3.6).

3.5. Mixtures of DP

In §3.3.1 we discussed the possibility of making the baseline measure random. In this section we formalize this construction through the introduction of mixtures of DPs (MDP). MDP's where first introduced by Antoniak (1974) as a generalization of the Dirichlet process. In contrast to the DPM, where the DP is the prior model for the mixing measure in a mixture of parametric distributions, the MDP arises when the baseline measure G_0 and/or the concentration parameter M are random. I.e., $G_0 = G_{0,\eta}$ and/or $M = M_\eta$ are indexed by η and we define a random probability measure by

$$G \mid \eta \sim \mathsf{DP}(M_\eta, G_{0,\eta}), \qquad \eta \sim H(\eta).$$

Then, G is said to follow a mixture of Dirichlet processes with precision M_η, baseline measure $G_{0,\eta}$ and mixing distribution H, or simply $G \sim \int \mathsf{DP}(M_\eta, G_{0,\eta}) dH(\eta)$.

Note that the MDP and the DPM are entirely different models. The earlier mixes some kernel with respect to a DP random measure. The latter mixes DP random measure with respect to a prior on hyperparameters. Nevertheless, there is a natural connection between both models (see Antoniak, 1974). The posterior law for the mixing distribution G in a DPM model follows a MDP distribution. Consider

$$y_i \mid \theta_i \sim p(y_i \mid \theta_i), \qquad \theta_i \mid G \sim G, \qquad G \sim \mathsf{DP}(M, G_0),$$

then, the law of $G \mid y_1, \ldots, y_n$ can be written as

$$G \mid y_1, \ldots, y_n \sim \int \mathsf{DP}\left(M + n, \frac{M G_0 + \sum_{i=1}^{n} \delta_{\boldsymbol{\theta}^\star_{r_i}}}{M + n} \right) dP(s_1, \ldots, s_n, \theta^\star_1, \ldots, \theta^\star_k \mid y_1, \ldots, y_n). \tag{3.26}$$

The posterior distribution over G induced by a DPM is simply a MDP!

3.6. Functionals of DPs

3.6.1. Inference for Non-linear Functionals of DP

We return to inference for the DPM model (3.9) again,

$$y_i \mid G \sim F = \int p(y_i \mid \theta) G(\mathrm{d}\theta), \qquad G \sim \mathsf{DP}(M, G_0).$$

Sometimes investigators are interested in posterior inference for functionals of the unknown distribution F. For example, in density estimation with an unknown distribution F one might be interested in computing $\mathsf{E}\{f \mid y_1, \ldots, y_n\}$ as a point

estimate of the density f associated with the unknown distribution F. Alternatively, we might be interested in providing posterior distributions for quantiles of F, i.e., $p(F^{-1}(\gamma) \mid y_1, \ldots, y_n)$ for some $\gamma \in (0,1)$ where F^{-1} is the inverse c.d.f. of F and $\gamma \in (0,1)$ is a prespecified percentile.

First, we consider inference for functionals of F under the collapsed sampler discussed in §3.3.1. For linear functionals of F, we can explicitly marginalize with respect to G and compute point estimators directly from the sampler output. For example, we can compute $\mathsf{E}\{f(y^\star) \mid y_1, \ldots, y_n\}$ by changing the order of integration in

$$\mathsf{E}\{f(y^\star) \mid y_1, \ldots, y_n\} = \mathsf{E}\left\{\int\int p(y^\star \mid \theta) G(\mathrm{d}\theta)\right\} = \int p(y^\star \mid \theta) G^\star(\mathrm{d}\theta),$$

where $G^\star = \mathsf{E}\{G \mid y_1, \ldots, y_n\}$. Since $G \mid y_1, \ldots, y_n$ is a MDP, as we discussed at the end of §3.5 we have

$$\begin{aligned}(3.27)\quad &\mathsf{E}\{f(y^\star) \mid y_1, \ldots, y_n\} \\ &= \int \left\{ \frac{M}{M+n} \int p(y^\star \mid \theta) G_0(\mathrm{d}\theta) + \sum_{j=1}^{k^{(b)}} \frac{n_j}{n+M} p(y^\star \mid \theta_j^\star) \right\} \\ &\qquad\qquad dp(s_1, \ldots, s_n, \theta_1^\star, \ldots, \theta_k^\star \mid y_1, \ldots, y_n).\end{aligned}$$

Given a Monte Carlo posterior sample $(s_1^{(b)}, \ldots, s_n^{(b)}, \theta_1^{\star(b)}, \ldots, \theta_{k^{(b)}}^{\star(b)})_{b=1}^B$ from

$$p(s_1, \ldots, s_n, \theta_1^\star, \ldots, \theta_k^\star \mid y_1, \ldots, y_n),$$

(3.27) can be approximated by the Monte Carlo estimator

$$\begin{aligned}(3.28)\quad &\mathsf{E}\{f(y^\star) \mid y_1, \ldots, y_n\} \\ &= \frac{1}{B} \sum_{i=1}^{B} \left\{ \frac{M}{M+n} \int p(y^\star \mid \theta) G_0(\mathrm{d}\theta) + \sum_{j=1}^{k^{(b)}} \frac{n_j^{(b)}}{n+M} p(y^\star \mid \theta_j^{\star(b)}) \right\}.\end{aligned}$$

Alternatively one could evaluate $E[f(y^\star) \mid \mathbf{y}]$ as the posterior predictive $p(y_{n+1} \mid \mathbf{y})$ in a random sample $y_i \sim F$,

$$p(y_{n+1} \mid \mathbf{y}) = E[p(y_{n+1} \mid F, \mathbf{y}) \mid \mathbf{y}] = E[f(y_{n+1}) \mid \mathbf{y}],$$

which leads to the same MCMC estimate (3.28).

For non-linear functionals, and more generally, if we want to obtain the full posterior distribution of a given functional, we need to deal with the infinite dimensional mixing distribution G. Gelfand and Kottas (2002) exploit the fact that $p(G, s_1, \ldots, s_n, \theta_1^\star, \ldots, \theta_k^\star \mid y_1, \ldots, y_n)$ can be factorized as

$$\begin{aligned}&p(G, s_1, \ldots, s_n, \theta_1^\star, \ldots, \theta_k^\star \mid y_1, \ldots, y_n) \\ &\qquad = p(G \mid s_1, \ldots, s_n, \theta_1^\star, \ldots, \theta_k^\star) p(s_1, \ldots, s_n, \theta_1^\star, \ldots, \theta_k^\star \mid y_1, \ldots, y_n),\end{aligned}$$

where $p(G \mid s_1, \ldots, s_n, \theta_1^\star, \ldots, \theta_k^\star)$ is simply a Dirichlet process with base measure $G_1 \propto MG_0 + \sum_{j=1}^k n_j \delta_{\theta_j^\star}$. Hence, given a realization $(s_1^{(b)}, \ldots, s_n^{(b)}, \theta_1^{\star(b)}, \ldots, \theta_{k^{(b)}}^{\star(b)})$ from the posterior distribution, a realization from the posterior for G can be constructed as

$$G^{(b)}(\cdot) = \sum_{h=1}^{\infty} \varpi_h \delta_{\widetilde{\theta}_h}(\cdot)$$

where $\varpi_h = z_h \prod_{k<h}(1 - z_h)$, $z_h \sim \mathsf{Beta}(1, M + n)$ and $\widetilde{\theta}_h \sim G_1$.

In practice, an ϵ truncation of the DP (recall our discussion from §3.4) is used, so that we generate only a random (but finite!) number of atoms H_ϵ such that $H_\epsilon = \inf\{m \in \mathbb{N} : \sum_{h=1}^{m} \varpi_h > 1 - \epsilon\}$. Given $G^{(b)}$, we can evaluate any functional of F. For example, a sample from $p(F^{-1}(\gamma) \mid y_1, \ldots, y_n)$, can be obtained by computing (for each iteration of the MCMC) the value $q_\gamma^{(b)}$ such that

$$\gamma = \sum_{h=1}^{H_\epsilon} \varpi_h^{(b)} P(q_\gamma^{(b)} \mid \theta_h^{\star(b)}).$$

Here $P(y) = \int_{-\infty} p(y \mid \theta)$ is the c.d.f. of the kernel in (3.9). The values $q_\gamma^{(1)}, \ldots, q_\gamma^{(B)}$ can then be used to perform posterior inference on $F^{-1}(\gamma)$.

Finally, we consider inference for functionals of F under the slice and retrospective samplers that are discussed in §3.3.2 and §3.3.3. Both of these samplers rely on an explicit representation of the mixing distribution G. Therefore inference for functionals is, in principle, straightforward. However, a word of caution is in order. Even though both the slice and the retrospective samplers perform dynamically adaptive truncations of G, the accuracy of these truncations (in terms of how well they approximate the infinite dimensional G) is not predetermined. Hence, in practice we might need to explicitly represent (and simulate) additional mixture components in order to ensure that the posterior inference for the functionals of G is sensible. Note that this is not the case if an almost sure truncation is used because the value of H is predetermined beforehand to ensure a good approximation.

3.6.2. Centering the DP

A particular example of inference for functionals of a DP random probability measure arises in applications to mixed effects models. Recall the discussion of mixed effects models in §2.3. To be specific, assume a linear model

$$y_{ij} = \beta_0 + \beta_1 t_{ij} + b_{0i} + b_{1i} t_{ij} + \epsilon_{ij}, \tag{3.29}$$

with $\epsilon_{ij} \sim \mathsf{N}(0, \sigma^2)$, i.i.d. For example, y_{ij} might be logarithm of the prostate-specific antigen (log PSA) measurements for the i-th patient at time t_{ij}. Here (β_0, β_1) are fixed effects including intercept and overall growth rate of PSA, and $\mathbf{b}_i = (b_{0i}, b_{1i})$ are patient-specific random effects. The random effects are assumed to arise from a random effects distribution $\mathbf{b}_i \sim G$. When the investigator is not willing or able to assume a parametric model for G then we might complete the model with a BNP prior, for example,

$$\mathbf{b}_i \sim G, \qquad\qquad G \sim \mathsf{DP}(M, G_0),$$

with $G_0(\mathbf{b}) = \mathsf{N}(\mathbf{b} \mid 0, D)$ and a (conditionally) conjugate hyperprior on D. Let $\mu_G = \int x dG(x)$ and $\Sigma_G = \int (x - \mu_G)(x - \mu_G)' \, dG(x)$ denote the first and (centered) second moment of G. For a random probability measure G the moments μ_G and Σ_G become random variables. Even when G_0 is chosen to have a zero mean $\mu_{G_0} =$

0, the random moments μ_G are almost surely not zero. This greatly complicates interpretation of the fixed effects (β_0, β_1). Similarly, the elements of Σ_G are the variance components; reporting D, as is often done, only approximates Σ_G.

Inference on fixed effects in (3.29) is best formalized as inference on $(\boldsymbol{\beta} + \boldsymbol{\mu}_G)$, and inference on variance components requires the posterior distribution of Σ_G. Fortunately both are easily available from standard Monte Carlo output for MCMC posterior simulation under model (3.29). Li *et al.* (2010) give explicit formulas for the first two posterior moments of $(\boldsymbol{\beta} + \boldsymbol{\mu}_G)$ and Σ_G. All can be evaluated by postprocessing with an available Monte Carlo sample.

Chapter 4

Pólya Trees

4.1. Definition

An intuitively attractive way to construct RPMs is as a random histogram, with a fixed set of bins and random probability mass associated with each bin (for example, see Loredo, 2011). In anticipation of the upcoming discussion, we assume that the bins define 2^m partitioning subsets and index the subsets by an m-digit binary number $\epsilon = e_1 \cdots e_m$, $e_j \in \{0, 1\}$. Let $\Pi_m = \{B_\epsilon,\ \epsilon = e_1 e_2 \cdots e_m\}$ denote this partition of the sample space into 2^m bins or partitioning subsets. A random histogram could define an RPM G by defining the joint distribution $p(G(B_\epsilon);\ \ B_\epsilon \in \Pi_m)$.

Pólya trees arise as an extension of this idea where the size of the bins is made sequentially smaller. More specifically, consider sequentially refining a partition Π_m to Π_{m+1} by splitting $B_{e_1 \cdots e_m}$ into $B_{e_1 \cdots e_m} = B_{e_1 \cdots e_m 0} \cup B_{e_1 \cdots e_m 1}$ (see Figure 4.1.). The problem now arises to define $G(B_\epsilon)$ coherently across nested partitions, with $G(B_\epsilon) = G(B_{\epsilon 0}) + G(B_{\epsilon 1})$. The elegant solution is to define the random $G(B_\epsilon)$ through a sequence of conditional probabilities as

$$G(B_\epsilon) = \prod_{k=1}^{m} G(B_{e_1 \cdots e_{k-1} e_k} \mid B_{e_1 \cdots e_{k-1}}),$$

with the understanding that $B_\emptyset$ denotes the entire sample space. An RPM $p(G)$ is then defined by specifying a prior for the random splitting probabilities $Y_\epsilon = G(B_{\epsilon 0} \mid B_\epsilon)$ for any m-digit index $\epsilon = e_1 \cdots e_m$, $m > 0$.

The resulting construction is called a Pólya tree (PT) prior (Lavine, 1992, 1994; Mauldin *et al.*, 1992). The PT model assumes that $Y_\epsilon \sim \mathsf{Beta}(a_{\epsilon 0}, a_{\epsilon 1})$, independently across m. In general, an RPM with independent splitting probabilities Y_ϵ is known as tail-free with respect to Π. The PT can thus be characterized as a tail-free process with respect to a nested partition sequence Π and beta distributed random splitting probabilities.

Notice the similarities between the PT prior the class of neutral to the right (NTR) priors introduced in §1.2.7. Recall that NTR refers to independence of the normalized increments $G(t_{i-1}, t_i]/G(-\infty, t_i]$ for a partition with partition boundaries $t_0 = -\infty < t_1 \ldots < t_n < \infty$. In contrast, the tailfree property of the PT refers to independence across two sets of partitions.

In summary, the PT prior defines an RPM by assigning any partitioning subset B_ϵ the random probability

$$G(B_{e_1 \cdots e_m}) = \underbrace{\prod_{j=1; e_j=0}^{m} Y_{e_1 \cdots e_{j-1} 0}}_{\text{all } \textit{left} \text{ splits}} \underbrace{\prod_{j=1; e_j=1}^{m} (1 - Y_{e_1 \cdots e_{j-1} 0})}_{\text{all } \textit{right} \text{ splits}},$$

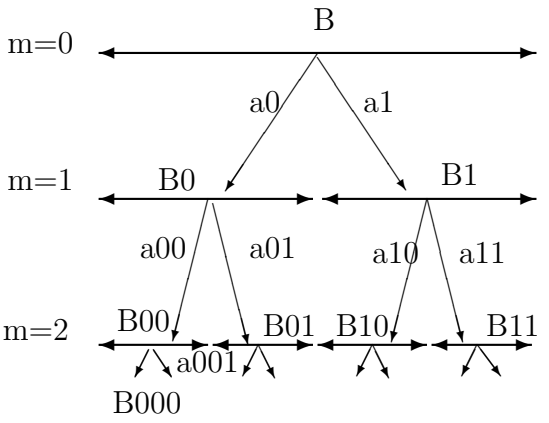


FIG 4.1. *PT: The diagram shows the nested sequence of partitions* $\Pi_m = \{B_\epsilon,\ \epsilon = e_1 \cdots e_m\}$ *with* $e_j \in \{0,1\}$. *The PT is defined by random splitting probabilities* $Y_{\epsilon 0} = G(B_{e_1 \cdots e_m 0} \mid B_{e_1 \cdots e_m})$ *with* $Y_{\epsilon 0} \sim \mathsf{Beta}(a_{e_1 \cdots e_m 0}, a_{e_1 \cdots e_m 1})$.

with independent beta priors $Y_{\epsilon 0} \sim \mathsf{Beta}(a_{\epsilon 0}, a_{\epsilon 1})$. The PT model is indexed with two sets of parameters, the nested sequence of partitions $\Pi = \{\Pi_m\}$ and the parameters $\mathcal{A} = \{a_\epsilon\}$ for the beta-distributed random splitting probabilities. Hence, we write $G \sim \mathsf{PT}(\Pi, \mathcal{A})$.

One of the important features of the PT prior is that it can generate continuous probability measures. A random probability measures $G \sim \mathsf{PT}(\mathcal{A}, \Pi)$ is absolutely continous with probability 1 when the $\alpha_{e_1 \cdot e_m}$ parameters increase suffiently fast with m. A popular choice is $\alpha_{\epsilon_1 \ldots \epsilon_m} = c\, m^2$. On the other hand, for decreasing α, the random probability measure can also be almost surely discrete. For example, for $\alpha_{e_1 \cdots e_m} = c/2^m$ the PT prior reduces to the special case of the DP prior.

One of the attractions of the PT model is the ease of centering the model at any desired prior mean G_0. One way to accomplish this centering is to fix the partitioning subsets B_ϵ as the dyadic quantiles of G_0. More specifically, let $z_\epsilon = \sum_{j=1}^m 2^{-e_j}$ and define $B_{e_1 \cdots c_m} = (G_0^{-1}(z_\epsilon), G_0^{-1}(z_\epsilon + 2^{-m})]$. At $m = 1$, the two subsets $\{B_0, B_1\}$ are simply below and above the median of G_0, at $m = 2$, the partitioning subsets are determined by the quartiles, etc. If the a_ϵ parameters are chosen to be symmetric, $a_{\epsilon 0} = a_{\epsilon 1}$, then it is easy to show that $E(G(B)) = G_0(B)$, i.e., the RPM is centered at G_0, as desired. Alternatively, the same centering can be achieved with an arbitrary nested partition sequence Π by taking $a_\epsilon = c_m G_0(B_\epsilon)$ for some sequence (c_m) (for example, $c_m = m^2$). This second method of prior centering might be preferable when $G_0 = G_{0,\eta}$ includes some unknown hyper-parameters η. It would be computationally awkward if one had to change the partitioning sequence each time a different value of η is being considered and, in most implementations of posterior inference, it is easier to change the parameters a_ϵ. By a slight abuse of notation we write $\mathsf{PT}(\mathcal{A}, G_0)$ to indicate a PT prior with partitioning subsets determined to achieve a desired prior mean G_0, and similarly we write $\mathsf{PT}(G_0, \Pi)$ for a PT prior with the beta parameters chosen to match a desired prior centering.

Figure 4.9 shows some random realizations from a finite PT prior. More specifically, we plot the random probabilities $G(B_{e_1 e_2})$ up to level $m = 2$ under a PT prior $G \sim \mathsf{PT}$. The plot highlights that realizations from a PT prior are discontinous at the partition boundaries, a feature that is often considered as a limitation of this class of priors. However, the effect of the discrete partition boundaries can trivially

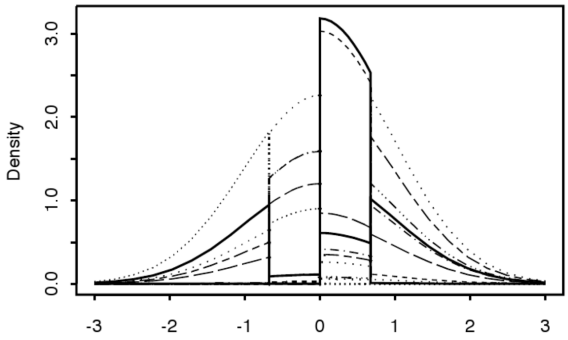


FIG 4.2. *Prior simulation of 10 random realizations* $G \sim \mathsf{PT}$ *with a standard Gaussian centering measure. Note the discontinuities at the boundaries of the partitioning sets.*

be removed. The most commonly used approach is to add a mixture with respect to Π. We will return to this problem §4.4.

For later reference we note that the split at each level of the nested partition sequence need not be binary. In general, each partitioning subset B_ϵ could be split into q subsets $B_{e_1 \cdots e_{m-1}} = \biguplus_{e_m=0}^{q-1} B_{e_1 \cdots e_{m-1} e_m}$, at the next level of the partitioning sequence. The digits e_j of the index ϵ are $e_j \in \{0, \ldots, q-1\}$ and the beta prior for the random splitting probability $G(B_{\epsilon 0} \mid B_\epsilon)$ is replaced by a Dirichlet distribution for the $q-$way splitting probabilities $(G(B_{\epsilon 0}), \ldots, G(B_{\epsilon,q-1}) \mid G(B_\epsilon)) \sim \mathsf{Dir}(\alpha_{\epsilon 0}, \ldots, \alpha_{\epsilon,q-1})$.

4.2. Posterior Inference

The PT is conjugate under i.i.d. sampling. Assume $x|G \sim G$ with a PT prior, $G \sim \mathsf{PT}(\mathcal{B}, \mathcal{A})$. Then the posterior on the unknown probability measure G is again a PT, $(G|x) \sim \mathsf{PT}(\mathcal{B}, \mathcal{A}^*)$ with

$$\alpha_\epsilon^* = \begin{cases} \alpha_\epsilon + 1 & \text{if } x \in B_\epsilon \\ \alpha_\epsilon & \text{otherwise.} \end{cases} \tag{4.1}$$

The α_ϵ parameters corresponding to the partitioning subsets are incremented by one for each subset B_ϵ that contains x. In practice, if a finite tree with T levels is used, (typically with $T \approx 7$) equation (4.1) allows for straightforward posterior updating.

Figure 4.3 shows a simple example of posterior updating for a PT prior. The prior model in the example is centered at a standard normal. The figure shows posterior inference conditional on observed data. Posterior updating for censored data introduces no additional difficulties if the partition boundaries are chosen to match the censoring times (Muliere and Walker, 1997).

The nature of the posterior PT also leads to straightforward posterior predictive simulation. To draw a new, future observation x_{n+1} from G conditional on observed data, $x_1, \ldots, x_n$ also drawn from G we only need to follow the posterior updating over finitely many levels. First generate an indicator $e_1 = I(x_{n+1} \in B_0)$ to determine whether the new observation x_{n+1} falls into B_0. To determine e_1 we generate $y_0 = G(B_0)$. The earlier discussion of the posterior process implies $p(y_0 \mid x_1, \ldots, x_n) = \mathsf{Beta}(\alpha_0^\star, \alpha_1^\star)$. Next let $e_2 = I(x_{n+1} \in B_{e_1 0})$. Again, $p(e_2 = 1 \mid$

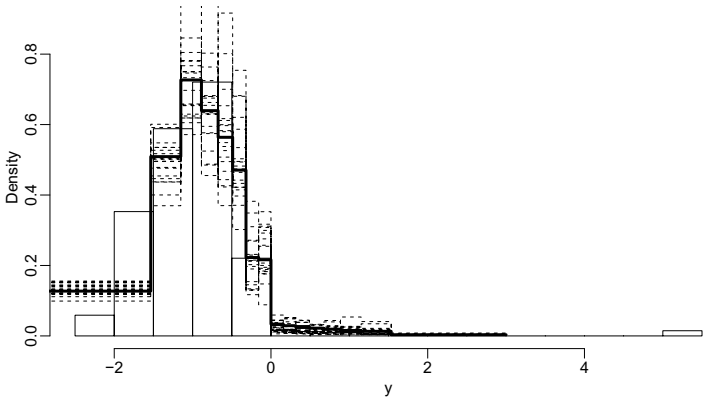


FIG 4.3. *Data (histogram), posterior mean $E(G \mid \boldsymbol{y})$ (thick line), random posterior draws $G \sim p(G \mid \boldsymbol{y})$ (dashed lines). The prior mean was $G_0 = \mathsf{N}(0, 1)$.*

$e_1, G) = y_{e_1 0}$ and $p(y_{e_1 0} \mid x_1, \ldots, x_n) = \mathsf{Beta}(\alpha^\star_{e_1 0}, \alpha^\star_{e_1 1})$, etc. The iteration ends when it reaches the first level m with $\alpha_{e_1 \cdots e_m} = \alpha^\star_{e_1 \cdots e_m}$, i.e., when the new draw is imputed to fall within a partitioning subinterval without earlier data. At that moment we simply generate x_{n+1} from the prior mean $G_0 = E(G)$ restricted to this subset, $x_{n+1} \sim G_0 I(x_{n+1} \in B_{e_1 \cdots e_m})$.

The described process of generating from the predictive distribution $p(x_{n+1} \mid x_1, \ldots, x_n)$ is beautifully illustrated by the following special case. Consider $n = 0$, i.e., marginal simulation for the first observation, assume that the sample space is the unit interval $B = [0, 1]$, the partition boundaries are the dyadic subintervales $[0, 1/2), [1/2, 1], [0, 1/4), [1/4, 1/2), \ldots$, and the centering measure is the uniform distribution. In that case the indicators e_j are simply the digits of x_{n+1} in a dyadic expansion and the process amounts to iteratively generating its digits.

Example 12 (A Survival Model with a Longitudinal Covariate) *Zhang* et al. *(2010) report a typical application of PT models in survival analysis. They discuss inference for data from a phase III trial of androgen ablation (AA) vs. chemohormonal (CH) therapy for patients with metastatic prostate cancer. Patients joined the trial with very diverse prior treatment histories, giving rise to a challenging statistical inference problem. The study enrolled $n = 286$ subjects, randomized to the two arms with $n_0 = 137$ patients assigned to the AA arm, and $n_1 = 149$ patients assigned to CH. The primary endpoint is time to progression (TTP) to androgen independent prostate cancer.*

An important covariate for TTP is the change of prostate specific antigen (PSA) over time. Let $\boldsymbol{y}_i$ denote the longitudinal trajectory of PSA measurements over time for patient i, let T_i denote the TTP, and let $x_i \in \{0, 1\}$ denote an indicator for assignment to AA (0) or CH (1). Figure 4.4 shows the data as Kaplan Meier plots arranged by treatment allocation.

We construct a joint probability model for $\boldsymbol{y}_i$ and T_i for a patient in treatment group x_i as a marginal model $G_{x_i}(T_i)$ and a conditional model $p(\boldsymbol{y}_i \mid T_i, x_i)$. This unusual factorization into a marginal model for TTP and a conditional model for the longitudinal covariate given TTP makes it easy to go nonparametric on the event time model. We assume $G_x \sim \mathsf{PT}(\mathcal{A}, \Pi)$, independently for $x = 0, 1$. Conditional on T_i the model for $\boldsymbol{y}_i$ is a non-linear regression. Details of the regression mean

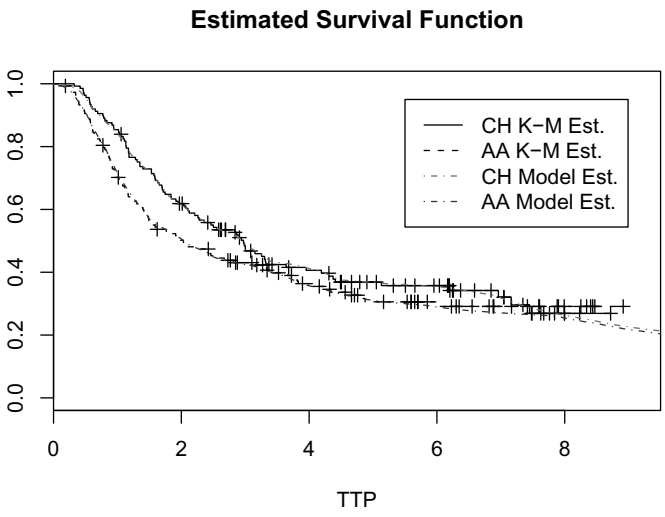


FIG 4.4. *Prostate cancer trial. Kaplan Meier plot of the data, arranged by the two treatment arms. The dot-dashed lines show the model based inference.*

function are motivated by the typical PSA profiles and the nature of the intervention. See Zhang et al. *(2010) for details.*

Figure 4.5a shows the estimated distributions of TTP under the two treatments. For comparison, Figure 4.5b shows the same inference using two independent Weibull models for G_x, $x = 0, 1$.

An important feature of density estimation with nonparametric Bayesian models is the coherent nature of inference as a posterior probability model on the unknown probability measure. This enables us to report uncertainties on any event or summary of interest. For example, Figure 4.6 shows the implied uncertainty on the hazard function.

4.3. The Marginal Model

The PT prior allows a closed form expression for the marginal distribution $p(x_1, \ldots, x_n)$ of a random sample $x_i \sim G$, i.i.d., under $G \sim \mathsf{PT}(\cdot)$. The PT shares this practically very useful property with the DP. For the DP prior the marginal model is determined by the Pólya urn in (7.1).

Lavine (1992) shows the expression of the marginal model for the PT prior. Let $\epsilon_m(x_i) = e_1 \ldots e_m$ denote the index of the level m subset that contains x_i, i.e., $x_i \in B_{e_1 \cdot e_m}$. Also, let $m^*(x_i)$ denote the lowest level m such that x_i is the only data point in $B_{\epsilon_m(x_i)}$. Formally, $m^\star(x_i) = \min_m\{x_j \notin B_{\epsilon_m(x_i)},\ j \neq i\}$. Then

$$p(x_1, \ldots, x_n \mid \eta) = \prod_{i=1}^{n} G_0(x_i \mid \eta) \prod_{j=2}^{n} \prod_{m=1}^{m^*(x_j)} \frac{\alpha^\star_{\epsilon_m(x_j)}}{\alpha_{\epsilon_m(x_j)}} \cdot \frac{\alpha_{\epsilon_{m-1}0(x_j)} + \alpha_{\epsilon_{m-1}1(x_j)}}{\alpha^\star_{\epsilon_{m-1}0(x_j)} + \alpha^\star_{\epsilon_{m-1}1(x_j)}}.$$

See Berger and Guglielmi (2001) and Hanson and Johnson (2002) for more discussion. In particular, Berger and Guglielmi (2001) use the marginal model to evaluate Bayes factors.

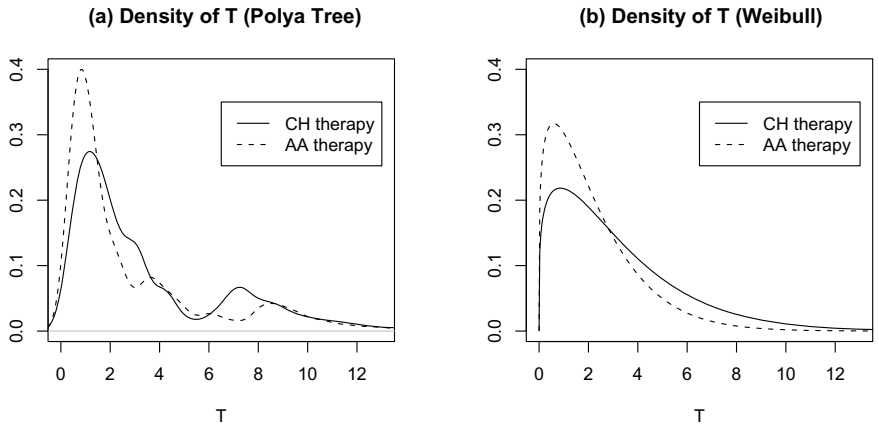


FIG 4.5. *Prostate cancer trial. Estimated $G_x(T)$ using PT (left) and Weibull (right).*

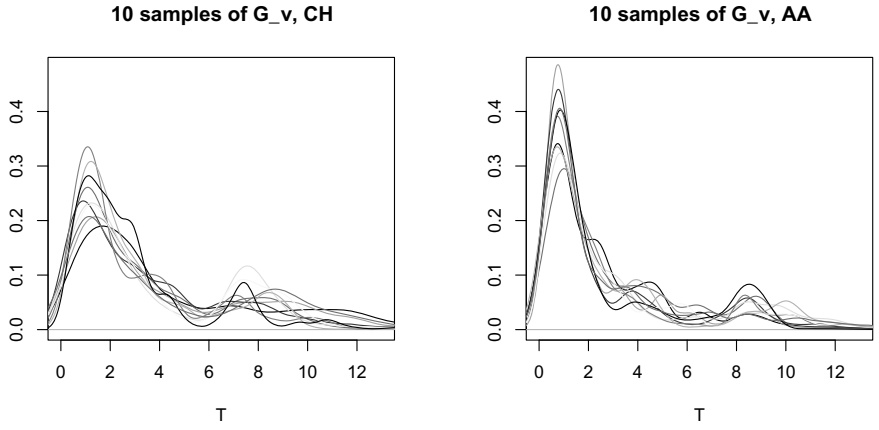


FIG 4.6. *Prostate cancer trial. Uncertainty $p(G(\cdot) \mid \boldsymbol{y})$ using PT (left) and Weibull (right).*

4.4. Mixtures of Pólya Trees

Figure 4.3 highlights a critical limitation of the PT prior, i.e., posterior draws $G \sim p(G \mid x_1, \ldots, x_m)$ as well as the posterior mean $\bar{G} = E(G \mid x_1, \ldots, x_m)$ show visible discontinuities at the partition boundaries. This sensitivity of posterior inference to the chosen partition is undesirable in most data analyses.

One possible fix is to consider PTs with random centering measures. Recall the discussion of the DP prior; a random probability measure which is assigned a DP prior, $G \sim \mathsf{DP}$, is almost surely discrete. The discrete nature of the DP greatly simplified many of the computational details of posterior simulation but is unappealing for most applications. In that context, we mitigated concerns related to the discrete nature of a DP random measure by convoluting G with an additional smooth kernel to define DP mixture models. Similarly, we can mitigate the undesirable sensitivity of the PT prior to partition boundaries by introducing additional mixing with respect to the centering measure. Assume that the PT prior is centered by defining the nested partition sequence Π to be determined by dyadic quantiles of

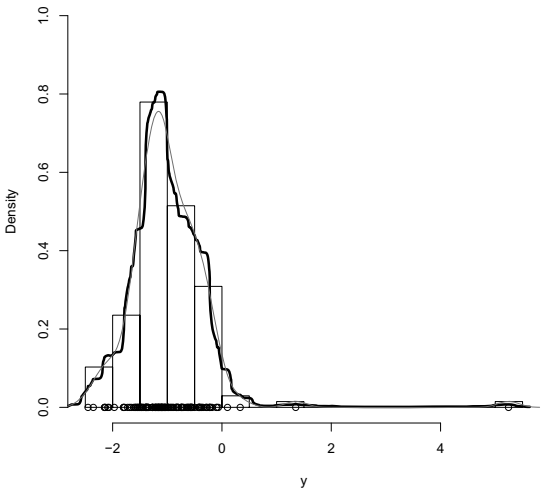


FIG 4.7. *Same data as in Figure 4.3, now with PT mixture (black). For comparison a kernel density estimate (light grey).*

a desired prior mean $G_0 = E(G)$. Hanson and Johnson (2002) propose to introduce additional hyperparameters η to index the centering model $G_{0,\eta}$ and extend the model with a hyperprior on η.

$$G \mid \eta \sim \mathsf{PT}(\mathcal{A}, G_{0,\eta}), \qquad \eta \sim p(\eta). \tag{4.2}$$

Mixing with respect to a hyperprior $p(\eta)$ smoothes out the undesired dependence on partition boundaries that appears in Figure 4.3. We refer to model (4.2) as a mixture of PT model. Note the difference to the DP mixture model that defined a mixture of a kernel with respect to a DP prior; hence, a mixture of PT models is analogous to the MDP model discussed in §3.5. Figure 4.7 shows how posterior inference is improved under a PT mixture prior.

4.5. Multivariate Pólya Trees

The definition of the PT prior is general, all we needed was a nested sequence of partitions and the beta priors for the random splitting probabilities. When $B = \mathbb{R}$, the partitions can be described by partition boundaries and can be naturally indexed by sequences of binary indicators for left versus right splits.

For higher dimensional sample spaces, it becomes awkward to specify and keep track of the nested partition sequence. For $B = \mathbb{R}^p$ this task becomes challenging, but not impossible. Jara *et al.* (2009) propose a possible construction that remains feasible even for moderately high dimensions, $p = 8$ and beyond. The construction works with a multivariate normal centering measure, $G_0 = \mathsf{N}(\boldsymbol{\mu}, \Sigma)$ where $\Sigma = UU'$, along with a split of each partitioning subset into 2^p partitioning subsets. More specifically, the partitioning subsets B_ϵ are indexed with sequences of base 2^p digits, i.e., $\epsilon = e_1 \cdot e_m$, $e_j \in \{0, \ldots, 2^p - 1\}$. For example, in Figure 4.1, instead of splitting each B_ϵ into two daughters, each partitioning subset B_ϵ is split into 2^p nested sets. The definition of these partitioning subsets B_ϵ starts with p-dimensional rectangles defined by standard normal dyadic quantiles. Let $B_0(m, k)$ denote the

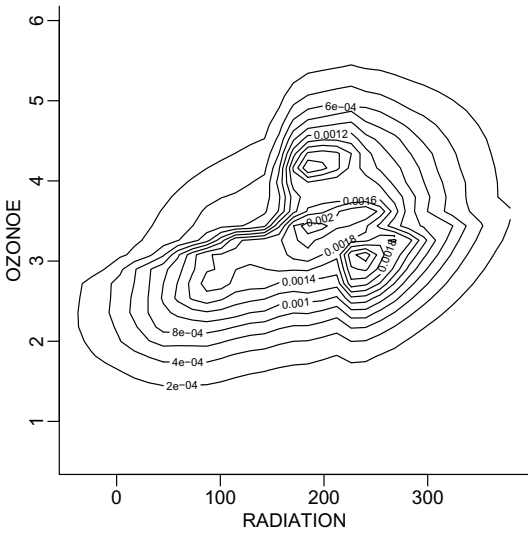


FIG 4.8. *Airquality data. Bivariate density estimate, using the R function* `PTdensity(·)` *from the R package* `DPpackage`. *The air quality data is available in R.*

k-th of 2^m dyadic (univariate) standard normal quantiles at level m. At level $m = 1$ the two subsets are defined by the partition boundary at 0, the standard normal median. At $m = 2$, the four subsets are defined by the partition boundaries at the 1/4, 1/2 and 3/4 quantiles, etc. Next define the p-dimensional product sets $\boldsymbol{B}_0(m, \boldsymbol{k}) = B_0(j, k_1) \times \ldots \times B_0(j, k_p)$. Finally, the partitioning subsets for the nested partition sequence Π are defined by an affine transformation $\boldsymbol{B}(m, \boldsymbol{k}) = \{\boldsymbol{\mu} + Uz;\ z \in \boldsymbol{B}_0(j, \boldsymbol{k})\}$.

The construction was easily explained, but the reader might be reluctant to venture into an implementation. Fortunately inference for the multivariate PT is implemented in `DPpackage`. Figure 4.8 shows an example of output from the function `PTdensity(·)`.

4.6. Rubbery Pólya Tree

Recall that an RPM G with PT prior includes discontinuities at the partition boundaries. These discontinuities can clearly be seen in Figure 4.9, and they persist in the posterior means. See, for example, see 4.3. This awkward property limits the use of the PT prior for many data analysis problems. In §4.4 we discussed a construction that can mitigate this awkward feature of the PT prior by adding uncertainty about the centering measure $G_{0\eta}$. Mixing over η smears out the partition boundaries. Alternatively, Paddock *et al.* (2003) introduce additional randomness in the model by jittering the cutoff points in a dyadic nested partition; the discontinuities in the RPM are then removed by averaging with respect to this additional jittering.

A third approach that directly addresses the cause of the discontinuities in the PT mode is the rubbery PT introduced by Nieto-Barajas and Müller (2012). Recall the independent random splitting probabilities

$$Y_{\epsilon 0} = G(B_{\epsilon 0} \mid B_\epsilon).$$

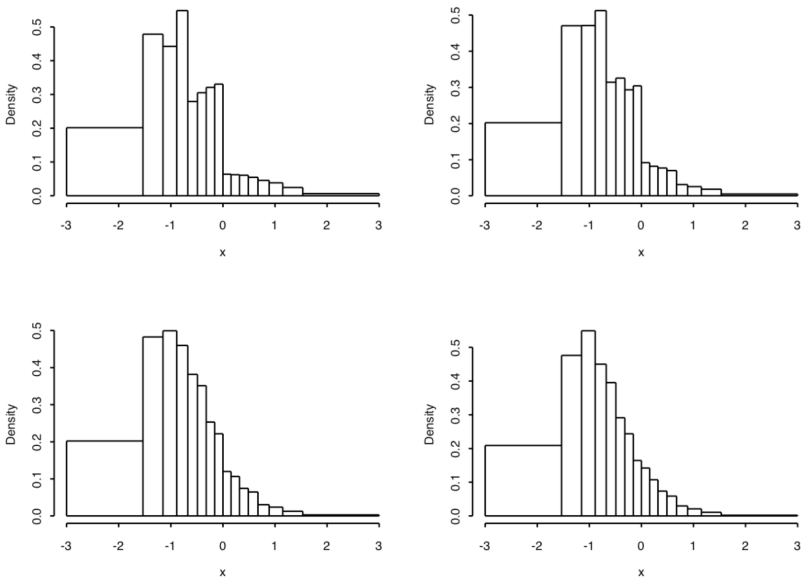


FIG 4.9. *Posterior predictive distributions for a rPT with a sample of size 1 at $x = -2$: Top left $\delta = 0$, top right $\delta = 1$, bottom left $\delta = 5$ and bottom right $\delta = 10$. The posterior predictive is identical to the posterior mean, $p(x_2 \mid x_1 = x) = E(G \mid x)$.*

The independence of these Y_ϵ, across ϵ is the source of the discontinuities in G at the partition boundaries; by introducing dependence among these probabilities it is possible to eliminate the discontinuities.

The construction of the rubbery PT is more easily explained in the context of a finite PT. For ease of exposition assume a PT_2 prior with only two levels and use decimal integers to index the partitioning subsets at each level:

$$\begin{array}{ccc} B_{11} & | & B_{12} \\ B_{21} \mid B_{22} & | & B_{23} \mid B_{24} \end{array} \tag{4.3}$$

A computationally easy way to introduce dependence between the two random probabilities Y_{21} and Y_{23} while leaving the marginal beta distribution unchanged is the use of a latent binomial variables Z_{21} sandwiched between them. We leave the marginal distribution of Y_{21} unchanged as $Y_{21} \sim \mathsf{Beta}(\alpha_2, \alpha_2)$ and $Y_{22} = 1 - Y_{21}$. The prior for Y_{23} is changed to

$$Z_{21} \mid Y_{21} \sim \mathsf{Bin}(\delta_{21}, Y_{21}), \qquad Y_{23} \mid Z_{21} \sim \mathsf{Beta}(\alpha_2 + Z_{21}, \alpha_2 + \delta_{21} - Z_{21}),$$

and $Y_{24} = 1 - Y_{23}$. It is easily verified that the implied marginal prior $p(Y_{23})$ remained unchanged as a $\mathsf{Beta}(\alpha_2, \alpha_2)$ while introducing the desired dependence of Y_{21} and Y_{23}. Also, the level 1 priors remain unchanged. The Binomial sample size parameter δ_{21} tunes the desired level of smoothing; large values imply more smoothing. We use $\mathsf{rPT}(\Pi, \mathcal{A}, \delta)$ to denote a rubbery PT with a sequence of latent binomial variables indexed by δ_{mj}, $m > 1$, $j = 1, 3, \ldots$. Figure 4.9 shows posterior predictive distributions for a future observation for different choices of δ.

Chapter 5

Dependent Dirichlet Processes and Other Extensions

5.1. Dependent Extensions of the DP

Many applications involve families of probability models $\mathcal{G} = \{G_x : x \in X\}$. For example, G_x could be the distribution of the time to progression for a patient with baseline covariates x. In that case, a prior $p(G_x : x \in X)$ would provide a nonparametric alternative to the popular but restrictive proportional hazards model. More generally, a nonparametric prior $p(\mathcal{G})$ can be used to define a fully non-parametric regression $p(y \mid x) = G_x(y)$. Another typical applications of prior models $p(\mathcal{G})$ is in the construction of mixed effects models, where $p(\mathcal{G})$ is used to define a random effects distribution $G_x(\cdot)$ for patients with covariates x.

Most popular prior models for $\mathcal{G}$ in the recent literature are based on extensions of the Dirichlet process (DP) model discussed in Chapter 3, which are collectively known as dependent Dirichlet process (DDP) models. We first consider the simplest case, with finitely many dependent RPMs $\mathcal{G} = \{G_j,\ j = 1, \ldots, J\}$ that are judged to be exchangeable, i.e., the prior model $p(\mathcal{G})$ should be invariant with respect to any permutation of the indices. This case could arise, for example, as a prior model for unknown random effects distributions G_j in related studies, $j = 1, \ldots, J$. To keep the upcoming discussion specific we will continue to refer to this motivating example. In words, we wish to define a prior probability model $p(\mathcal{G})$ that allows us to borrow strength across the J studies. Patients under study j_1 should inform inference about patients enrolled in another related study $j_2 \neq j_1$. Two extreme modeling choices would be (i) to pool all patients and assume one common random effects distribution, or (ii) to assume J distinct random effects distributions with independent priors. Formally the earlier choice assumes $G_j \equiv G$, $j = 1, \ldots, J$ with a prior $p(G)$. The latter assumes $G_j \sim p(G_j)$, independently, $j = 1, \ldots, J$. We refer to the two choices as extremes since the first choice implies maximum borrowing of strengths, and the other choice implies no borrowing of strength. In most applications, the desired level of borrowing strength is somewhere in-between these two extremes.

Figure 5.1 illustrates the two modeling approaches. Note that in Figure 5.1 we added a hyperparameter η to index the prior model $p(G_j \mid \eta)$ and $p(G \mid \eta)$, which was implicitly assumed fixed. The use of a random hyperparameter η allows for some borrowing of strength even in the case of conditionally independent $p(G_j \mid \eta)$. Learning across studies can happen through learning about the hyperparameter η. This is exactly the construction in Cifarelli and Regazzini (1978), which was used in, among others, Muliere and Petrone (1993) and Mira and Petrone (1996). However, the nature of the learning across studies is determined by the parametric form of η. This is illustrated in Figure 5.2. Assume $G_j \sim \mathsf{DP}(\alpha, G^\star_\eta)$, independently, $j = 1, 2$ and a base measure $G^\star_\eta = \mathsf{N}(m, B)$ with unknown hyperparameter $\eta = (m, B)$.

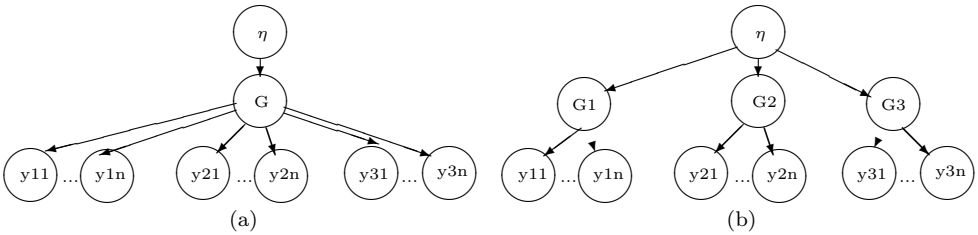


FIG 5.1. *One common RPM G (panel a) versus distinct RPMs G_j, independent across studies (panel b).*

In this case, prediction for a future study G_3 can not possibly learn about the multimodality of G_1 and G_2, beyond general location and orientation.

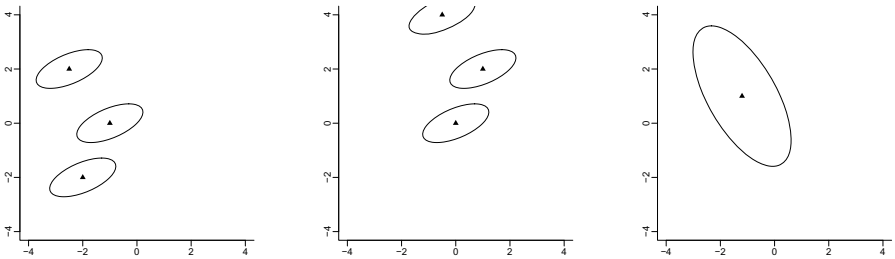
A natural next step in the model elaboration would now be to consider more complex choices for the hyperparameter η. Ideally, when $G^\star = \eta$ is an RPM itself, then we could potentially achieve arbitrary learning across the studies. This is exactly the construction of the hierarchical and nested DPs. See §5.4.2 and §5.4.3. However, these approaches are not suitable to model more general types of data such as spatial and/or temporal data. In §5.2 we introduce a still more general and widely used extensions of the DP that achieves the desired borrowing of strength while preserving the computational advantages of the DP.

5.2. Dependent DP (DDP)

MacEachern (1999) introduced what has meanwhile become by far the most commonly used prior for dependent RPMs, $p(G_x : x \in X)$. The model is known as the dependent DP (DDP). The beauty of the model is the elegance and simplicity of the construction. Recall the stick breaking representation of a DP random measure $G \sim \mathsf{DP}(M, G^\star)$, where

$$G(\cdot) = \sum_{h=1}^{\infty} w_h \delta_{m_h}(\cdot),$$

$m_h \sim G^\star$, independently across h and $w_h = v_h \prod_{g<h}(1-v_h)$ with $v_h \sim \mathsf{Beta}(1, M)$,



(a) $E(G_1 \mid \boldsymbol{y})$ (b) $E(G_2 \mid \boldsymbol{y})$ (c) Prediction for G_3

FIG 5.2. *$G_j \sim \mathsf{DP}(M, G^\star)$ with common $G^\star = \mathsf{N}(m, B)$. Learning across studies is restricted to the parametric form of η.*

i.i.d. The DDP uses the same construction for each G_x,

$$G_x(\cdot) = \sum_{h=1}^{\infty} w_h \delta_{m_{x,h}(\cdot)}. \tag{5.1}$$

Here $\boldsymbol{m}_h = \{m_{x,h} : x \in X\}$ are independent realizations from a stochastic process on X (such as a Gaussian process), and the w_hs are constructed as before. Keeping $m_{x,h}$ independent across h ensures that each G_x marginally follows a DP prior. The simple, yet powerful idea of the DDP construction is to introduce dependence over x, i.e., to link the G_x through dependent locations of the point masses. Implicit in the notation used in (5.1) is the definition of weights w_h that are common across x. This variation of the DDP model is sometimes referred to as "common weight" (or "single p") DDP. However, the proposal in MacEachern (1999) is more general than (5.1), allowing also varying weights $w_{x,h}$. This more general construction is used, for example, in the time series DDP proposed in Nieto-Barajas *et al.* (2008) who define a DDP prior for a time series $\{G_t,\ t = 1, \ldots, T\}$ of random probability measures by introducing dependence of the weights $w_{t,h}$ (see §5.5.2).

Griffin and Steel (2006) define another interesting variation of the basic DDP by keeping both sets of parameters, locations and weights, unchanged across x. Instead they use permutations of how the weights are matched with locations. The permutations change with x. One advantage of such models is the fact that the support of G_x remains constant over x, a feature that can be important for extrapolation beyond the observed data.

5.3. ANOVA DDP

De Iorio *et al.* (2004) and De Iorio *et al.* (2009) define the ANOVA DDP as a variation of the DDP that is particularly useful for multivariate categorical covariates x. For illustration, assume $x = (u, v)$ for two categorical factors u and v. For example $G_{u,v}$ could be the random effects distribution for patients who are treated in related multi-arm clinical studies. Here u could be indexing related studies and v could be the different treatment arms.

The simplest form of dependence for a set $\{m_{x,h}\}$ of random variables indexed by two categorical covariates $x = (u, v)$ is an ANOVA model with main effects for u and v. This is exactly the model used in De Iorio *et al.* (2004). In particular, we assume $m_{x,h} = \mu_h + \alpha_{h,u} + \beta_{h,v}$ for $u \in \{0, \ldots, U\}$ and $v \in \{0, \ldots, V\}$, and assign normal priors on $\mu_h, \alpha_{h,u}$ and $\beta_{h,v}$, with $\alpha_{h,0} = \beta_{h,v} = 0$ for identifiability. Also, let $\boldsymbol{\theta}_h = (\mu_h, \alpha_{h,u}, \beta_{h,v},\ u = 1, \ldots, U,\ v = 1, \ldots, V)'$ denote the column vector of all ANOVA effects. Finally, let $G^\star(\boldsymbol{\theta}_h)$ denote the joint normal prior on the ANOVA effects. We write $\{G_x,\ x \in X\} \sim \mathsf{ANOVA\ DDP}(G^\star, M)$.

Implementation becomes particularly easy when the ANOVA DDP model is used in a DPM model, i.e., the random G_x is convoluted with an additional kernel, for example

$$y_i \mid m_i \sim \mathsf{N}(m_i, s^2), \qquad m_i \mid x_i = x, \mathcal{G} \sim G_x,$$

with $\{G_x,\ x \in X\} \sim \mathsf{ANOVA\ DDP}(G^\star, M)$. In this case, inference can be reduced to a standard DP mixture of normal model. To do so, let $\boldsymbol{d}_i$ denote a design vector that selects the relevant ANOVA factors corresponding to x_i for observation y_i. We can equivalently write

$$y_i \mid \boldsymbol{\theta}_i \sim \mathsf{N}(\boldsymbol{d}_i'\boldsymbol{\theta}_i, s^2), \qquad \boldsymbol{\theta}_i \mid F \sim F, \tag{5.2}$$

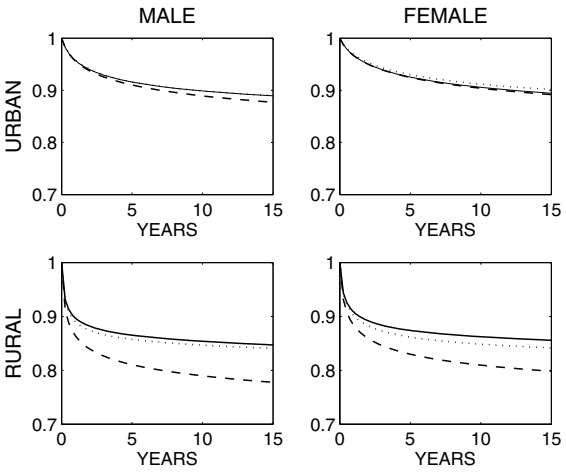


FIG 5.3. *Posterior estimated survival functions $E(S_x \mid \text{data})$, arranged by sex and rural vs. urban birth place. The three curves in each panel correspond to the three birth cohorts.*

with $F \sim \mathsf{DP}(F^\star, M)$, a DP mixture of normal linear models. The mixing measure G in the DP mixture is a probability model for complete vectors of ANOVA factors, $F = \sum_h w_h \delta_{\theta_h}$. Inference can therefore proceed like in a standard DP mixture model. Although the description implicitly assumed univariate outcomes y_i, extending the model to multivariate outcomes is straightforward using corresponding multivariate ANOVA models for $m_{x,h}$.

Example 13 (ANOVA DDP) *De Iorio* et al. *(2009) use an ANOVA DDP prior to implement non-parametric survival regression with multiple covariates. We apply their model to analyze data on childhood mortality in Columbia (Somoza, 1980). The dataset includes observations for 1437 children (using only the oldest child for each mother) and covariates including gender (binary), birth cohort (categorical with 3 levels), and an indicator for the child being born in a rural area (binary). The dataset includes extensive censoring, with 87% of the children alive at the time of observation.*

Let $S_x(t)$ denote the probability of a child with covariates x surviving beyond time t. The ANOVA DDP defines a prior probability model on $\mathcal{G} = \{S_x;\ x \in X\}$. Figure 5.3 shows point estimates of the survival functions for all combinations of gender, rural and birth cohort as $E(S_x \mid \text{data})$. However, posterior inference under the nonparametric ANOVA DDP model delivers more than point estimates. Figure 5.4 shows pointwise central 95% posterior probability intervals for $S_x(t)$.

5.4. Multilevel Modeling of Exchangeable RPMs

5.4.1. Weighted Mixtures of DPs

We consider now the problem of modeling exchangeable collections of RPMs, i.e., $\mathcal{G} = \{G_j : j = 1, \ldots, n\}$, where the prior $p(\mathcal{G})$ is invariant with respect to the order in which the G_js are included in the model. The data are $(y_{i,j})$, where $j = 1, \ldots J$ denotes the study under which the observations were generated and $i = 1, \ldots, I_j$ indexes observations within study j.

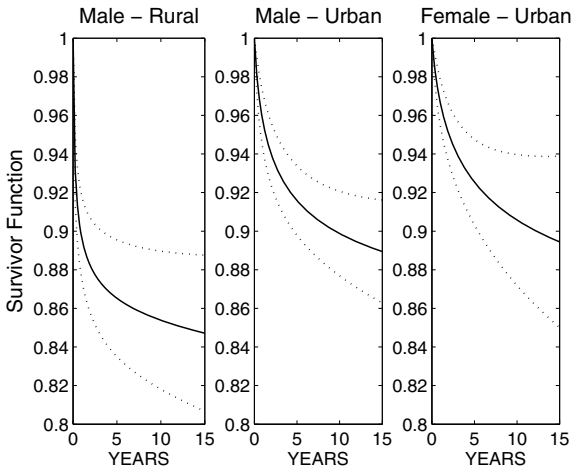


FIG 5.4. *Posterior estimated survival functions (solid lines) with 95% credible interval (dotted lines). All three panels represent children belonging to the third birth cohort. The labels indicate the levels of the other two covariates.*

Müller *et al.* (2004) and more recently Griffin *et al.* (2010) and Kolossiatis *et al.* (2012) use a construction based on the superposition of random measures. By sharing part of the probability mass across different studies the construction creates the desired dependence. Figure 5.5 illustrates the idea.

In Müller *et al.* (2004) each of the RPMs G_j is defined as a combination of a common F_0 and a study-specific F_j. Let

$$G_j \mid \epsilon, F_j, F_0 = \epsilon F_0 + (1-\epsilon)F_j, \qquad F_0 \sim (M, H), \qquad F_j \sim \mathsf{DP}(M, H), \tag{5.3}$$

with $y_{i,j} \sim \int p(y_{i,j} \mid \theta) G_j(\mathrm{d}\theta)$. The model is completed with a prior on ϵ,

$$p(\epsilon) = \pi_0 \delta_0 + \pi_1 \delta_1 + (1 - \pi_0 - \pi_1)\mathsf{Beta}(a, b),$$

where $\mathsf{Beta}(x; a, b)$ denotes a beta distributed random variable x with parameters (a, b). Note that this prior on ϵ includes point masses on 0 and 1, allowing for the two extreme cases of common and conditionally independent G_j across studies.

Example 14 (Dependent RPMs) *Müller* et al. *(2004) use the hierarchical model* (5.3) *as a prior probability model for random effects distributions in two related studies. The data are log white blood cell counts over time for breast cancer patients in*

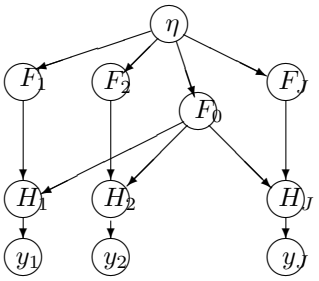


FIG 5.5. *Hierarchical composition of RPMs.*

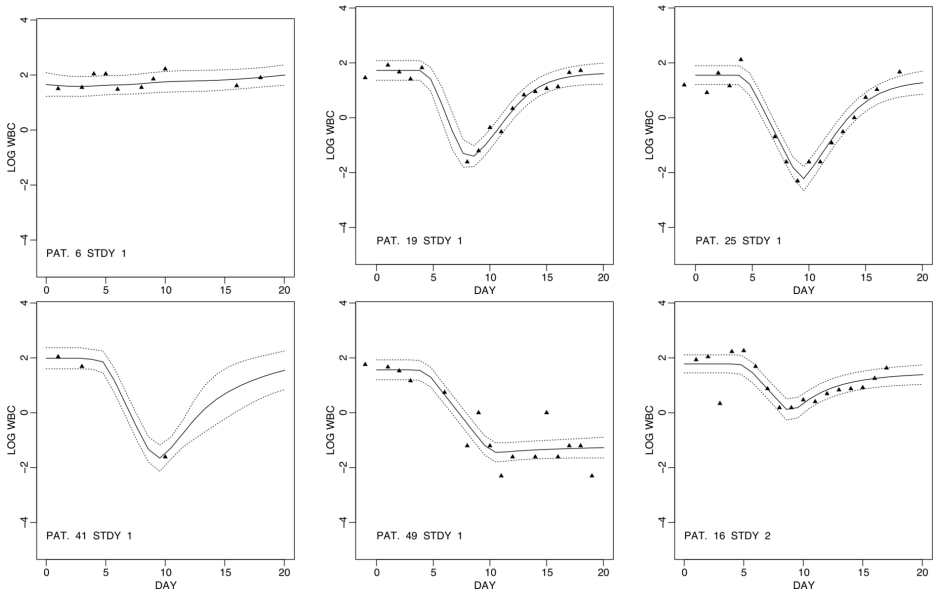


FIG 5.6. *Some typical patients. The data show y_{ijk} for 6 arbitrarily selected patients from studies $j = 1$ and $j = 2$. The triangles are the observed WBC. The solid line shows the posterior fitted mean curve, and the dotted lines show 95% central HPD intervals for the mean curve.*

two related studies. Figure 5.6 shows the data for some selected patients from the two studies.

The model includes a non-linear regression mean curve $f(t; \theta_{ji})$ for blood count data for patient i, in study j, $i = 1, \ldots, n_i$ and $j = 1, 2$. The mean curve is indexed with patient-specific random effects θ_{ij}. The random effects θ_{ij} are 9-dimensional and are assumed to arise from a study-specific random effects distribution G_j. The model is completed with the hierarchical prior in (5.3) for $\{G_1, G_2, G_3\}$, including a future third study $j = 3$. Figure 5.7 shows posterior inference for the unknown distributions F_0, F_1 and G_1 as bivariate scatterplots of random draws from the posterior means $E(F_0 \mid Data)$, $E(F_1 \mid Data)$ and $E(G_1 \mid Data)$. Figure 5.8 shows posterior predictive inference for a patient from a future third study $j = 3$.

Note that, in equation (5.3), the marginal prior for G_j is constructed as a sum of two RPMs that follow DP priors. Hence, the implied marginal prior $p(G_j)$ is not in general a DP itself. For many applications this might not be a concern. If desired, however, it is possible to construct the combination of the two RPMs F_0 and F_j such that the implied marginal prior $p(G_j)$ is again in the same family as $p(F_j)$; such a model is developed in Kolossiatis *et al.* (2012). In particular, assuming a DP prior for $p(F_j)$ it is possible to choose $p(\epsilon)$ such that $p(G_j)$ is a DP prior again.

The construction is easiest described by using a representation of the DP prior as a normalized gamma process. Let $\mu \sim \mathsf{GaP}(M\, G^\star)$ denote a gamma process, i.e., $\mu(B) \sim \mathsf{Gamma}(M\, G^\star(B), 1)$ for any measurable set B. Without loss of generality assume $J = 2$. Kolossiatis *et al.* (2012) use independent gamma processes $\mu_j \sim \mathsf{GaP}(M\, G^\star)$, $j = 0, 1, \ldots, 2$. Then

$$F_j(B) = \frac{\mu_j(B)}{\mu_j(X)}, \qquad j = 0, 1, 2$$

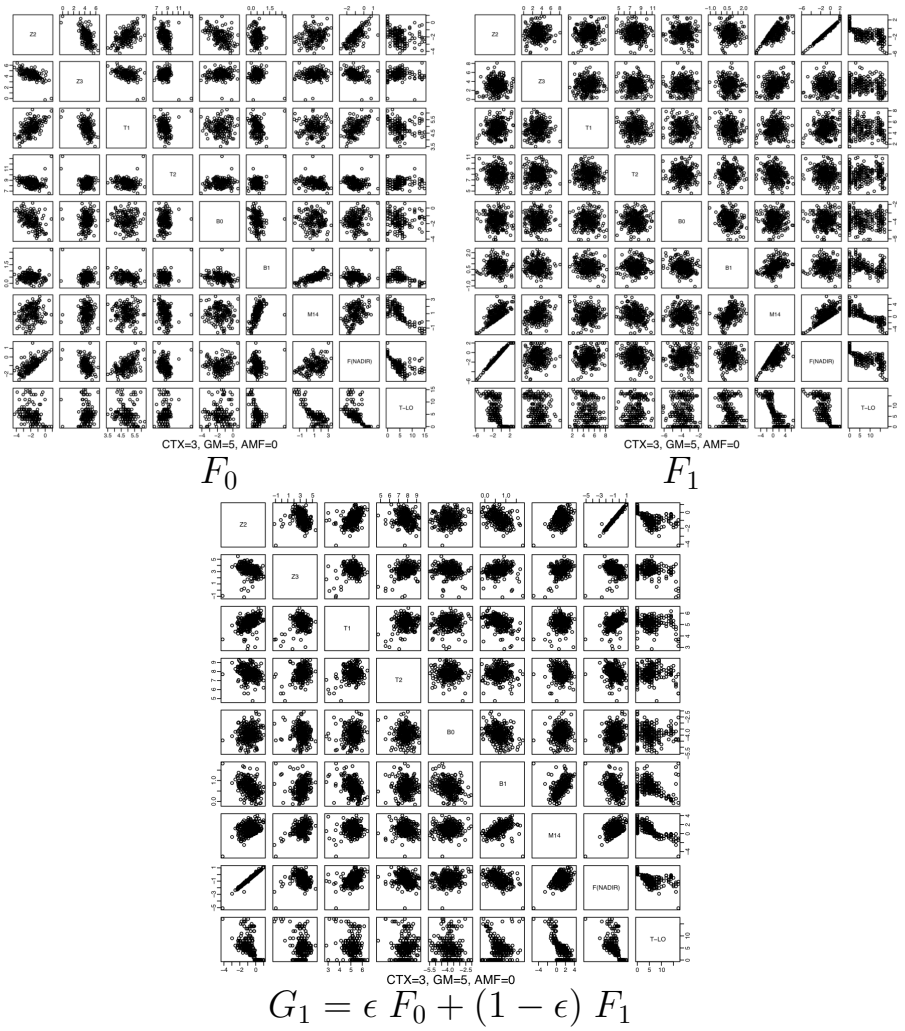
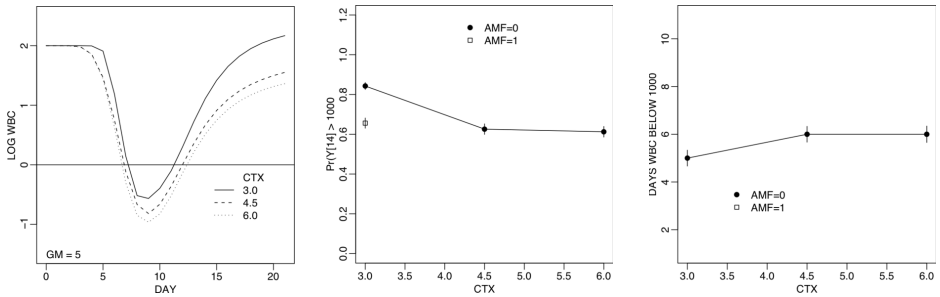


FIG 5.7. *Posterior estimated distributions* $E(F_0 \mid Data)$, $E(F_1 \mid Data)$ *and* $E(G_1 \mid Data)$.



(a) Estimated WBC profile

(b) $Pr\{WBC > 1000\}$ on day 14

(c) $E(T_{14} \mid y)$ T_{14} = days WBC below 1000.

FIG 5.8. *Posterior predictive inference for a hypothetical future patient in a future study* $j = 3$.

is a DP random probability. If we define the prior for ϵ as

$$\epsilon = \mu_0(X)/\left(\mu_0(X) + \mu_1(X)\right),$$

then

$$G_j(B) = \epsilon F_0(B) + (1-\epsilon)F_1(B) = \frac{\mu_0(B) + \mu_1(B)}{\mu_0(X) + \mu_1(X)}$$

follows marginally a DP prior again since the sum of the two gamma processes μ_1 and μ_2 is a gamma process again, $\mu_0 + \mu_1 \sim \mathsf{GaP}\left((M_0 + M_1)\, G^\star\right)$.

5.4.2. Hierarchical DP

Consider conditionally independent DP priors for each $G_j \in \mathcal{G} = \{G_j;\ j = 1, \ldots, J\}$, i.e., $G_j \sim \mathsf{DP}(M, G_0)$, independently. Recall the discussion in §5.1. The simplest way to borrow strength across G_js is through the base measure G_0. However, if G_0 is modeled parametrically, then the specific form of that parametric family determines and limits how information can be shared across the G_js. For example, if $G_0 = \mathsf{N}(\phi, 0)$ is indexed with a location parameter ϕ then borrowing strength across the G_j can only be through that location parameter. To avoid this limitation we could instead use a nonparametric prior for G_0, e.g., $G_0 \sim \mathsf{DP}(B, H)$. This is exactly the construction of the hierarchical Dirichlet process (HDP) of Teh *et al.* (2006). An early version of the same model appears in Escobar and Tomlinson (1999).

As a DP random measure, G_0 is discrete $G_0(\cdot) = \sum_{h=1}^{\infty} w_l \delta_{\widetilde{\theta}_h}(\cdot)$. Any θ drawn from G_0 is necessarily equal to one of the $\widetilde{\theta}_h$. In other words, the atoms of G_j agree with those of G_0. Hence, G_j can be written as

$$G_j(\cdot) = \sum_{h=1}^{\infty} \varpi_{j,h} \delta_{\widetilde{\theta}_h}(\cdot),$$

where $p\{(\varpi_{j1}, \varpi_{j2}, \ldots, \varpi_{jJ}) \mid (w_1, w_2, \ldots, w_J)\} = \mathsf{Dir}(w_1, w_2, \ldots, w_J)$ for any finite J. In other words, all the G_js use the same set of atoms but assign different (albeit related) weights to them (see Figure 5.9).

One implication of sharing the same atoms $\widetilde{\theta}_h$ across all G_js is that HDP mixture (HDPM) models allow co-clustering across different groups. Similar to (3.9) the HDPM can be written as a hierarchical model

$$y_{i,j} \mid \theta_{i,j} \sim p(y_{i,j} \mid \theta_{i,j}), \quad \theta_{i,j} \mid G_j \sim G_j, \quad G_j \mid G_0 \sim \mathsf{DP}(M, G_0), \quad G_0 \sim \mathsf{DP}(B, H),$$

where $j = 1, \ldots, J$ and $i = 1, \ldots, I_j$. Let $\{\theta_r^{\star\star};\ r = 1, \ldots, k\}$ denote the unique values among all θ_{ij}, $j = 1, \ldots, J$, $i = 1, \ldots, I_j$. Using ties to define clusters $S_r = \{(ij) :\ \theta_{ij} = \theta_r^{\star\star}\}$ we get a random partition model where $y_{i,j}$ and $y_{i',j'}$ can be assigned to the same cluster, even if they belong to different studies $j \neq j'$.

An appealing feature of the HDP is that it inherits the simple form of the predictive probability distribution from the DP prior. Conditional on G_0, the predictive probability distribution for $\theta_{ij} \sim G_j$ is unchanged from (3.3). Let k_i^j denote the number of distinct values $\{\theta_{hj}^\star, h = 1, \ldots, k_i^j\}$ among the draws $\{\theta_{1,j}, \ldots, \theta_{i-1,j}\}$ and $n_{i-1,h}^j = \sum_{\ell=1}^{i-1} I(\theta_{\ell j} = \theta_{hj}^\star)$. Then

$$\theta_{i,j} \mid \theta_{i-1,j}, \ldots, \theta_{1,j} \sim \sum_{h=1}^{k_i^j} \frac{n_{i-1,h}^j}{M + i - 1} \delta_{\theta_{h,j}^\star} + \frac{M}{M + i - 1} G_0, \qquad 1 \leq i \leq I_j,$$

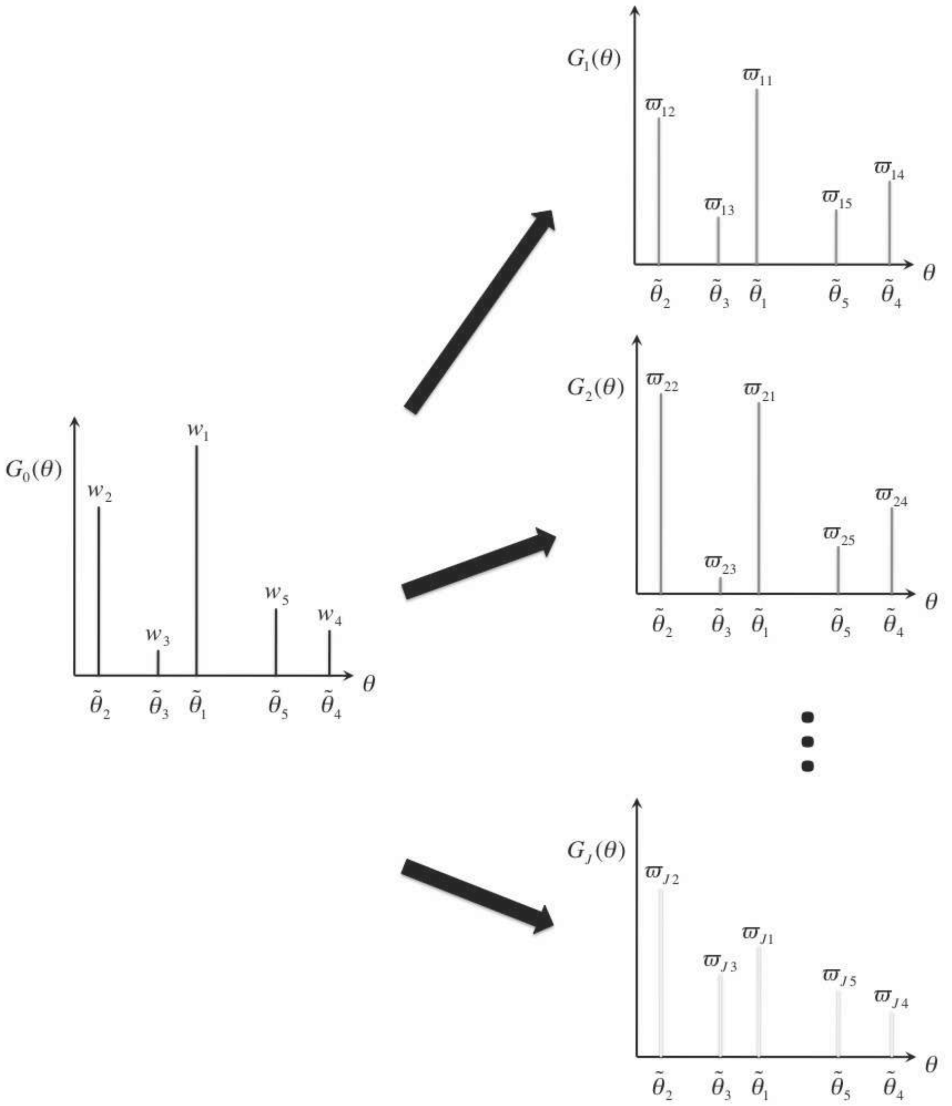


FIG 5.9. *Stylized representation of the hierarchical Dirichlet process. For each distribution, the location of the vertical lines on the horizontal axis corresponds to the value of the atoms* ($\tilde{\theta}_h$)*, while the height corresponds to the weight associated with it. The group-specific distributions* $G_1, \ldots, G_J$ *are conditionally independent draws from a Dirichlet process with baseline measure* G_0*, so the atoms are drawn from it. But, since the baseline measure* G_0 *is also drawn from a DP, it is discrete, and the atoms of the* G_j*s have to be identical to those originally drawn to construct* G_0*.*

where the unique values $\theta^{\star}_{hj}$ in turn are draws from G_0. We have a second instance of the Pólya urn (3.3) for a sequence of draws $\theta^{\star}_{hj} \sim G_0$. A minor notational complication arises by the double index $_{hj}$. Index the $\theta^{\star}_{hj}$ in sequence by running the first index h faster than the second index j, i.e., first we list all unique values among $\{\theta_{i1},\ i = 1, \ldots, I_1\}$, then all additional unique values among $\{\theta_{i2},\ i = 1, \ldots, I_2\}$ that have not yet been recorded, etc. Recall that $\theta^{\star\star}_r$, $r = 1, \ldots, k^{\star}$, are the unique

values among the $\theta^\star_{hj}$. Let $m_{h,j,r}$ be the number of elements among $\{\theta^\star_{11}, \ldots, \theta^\star_{h-1,j}\}$ equal to $\theta^{\star\star}_r$, and let $R_{h,j} = h + \sum_{j'<j} k_{j'}$. We get the predictive probability function

$$\theta^\star_{h,j} \mid \theta^\star_{h-1,j}, \ldots, \theta^\star_{1,1} \sim \sum_{r=1}^{k^\star_{h-1,j}} \frac{m_{h-1,j,r}}{B + R_{h-1,j} - 1} \delta_{\theta^{\star\star}_r} + \frac{B}{B + R_{h-1,j} - 1} H,$$

where $k^\star_{h,j}$ is the number of unique values among $\theta^\star_{11}, \ldots, \theta^\star_{h-1,j}, \theta^\star_{hj}$. The two Pólya urns can be combined to define a collapsed Gibbs sampler similar to §3.3.1, for details, see Teh *et al.* (2006).

Example 15 (Modeling documents using bag of words models) *One of the most illuminating applications of the hierarchical Dirichlet process mixture model is in modeling a collection of documents (corpora) using bag-of-words models.*

Bag-of-words models ignore the order in which observations appear in the text. Individual words are treated as categorical data with a document-specific distribution. Let $y_{i,j} \in \{1, \ldots, D\}$ be a categorical variable such that $y_{i,j} = d$ indicates that the i-th word in the j-th document is the d-th word in a dictionary of size D. In the simplest bag-of-words model, words from document j are assumed i.i.d. with $p(y_{i,j} = d \mid \theta_j) = \theta_{j,d}$. Information is shared across documents through a random-effects distribution on the θ_js, so that $\theta_j \mid G \sim G$. For example, for a nonparametric specification we could set $G \sim \mathsf{DP}(M, G_0)$ with G_0 being a (finite) Dirichlet distribution. This type of models can be considered "single topic" models because all words within a document come from the same probability distribution over words.

A natural extension of this idea is to treat each document as being composed of multiple topics, with topics being shared across documents. For example, a document on the effect of singing on child health might deal with the topics "music" (which places high probability on words such as "song," "melody" and "piano") and "medicine" (which emphasizes words such as "health," "symptom" and "treatment"), while another document about the entertainment options available in San Francisco this weekend might involve again the topic "music," along with the topic "outdoor activities" (focusing on words such as "hike," "ocean" and "sun"). In this extended model, topics corresponds to a different probability distribution over the dictionary, and words are still independently drawn from one of the multiple topics in the document. Such a model can be implemented using a HPDM with sampling model

$$p(y_{i,j} = d \mid \theta_{i,j}) = \theta_{i,j,d}$$

and HPDM prior

$$\theta_{i,j} \mid G_j \sim G_j, \quad G_j \mid G_0 \sim \mathsf{DP}(M, G_0), \quad G_0 \sim \mathsf{DP}(B, \mathsf{Dir}(\eta)).$$

Here $G_0 = \sum w_h \widetilde{\theta}_h$ is a distribution of multinomial probability vectors (over the dictionary) $\widetilde{\theta}_h$. Each $\widetilde{\theta}_h$ corresponds to a topic. Each $G_j = \sum \varpi_{jh} \widetilde{\theta}_h$ is a mixture of topic-specific multinomial probabilities and is the distribution over words for document j. The weights ϖ_{jh} are the relative weights of the topics. To observe a word in document j we first select a topic by drawing $\theta_{ij} \sim G_j$, and then an actual word with $p(y_{ij} = d \mid \theta_{ij}) = \theta_{ijd}$.

5.4.3. Nested DP

In the HPD, information is shared across the distributions $G_1, \ldots, G_J$ by sharing the atoms of the stick-breaking construction. This allows us to cluster draws

across groups, but tells us nothing about how the distributions themselves should be grouped together. There are no ties, $p(G_j = G_\ell) = 0$ for $j \neq \ell$. The nested Dirichlet process (NDP), first introduced in Rodríguez *et al.* (2008), is an alternative construction for the collection $G_1, \ldots, G_J$ which does allow for clustering of the groups as well as clustering of the observations themselves.

Like the HDP, the NDP is a hierarchical model involving two Dirichlet processes,

$$G_j \mid Q \sim Q, \qquad\qquad Q \sim \mathsf{DP}(M, \mathsf{DP}(B, G_0)).$$

Hence, in the NDP the baseline measure for the first Dirichlet process is given by the second Dirichlet process *rather than by a random distribution drawn from it.* The random probability measure Q is a distribution on distributions. Alternatively, we could write NDP in terms of a stick-breaking construction,

$$G_j \mid Q \sim Q, \qquad\qquad Q = \sum_{k=1}^{\infty} w_k \delta_{\widetilde{G}_k},$$

where $w_k = z_k \prod_{h<k}(1 - z_h)$, $z_k \sim \mathsf{Beta}(1, M)$, and $\widetilde{G}_k \sim \mathsf{DP}(B, G_0)$. Hence the first level of the hierarchy generates a distribution on RPMs with point masses corresponding to the random distributions $\widetilde{G}_1, \widetilde{G}_2, \ldots$. These random distributions are then specified nonparametrically through draws from a common Dirichlet process, so that $\widetilde{G}_k = \sum_{l=1}^{\infty} \varpi_{k,l} \delta_{\widetilde{\theta}_{k,l}}$, with $\varpi_{k,l} = v_{k,l} \prod_{h<l}(1 - v_{k,l})$, $v_{k,l} \sim \mathsf{Beta}(1, B)$, and $\widetilde{\theta}_{k,l} \sim G_0$ independently for every k and l.

Writing the NPD in terms of its stick-breaking construction highlights the nature of the NDP as two-level clustering. First, the model clusters similar distributions together, by sampling from $G_j \sim Q = \sum w_k \delta_{\widetilde{G}_k}$. Then the model clusters observations only across distributions that have already been clustered together.

An alternative characterization for the NDP is as a model for random partitions of a set of random distributions. To see this consider the partition $\rho = \{S_1, \ldots, S_K\}$ of $\{G_1, \ldots, G_J\}$, where $S_k = \{j : G_j = G_k^\star\}$, so that the $G_k^\star$s denote a set of distinct random random measures. Then, the definition for Q implies that

$$p(\rho) = \frac{M^K (M-1)! \prod_{k=1}^{K} (n_k - 1)!}{(M + J - 1) K!},$$

where n_k denotes the umber of distributions in the set S_k. This is (3.5), with an additional $K!$ in the denominator to reflect the lack of ordering of the clusters in ρ. From there we can write

$$\begin{aligned} p(G_1, \ldots, G_J) &= p(G_1^\star, \ldots, G_K^\star \mid \rho) p(\rho) \\ &= \left\{ \prod_{k=1}^{K} \mathsf{DP}(G_k^\star \mid B, G_0) \right\} \left\{ \frac{M^K (M-1)! \prod_{k=1}^{K} (n_k - 1)!}{(M + J - 1) K!} \right\}. \end{aligned} \tag{5.4}$$

Finally, since sampling from G_j induces clusters we get two-level clustering.

Computation for NDP mixtures can be easily carried out by replacing the DP with almost sure truncations as the ones discussed in §3.4 (see Rodríguez *et al.*, 2008 for details). Alternatively, we can derive MCMC algorithms that avoid truncating the process by extending the representation in (5.4). To do so, condition on $\rho = \{S_1, \ldots, S_K\}$ and define K sets of partitions $\sigma_1, \ldots, \sigma_K$ with $\sigma_k = \{R_{k,1}, \ldots, R_{k,L_k}\}$, such that the k-th set is associated with the observations drawn from the distribution assigned to S_k. In other words, $R_{k,l} = \{\theta_{i,j} : \theta_{i,j} = \theta_{k,l}^\star\}$ where $\theta_{k,1}^\star, \ldots, \theta_{k,L_k}^\star$

denotes a set of L_k unique draws from $G_k^\star$. Since each $G_k^\star$ is independently drawn from a DP, the implied prior on each σ_k is

$$p(\sigma_k) = \frac{B^{L_k}(B-1)!\prod_{l=1}^{L_k}(m_{k,l}-1)!}{(B+\bar{I}_k-1)L_k!},$$

where $\bar{I}_k$ is the number of observations generated from distributions in group R_k, and $L_k < \bar{I}_k$ is the number of clusters associated with them. Hence, the joint distribution on $G_1, \ldots, G_K$ can be written in terms of ρ, (σ_k) and $(\theta^\star_{k,l})$,

$$p(\rho, (\sigma_k), (\theta^\star_{k,l}) \mid \mathbf{y}) = \left\{\prod_{j=1}^{J}\prod_{i=1}^{I_j} p(y_{i,j} \mid \theta_{i,j})\right\}\left\{\prod_{k=1}^{K}\prod_{l=1}^{L_k} p(\theta^\star_{k,l})\right\}\left\{\prod_{k=1}^{K} p(\sigma_k)\right\} p(\rho).$$

MCMC algorithms can be generated by devising proposal distributions that modify ρ, (σ_k) and/or $(\theta^\star_{k,l})$. One such proposal distribution is discussed in Müller and Nieto-Barajas (2008).

Example 16 (Assesing quality of care in US hospitals) *The Department of Health and Human Services makes available to the public a series of (self-reported) quality of care measures for U.S. hospitals at http://www.hospitalcompare.hhs.gov/. To illustrate the characteristics of the NPM, we consider a model for one of these measures (the percentage of patients who received the appropriate initial antibiotic). The information on hospitals is nested within states, so a NDP mixture is a natural alternative to identify underperforming/overperforming states, as well as underperforming/overperforming hospitals within each group of states. The model clusters states j into sets of states with matching distribution of hospitals. All hospitals within each such set of states are then clustered into sets of hospitals with matching distribution of quality of care measures.*

More specifically, let $y_{i,j}$ be the (suitable transformed) quality of care measurement in hospital $i = 1, \ldots, I_j$ of state $j = 1, \ldots, J$. Then the model becomes

$$y_{i,j} \mid \theta_{i,j} \sim \mathsf{N}(\mu_{i,j}, \tau^2_{i,j}), \quad (\mu_{i,j}, \tau^2_{i,j}) \mid G_j \sim G_j, \quad G_j \mid Q \sim Q, \quad Q = \sum_{k=1}^{\infty} w_k \delta_{\widetilde{G}_k},$$

where $\widetilde{G}_k \sim \mathsf{DP}\left\{B, \mathsf{N}(\mu \mid \mu_0, \sigma/^2\kappa_0)\mathsf{IGamma}(\sigma^2 \mid \nu_0, \tau_0^2)\right\}$.

Figure 5.10 presents the resulting density estimates for four states representative of the clusters generated by the NDP. Note that the estimates demonstrate slightly different levels of skewness in addition to different means.

Example 17 (Document clustering in multi-topic models) *To help clarify the differences between the NDP and the HDP, consider an alternative extension of the bag-of-words model discussed in Example 15, where a nested DP is used to model G_j, the document-specific topic distribution,*

$$y_{i,j} \mid \theta_{i,j} \sim \mathsf{Multinom}(\theta_{i,j}), \quad \theta_{i,j,} \mid G_j \sim G_j, \quad G_j \mid Q \sim Q, \quad Q = \sum_{k=1}^{\infty} w_k \delta_{\widetilde{G}_k},$$

and $\widetilde{G}_k \sim \mathsf{DP}(B, \mathsf{Dir}(\eta))$. The structure on Q implies that documents with matching distributions G_j will be clustered together, something that does not happen under the HDP model. On the other hand, since the $\widetilde{G}_k$s are drawn independently, topics are shared only among documents assigned to a common cluster, but not across clusters of documents.

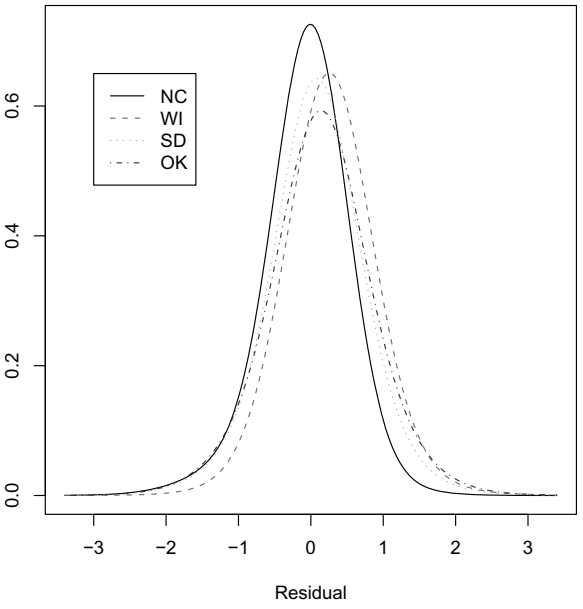


FIG 5.10. *Mean predictive density for four states under the NDP model: North Carolina (NC), Wisconsin (WI), South Dakota (SD) and Oklahoma (OK).*

5.5. DP Models for Time Course Data

5.5.1. Dynamic DP

The "single-p" DDP model can be used to construct collections of random distributions that evolve in discrete time. Consider a setting where at each time $t = 1, \ldots, T$ we collect observations $y_{t,1}, \ldots, y_{t,n_t}$ from a model that is written as a convolution of a normal linear model with a mixing measure G_t on the linear regression parameters:

$$y_{t,i} \mid \theta_{t,i} \sim \mathsf{N}(y_{t,i} \mid x_{t,i}\theta_{t,i}, \sigma^2), \qquad \theta_{t,i} \mid G_t \sim G_t,$$

where $x_{t,i}$ is a row vector. To create a flexible model for the mixing distribution we let

$$G_t(\cdot) = \sum_{h=1}^{\infty} w_h \delta_{\widetilde{\theta}_{t,h}},$$

and define the atoms sequentially by setting

$$\widetilde{\theta}_{0,h} \sim \mathsf{N}(\mathbf{m}_0, \mathbf{C}_0), \qquad \widetilde{\theta}_{t,h} \mid \widetilde{\theta}_{t-1,h} \sim \mathsf{N}(\mathbf{B}_t\widetilde{\theta}_{t-1,h}, \mathbf{W}_t). \tag{5.5}$$

If the weights are defined by the stick breaking prior, as in (5.1), then the model becomes a DDP with point masses $\boldsymbol{m}_h$ replaced by the stochastic process $\widetilde{\theta}_h = (\widetilde{\theta}_{th} : t = 1, 2, \ldots)$ defined by the Markov model (5.5).

This type of dynamic DDP was introduced in Rodríguez and Ter Horst (2008). It can be interpreted as a type II multiprocess model, in the sense of West and Harrison (1997). Since the weights (w_h) are independent of t, the model can be rewritten as a regular DP mixture model where $\mathbf{\Theta}_i = (\theta'_{0,i}, \theta'_{1,i}, \ldots, \theta'_{T,i})$ and

$$y_{t,i} \mid \theta_{t,i} \sim \mathsf{N}(y_{t,i} \mid x_{t,i}\theta_{t,i}, \sigma^2), \qquad \mathbf{\Theta}_t \mid \widetilde{G} \sim \widetilde{G}, \qquad \widetilde{G} \sim \mathsf{DP}(M, \widetilde{G}_0),$$

where the baseline measure $\widetilde{G}_0$ is the multivariate normal distribution induced by (5.5). Using this representation in terms of a single DPM allows us to create a slight generalization where we also mix over the observational variance σ^2. In that case,

$$y_{t,i} \mid \theta_{t,i} \sim \mathsf{N}(y_{t,i} \mid x_{t,i}\theta_{t,i}, \sigma_i^2), \qquad (\mathbf{\Theta}_t, \sigma_i^2) \mid \widetilde{G} \sim \widetilde{G}, \qquad \widetilde{G} \sim \mathsf{DP}(M, \widetilde{G}_0),$$

where $\widetilde{G}_0$ is defined as

$$\widetilde{\theta}_0 \mid \sigma^2 \sim \mathsf{N}(\mathbf{m}_0, \sigma^2\mathbf{C}_0), \quad \widetilde{\theta}_t \mid \widetilde{\theta}_{t-1}, \sigma^2 \sim \mathsf{N}(\mathbf{B}_t\widetilde{\theta}_{t-1}, \sigma^2\mathbf{W}_t), \quad \sigma^2 \sim \mathsf{IGamma}(\nu_0, V_0).$$

Another generalization, in which σ^2 is constant across components but allowed to evolve with time, is presented in Rodríguez and ter Horst (2010).

As with other DDP models, the representation in terms of a simple DPM also allows us to employ all the computational tools described in §3.3 to make inferences on this model. An important consideration, no matter which algorithm is used, is that efficient sampling for the atoms can be accomplished using Forward-Backward algorithms (Carter and Kohn, 1994; Frühwirth-Schnatter, 1994).

An appealing feature of this type of dynamic DDP models is their flexibility. By appropriately choosing the structural parameters x_t, $\mathbf{B}_t$ and $\mathbf{W}_t$ a number of different evolution patterns can be accommodated, including trends, periodicities and dynamic regressions.

Example 18 (Autoregressive models for distributions) *Consider a mixture of order-p autoregressive processes where $x_{it} = (1, 0, 0, \ldots, 0)$ is a vector of length p and*

$$\mathbf{B}_t = \begin{pmatrix} \phi_1 & \phi_2 & \phi_3 & \cdots & \phi_{p-1} & \phi_p \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{pmatrix}$$

is a $p \times p$ matrix. The model is completed by setting a prior on the vector of autoregressive coefficients $(\phi_1, \ldots, \phi_p)$. To simplify computation, this can be chosen as a multivariate Gaussian distribution.

Example 19 (Modeling the evolution of claim distributions) *Rodríguez and Ter Horst (2008) use the dynamic DDP to model the value of travel reimbursement claims in a major international development bank between January 2005 and May 2007. They use a simple random walk model where $y_{t,i} \sim \mathsf{N}(y_{t,i} \mid \theta_{t,i}, \sigma_i^2)$ and $\widetilde{\theta}_t \mid \widetilde{\theta}_{t-1}, \sigma^2 \sim \mathsf{N}(\widetilde{\theta}_{t-1}, \sigma^2 U)$. Figure 5.11 shows the smoothed and one-step-ahead density estimates generated by the model for five months (January to May, 2007).*

5.5.2. Time Series DDP

Nieto-Barajas *et al.* (2008) introduce another variation of DDP models that is suitable as a prior for a time series of random probability measures.Their construction

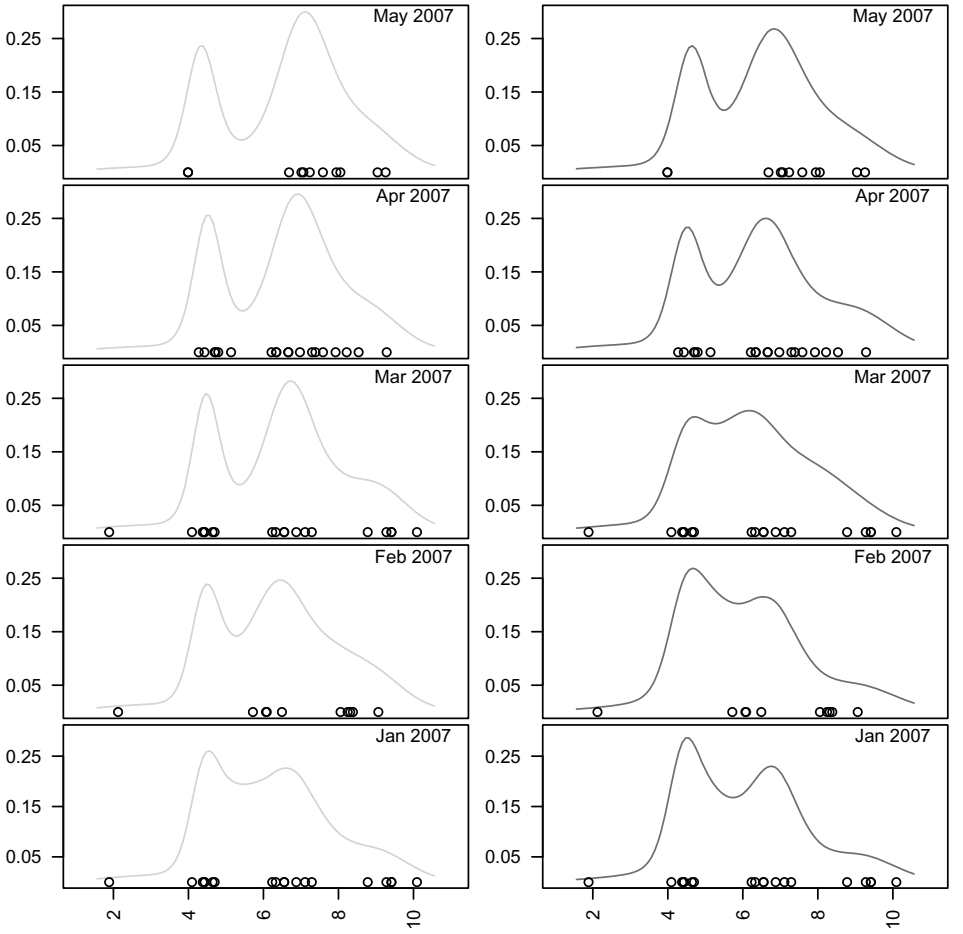


FIG 5.11. *Dynamic density estimates* $p(y_t \mid D_T)$ *and one-step ahead predictive distributions,* $p(y_t \mid D_{t-1})$ *for claims in 2007. Dots correspond to actual observations.*

is very straightforward. Recall the representation of a DP random measure by the stick breaking construction

$$G_t = \sum_{h=1}^{\infty} w_{th}\delta_{\widetilde{\theta}_h}, \tag{5.6}$$

where w_{th} are weights specific to G_t and $\widetilde{\theta}_h$ are point masses. Recall that the weights are defined by iterative stick breaking as $w_{th} = v_{th}\prod_{g<h}(1 - v_{tg})$ with beta distributed fractions v_{th}. The locations of the point masses are assumed to be common across all t. The use of the representation (5.6) already reveals that the proposed construction will be a variation of a common location DDP, i.e., all RPMs G_t have the same atoms $\widetilde{\theta}_h$ and only differ by the weights w_{ht}.

Nieto-Barajas *et al.* (2008) achieve the desired serial dependence by introducing a sequence of latent binomial variables $z_{th} \sim \mathsf{Bin}(k, v_{th})$ and replacing the prior for v_{th} in the stickbreaking construction of the DP prior by

$$v_{th} \mid z_{t-1,h} \sim \mathsf{Beta}\left(1 + z_{t-1,h}, M + \{k - z_{th}\}\right),$$

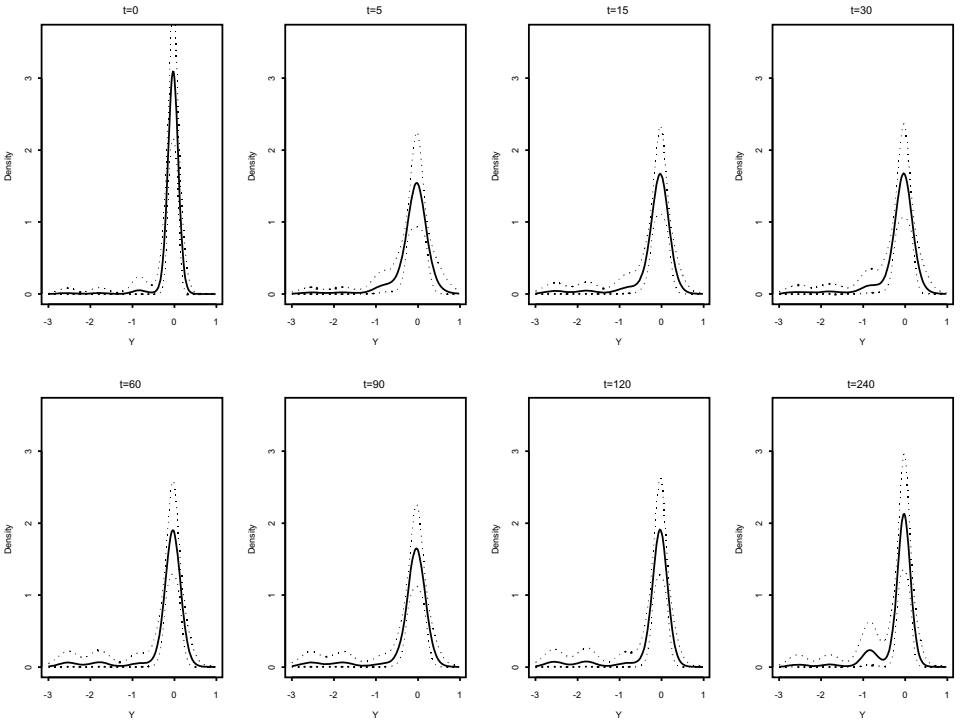


FIG 5.12. *Estimated distributions for* $t \in \{0, 5, 15, \ldots, 240\}$ *under a tsDDP model. The dotted curves show pointwise central 95% credible intervals.*

$t = 2, \ldots, T$. The marginal distribution of v_{th} remains unchanged $v_{th} \sim \mathsf{Beta}(1, M)$, and thus $G_t \sim \mathsf{DP}$, remains unchanged. The choice of k controls the level of dependence, with larger k implying higher dependence. Nieto-Barajas *et al.* (2008) use the model to analyze protein activation over time after an initial intervention. Figure 5.12 shows the estimated distributions G_t for an application of the tsDDP model to inference for protein activations over time after an intervention. Most of the proteins are not impacted by the intervention, only some are. This is reflected in a stable peak around 0 over time, and varying weight in the left tail, corresponding to inhibition of some proteins.

5.6. Spatial DDP

A version of the "single p" DDP model that is suitable for point-referenced spatial data is developed in Gelfand *et al.* (2005), and later extended in Duan *et al.* (2007). In the original definition of the spatial DDP, realizations from a Gaussian process are used as atoms in the stick breaking construction,

$$G_S = \sum_{h=1}^{\infty} w_h \delta_{\widetilde{\theta}_{h,S}}, \qquad \widetilde{\theta}_{h,S} = \left\{\widetilde{\theta}(s) : s \in S\right\} \sim \mathsf{GP}\left\{\mu(s), \gamma(s, s')\right\},$$

with $w_h = v_h \prod_{k<h}\{1-v_k\}$ and $v_h \sim \mathsf{Beta}(1, M)$. In this definition, $\mathsf{GP}\left\{\mu(s), \gamma(s, s')\right\}$ denotes a Gaussian process prior with mean function $\mu(s)$ and covariance function $\gamma(s, s')$. See §1.3.1.

The random distributions G_s can be used, for example, to model the distribution associated with a spatial random-effects. Assume that T independent realizations $y_1, \ldots, y_T$ with $y_t = (y_t(s_1), \ldots, y_t(s_m))'$ are available at locations $s_1, \ldots, s_m$. The model in Gelfand *et al.* (2005) implies that

$$y_t \mid \theta_t \sim \mathsf{N}(\theta_t, \sigma^2 I),$$

where $\theta_t = (\theta_t(s_1), \ldots, \theta_t(s_m))'$, $\theta_t \mid \widetilde{G} \sim \widetilde{G}$, and $\widetilde{G}(\cdot) = \sum_{h=1}^{\infty} w_h \delta_{\widetilde{\theta}_h}(\cdot)$. The GP prior implies

$$\widetilde{\theta}_h = \begin{pmatrix} \widetilde{\theta}_h(s_1) \\ \widetilde{\theta}_h(s_2) \\ \vdots \\ \widetilde{\theta}_h(s_m) \end{pmatrix} \sim \mathsf{N}\left(\begin{pmatrix} \mu(s_1) \\ \mu(s_2) \\ \vdots \\ \mu(s_m) \end{pmatrix}, \begin{pmatrix} \gamma(s_1,s_1) & \gamma(s_1,s_2) & \cdots & \gamma(s_1,s_m) \\ \gamma(s_2,s_1) & \gamma(s_2,s_2) & \cdots & \gamma(s_2,s_m) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(s_m,s_1) & \gamma(s_m,s_2) & \cdots & \gamma(s_m,s_m) \end{pmatrix} \right).$$

In other words, for any finite sample, the spatial DDP reduces to a multivariate DP mixture with a multivariate Gaussian baseline measure whose mean and covariance matrix are structured according to $\mu(s)$ and $\gamma(s, s')$.

An appealing feature of the spatial DPM is that, although it can be centered around a stationary model a priori, it produces a non-stationary model a posteriori. Indeed, note that

$$\mathsf{E}\{y(s) \mid G_S\} = \sum_{h=1}^{\infty} w_h \widetilde{\theta}_h(s),$$

and

$$\begin{aligned} \mathsf{Cov}\{y(s), y(s') \mid G_S\} &= \left\{ \sum_{h=1}^{\infty} \sum_{k=1}^{\infty} w_h w_k \widetilde{\theta}_h(s) \widetilde{\theta}_k(s') \right\} \\ &\quad - \left\{ \sum_{h=1}^{\infty} w_h \widetilde{\theta}_h(s) \right\} \left\{ \sum_{h=1}^{\infty} w_h \widetilde{\theta}_h(s') \right\}, \end{aligned}$$

while, a priori,

$$\mathsf{E}\{y(s)\} = \mu(s), \qquad \mathsf{Cov}\{y(s), y(s')\} = \frac{1}{M+1} \gamma(s, s').$$

An interesting alternative to the spatial DDP is the hybrid DP of Petrone *et al.* (2009). They achieve a more parsimonious representation than the spatial DDP by considering a mixture of unique processes, but with local mixture weights. Each realization can pick up different unique elements at different locations. A related model is discussed in Rodríguez *et al.* (2010).

5.7. Other Dependent Extensions of the DP

The previous sections have focused mostly on "single p" DDPs. The popularity of this class of models is due to the fact that introducing dependence in the weights of the process, and performing posterior computation in the resulting constructions, is typically difficult. This section discusses dependent generalizations of DP mixtures that induce dependence in the weights of stick-breaking representation by replacing the beta-distributed random variables with more general random variables.

5.7.1. Probit Stick-Breaking Processes

Recall the stick-breaking construction of the DP,

$$G(\cdot) = \sum_{h=1}^{\infty} w_h \delta_{\widetilde{\theta}_h}(\cdot),$$

where $\widetilde{\theta}_h \sim G_0$, $w_h = v_h \prod_{s<h}(1 - v_s)$ and $v_h \sim \mathsf{Beta}(1, M)$. Instead, consider stick-breaking ratios where $v_h = \Phi(\alpha_h)$ and $\alpha_h \sim \mathsf{N}(\mu, \sigma^2)$. Here $\Phi(x)$ is a standard normal c.d.f. In that case, we say that G follows a probit stick-breaking process (PSBP) with baseline measure G_0 and shape parameters μ and σ, denoted $G \sim \mathsf{PSBP}(\mu, \sigma, G_0)$. In words, the beta prior for the stick breaking model in the DP prior is replaced by a probit model. This simple change greatly simplifies extensions to dependent priors across families of probabilities measures, similar to the DDP.

The probit stick-breaking process has been discussed by Rodríguez *et al.* (2009), Chung and Dunson (2009) and Rodríguez and Dunson (2011), among others. The random distribution G is well defined (in the sense that $\sum_{h=1}^{\infty} w_h = 1$ almost surely), and very flexible. Indeed, from Proposition 3 and Corollary 1 in Ongaro and Cattaneo (2004), the support of the PSBP with respect to the topology of pointwise convergence is the set of absolutely continuous measures with respect to the baseline measure G_0.

The interpretation of the parameters of the PSBP is similar to those in the DP. Indeed, if $\mu = 0$ and $\sigma = 1$ then $v_h \sim \mathsf{Uni}[0, 1]$ and G follows a DP with $M = 1$; also, as $\mu \to \infty$ then $w_1 \to 1$ and G becomes a degenerate distribution at a random location $\widetilde{\theta}_1 \sim G_0$. More generally, for any measurable set B, $\mathsf{E}\{G(B)\} = G_0(B)$ and

$$\mathsf{Var}\{G(B)\} = \frac{\beta_2}{2\beta_1 - \beta_2} G_0(B)\{1 - G_0(B)\},$$

where $\beta_1 = \mathsf{Pr}(T_1 > 0) = \Phi(\mu/\sqrt{1+\sigma^2})$ and $\beta_2 = \mathsf{Pr}(T_1 > 0, T_2 > 0)$, where $(T_1, T_2)'$ follows a bivariate joint distribution with mean $\mathsf{E}(T_i) = \mu$, $\mathsf{Var}(T_i) = 1+\sigma^2$ and $\mathsf{Cov}(T_1, T_2) = \sigma^2$. Hence, G_0 represents the mean of the process, while μ and σ control the variability of G around G_0. Figure 5.13 presents various random samples from PSBPs that illustrate the effect of the parameters on the realizations.

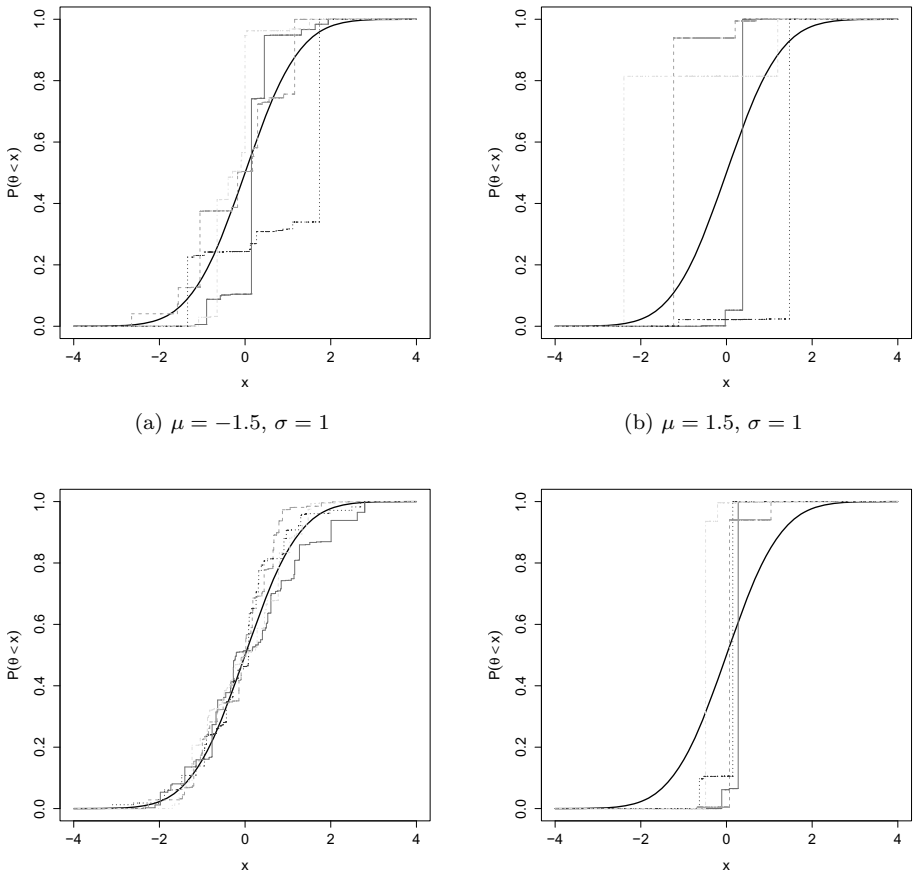
One appealing feature of the PSBP is the computational tractability of PSBP mixture models. In particular, consider a truncated version model

$$y_i \sim p(y_i \mid \theta_i), \qquad \theta_i \mid G \sim G^H,$$

where $G^H(\cdot) = \sum_{h=1}^{H} w_h \delta_{\widetilde{\theta}_h}(\cdot)$ and (w_h) and $(\widetilde{\theta}_h)$ are defined as before but setting $v_H = 1$. In that case, we can introduce random variables $(z_{i,1}), \ldots, (z_{i,H-1})$ such that $z_{i,j} \sim \mathsf{N}(\alpha_h, 1)$ and $\theta_i = \widetilde{\theta}_h$ if and only if $z_{ik} < 0$ for $k < h$ and $z_{ih} \geq 0$. Note that, if we let $s_i = h$ if and only if $\theta_i = \widetilde{\theta}_h$, by integrating out the z_{ih}s we get

$$\mathsf{Pr}(s_i = h) = \mathsf{Pr}(z_{i,1} < 0, \ldots, z_{i,h-1} < 0, z_{i,h} > 0) = \Phi(\alpha_{ih}) \prod_{k<h} \{1 - \Phi(\alpha_{ik})\} = w_h.$$

Conditionally on the auxiliary variables (z_{ih}), the full conditional distribution for α_h is a Gaussian distribution, while conditionally on α_h and the component indicators (s_i), the z_{ih}s are independent and follow (truncated) normal distributions. A similar data augmentation algorithm was originally proposed in the survival analysis literature to fit continuation ratio probit models Albert and Chib (2001).



(a) $\mu = -1.5$, $\sigma = 1$ (b) $\mu = 1.5$, $\sigma = 1$

(c) $\mu = -1.5$, $\sigma = 0.1$ (d) $\mu = 1.5$, $\sigma = 0.1$

FIG 5.13. *Realizations of probit stick-breaking process. The thick line on each Figure corresponds to the same baseline measure G_0 (in this case, a standard normal distribution). The plots demonstrate the effect of the parameters μ and σ (which control how close the realizations are to G_0).*

Alternatively, instead of explicitly truncating the mixing distribution G, designing a slice sampling algorithm similar to the one described §3.3.2 is also possible.

The PSBP can be easily generalized to create dependent probit stick-breaking processes (DPSBP) where dependence is introduced through the weights of the distributions. This is done by replacing the random draws (α_h) by independent realizations $(\alpha_h(x))$ from an appropriate Gaussian process. We illustrate these ideas with two examples.

Example 20 (An alternative to the hierarchical DP) *Consider a situation like the one we described in §5.4.2, where a partially exchangeable set of observations is collected. As before we model*

$$y_{i,j} \mid \theta_{i,j} \sim p(y_{i,j} \mid \theta_{i,j}), \qquad \theta_{i,j} \mid G_j \sim G_j,$$

for $i = 1, \ldots, I_j$ and $j = 1, \ldots, J$. Consider modeling the mixing distributions as

$$G_j(\cdot) = \sum_{h=1}^{\infty} \varpi_{j,h} \delta_{\widetilde{\theta}_h}(\cdot),$$

where $\widetilde{\theta}_h \sim G_0$, $\varpi_{j,h} = \Phi(\alpha_{j,h}) \prod_{k<h}\{1 - \Phi(\alpha_{j,k})\}$, $\alpha_{j,h} \sim \mathsf{N}(\mu_h, \sigma^2)$ and $\mu_h \sim \mathsf{N}(\mu_0, \tau^2)$. This model shares a number of features with the HDP. For example, the collection $G_1, \ldots, G_J$ is exchangeable because the atoms and weights are conditionally independent from each other. Also the distributions are centered around a common mean, in the sense that the transformed weights $(\alpha_{j,1}, \alpha_{j,2}, \ldots)$ are centered around a common value $(\mu_1, \mu_2, \ldots)$, i.e., $\mathsf{E}(\alpha_{j,h}) = \mu_h$ (remember Figure 5.9).

Sampling from this model is straightforward using auxiliary variables. As before, we introduce $z_{i,jh} \sim \mathsf{N}(\alpha_{j,h}, 1)$ and let $s_{i,j} = h$ if and only if $z_{i,j,s} < 0$ for $s < h$ and $z_{i,j,h} \geq 0$. Then, the full conditional distribution for $\alpha_{j,h}$ is normally distributed, i.e.,

$$\alpha_{j,h} \mid \cdots \sim \mathsf{N}\left(\left\{ \frac{1}{\frac{1}{\sigma^2} + n} \right\}^{-1} \left\{ \frac{\mu_h}{\sigma^2} + \sum_{i=1}^{I_j} z_{i,j,h} \right\}, \left\{ \frac{1}{\frac{1}{\sigma^2} + n} \right\}^{-1} \right).$$

On the other hand, the latent variables $z_{i,j,h}$ can be sampled from truncated normal distributions

$$z_{i,j,h} \mid \cdots \sim \begin{cases} \mathsf{N}(\alpha_{j,h}, 1) I(z_{i,j,h} < 0) & h < s_i \\ \mathsf{N}(\alpha_{j,h}, 1) I(z_{i,j,h} \geq 0) & h = s_i \\ \mathsf{N}(\alpha_{j,h}, 1) & h > s_i, \end{cases}$$

where $\mathsf{N}(a, b^2) I(A)$ represents the normal distribution with mean a and variance b^2 truncated to the set A.

Example 21 (Modeling an uncountable collection of distributions) *Consider an index space $\mathcal{X} \in \mathbb{R}^d$ and an uncountable collection of distributions $G_{\mathcal{X}} = \{G_x : x \in \mathcal{X}\}$. Define*

$$G_x(\cdot) = \sum_{h=1}^{\infty} w_h(x) \delta_{\theta_h}, \qquad w_h(x) = \Phi(\alpha_h(x)) \prod_{k<h} \{1 - \Phi(\alpha_k(x))\},$$

and $\alpha_h(x)$ is a Gaussian process over $\mathcal{X}$ with mean μ and covariance function $\sigma^2 \gamma(x, x')$. Given observations associated with locations $x_1, \ldots, x_n$, the joint distribution for the realizations of the latent processes $\alpha_h(x)$ at these locations is given by

$$\begin{pmatrix} \alpha_h(x_1) \\ \alpha_h(x_2) \\ \vdots \\ \alpha_h(x_n) \end{pmatrix} \sim \mathsf{N}\left(\begin{pmatrix} \mu \\ \mu \\ \vdots \\ \mu \end{pmatrix}, \sigma^2 \begin{pmatrix} 1 & \gamma(x_1, x_2) & \ldots & \gamma(x_1, x_n) \\ \gamma(x_2, x_1) & 1 & \ldots & \gamma(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(x_n, x_1) & \gamma(x_n, x_2) & \ldots & 1 \end{pmatrix} \right).$$

Models of this type can be used for time series observed in continuous time ($\mathcal{X} = \mathbb{R}^+$), or to construct models for spatial data ($\mathcal{X} \subset \mathbb{R}^2$). In particular, this construction allows us to easily generate spatial processes for discrete and non-Gaussian distributions. Even more, we can introduce multivariate atoms, leading to a simple

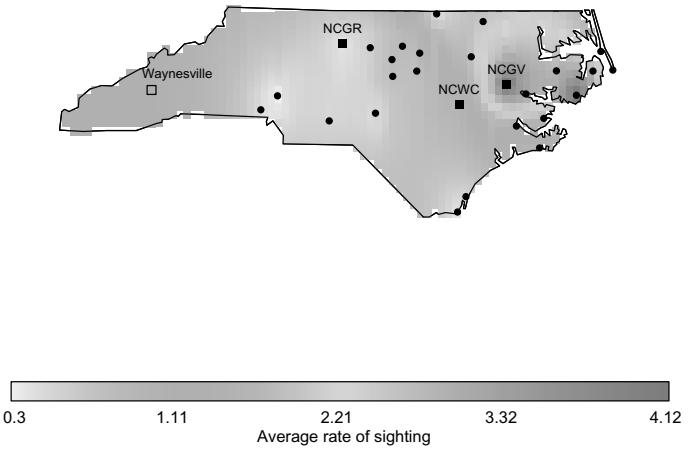


FIG 5.14. *Estimated expected rate of sightings (per man-hour) for the Mourning Dove. Filled dots correspond to the 27 locations where observations were collected. Squared dots represent locations where density estimation is carried out, filled squares represent locations for in-sample-predictions, while the empty square corresponds to a point of out-of-sample prediction.*

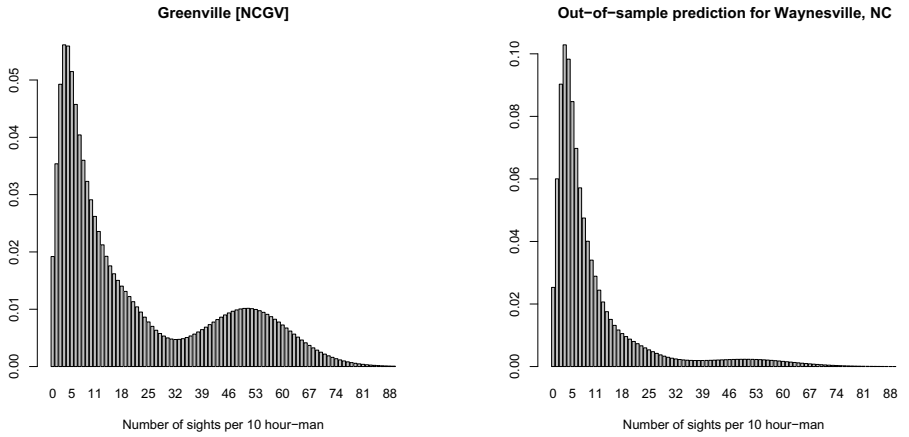


FIG 5.15. *Density estimates for two NC locations. The left panel corresponds an in-sample predictions at Greenville, NC (see also Figure 5.14), while the right panel corresponds to an out-of-sample prediction for a location in the Blue Ridge mountains next to Waynesville, NC.*

procedure to construct non-stationary, non-separable multivariate spatial-temporal processes. By interpreting $\mathcal{X}$ as a space of predictors, this construction also allows us to generate flexible nonparametric regression models with heteroscedastic errors.

Rodríguez and Dunson (2011) use this approach to generate a flexible spatial model for count data, which is used to model bird abundance in North Carolina. Figure 5.14 presents estimates of the expected rate of sightings (per man-hour) for the Mourning Dove, while Figure 5.15 presents predictive distributions for the number of sightings at two different locations.

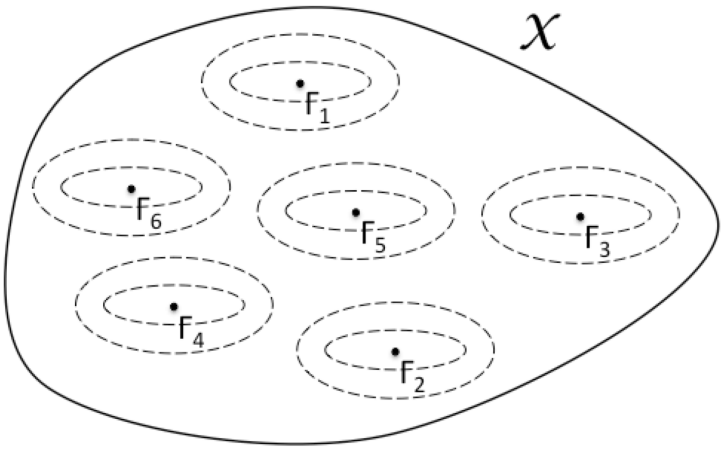


FIG 5.16. *Idealized representation for the construction of the kernel stick-breaking process.*

5.7.2. Kernel Stick-Breaking Processes

The kernel stick-breaking process (KSBP) (Dunson and Park, 2007) is another approach to create a prior over an uncountable collection of distributions $G_{\mathcal{X}} = \{G_x : x \in \mathcal{X} \in \mathbb{R}^d\}$.

In its simplest version, the KSBP is constructed by rebalancing its weights according to the distance between the value of the covariate x and a set of (random) fixed basis locations $\Gamma_1, \Gamma_2, \ldots$. More specifically, given a kernel $K(\cdot,\cdot) : \mathcal{X} \times \mathcal{X} \to [0,1]$ and a probability measure Q defined on the index space $\mathcal{X}$, draw $\Gamma_h \sim Q$ for $h = 1, 2, \ldots$ and, for every $x \in \mathcal{X}$, create the countable collection $(K(x, \Gamma_h))$ (see Figure 5.16). The distribution G_x is then defined as

$$G_x(\cdot) = \sum_{h=1}^{\infty} w_h(x)\delta_{\widetilde{\theta}_h},$$

where $\widetilde{\theta}_h \sim G_0$, $w_h(x) = u_h(x)\prod_{k<h}\{1 - u_k(x)\}$, $u_h(x) = v_h K(x, \Gamma_h)$ and $v_h \sim \mathsf{Beta}(1, M)$.

Example 22 (KSBP with Gaussian kernels) *Consider a KSBP on $\mathcal{X} = \mathbb{R}^d$ where $K(x, x')$ is a Gaussian kernel, i.e.,*

$$K(x, x') = \exp\left\{-\lambda||x - x'||^2\right\},$$

in which case

$$G_x(\cdot) = \sum_{h=1}^{\infty}\left\{v_h K(x, \Gamma_h)\prod_{k<h}[1 - v_k K(x, \Gamma_k)]\right\}\delta_{\widetilde{\theta}_h}$$

and, for example, $\Gamma_h \sim \mathsf{N}(0, \tau^2)$. Note that, if $\lambda \to 0$, then $K(x, \Gamma_h) = 1$ for every pair (x, Γ_h), and the model reduces to a DP prior. Similarly, if $M \to 0$, G_x becomes a degenerate distribution at a random location $\widetilde{\theta}_1$ for every $x \in \mathcal{X}$. As before, a model of this type can be used for nonparametric regression, and well as for modeling non-stationary, non-separable temporal ($\mathcal{X} = \mathbb{R}^+$), spatial ($\mathcal{X} \subset \mathbb{R}^2$) or spatio-temporal ($\mathcal{X} \subset \mathbb{R}^3$) processes with non-Gaussian marginals.

A slightly more general version of this model can be obtained by replacing the point masses by random distributions drawn from a Dirichlet process and/or by replacing the prior on the v_hs with a more general beta distribution, so that $v_h \sim \mathsf{Beta}(a_h, b_h)$. In any case, the weights of the KSBP satisfy $\sum_{h=1}^{\infty} w_h(x) = 1$ for all $x \in \mathcal{X}$, and each member G_xs is therefore well defined.

Consider now a conditionally independent sequence where $\theta_i \mid \mathcal{G}, x_i \sim G_{x_i}$ and $\mathcal{G}_{\mathcal{X}} = \{G_x : x \in \mathcal{X}\}$ is assigned a KSBP prior. An interesting feature of the KSBP is that the joint distribution for $\theta_1, \ldots, \theta_n \mid x_1, \ldots, x_n$ obtained after integrating the random elements in $\mathcal{G}_{\mathcal{X}}$ can be obtained in closed form. As with the Dirichlet process, this joint distribution is obtained as a product of predictive distributions, each one corresponding to a generalized Pólya urn.

Posterior inference for the KSBP can be accomplish using through a Markov chain Monte Carlo algorithm that combines retrospective sampling and generalized Pólya urn sampling steps. Details can be seen in Dunson and Park (2007).

Chapter 6

Dependent Tailfree Process and Dependent Multivariate PT

6.1. Linear Dependent Tailfree Process (LDTP)

The popular DDP models for families of random probability measures $\mathcal{G} = \{G_x,\ x \in X\}$ inherits a limitation from the underlying DP prior. The probability measures G_x are a.s. discrete. We earlier discussed a simple fix by convolution with continuous kernels. Alternatively, Jara and Hanson (2011) define a nonparametric Bayesian prior model $p(\mathcal{G})$ that builds on the PT construction and allows to generate absolutely continuous distributions G_x. Recall the construction of the PT prior by defining

$$G(B_{\epsilon 0} \mid B_\epsilon) \equiv Y_{\epsilon 0} \sim \mathsf{Be}(a_{\epsilon 0}, a_{\epsilon 1}) \tag{6.1}$$

for B_ϵ and $B_{\epsilon 0}$ in two adjacent levels of the nested partition. By definition of the PT prior, $Y_{\epsilon 0}$ are independent *across* ϵ. Jara and Hanson (2011) build on (6.1) to define a prior for $\mathcal{G}$. Similar to (6.1) they define

$$Y_{x,\epsilon 0} = G_x(B_{\epsilon 0} \mid B_\epsilon),$$

and introduce dependence *across* x by a simple logistic regression

$$Y_{x\epsilon 0} = \frac{\exp(x'\beta_{\epsilon 0})}{1 + \exp(x'\beta_{\epsilon 0})}, \tag{6.2}$$

leaving independence across ϵ intact. They recommend a g-prior $\beta_{\epsilon 0} \sim N[0, g(X'X)^{-1}]$, with $g = 2n/c$.

Figure 6.1 shows inference in an example. Model (6.2) is a natural equivalent of the PT model for $G \sim \mathsf{PT}$ to families of random probability measures $\{G_x,\ x \in X\}$. However, we should note that, because the logistic model (6.2) for a fixed x does not reduce a beta prior, the implied marginal for G_x is not a PT prior and the model is not, strictly speaking, an extension of the PT model discussed in Chapter 4.

6.2. Dependent PTs

Trippa *et al.* (2011) develop a generalization of the PT prior to a model for related RPMs $\mathcal{G} = \{G_x,\ x \in X\}$ for applications similar to the DDP model. Trippa *et al.* (2011) define a dependent multivariate PT prior (MPT) for $\mathcal{G}$. The advantage of the MPT is the possibility to restrict the model to continuous random distributions and an elegant construction to introduce the dependence. In contrast to the DDP there is no need to track point masses across covariates, and covariates can be of any data format, continuous, categorical or count variables. To avoid confusion we

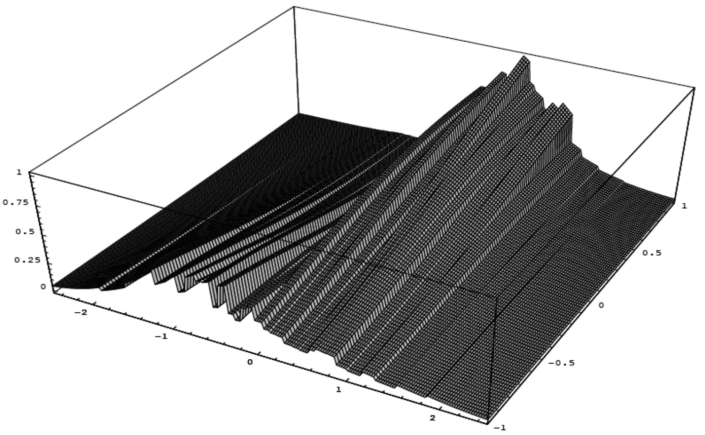


FIG 6.1. *Posterior estimates* $\overline{G}_x = E(G_x \mid data)$. *The figure plots* $\overline{G}_x(t)$ *against* $x \in (-1, 1)$ *and* $-2.5 \leq t \leq 2.5$.

note that this construction is different from the multivariate PT of §4.5. The latter defines a single random probability measure G on a multivariate sample space, while the former constructs a BNP prior for a family $\mathcal{G}$ of random probability measures indexed by x.

Recall the independent random splitting probabilities for a PT random measure $G \sim \mathsf{PT}$,

$$Y_\epsilon = G(B_{\epsilon 0} \mid B_\epsilon) \sim \mathsf{Be}(a_{\epsilon 0}, a_{\epsilon 1}).$$

Note that in anticipation of the upcoming construction we index the random splitting probabilities by ϵ, rather than $(\epsilon, 0)$, as before. By definition of the PT, Y_ϵ are independent across ϵ. The construction of the MPT is conceptually very straightforward. We simply replace the independent beta random variables by random processes $Y_{x,\epsilon}$ with unchanged marginal beta prior, but now with dependence across x.

6.2.1. Multivariate Beta Process

We refer to the desired process $Y_{x,\epsilon}$ as a multivariate beta process (MPT). For clarification we note that the MPT is unrelated to the beta process of Hjort (1990). The MPT uses one realization of the MBP for each ϵ. In the upcoming brief definition of the MBP we simplify notation by dropping the ϵ index in $Y_{x,\epsilon}$.

The construction starts with the representation of a beta random variable Y_x as a ratio of gamma random variables. Let G^o_x, G^1_x denote two independent gamma random variables. Then $Y_x = G^o_x/(G^o_x + G^1_x)$ is a beta random variable. Next we generate the gamma random variables indirectly as illustrated in Figure 6.2 as the random measures $\Gamma(S^o_x)$ and $\Gamma(S^1_x)$ assigned by a gamma process $\Gamma(\cdot)$ to the area circumscribed by kernels centered at x under a gamma process $\Gamma(\cdot)$.

Let X denote the covariate space. The gamma process is defined on $X \times \Re$; if $X = \Re^k$, then the gamma process is indexed by $(k + 1)$ dimensional Borel sets. The trick is that the same construction works just as well to define Y_{x_1} and Y_{x_2} for two covariate values x_1, x_2. The magic is that the overlap of the kernels centered at x_1 and x_2 induces exactly the kind of desired dependence between Y_{x_1} and Y_{x_2}. If the kernels are $N(x, \sigma)$ Gaussian kernels, then the choice of the scale σ determines

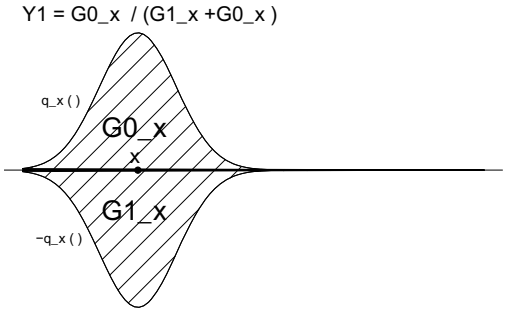


FIG 6.2. *Generating the beta r.v.* Y_x *as ratio of gamma random measures. The gamma random variables are created as random measures* $G_x^0 = \Gamma(S_x^0)$ *and* $G_x^1 = \Gamma(S_x^1)$ *of two the sets* S_x^0 *and* S_x^1 *under a gamma process* $\Gamma(\cdot)$.

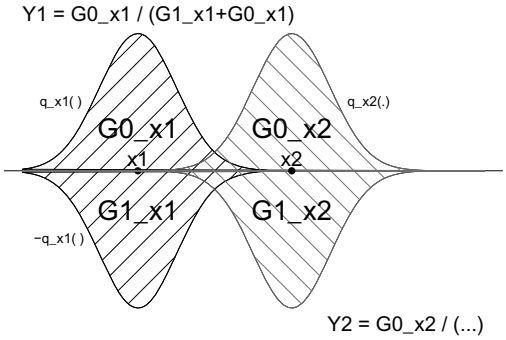


FIG 6.3. *Ratios of gamma random variables define two dependent beta random variables,* $Y_{x_1} = G_{x_1}^o/(G_{x_1}^o + G_{x_2}^1)$ *and* $Y_{x_2} = G_{x_2}^o/(G_{x_2}^o + G_{x_2}^1)$. *The overlap of the kernels determines the correlation. The construction generalizes to families* $\{Y_x;\ x \in X\}$.

the level of dependence. Finally, the construction generalizes to $\{Y_x;\ x \in X\}$, as desired.

In summary, we define the MBP as follows. Let $Q = \{q_x(\cdot)\}$ denote a family of kernels indexed by $x \in X$. For continuous covariates the kernels could be, for example, Gaussian kernels centered at x. Let $S_x^0 = \{(\xi, \zeta),\ \zeta \in X, 0 < \zeta < \alpha_0 q_x(\xi)\}$ denote the area circumscribed by $\alpha_0 q_x(\cdot)$, and similarly for S_x^1 for $-\alpha_1 q_x(\cdot)$. Define $Y_x = \frac{G(S_x^0)}{G(S_x^0)+G(S_x^1)}$ for $x \in X$. In this case we say that $\{Y_x;\ x \in X\} \sim \mathsf{MBP}(\alpha_0, \alpha_1, Q)$.

For use in posterior simulation we note an alternative equivalent construction. For a detailed description of posterior inference see Trippa *et al.* (2011). Here we only introduce the main trick of constructing posterior simulation for the MBP (and thus in the MPT) as standard posterior simulation in DP models. Assume $(Y_{x_1}, \ldots, Y_{x_m}) \sim \mathsf{MBP}(\alpha_0, \alpha_1, Q)$ indexed by x_i, $i = 1, \ldots, m$. In words, we will construct $(Y_{x_1}, \ldots, Y_{x_m})$ as ratios of random measures generated under a DP prior. The construction hinges on the fact that a DP random measure can be written as a normalized gamma process. Thus the ratio of random measures under a DP prior takes the form of a ratio of gamma distributed random variables.

Let $S \equiv \bigcup_{i=1}^m (S_{x_i}^0 \cup S_{x_i}^1)$ denote the region bounded by the $2 \cdot m$ kernels and let $\nu_{x_1 \cdots x_m} \equiv$ denote Lebesgue measure on S. The region S is shown in Figure 6.4.

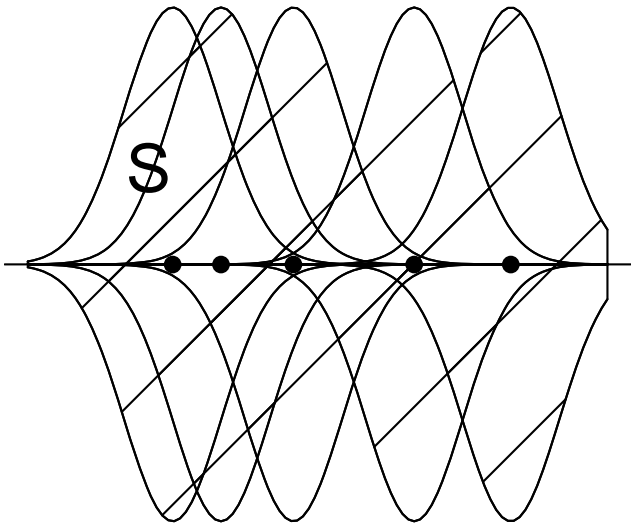


FIG 6.4. *Region S cicumscribed by the union of the* 2 *m kernels.*

Consider a DP random measure

$$D_{x_1 \cdots x_m} \sim \mathsf{DP}(\nu_{x_1 \cdots x_m}).$$

The representation of the DP as a normalized gamma process implies

$$Y_{x_i} = \frac{G(S^0_x)}{G(S^0_x) + G(S^1_x)} \stackrel{d}{=} \frac{D_{x_1 \cdots x_m}(S^0_{x_i})}{D_{x_1 \cdots x_m}(S^0_{x_i} \cup S^1_{x_i})}.$$

This representation can be used to construct a Pólya urn scheme to simulate draws from the MBP. Assume that $Z_i \mid Y_{x_i} \sim \mathsf{Ber}(Y_{x_i})$ and $\{Y_{x_i}\} \sim \mathsf{MBP}$. Then $\boldsymbol{Z} = (Z_1, \ldots, Z_m)$ can be generated as follows. First generate a sequence $(\xi_h, \zeta_h) \sim D_{x_1 \cdots x_m}$. These are points in S (area between the kernels). Find the first pair $(\xi_h, \zeta_h) \in S^0_{x_i} \cup S^1_{x_i}$ and record $Z_i = I(\zeta > 0)$ for that pair. Then we repeat the same for Z_2, etc. The key feature is that the (ξ, ζ) sequence can be generated by the Pólya urn scheme for a marginal sample from a DP random measure, marginalizing with respect to $D_{x_1 \cdots x_m}$.

Later, when we use the MBP to define a prior for the random splitting probabilities $Y_{x_i,\epsilon}$ in the MPT, then the Z_i will be the binary digits of observations $y_i \sim G_{x_i}$, with $p(G_x)$ defined by the binary splitting probabilities $Y_{x,\epsilon}$. Details of this construction are described next.

6.2.2. Dependent Multivariate Pólya Tree

The MBP can be used to generate the beta random variables $Y_{x,\epsilon}$ for a PT prior $G_x \sim \mathsf{PT}$ for $x \in X$, with the dependence across x induced by the dependence of the MBP. We first discuss the construction for a uniform centering distribution, i.e., $E(G_x) = \mathsf{Uni}[0,1]$ for all x. We use the dyadic quantiles of the uniform on $[0,1]$ to define the nested partition sequence, i.e., B_ϵ are the sets $[0, \frac{1}{2}), [\frac{1}{2}, 1], [0, \frac{1}{4})$, etc.

Then, we use the MBP to define the random splitting probabilities. For each ϵ define $\{Y_{\epsilon,x}\} \sim \mathsf{MBP}(\alpha_\epsilon, \alpha_\epsilon, Q)$, using one MBP for each ϵ. We define a family of

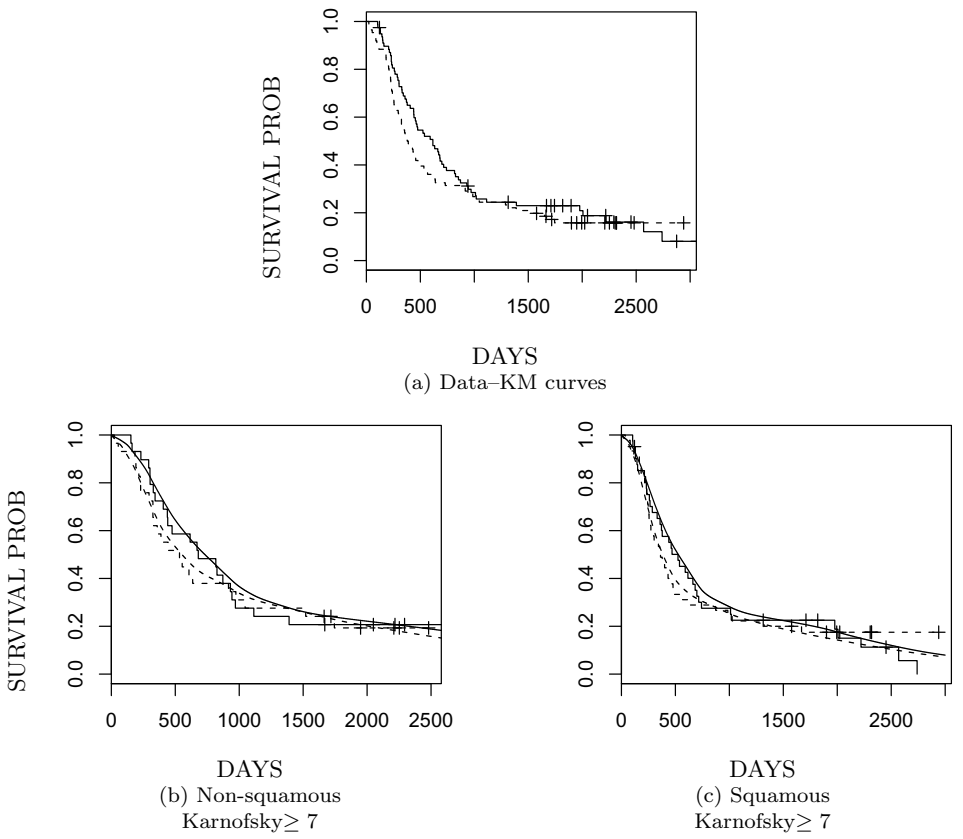


FIG 6.5. *Lung cancer trial. Panel (a) shows the data as Kaplan–Meier curves for treatment (solid line) and control (dashed line). Panels (b) and (c) show the model-based estimates of survival curves arranged by cancer histology, together with the corresponding KM curves.*

RPMs by

$$P_x(B_{\epsilon_1\cdots\epsilon_m}) = \prod_{j=1;\epsilon_j=0}^{m} Y_{\epsilon_1\cdots\epsilon_{j-1}0,x} \prod_{j=1;\epsilon_j=1}^{m} (1 - Y_{\epsilon_1\cdots\epsilon_{j-1}0,x}).$$

We write $\{P_x;\ x \in X\} \sim \mathsf{MPT}(\mathcal{A}, Q, \mathsf{Uni}[0,1])$. The third argument, marks the centering $E(P_x) = \mathsf{Uni}[0,1]$ at the uniform distribution.

Arbitrary centering distributions F_x are easily achieved by modifying the definition to

$$P_x(F_x^{-1}(B_{\epsilon_1\cdots\epsilon_m}) = \prod_{j=1;\epsilon_j=0}^{m} Y_{\epsilon_1\cdots\epsilon_{j-1}0,x} \prod_{j=1;\epsilon_j=1}^{m} (1 - Y_{\epsilon_1\cdots\epsilon_{j-1}1,x}).$$

We write $\{G_x;\ x \in X\} \sim \mathsf{MPT}(\mathcal{A}, Q, F_x)$.

Example 23 *Lung Cancer Trial. Lad* et al. *(1988) report a clinical trial for lung cancer patients. The trial compared radiotherapy versus radiotherapy plus adjuvant chemotherapy. The overall survival data for this study are published in Piantadosi (1997) and is shown in Figure 6.2a. Notice the crossing survival functions The trial enrolled* $n = 164$ *patients, of whom 28 were alive at the end of the followup period.*

The two most important baseline covariates were indicators for squamous versus non-squamous (x_{i1}) histology and performance status at enrollment (x_{i2}). The latter is dichotomized Karnofsky score, with $x_{2i} = 1$ for Karnofsky score ≥ 7. We define a third covariate x_{0i} for treatment assignment with $x_{0i} = 1$ for radiotherapy plus chemotherapy. Trippa et al. *(2011) use the MPT to analyze the data. The MPT model with covariates $\mathbf{x}_i = (x_{i0}, x_{i1}, x_{i2})$ and overall survival times y_i as outcomes implements a fully nonparametric regression for these survival data. Figure 6.2 shows the data and the estimated survival curves.*

Chapter 7

Species Sampling Models

7.1. Introduction

One of the reasons for the widespread popularity of the DP prior model is the computational simplicity of posterior simulation and posterior predictive inference. This simplicity is in part due to the almost sure discrete nature of a random probability measure G with DP prior.

Recall from the discussion in §2.1 that the discrete nature of G naturally induces a prior on random partitions, in the following sense. Consider a random sample, $x_i \mid G \sim G$, $i = 1, \ldots, n$, with $G \sim \mathsf{DP}(M, G_0)$. The discreteness of G implies a positive probability of ties among the x_i. We can use these ties to partition experimental units into clusters defined by the unique values. Let $x_j^\star$, $j = 1, \ldots, k$, define the $k \leq n$ unique values among the x_i and define clusters $S_j = \{i : x_i = x_j^\star\}$. Also, let $n_j = |S_j|$ denote the size of the j-th cluster and let $\boldsymbol{n}_n = (n_{1n}, \ldots, n_{kn})$. When the number n of experimental units is understood from the context we drop the index $_n$.

Finally, recall posterior predictive inference under i.i.d. sampling from a DP random measure:

$$p(x_{n+1} \mid x_1, \ldots, x_n) = \begin{cases} \delta_{x_j^\star}(x_{n+1}) & \text{w. prob } \frac{n_j}{n+\alpha} \equiv p_j(\boldsymbol{n}) \\ G_0(x_{n+1}) & \text{w. prob } \frac{\alpha}{n+\alpha} \equiv p_{k+1}(\boldsymbol{n}). \end{cases} \tag{7.1}$$

In anticipation of the upcoming discussion, the posterior predictive distribution can also be written as

$$x_{n+1} \mid x_1, \ldots, x_n \sim \sum_{j=1}^{k} p_j(\boldsymbol{n}) \delta_{x_j^\star} + p_{k+1}(\boldsymbol{n}), \tag{7.2}$$

with weights $p_j(\boldsymbol{n})$. This is known as the predictive probability function (PPF). The first important feature to see in (7.1) is what is not seen. The posterior predictive is integrated with respect to the random probability measure G. This is important for computation. It would be impossible to keep an infinite dimensional quantity G in computer memory. In other words, the Pólya urn (7.1) characterizes the marginal distribution $p(x_1, \ldots, x_n)$, simply by multiplying the posterior predictive $p(x_{i+1} \mid x_1, \ldots, x_i)$ for $i = 1, \ldots, n-1$ and the marginal $p(x_1) = G_0(x_1)$. The second important feature to note is that the weights $p_j(\cdot)$ of the clusters are a function of only the cluster size n_j. Neither the actual observations $x_i \in S_j$, nor the number and sizes of other clusters enter the expression.

7.2. Predictive Probability Functions

We introduced the PPF in (7.2) as the predictive rule that is implied by i.i.d. sampling from a probability measure with DP prior. However, also the opposite is

true; the PPF characterizes the DP prior and is a defining property of the process. This is not immediately obvious; to formally state this we introduce the notion of a species sampling sequence.

An *exchangeable* sequence of random variables $x_1, x_2, \ldots$, is called a species sampling sequence (SSS) if

$$x_{n+1} \mid x_1, \ldots, x_n \sim \sum_{j=1}^{k_n} p_j(\boldsymbol{n}_n)\delta_{x_j^\star} + p_{k_n+1}(\boldsymbol{n}_n)G_0,$$

with weights depending on the data only indirectly through the cluster sizes n_j.

The family of functions $\{p_j(\boldsymbol{n}_n)\}$ is called the predictive probability function (PPF). As the weights in the posterior predictive distribution any PPF needs to satisfy

$$p_j(\boldsymbol{n}_n) \geq 0, \qquad \sum_{j=1}^{k_n+1} p_j(\boldsymbol{n}_n) = 1, \tag{7.3}$$

for all $\boldsymbol{n}_n$. In the definition of the SSS and PPF, the restriction to exchangeable sequences $(x_n)_{n\geq 1}$ is important. Not every family of functions $\{p_j(\mathbf{n})\}$ that satisfies (7.3) is a PPF. In fact, most such families are not.

The expression for the weights $p_j(\cdot)$ in the Pólya urn for the DP appear particularly simple with $p_j \propto n_j$. It can be shown (Gnedin and Pitman, 2006; Lee *et al.*, 2008) that any PPF with weights that are functions of the cluster size only must be of essentially that form. More specifically, if $p_j = f(n_j)$ for some function of the cluster size, $j = 1, \ldots, k_n$ and $p_{k_n+1} = \theta$, then $f(n_j) = an_j$ for some $a > 0$. Actually, for finite exchangeable sequences $x_1, x_2, \ldots, x_{n+1}$ of categorical random variables, i.e., possible outcomes $x_i \in \{1, \ldots, t\}$, the same result is known as Johnson's sufficientness postulate (Zabell, 1982).

We are now ready to state how the predictive rule of a SSS characterizes a random probability measure. An exchangeable sequence of random variables (x_n) is a SSS if and only if $x_i \sim G$, i.i.d., for some random distribution G that admits a representation of the form

$$G = \sum_{h=1}^{\infty} p_h \delta_{m_h} + RG_0,$$

with $m_h \sim G_0$, i.i.d., and some sequence of positive random weights p_h such that $\sum_{h=1}^{\infty} p_h \leq 1$ (Pitman, 1996, Proposition 11). The random probability measure G is called the species sampling model (SSM) of the SSS (x_n).

A SSM can be defined directly by specifying a prior for the weights p_h, and a distribution for the point masses m_h. The only constraint is the positivity of the p_h and the constraint on the sum of the weights. Alternatively an SSM can be (implicitly) defined through its PPF. The characterization is very useful for computational purposes, but of little use to construct an SSM, because of the difficult constraint that the implied sequence x_n be exchangeable.

A third characterization of an SSM is through the implied prior on the sequence of random partitions. A sequence of discrete random variables (x_n) defines a partition of $\{1, \ldots, n\}$ into clusters $S_j = \{i : \ x_i = x_j^\star\}$ of tied observations. Thus the SSM indirectly defines a sequence of priors for partitions. As before, let $\boldsymbol{n}_n = (n_1, \ldots, n_{k_n})$ denote the cluster sizes of the partition of $\{1, \ldots, n\}$. Since the

sequence x_n is exchangeable it suffices to specify the probability of $\boldsymbol{n}_n$. The probability for any two partitions with the same cluster sizes $\boldsymbol{n}_n$ must be the same. The implied prior $p(\boldsymbol{n}_n)$ is known as the exchangeable partition probability function (EPPF). Let $\mathsf{N}^\star = \cup_{k=1}^{\infty} \mathsf{N}^k$ and let $\boldsymbol{n}^{j+}$ denote $\boldsymbol{n}$ with the the j-th cluster size incremented by 1. Formally an EPPF $p(\cdot)$ is a symmetric function $p : \mathsf{N}^\star \to [0, 1]$ with

$$p(\boldsymbol{n}) = \sum_{j=1}^{k_n+1} p(\boldsymbol{n}^{j+}) \text{ for all } \boldsymbol{n} \in \mathsf{N}^\star \tag{7.4}$$

and $p(1) = 1$. The condition simply formalizes coherence across sample sizes. The probability of partitions for the first n elements of a SSS must match the appropriate marginal of the probabilities for partitions of the first $n + 1$ elements.

The converse is also true. For any function that could be interpreted as an EPPF, i.e., that satisfies the above condition, there is a SSS that gives rise to it (Pitman, 1996, Proposition 13). Again, similar to the characterization of a SSM by PPF, the definition through the EPPF is of little practical use. It is difficult to directly elicit and specify a legitimate EPPF that satisfies (7.4).

Finally, there is an obvious link between the EPPF and the PPF. Every EPPF defines a PPF through

$$p_j(\boldsymbol{n}) \equiv p(\boldsymbol{n}^{j+})/p(\boldsymbol{n}).$$

7.3. More SSMs

We used the DP prior to introduce the notion of the PPF. Some other examples of SSMs are the Pitman Yor (PY) process, the normalized inverse Gaussian (NIG) and Gibbs type priors.

Pitman-Yor Process

The PY process (Pitman, 1995; Pitman and Yor, 1997) is more easily introduced as a stick breaking prior. A random probability measure $G = \sum_h w_h \delta_{\theta_h}$ has a $\mathsf{PY}(\sigma, \alpha, G_0)$ prior if $w_h = \prod_{\ell<h}(1-v_\ell)v_h$ for $v_h \sim \mathsf{Be}(1-\sigma, \alpha+h\sigma)$, independently with $0 \le \sigma < 1$ and $\alpha > -\sigma$, and the locations θ_h are a random sample from the base measure, $\theta_h \sim G_0$. See Ishwaran and James (2001) for a discussion of this construction and a larger class of random probability measures defined by similar stick breaking algorithms. The PPF implied by the PY process is simply

$$x_{n+1} \mid x_1, \ldots, x_n \sim \sum_{j=1}^{k_n} \frac{n_j - \sigma}{n + \theta} \delta_{x_j^\star}(x_{n+1}) + \frac{\theta + k_n\sigma}{n + \theta} G_0(x_{n+1}),$$

while the EPPF reduces to

$$p(\boldsymbol{n}) = \frac{\Gamma(\theta + 1)}{(\theta + k_n\sigma)\Gamma(\theta + n)} \prod_{j=1}^{k_n} \left\{ (\theta + j\sigma) \frac{\Gamma(n_j - \sigma)}{\Gamma(1 - \sigma)} \right\}.$$

Homogeneous NRMIs

Many more SSMs exist. Any homogeneous NRMI is a SSM. Recall from §1.2.6 the construction of NRMI's as normalized CRM, which in turn can be constructed with

a Poisson process with intensity $\nu(x,s)$ on $X \times \Re^+$. An NRMI is called homogeneous when the intensity factors as $\nu(x,s) = \rho(s)G_0(x)$. While all homogeneous NRMIs define a SSM, most do not allow a closed form expression for the weights in the PPF. The most prominent exception is the DP. Another is the normalized inverse Gaussian process (NIG) that was already briefly introduced in §1.2.6. The NIG is in many ways similar to the DP prior. Recall the characterization of a DP for G as assigning a Dirichlet prior to $(G(A_1), \ldots, G(A_k))$ for any partition $\{A_1, \ldots, A_k\}$ of the sample space. Similarly a NIG prior for random probability measure G can be defined by requiring a normalized inverse Gaussian distribution for $(G(A_1), \ldots, G(A_k))$. See §1.2.6 for a statement of the normalized inverse Gaussian distribution. Like the DP the NIG allows closed form expressions for the PPF. See, for example, Lijoi *et al.* (2007b) (using $\sigma = 1/2$) or Lijoi *et al.* (2005). The NIG is a special case of the more general normalized generalized gamma (NGG) process.

Gibbs Type Priors

Another large class of SSMs are Gibbs type priors (Gnedin and Pitman, 2006). Gibbs type priors can be defined by the EPPF. Let $a_k = a(a+1)\cdot\ldots\cdot(a+k-1)$ define a rising factorial. A Gibbs type prior is a prior for a discrete random probability measure with EPPF

$$p(\boldsymbol{n}) = V_{n,k} \prod_{j=1}^{k} (1-\sigma)_{n_j-1}.$$

with $\sigma < 1$, $V_{1,1} = 1$ and $V_{n,k} = V_{n+1,k}(n - k\sigma) + V_{n+1,k+1}$. The last condition is simply (7.4). The implied weights in the PPF are

$$p_j(\boldsymbol{n}) \propto V_{n+1,k}(n_j - \sigma), \qquad p_{k+1}(\boldsymbol{n}) \propto V_{n+1,k+1}.$$

Lijoi *et al.* (2007a) discuss some results about the predictive distribution under this model.

Chapter 8

Random Partition Models

8.1. Introduction

In earlier chapters we discussed nonparametric Bayesian priors $p(G)$ for random probability measures. The most commonly used model is the DP prior and its variations and extensions. One of the many interesting properties of the DP is the almost sure discrete nature of a random probability measure G with DP prior, $G \sim \mathsf{DP}(M, G_0)$. The discrete nature of G naturally induces a prior on random partitions, as we have seen many times before in earlier chapters. Consider a random sample, $x_i \mid G \sim G$, $i = 1, \ldots, n$, generated from a probability model with DP prior, $G \sim \mathsf{DP}(M, G_0)$. The discreteness of G implies a positive probability of ties among the x_i. We can use these ties to define a partition of the experimental units $\{1, \ldots, n\}$ as

$$\{1, \ldots, n\} = \bigcup_{j=1}^{k} \underbrace{\{i : \; x_i = x_j^\star\}}_{S_j}$$

defined by the unique values $x_1^\star, \ldots, x_k^\star$. In other words, the DP prior induces a prior on clusters defined by the $k \leq n$ unique values of the random sample.

Many applications of nonparametric Bayesian models focus on this implied clustering. The inference on the unknown probability measure G is often of less interest than the implied clustering. In this chapter we focus on this aspect of nonparametric Bayes models and introduce some alternative models for random partitions. We start by introducing useful notation. Let $S = \{1, \ldots, n\}$ denote the experimental units that are being clustered. Let $\rho_n = \{S_1, \ldots, S_k\}$ denote a partition with non-overlapping subsets S_j that cover S. When the sample size n is obvious from the context we drop the subindex $_n$. Sometimes it is technically more convenient to use alternative equivalent notation with cluster membership indicators $s_i = j$ if $i \in S_j$. Let $y_j^\star = (y_i, \; i \in S_j)$ denote outcomes arranged by clusters. For some models we will make use of available covariates x_i and use $x_j^\star$ to denote covariates arranged by clusters. In this chapter we discuss probability models $p(\rho_n)$ for random partitions and extensions of such models that include a regression on covariates by defining $p(\rho_n \mid \boldsymbol{x})$.

8.2. Random Partition Models

Product Partition Model

Hartigan (1990), Barry and Hartigan (1993), and Crowley (1997) propose and develop the product partition model (PPM) for random partitions. In contrast to the prior on clustering that is implied by the DP prior, the PPM explicitely defines a probability distribution $p(\rho_n)$ over alternative partitions. The PPM uses a

non-negative function $c(S_j)$, known as the cohesion function to define a product partition probability

$$p(\rho_n) = K \prod_{j=1}^{k} c(S_j). \tag{8.1}$$

Conditional on a given partition, the PPM assumes independent sampling across clusters,

$$p(\boldsymbol{y} \mid \rho) = \prod_j p(y_j^\star \mid \mu_j^\star), \tag{8.2}$$

where $\mu_j^\star$ are cluster specific parameters. Applications of the PPM often use exchangeability of y_i across $i \in S_j$ by assuming that y_i, $i \in S_j$ are independent given $\mu_j^\star$. One of the attractions of the PPM is the conjugate nature. The posterior $p(\rho_n \mid \boldsymbol{y})$ is again a product partition model, with updated cohesion functions $c(S_j)p(y_j^\star)$, where $p(y_j^\star)$ is the marginal sampling model for y_i, $i \in S_j$ under partition ρ_n.

The Pólya urn implied by the DP prior, $\mathsf{DP}(M, G_0)$, is a special case of a PPM, with cohesion function $c(S_j) = M(n_j - 1)!$. Another example of a PPM are Gibbs type priors. Recall from §7.3 the format of the EPPF for Gibbs type priors. Let $n_j = |S_j|$ denote the cardinality of the $j-$th cluster. The EPPF of a Gibbs type prior takes the form (8.1) with cohesion function $c(S_j) = (1-\sigma)_{n_j-1}$. Here $a_k = \Gamma(a+k)/\Gamma(a)$ denotes a rising factorial.

Some applications use constrained partition models. For example, when observations are ordered in time it might be desireable to restrict clusters to contiguous sequences of objects (Barry and Hartigan, 1993; Monteiro *et al.*, 2010; Yao, 1984).

Species Sampling Model

We already discussed the species sampling model (SSM) as a large class of prior models for random distributions (Ishwaran and James, 2003; Pitman, 1996) that includes many popular models as special cases. One of the characterizations of the SSM is through the EPPF, the implied probability model for the induced partition of $\{1, \ldots, n\}$. Recall that the EPPF is a symmetric function $f(n_1, \ldots, n_k)$, symmetric in its arguments,

$$p(\rho_n) = f(\boldsymbol{n}).$$

Again the partition that is implied by i.i.d. sampling from a random probability measure with DP prior is a special case with $f(\boldsymbol{n}) \propto \prod_{j=1}^{n} M(n_j - 1)!$.

Model-Based Clustering

In data analysis, when formal probability models are used for clustering, perhaps the most commonly used approach is model-based clustering. Model-based clustering defines a prior $p(\rho_n)$ implicitly through a mixture model for the observed data. Let y_i, $i = 1, \ldots, n$, denote responses for n experimental units. A mixture model $p(y_i \mid k, (\theta_j), (\pi_j)) = \sum_{j=1}^{k} \pi_j f_j(y_i \mid \theta_j)$ can be equivalently written as a hierarchical model with latent indicators $s_i \in \{1, \ldots, k\}$,

$$p(y_i \mid k, (\theta_j), s_i = j) = f_j(y_i \mid \theta_j), \qquad \mathsf{Pr}(s_i = j) = \pi_j. \tag{8.3}$$

When the latent indicators s_i are interpreted as cluster membership indicators, then (8.3) implicitely defines $p(\rho_n)$. Inference for such models is discussed, among others, in Fraley and Raftery (2002), Richardson and Green (1997) and Green and Richardson (2001).

Pólya Urn

Recall the predictive rule for cluster allocation under i.i.d. sampling $x_i \mid G \sim G$ from a random probability measure G with a DP prior, $G \sim \mathsf{DP}(M, G_0)$. Let $s_i = j$ when the i-th observation is equal to the j-th unique value, i.e., when $x_i = x_j^\star$. The Pólya urn (Chinese restaurant process) specifies

$$p(s_{n+1} \mid s_1, \ldots, s_n) = \begin{cases} n_h & \text{with prob } 1/(M+n) \\ k_n + 1 & \text{with prob } M/(M+n). \end{cases} \tag{8.4}$$

The prior $p(\rho_n)$ implied by (7.1) is a special case of the PPM, a special case of the SSM, as well as a special limiting case of model-based clustering. We already mentioned the earlier two special cases. The Pólya urn arises as a limiting case of model-based clustering when $p(\pi_1, \ldots, \pi_k)$ is assumed as a symmetric Dirichlet distribution, $\mathsf{Dir}(\delta, \ldots, \delta)$ and one considers the limiting case $\delta \to 0$ and $k \to \infty$ subject to $k\delta \to M$ (Green and Richardson, 2001). The nature of the DP as a special case of many other models is one of the reasons for the undying popularity of the model.

8.3. Covariate-Dependent Clustering

Covariate-Dependent PPM

The previously discussed clustering models are useful for inference about clusters and subpopulations in observed data, but of little use for predictive inference. In Example 24 we are interested in predicting overall survival time y_{n+1} for a future patient $i = n + 1$ on the basis of data for $n = 763$ patients in a clinical trial. Let $\boldsymbol{y} = (y_1, \ldots, y_n)$. Clustering patients on the basis of the outcome would allow us to predict survival time for a future patient in the same population of patients who were eligible for this trial as

$$p(y_{n+1} \mid \boldsymbol{y}) = \int p(y_{i+1} \mid s_{n+1}, \rho_n, \boldsymbol{y}) dp(s_{n+1} \mid \rho_n) dp(\rho_n \mid \boldsymbol{y}), \tag{8.5}$$

where integration with respect to s_{n+1} is simply averaging over the $k_{n_1} + 1$ possible choices and integration with respect to ρ_n is averaging with respect to the posterior distribution on possible cluster arrangements of the first n patients. This is density estimation for the survival time of women in this population. We would report the same inference for any future patient, independently of the patient's baseline characteristics. This limits the use of (8.5) for prediction in this scenario.

More relevant would be inference of overall survival for a woman with particular baseline covariates x_i. A convenient and often used implementation is to consider an augmented outcome vector $\mathbf{z}_i = (y_i, x_i)$, implement clustering on the basis of $\mathbf{z}_i$ and report

$$p(y_{n+1} \mid x_{n+1}, \boldsymbol{y}, \boldsymbol{x}) \tag{8.6}$$

as the desired inference. The problem is that the covariate vector x_i often involves a mix of data formats, complicating the specification of a sampling model. Also, some of the covariates such as treatment assignment are not random at all, making it awkward to model a distribution for these variables.

Müller *et al.* (2011) propose to instead use a model $p(\rho_n \mid \boldsymbol{x})$, together with a sampling model $p(\boldsymbol{y} \mid \rho_n)$. Here $p(\rho_n \mid \boldsymbol{x})$ is a regression of the random partition ρ_n on the known covariates $\boldsymbol{x}$. The idea is to specify a probability model for random partitions that favors clusters that are homogeneous in the covariates x_i. Predicting the outcome for a future subject is then based on averaging over all clusters, with the weights determined by the respective probability of cluster membership $p(s_{n+1} \mid x_{n+1}, \rho_n, \boldsymbol{x})$. In words, the prediction weighs clusters of earlier patients with similar covariates higher than others.

Formally, let $x_j^\star = (x_i,\ i \in S_j)$ denote covariates of experimental units in the j-th cluster, and let $g(x^\star)$ denote a non-negative function that formalizes homogeneity of a cluster with covariates $x^\star$. For example, $g(x^\star)$ could be the determinant of the empirical precision matrix of the x_i. For a categorical covariate x the similarity function could be related to the number of distinct values in a cluster. For example, a cluster with all women with the same prior treatment history is clinically more meaningful than a cluster that includes a large diversity of prior treatment histories. A simple application of the PPM provides the desired random partition model

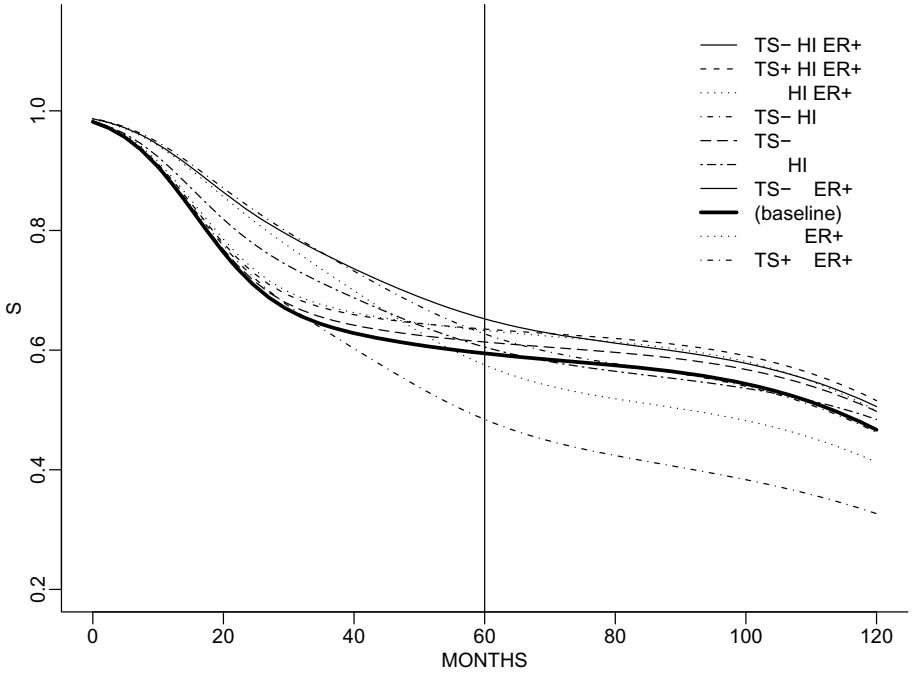
$$p(\rho_n \mid \boldsymbol{x}) \propto \prod_{j=1}^{k} g(x_j^\star) c(S_j). \tag{8.7}$$

The choice of the similarity function depends on the application. As a generic choice, Müller *et al.* (2011) define $g(x^\star)$ on the basis of an auxiliary probability model $q(\cdot)$:

$$g(x_j^\star) \equiv \int \prod_{i \in S_j} q(x_i \mid \xi_j)\, q(\xi_j) d\xi_j.$$

Chosing $q(x_i \mid \xi_j)$ and $q(\xi_j)$ as a conjugate pair simplifies analytic evaluation of $g(x^\star)$.

Example 24 (Survival Time Model with Clustering) *Müller* et al. *(2011) consider data from a high-dose chemotherapy treatment of $n = 763$ women with breast cancer. The response of interest is overall survival y_i. Let $\boldsymbol{y} = (y_1, \ldots, y_n)$ denote the observed data. There are six patient-specific covariates $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{i6})$, including a binary indicator for high dose chemotherapy, age in years, number of positive lymph nodes, tumor size, indicator for estrogen or progesterone receptor positive tumor, and an indicator for the woman's menopausal status. indicator. Let $x_{j,\ell}^\star = (x_{i\ell}, i \in S_j)$ denote the values for the ℓ-th covariate in cluster j. Müller* et al. *(2011) define a similarity function $g(x_j^\star) = \prod_{\ell=1}^{6} g_\ell(x_{j,\ell}^\star)$ using default similarity function g_ℓ for each data format, including a beta-binomial for the binary covariates, a normal-normal for continuous covariates and a poisson-gamma model for the count covariate. Figure 8.1 summarizes prediction for a future patient as a function of baseline covariates.*



$S(t \mid x)$

FIG 8.1. *Posterior predictive summarized by survival functions* $S(t \mid x) \equiv p(y_{n+1} \geq t \mid x_{n+1} = x, data)$. *In the legend TS− and TS+ indicates tumor size equal to the first and third empirical quantile, ER+ indicates ER-positive tumor, and HI indicates high dose.*

.

Alternative Constructions

Model-based clustering (8.3) allows an easy extension to include covariates in the implied prior on random partitions. Consider

$$y_i \mid x_i \sim \sum_{j=1}^{k} \pi_j(x_i;\, \alpha_j)\, f_j(\cdot \mid \theta_j).$$

The generalization is the explicit inclusion of covariates in the weights of the component models. As before, rewriting the mixture as a hierarchical model with latent indicators defines the desired covariate-dependent random partition model:

$$p(y_i \mid k, (\theta_j), s_i = j) = f_j(y_i \mid \theta_j), \qquad p(s_i = j \mid x_i) = \pi_j(x_i;\, \alpha_j). \tag{8.8}$$

The regression $\pi_j(x_i;\, \alpha_j)$ could be, for example, a logistic regression. This is essentially the hierarchical mixture of experts model (Bishop and Svensén, 2003; Jordan and Jacobs, 1994). The model is very useful for flexible non-parametric regression, especially when the focus is prediction. The limiations of the approach are the use of a fixed number of component models k, which becomes an upper bound for the number of clusters, and the restriction of covariate dependence to the particular parametric form chosen for $\pi_j(x_i;\, \alpha_j)$.

Dahl (2008) defines another interesting probability model for covariate-based clustering. Let $s_{-i} = (s_\ell,\, \ell \neq i)$ denote the partition of all but the i-th object. He defines the desired $p(\rho_n \mid \boldsymbol{x})$ by modifying the complete conditional probabilities $p(s_i \mid s_{-i}, \boldsymbol{x})$. The modification of complete conditionals needs care to assure the existence of a well defined probability model.

Clustering with DDP and Related Models

We earlier introduced the dependent DP model as prior for families of random probability measures $\mathcal{G} = \{G_x,\ x \in X\}$. In particular, recall the definition of the DDP, here with common locations and variable weights

$$G_x = \sum_{h=1}^{\infty} \pi_{hx} \delta_{m_h}. \tag{8.9}$$

Note that here the locations m_h of the point masses are common across x, and only the weights vary.

Similar to how the DP induces implicitely a prior for random partitions $p(\rho_n)$, the DDP can be used to implicitely define a prior $p(\rho_n \mid \boldsymbol{x})$ for random partitions with a regression on covariates. In particular, assume $y_i \mid x_i = x, \mathcal{G} \sim G_x$, independently for experimental units $i = 1, \ldots, n$, with known covariates x_i. In addition, let $y_j^\star$, $j = 1, \ldots, k$ denote the $k \leq n$ unique values among the y_i and define clusters $S_j = \{i :\ y_i = y_j^\star\}$. The construction is almost identical to before, when we used the DP to define a random partition. However, the probabilities for cluster membership now depend on x, as desired. To our knowledge, this construction itself, i.e., the clustering implied by sampling from a DDP family of random probability measures, has not be used in the literature before. However, several proposed approaches can be interpreted as approximations to this natural construction. Note that other variants of dependent stick-breaking prior such as the probit stick-breaking process (see §5.7.1) and the kernel stick-breaking process (see §5.7.2) can be used to generate prior on dependence partitions in a similar fashion.

Appendix

A.1. Implementing DP Mixtures in R

We introduce some R macros to implement inference in a DP mixture in (3.9). Using Gaussian kernels the DP mixture model becomes

$$y_i \mid G \sim F(y_i) = \int N(y_i;\ \theta_i, \sigma)\ dp(G). \tag{A.10}$$

We will use $f(x)$ to denote the p.d.f. The moel can equivalently be written as a hierarchical model

$$y_i \mid \theta_i \sim N(\theta_i, \sigma), \qquad \theta_i \mid G \sim G. \tag{A.11}$$

where $G \sim \mathsf{DP}(M, G_0)$. For the centering measure, we use $G_0 = N(m_0, B_0)$, and we take $m_0 = 0$ and $B_0 = 4$, while the precision parameter is set to $M = 1$. We complete the model with a gamma prior for the kernel width, $1/\sigma^2 \sim \mathsf{Ga}(a, b)$; in the example below we take $a = b = 1$. We implement posterior MCMC simulation using the methods described in §3.3. The complete R code is available at

`http://www.math.utexas.edu/users/pmueller/prog/BNPnotes/`

We briefly explain the main steps in the macros. We use the data from the Old Faithful geyser data in R and fix the hyperparameters for the model. The hyperparameters and the data are saved as global variables:

```
## DATA: Old Faithful geyser data
y <- round(faithful$eruptions, digits=2)
n <- length(y)
## hyperparameters
a  <- 1;   b <- 1    # 1/sig ~ Ga(a,b)
m0 <- 0;  B0 <- 4    # G0 = N(m0,B0)
M  <- 1
```

Collapsed Gibbs Sampler – Conjugate Models

We first implement the Gibbs sampler from §3.3.1. The algorithm consists of the following steps:

0. Initialization: We initialize s_i using (deterministic) hierarchical clustering.
Initialize a plot by plotting a kernel density estimate of $f(y)$.
1. Update θ_i: sample from $p(\theta_i \mid \boldsymbol{\theta}_{-i}, \sigma, \mathbf{y})$, $i = 1, \ldots, n$. See (3.13).
2. Update $\theta_j^\star$: sample from $p(\theta_j^\star \mid s_i, \sigma, \mathbf{y})$, $j = 1, \ldots, k$, as in (3.14).
3. Update σ^2: sample from $p(\sigma^2 \mid \mathbf{s}, \boldsymbol{\theta}^\star, \mathbf{y})$.
4. Generate f: generate $f \sim p(f \mid \boldsymbol{\theta}, \phi, \sigma)$; add f to the plot.

In Step 4 we sample f conditional on the currently imputed values of $\boldsymbol{\theta}, \phi, \sigma$. For plotting it suffices to evaluate f on a grid that is fine enough to get a smooth plot. To sample f (on the grid) we use a minor approximation. Essentially we (i) sample $G \sim p(G \mid \boldsymbol{\theta})$, and (ii) evaluate $F = \int \mathsf{N}(\theta, \sigma)\, dG(\theta)$. The approximation

is applied in step (i); recall that $p(G \mid \theta_1, \ldots, \theta_n) = \mathsf{DP}(M+n, G_1)$ with $G_1 \propto MG_0 + \sum_{i=1}^{n} \delta_{\theta_i}$. For large n, the total mass (precision) $M_1 = M+n$ is large. In the limiting case $(M+n) \to \infty$ little uncertainty is left, and $G \approx G_1$. The code below uses the limiting case as an approximation, hence, we (approximately) generate $F \sim p(F \mid \boldsymbol{\theta}, \sigma)$ as

$$F \propto \sum n_j p(y \mid \theta_j^\star) + M \int p(y \mid \theta) dp G^\star(\theta).$$

This is a poor man's version of the algorithm proposed by Gelfand and Kottas (2002).

Steps 0 through 4 are implemented in the following R macros. We first define macros for each of the transition probabilities and for sampling $f \sim p(f \mid \boldsymbol{\theta})$. A final macro `gibbs()` implements the loop across iterations and calls each of the transition probabilities in turns.

Step 0. As with any MCMC algorithm, a good initialization can significantly speed up convergence. Good initializations might use (for example) exploratory data analysis estimates, empirical Bayes estimates or maximum likelihood estimates. In this case we use an initialization with a partition that is created by cutting a (deterministic) hierarchical clustering tree, for say $k = 10$ clusters. The macro `init.DPk()` implements this initialization. Recall that the data `y` is available as a global variable.

```
init.DPk <- function()
  { ## inital EDA estimate of th[1..n]

    ## cluster data, and cut at height H=10, to get 10 clusters
    hc <- hclust(dist(y)^2, "cen")
    s  <- cutree(hc, k = 10)          # cluster membership indicators
    ths <- sapply(split(y,s),mean)    # cluster specific means
    th <-  ths[s]                     # return th_i = ths[ s_i ]
    return(th)
}
```

Step 1. The first transition probability generates from $p(\theta_i \mid \boldsymbol{\theta}_{-i}, \sigma, \mathbf{y})$, $i = 1, \ldots, n$.

```
sample.th <- function(th,sig)
  { ## sample
    ##    th[i]  ~ p(th_i | th[-i],sig,y)
    ## returns updated th vector

    for(i in 1:n){
      ## unique values and counts
      nj <- table(th[-i])             # counts
      ths <- as.numeric(names(nj))    # unique values
      k <- length(nj)
      ## likelihood
      fj <- dnorm(y[i], m=ths, s=sig)           # p(y_i | th*_j, sig), j=1..k
      f0 <- dnorm(y[i], m=0,   s=sqrt(4+sig^2)) # q0
      pj <- c(fj*nj,f0*M)                       # p(s[i]=j | ...), j=1..k, k+1
      s <- sample(1:(k+1), 1, prob=pj)          # sample s[i]
      if (s==k+1){ ## generate new th[i] value
        v.new <- 1.0/(1/B0 + 1/sig^2)
        m.new <- v.new*(1/B0*m0 + 1/sig^2*y[i])
        thi.new   <- rnorm(1,m=m.new,sd=sqrt(v.new))
        ths <- c(ths,thi.new)
      }
```

```
    th[i] <- ths[s]                              # record new th[i]
  }
  return(th)
}
```

Step 2. Next we update $\theta_j^\star$.

```
sample.ths <- function(th,sig)
  { ## sample ths[j] ~ p(ths[j] | ....)
    ##               = N(ths[j]; m0,B0) * N(ybar[j]; ths[j], sig2/n[j])
    ##               = N(ths[j]; mj,vj)

    ## unique values and counts
    nj <- table(th)                 # counts
    ths <- sort(unique(th))         # unique values
           ##use sort(.) to match table counts
    k <- length(nj)
    for(j in 1:k){
      ## find Sj={i: s[i]=j} and compute sample average over Sj
      idx <- which(th==ths[j])
      ybarj <- mean(y[idx])
      ## posterior moments for p(ths[j] | ...)
      vj <- 1.0/(1/B0 + nj[j]/sig^2)
      mj <- vj*(1/B0*m0 + nj[j]/sig^2*ybarj)
      thsj <- rnorm(1,m=mj,sd=sqrt(vj))
      ## record the new ths[j] by replacing all th[i], i in Sj.
      th[idx] <- thsj
    }
    return(th)
  }
```

Step 3. The last transition probability in each iteration updates σ^2 using $p(1/\sigma^2 \mid \boldsymbol{\theta}, \boldsymbol{y}) \propto \mathsf{Ga}(a_1, b_1)$ with $a_1 = a + 0.5n$ and $b_1 = b + 0.5s^2$, where $s^2 = \sum_i (y_i - \theta_i)^2$ is the residual sum of squares.

```
sample.sig <- function(th)
  { ## sample
    ##   sig ~ p(sig | ...)
    ## returns: sig

    s2 <- sum( (y-th)^2 )    # sum of squared residuals
    a1 <- a+0.5*n
    b1 <- b+0.5*s2
    s2.inv <- rgamma(1,shape=a1,rate=b1)
    return(1/sqrt(s2.inv))
  }
```

Step 4. The macro `fbar()` implements the draw from $f \sim p(f \mid \theta^\star, \sigma)$.

```
fbar <- function(x,th,sig)
  { ## conditional draw F ~ p(F | th,sig,y) (approx -- will talk about this..)
    ##

    nj <- table(th)              # counts
    ths <- as.numeric(names(nj)) # unique values
    k <- length(nj)
    fx <- M/(n+M)*dnorm(xgrid,m=m0,sd=sqrt(B0+sig))
    for(j in 1:k)
      fx <- fx + nj[j]/(n+M)*dnorm(xgrid,m=ths[j],sd=sig)
    return(fx)
  }
```

Gibbs loop: The macro `gibbs()` implements `n.iter` steps of the MCMC, calling

in turn macros for each of the transition probabilities. Before the actual for loop, the macro initializes several lists that will accumulate the states in each iteration, and starts a plot of a simple kernel density estimate, to which the draws $f \sim p(f \mid \boldsymbol{y})$ will be added.

```
gibbs <- function(n.iter=100)
  {
    th <- init.DPk()              ## initialize th[1..n]
    sig <- sqrt( mean((y-th)^2))  ## and sig
    ## set up data structures to record imputed posterior draws..1
    xgrid <- seq(from=0,to=6,length=50)
    fgrid <- NULL     ## we will record imputed draws of f
    njlist <- NULL    ## record sizes of 8 largest clusters
    klist <- NULL
    ## start with a plot of a kernel density estimate of the data
    plot(density(y),xlab="X",ylab="Y",bty="l",type="l",
         xlim=c(0,6),ylim=c(0,0.7), main="")
    ## now the Gibbs sampler
    for(iter in 1:n.iter){
      th  <- sample.th(th,sig)    ## 1. [th_i  | ...]
      sig <- sample.sig(th)       ## 2. [sig   | .. ]
      th  <- sample.ths(th,sig)   ## 3. [ths_j | ...]
      ## update running summaries ##################
      f   <- fbar(xgrid,th,sig)
      lines(xgrid,f,col=iter,lty=3)
      fgrid <- rbind(fgrid,f)
      nj <- table(th)                 # counts
      njlist <- rbind(njlist,sort(nj,decr=T)[1:8])
      klist <- c(klist,length(nj))
    }
    ## report summaries ############################
    fbar <- apply(fgrid,2,mean)
    lines(xgrid,fbar,lwd=3,col=2)
    njbar <- apply(njlist,2,mean,na.rm=T)
    cat("Average cluster sizes:\n",format(njbar),"\n")
    pk <- table(klist)/length(klist)
    cat("Posterior probs p(k): (row1 = k, row2 = p(k) \n ")
    print(pk/sum(pk))
    return(list(fgrid=fgrid,klist=klist,njlist=njlist))
  }
```

The defined macros could be called as follows:

```
mcmc <- gibbs()
names(mcmc)

## report summaries
njbar <- apply(mcmc$njlist,2,mean,na.rm=T)
cat("Average cluster sizes:\n",format(njbar,digits=1),"\n")
pk <- table(mcmc$klist)/length(mcmc$klist)
cat("Posterior probs p(k): (row1 = k, row2 = p(k) \n ")
print(pk/sum(pk))
```

A blocked Gibbs sampler for the finite DP

Alternatively we implement the Gibbs sampler for a DP mixture of normals (A.10) as before, but now with a finite DP prior, $G \sim \mathsf{DP}_H(M, G_0)$. We implement Gibbs sampling posterior simulation using (3.23) through (3.25). We need an additional hyperparameter to fix H in the finite DP. The initialization proceeds in the same

way as before using a hierarchical clustering. However, now we use the solution from the hierarchical clustering tree to initialize the $\widetilde{\theta}_h$ and w_h parameters.

```
H <- 10    # additional hyperpar

init.DPk <- function()
  { ## inital EDA estimate of G = sum_{h=1..10} w_h delta(m_h)
    ## returns:
    ##   list(mh,wh)
    ## use (mh,wh) to initialize the blocked Gibbs

    ## cluster data, and cut at height H=10, to get 10 clusters
    hc <- hclust(dist(y)^2, "cen")
    r  <- cutree(hc, k = 10)
    ## record cluster specific means, order them
    mh1 <- sapply(split(y,r),mean)    # cluster specific means == m_h
    wh1 <- table(r)/n
    idx <- order(wh1,decreasing=T)    # re-arrange in deceasing order
    mh <- mh1[idx]
    wh <- wh1[idx]
    return(list(mh=mh,wh=wh))
}
```

The next three R macros implement sampling from distributions (3.23) through (3.25). A fourth transition probability to update σ^2 remains unchanged from before, and we continue to use the earlier defined macro `sample.sig()`. A final macro `gibbs.H()` implements the loop over iterations.

```
sample.r <- function(wh,mh,sig)
  { ## samle allocation indicators

    r <- rep(0,n)
    for(i in 1:n){
      ph <-   dnorm(y[i],m=mh,sd=sig)*wh # likelihood   * prior
                                         ## p(yi | ri=h) * w_h
      r[i] <- sample(1:H,1,prob=ph)
    }
    return(r)
  }


sample.mh <- function(wh,r)
  { ## sample mh ~ p(mh | ...)
    ##

    mh <- rep(0,H)     # initialize
    for(h in 1:H){
      if(any(r==h)){      # some data assigned to h-th pointmass
        Sh <- which(r==h) # Sh = {i: r[i]=h
        nh <- length(Sh)
        ybarh <- mean(y[Sh])
        varh   <- 1.0/(1/B0 + nh/sig^2)
        meanh  <- varh*(1/B0*m0 + nh/sig^2*ybarh)
      } else {            # no data assinged to h-th pointmass
        varh  <- B0       # sample from base measure
        meanh <- m0
      }
      mh[h] <- rnorm(1,m=meanh,sd=sqrt(varh))
    }
    return(mh)
  }
```

```
sample.vh <- function(r)
  {## sample vh ~ p(vh | ...)
   ## returns: wh

    vh <- rep(0,H)  # initialize
    wh <- rep(0,H)
    V <-  1         # record prod_{g<h} (1-vh_h)
    for(h in 1:(H-1)){
      Ah <- which(r==h)
      Bh <- which(r>h)
      vh[h] <-  rbeta(1, 1+length(Ah), M+length(Bh))
      wh[h] <- vh[h]*V
      V <- V*(1-vh[h])
    }
    vh[H] <- 1.0
    wh[H] <- V
    return(wh)
  }
```

Imputing the density f is much simpler now. We can literally evaluate the mixture of normals.

```
fbar.H <- function(xgrid,wh,mh,sig)
  { \#\# return a draw F ~ p(F | ...) (approx)

    fx <- rep(0,length(xgrid))
    for(h in 1:H)
      fx <- fx + wh[h]*dnorm(xgrid,m=mh[h],sd=sig)
    return(fx)
  }
```

An outer loop over iterations implements the Gibbs sampler. The macro looks very similar to before. The Gibbs sampler is executed by calling `fgrid <- gibbs.H()`.

```
gibbs.H <- function(n.iter=100)
  {

    DPk <- init.DPk()
    sig <- 0.11
    wh <- DPk$wh
    mh <- DPk$mh

    ## data structures to save imputed F ~ p(F | ...)
    xgrid <- seq(from=0,to=6,length=50)
    fgrid <- NULL
    plot(density(y),xlab="X",ylab="Y",bty="l",type="l",
         xlim=c(0,6),ylim=c(0,0.7), main="")
    ## Gibbs
    for(iter in 1:n.iter){
      r <-  sample.r(wh,mh,sig)   # 1. r_i ~ p(r_i | ...), i=1..n
      mh <- sample.mh(wh,r)       # 2. m_h ~ p(m_h | ...), h=1..H
      vh <- sample.vh(r)          # 3. v_h ~ p(v_h | ...), h=1..H
      th <- mh[r]                 # record implied th[i] = mh[r[i]]
      sig <- sample.sig(th)       # 4. sig ~ p(sig | ...)

      ## record draw F ~ p(F | th,sig,y) (approx)
      f   <- fbar.H(xgrid,wh,mh,sig)
      lines(xgrid,f,col=iter,lty=3)
      fgrid <- rbind(fgrid,f)
    }
    ## add overall average (= posterior mean) to the plot
    fbar <- apply(fgrid,2,mean)
```

```
    lines(xgrid,fbar,lwd=3,col=2)
    return(fgrid)
  }
```

DPpackage

The detailed R code is useful to understand the algorithm. However, for actual data analysis the use of implementations in available public domain R packages is preferable. For example, `DPpackage` implements MCMC for DP mixtures. The following R fragment shows the use of `DPpackage` with the same Old Faithful geyser data. For more detail see Jara *et al.* (2011).

```
require ("DPpackage")

##### set up parameters for call to DPdensity(.) below:
state <- NULL                                          ## Initial state
nburn<-10; nsave<-1000; nskip<-10; ndisplay<-100       ## MCMC parameters
mcmc <- list(nburn=nburn,nsave=nsave,nskip=nskip,ndisplay=ndisplay)

## Prior 1: fixed alpha, m1, and Psi1
prior1<-list(alpha=1,m1=rep(0,1),psiinv1=diag(0.5,1),nu1=4,tau1=1,tau2=100)

## Prior 4: everything is random
prior4<-list(a0=2,b0=1,m2=rep(0,1),s2=diag(100000,1),
             psiinv2=solve(diag(0.5,1)),
             nu1=4,nu2=4,tau1=1,tau2=100)

## Fit the models and plot the density estimates
fit1 <-DPdensity(y=y,prior=prior1,mcmc=mcmc,state=state,status=TRUE)
fit4 <-DPdensity(y=y,prior=prior4,mcmc=mcmc,state=state,status=TRUE)
plot(fit1,ask=FALSE)
plot(fit4,ask=FALSE)
```

A.2. Implementing PTs in R

One of the attractions of using PT priors is the straightforward implementation of posterior inference. We show some R macros that implement inference in a density estimation problem

$$x_i \mid G \sim G, \qquad\qquad G \sim \mathsf{PT}(\mathcal{A}, G_0).$$

The nested sequence of partitions is defined by the quantiles of a centering measure G_0 (recall the discussion in §4.1). We use a Gaussian centering distribution $G_0(x) = N(\mu, \sigma)$ with fixed hyperparameters $\mu = 0$ and $\sigma = 1$. We implement inference for a Pólya tree up to $M = 5$ levels of nested partitions.

We first generate simulated data from a log normal distribution with some outliers and fixed hyperparameters. After generating the data we produce a histogram.

```
M  <- 5                ## number of levels in PT prior
mu <- 0; sig <- 1      ## centering measure G0 = N(mu,sig)
alpha <- 1
y <- make.dta()

make.dta <- function(plt=T)
  { ## prepares data
    ##
    w <- rnorm(134,m= -1, s=0.5)
    x <-  exp(w)
    x <-  c(x,3.866, 189.3)
    y <- log(x)
    n <-  length(y)
    if (plt)
      hist(log(x),nclass=15)
    return(y)
  }
```

R macros for posterior simulation and means.

We record the boundaries of the partitioning subsets B_ϵ, defined by dyadic quantiles of G_0. This is done in the R macro `prior.pars()`.

```
## record nested partition sequence B[eps]
prior.pars <- function()
  { ## sets up (right) boundaries of partitioning subsets for
    ## PT centered at N(mu,sig)
    B <- as.list(1:M)
    for(m in 1:M){
      q <- 1/(2**m)*(1:(2**m-1))     ## part sets (right boundaries of sets)
      B[[m]] <- c(qnorm(q,m=mu,sd=sig),99)
    }
    return(B)
  }
```

Another macro updates the prior parameters We use prior parameters $\alpha_\epsilon = \alpha m^2$ and compute posterior parameters as in (4.1).

```
## update PT parameters
post.pars <- function(B,alpha,y)
  { ## posterior pars for a PT prior with
    ## PT(B, A) with (right) boundaries of the partitioning sequence B
    ##          defined in B[[1]]..B[[M]] for levels 1..M
    ##          alpha[ep1...epsm] = alpha*m^2
```

```
  M <- length(B)
  neps <- list(1:M)
  a <- list(1:M)
  for(m in 1:M){
    neps[[m]] <- 0*(1:2**m)              ## initialize counts
    a[[m]]    <- alpha*m**2*rep(1,2**m)  ## prior beta parameters
    for(i in 1:n){
      j <- which(B[[m]]>y[i])[1]
      neps[[m]][j] <-  neps[[m]][j]+1    ## update the counts
    }
    a[[m]] <- a[[m]]+neps[[m]]           ## updated beta pars
  }
  return(a)
}
```

Now all parameters are set to proceed with posterior simulation. We define one macro, `post.sim()`, to carry out both, posterior simulation of $G \sim p(G \mid \overline{x})$ and the evaluation of $\overline{G} = E(G \mid \overline{x})$. An argument `sim` $\in \{T, F\}$ selects posterior simulation (T) or expectation (F). In the latter case the simulation from the beta random variables for the conditional splitting probabilities Y_ϵ are replaced by their expectations. Since all splitting probabilities are independent, by definition of the PT, the expectation of the product is the product of the expectations.

```
## posterior draw and posterior mean prob's
post.sim <- function(B,a,sim=T)
  { ## posterior simulation (sim=T) or mean (sim=F)
    M <- length(B)

    Y <- as.list(1:M)    ##  random splitting probs Y[eps]     (sim=T)
                         ##  mean splitting prob E(Y[eps] | y) (sim=F)
    P <- as.list(1:M)    # (random) prob's G(B[eps]) of partitioning subsets
                         ## when (sim=F):
                         ## Y = E(Y[eps] | y) and P = E{ G(B[eps]) | y}
    ## loop over all levels, m=0..M-1
    ## generate random splitting prob's starting with level 1,
    ## i.e. splitting sample space from level m=0
    ## split at level m=M-1 creates lowest level sets at m=M
     for(m in 0:(M-1)){
      for(j in 1:2**m){    # note, for m=0, only j=1 for the sample space
        j0 <- (j-1)*2 + 1  # index of left descendant set B[eps0]
        j1 <- (j-1)*2 + 2  # ... right set B[eps1]
        a0 <- a[[m+1]][j0]
        a1 <- a[[m+1]][j1]
        Y0 <- ifelse(sim, rbeta(1,a0,a1), a0/(a0+a1))
              ## (sim=T): generate random Beta splitting prob
              ## (sim=F): record mean splitting prob E(Y[eps] | dta)
        Y[[m+1]][j0] <- Y0
        Y[[m+1]][j1] <- 1-Y0
        if (m>0){
          P[[m+1]][j0] <- Y[[m+1]][j0] * P[[m]][j]
          P[[m+1]][j1] <- Y[[m+1]][j1] * P[[m]][j]
        } else {
          P[[m+1]][j0] <- Y[[m+1]][j0]
          P[[m+1]][j1] <- Y[[m+1]][j1]
        }
      }
    }
    return(list(Y=Y,P=P))
  }
```

The function `plt.G()` plots a simulated distribution G or a posterior mean $\overline{G}$ when

the distribution is given by probabilities over the partitioning subsets.

```
## plotting a distribution with probabilities P[eps]=p(B[eps])
plt.G <- function(P,B,col=1,lwd=3,lty=1)
    ## We build up the density P as a line (xx,yy)
    xx <- NULL
    yy <- NULL
    for(j in 1:2**M){ # loop over all part subsets B[eps] at level M
      ## we record endpoints: B[eps] = [x0,x1]
      if (j==1)       # left endpoint
        x0 <- -3
      else
        x0 <- B[[M]][j-1]
      if (j==2**M)    # right endpoint
        x1 <- 3
      else
        x1 <- B[[M]][j]
      ## and the density value = probability / length of interval
      y01 <- P[[M]][j]/(x1-x0)
      ## and now add the segment over B[eps] to (xx,yy)
      xx <- c(xx,x0,x1)
      yy <- c(yy,y01,y01)
    }
    ## finally, plot it..
    lines(xx,yy,lwd=lwd,col=col,lty=lty)
  }
```

The defined macros can be executed, for example, by the following sequence of calls.

```
    y  <- make.dta()            # prepare data set
    B <- prior.pars()           # partition boundaries
    a <- post.pars(B,alpha,y)   # posterior parameters

    ## plot the data
    hist(y,breaks=15,prob=T,ylim=c(0,0.9),main="")

    ## 20 random posterior draws
    for(i in 1:20){
      G <- post.sim(B,a)
      plt.G(G$P,B,col=1,lwd=1,lty=2)
    }
    ## posterior mean E(G | y)
    Gbar <- post.sim(B,a,sim=F)
    plt.G(Gbar$P,B,col=2,lwd=3)
```

DPpackage

The detailed R code is useful to understand the algorithm. For actual data analysis the use of implementations in available public domain R packages is much preferable. For example `DPpackage` implements MCMC for PT priors. In this case we use a PT mixture, as in §4.4. The following R fragment shows the use of `DPpackage` with the same data set. For more detail see Jara *et al.* (2011).

```
y  <- make.dta()                                ## prepare data set
state <- NULL                                   ## MCMC parameters
mcmc <- list(nburn=1000,nsave=1000,nskip=50,ndisplay=100,
             tune1=0.15,tune2=1.1,tune3=1.1)
prior<-list(alpha=1,M=6)                        ## Prior information
                                                ## Fitting the model
```

```
fit1 <- PTdensity(y=y,ngrid=1500,prior=prior,mcmc=mcmc,
                  state=state,status=TRUE)
hist(y,breaks=15,prob=T,ylim=c(0,0.8),main="") ## plot the data
lines(fit1$x1,fit1$dens,lty=1,lwd=3)           ## add PT fit
lines(density(y),col=2,type="l")               ## kernel density fit
```

Using the R package we can make use of many built-in additional features. For example, the package includes the evaluation of pseudo marginal likelihood values that can be used for model comparison.

Bibliography

Albert, J. H. and Chib, S. (2001). Sequential ordinal modeling with applications to survival data. *Biometrics*, **57**, 829–836.

Antoniak, C. E. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, **2**, 1152–1174.

Banerjee, S. and Gelfand, A. E. (2002). On smoothness properties of spatial processes. *Journal of Multivariate Analysis*, **84**, 85–100.

Banerjee, S., Carlin, B. P., and Gelfand, A. E. (2004). *Hierarchical modeling and analysis for spatial data*. Chapman & Hall/CRC, Boca Raton, FL.

Barnes, T. G., Jefferys, W. H., Berger, J. O., Müller, P., Orr, K., and Rodríguez, R. (2003). A Bayesian Analysis of the Cepheid Distance Scale. *The Astrophysical Journal*, **592**, 539.

Barrios, E., Nieto-Barajas, L. E., and Prünster, I. (2011). A study of normalized random measures mixture models. Technical report, ITAM, Mexico City.

Barry, D. and Hartigan, J. A. (1993). A Bayesian analysis for change point problems. *Journal of the American Statistical Association*, **88**, 309–319.

Berger, J. and Guglielmi, A. (2001). Bayesian testing of a parametric model versus nonparametric alternatives. *Journal of the American Statistical Association*, **96**, 174–184.

Berger, J., Jefferys, W. H., and Müller, P. (2012). Bayesian nonparametric shrinkage applied to cepheid star oscillations. *Statistical Science*, **27**, 3–10.

Bishop, C. M. and Svensén, M. (2003). Bayesian hierarchical mixtures of experts. In U. Kjaerulff and C. Meek, editors, *2003 Proceedings Nineteenth Conference on Uncertainty in Artificial Intelligence*, pages 57–64.

Blackwell, D. and MacQueen, J. B. (1973). Ferguson distributions via Pólya urn schemes. *The Annals of Statistics*, **1**, 353–355.

Blei, D. M. and Jordan, M. I. (2006). Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, **1**, 121–144.

Box, G. E. P. (1979). Some problems of statistics and everyday life. *Journal of the American Statistical Association*, **74**, 1–4.

Carter, C. K. and Kohn, R. (1994). On Gibbs sampling for state space models. *Biometrika*, **81**, 541–553.

Carvalho, C. M., Lopes, H., Polson, N. G., and Taddy, M. A. (2010). Particle learning for general mixtures. *Bayesian Analysis*, **5**, 709–740.

Chung, Y. and Dunson, D. B. (2009). Nonparametric Bayes conditional distribution modeling with variable selection. *Journal of the American Statistical Association*, **104**, 1646–1660.

Cifarelli, D. and Regazzini, E. (1978). Problemi statistici non parametrici in condizioni di scambialbilita parziale e impiego di medie associative. Technical report, Quaderni dell'Istituto di Matematica Finanziaria, Torino.

Cressie, N. A. C. (1993). *Statistics for Spatial Data*. Wiley-Interscience, New York.

Crowley, E. M. (1997). Product partition models for normal means. *Journal of the American Statistical Association*, **92**, 192–198.

Dahl, D. (2003). An improved merge-split sampler for conjugate Dirichlet process mixture models. Technical report, Department of Statistics, University of Wisconsin.

Dahl, D. B. (2008). Distance-based probability distribution for set partitions with applications to Bayesian nonparametrics. In *JSM Proceedings*. Section on Bayesian Statistical Science, American Statistical Association, Alexandria, VA.

De Iorio, M., Müller, P., Rosner, G. L., and MacEachern, S. N. (2004). An ANOVA model for dependent random measures. *Journal of the American Statistical Association*, **99**, 205–215.

De Iorio, M., Johnson, W. O., Müller, P., and Rosner, G. L. (2009). Bayesian nonparametric nonproportional hazards survival modeling. *Biometrics*, **65**, 762–771.

Diaconis, P. and Freedman, D. (1986a). On the consistency of Bayes estimates (with discussion). *The Annals of Statistics*, **14**, 1–67.

Diaconis, P. and Freedman, D. (1986b). On the inconsistency of Bayes estimates. *The Annals of Statistics*, **14**, 68–87.

Doksum, K. (1974). Tailfree and neutral random probabilities and their posterior distributions. *Annals of Probability*, **2**, 183–201.

Duan, J. A., Guindani, M., and Gelfand, A. E. (2007). Generalized spatial Dirichlet process models. *Biometrika*, **94**, 809–825.

Dunson, D. B. and Park, J.-H. (2007). Kernel stick-breaking processes. *Biometrika*, **95**, 307–323.

Escobar, M. (1988). *Estimating the means of several normal populations by estimating the distribution of the means*. Ph.D. thesis, Yale University.

Escobar, M. and Tomlinson, G. (1999). Analysis of densities. Technical report, University of Toronto.

Escobar, M. D. and West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, **90**, 577–588.

Ferguson, T. S. (1973). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, **1**, 209–230.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, **97**, 611–631.

Frühwirth-Schnatter, S. (1994). Data augmentation and dynamic linear models. *Journal of Time Series Analysis*, **15**, 183–202.

Gelfand, A. E. and Kottas, A. (2002). A Computational Approach for Full Nonparametric Bayesian Inference under Dirichlet Process Mixture Models. *Journal of Computational and Graphical Statistics*, **11**, 289–305.

Gelfand, A. E., Kottas, A., and MacEachern, S. N. (2005). Bayesian nonparametric spatial modeling with Dirichlet process mixing. *Journal of the American Statistical Association*, **100**, 1021–1035.

Ghosh, J. and Ramamoorthi, R. (2003). *Bayesian Nonparametrics*. Springer-Verlag, New York.

Ghoshal, S. (2010). The Dirichlet process, related priors and posterior asymptotics. In P. M. Nils Lid Hjort, Chris Holmes and S. G. Walker, editors, *Bayesian Nonparametrics*, pages 22–34. Cambridge University Press.

Gnedin, A. and Pitman, J. (2006). Exchangeable Gibbs partitions and Stirling triangles. *J. Math. Sci.*, **138**, 5674–5685.

Green, P. J. and Richardson, S. (2001). Modelling heterogeneity with and without the Dirichlet process. *Scandinavian Journal of Statistics*, **28**, 355–375.

Griffin, J., Kolossiatis, M., and Steel, M. F. J. (2010). Comparing distributions using dependent normalized random measure mixtures. Technical report, University of Kent, Canterbury.

Griffin, J. E. and Steel, M. F. J. (2006). Order-based dependent Dirichlet processes. *Journal of the American Statistical Association*, **101**, 179–194.

Hanson, T. and Johnson, W. (2002). Modeling regression error with a mixture of polya trees. *Journal of the American Statistical Association*, **97**, 1020–1033.

Hartigan, J. A. (1990). Partition models. *Communications in Statistics: Theory and Methods*, **19**, 2745–2756.

Hjort, N. L. (1990). Nonparametric Bayes estimators based on beta processes in models for life history data. *The Annals of Statistics*, **18**, 1259–1294.

Ishwaran, H. and James, L. F. (2001). Gibbs Sampling Methods for Stick-Breaking Priors. *Journal of the American Statistical Association*, **96**, 161–173.

Ishwaran, H. and James, L. F. (2002). Approximate Dirichlet process computing in finite normal mixtures: Smoothing and prior information. *Journal of Computational and Graphical Statistics*, **11**, 508–532.

Ishwaran, H. and James, L. F. (2003). Generalized weighted Chinese restaurant processes for species sampling mixture models. *Statistica Sinica*. In press.

Jain, S. and Neal, R. M. (2004). A split-merge Markov chain Monte Carlo procedure for the Dirichlet process mixture model. *Journal of Graphical and Computational Statistics*, **13**, 158–182.

Jain, S. and Neal, R. M. (2007). Splitting and merging components of a nonconjugate Dirichlet process mixture model. *Bayesian Analysis*, **2**, 445–472.

James, L. F., Lijoi, A., and Prünster, I. (2009). Posterior analysis for normalized random measures with independent increments. *Scandinavian Journal of Statistics*, **36**, 76–97.

Jara, A. and Hanson, T. (2011). A class of mixtures of dependent tail-free processes. *Biometrika*, **98**, 553–566.

Jara, A., Hanson, T. E., and Lesaffre, E. (2009). Robustifying generalized linear mixed models using a new class of mixtures of multivariate Polya trees. *Journal of Computational and Graphical Statistics*, **18**, 838–860.

Jara, A., Hanson, T., Quintana, F., Müller, P., and Rosner, G. (2011). DPpackage: Bayesian semi- and nonparametric modeling in R. *Journal of Statistical Software*, **40**, 1–30.

Jordan, M. and Jacobs, R. (1994). Hierarchical mixtures-of-experts and the em algorithm. *Neural Computation*, **6**, 181–214.

Karhunen, K. (1947). Über lineare methoden in der wahrscheinlichkeitsrechnung. *Ann. Acad. Sci. Fennicae. Ser. A. I. Math.-Phys.*, **37**, 1–79.

Klein, J. P. and Moeschberger, M. L. (2003). *Survival Analysis: Techniques for Censored and Truncated Data*. Springer-Verlag, New York.

Kolossiatis, M., Griffin, J., and Steel, M. F. J. (2012). On Bayesian nonparametric modelling of two correlated distributions. *Statistics and Computing*.

Lad, T., L., R., and Sadeghi, A. (1988). The benefit of adjuvant treatment for resected locally advanced non-small-cell lung cancer. *Journal of Clinical Oncology*, **6**, 9–17.

Lavine, M. (1992). Some aspects of Polya tree distributions for statistical modelling. *The Annals of Statistics*, **20**, 1222–1235.

Lavine, M. (1994). More aspects of Polya tree distributions for statistical modelling. *The Annals of Statistics*, **22**, 1161–1176.

Lee, J., Quintana, F., Mueller, P., and Trippa, L. (2008). Defining predictive probability functions for species sampling models. Technical report, UT Austin.

Li, Y., Lin, X., and Müeller, P. (2010). Bayesian inference in semiparametric mixed models for longitudinal data. *Biometrics*, **66**, 70–78.

Lijoi, A., Mena, R. H., and Prünster, I. (2005). Hierarchical mixture modeling with normalized inverse-gaussian priors. *Journal of the American Statistical Association*, **100**, 1278–1291.

Lijoi, A., Mena, R. H., and Prünster, I. (2007a). Bayesian nonparametric estimation of the probability of discovering new species. *Biometrika*, **94**, 769–786.

Lijoi, A., Mena, R. H., and Prünster, I. (2007b). Controlling the reinforcement in Bayesian non-parametric mixture models. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, **69**, 715–740.

Lo, A. Y. (1984). On a class of Bayesian nonparametric estimates: I. Density estimates. *The Annals of Statistics*, **12**, 351–357.

Lòeve, M. (1978). *Probability Theory*, volume II, 4th edition. Springer-Verlag, New York.

Loredo, T. J. (2011). Rotating stars and revolving planets: Bayesian exploration of the pulsating sky. In J. M. Bernardo, M. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West, editors, *Bayesian Statistics 9*, pages 361–392. Oxford University Press, Oxford.

Lubischew, A. (1962). On the use of discriminant functions in taxonomy. *Biometrics*, **18**, 455–477.

MacEachern, S. (1999). Dependent nonparametric processes. In *ASA Proceedings of the Section on Bayesian Statistical Science*. American Statistical Association, Alexandria, VA.

MacEachern, S. N. and Müller, P. (1998). Estimating mixture of Dirichlet process models. *Journal of Computational and Graphical Statistics*, **7**, 223–338.

MacEachern, S. N., Clyde, M., and Liu, J. S. (1999). Sequential importance sampling for nonparametric Bayes models: The next generation. *The Canadian Journal of Statistics*, **27**, 251–267.

Malec, D. and Müller, P. (2008). A Bayesian Semi-Parametric Model for Small Area Estimation. In S. Ghoshal and B. Clarke, editors, *Festschrift in Honor of J.K. Ghosh*, pages 223–236. IMS, Hayward, CA.

Mauldin, R. D., Sudderth, W. D., and Williams, S. C. (1992). Polya trees and random distributions. *The Annals of Statistics*, **20**, 1203–1221.

Mira, A. and Petrone, S. (1996). Bayesian hierarchical nonparametric inference for change-point problems. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, editors, *Bayesian Statistics 5*. Oxford University Press, Oxford.

Monteiro, J., Assuncao, R., and Loschi, R. (2010). Product partition models with correlated parameters. Technical report, UFMG.

Muliere, P. and Petrone, S. (1993). A Bayesian predictive approach to sequential search for an optimal dose: Parametric and nonparametric models. *Journal of the Italian Statistical Society*, **2**, 349–364.

Muliere, P. and Tardella, L. (1998). Approximating distributions of random functionals of Ferguson-Dirichlet priors. *Canadian Journal of Statistics*, **26**, 283–297.

Muliere, P. and Walker, S. (1997). A Bayesian non-parametric approach to survival analysis using Polya trees. *Scandinavian Journal of Statistics*, **24**, 331–340.

Müller, P. and Nieto-Barajas, L. (2008). Discussion of "The Nested Dirichlet Process" by A. Rodríguez, D. B. Dunson and A. E. Gelfand. *Journal of American Statistical Association*, **103**, 1146.

Müller, P. and Vidakovic, B., editors (1999). *Bayesian Inference in Wavelet-Based Models*. Springer-Verlag, New York.

Müller, P., Quintana, F., and Rosner, G. (2004). A method for combining inference across related nonparametric Bayesian models. *Journal of the Royal Statistical Society, Series B: Statistical Methodology*, **66**, 735–749.

Müller, P., Quintana, F., and Rosner, G. (2011). A product partition model with regression on covariates. *Journal of Computational and Graphical Statistics*, **20**, 260–278.

Neal, R. M. (2000). Markov Chain Sampling Methods for Dirichlet Process Mixture Models. *Journal of Computational and Graphical Statistics*, **9**, 249–265.

Nieto-Barajas, L., Müller, P., Ji, Y., and Mills, G. (2008). Time Series Dependent Dirichlet Process. Technical report, M.D. Anderson Cancer Center.

Nieto-Barajas, L. E. and Müller, P. (2012). Rubbery Polya tree. *Scandinavian Journal of Statistics*, **39**, 166–184.

Nieto-Barajas, L. E., Müller, P., Ji, Y., Lu, Y., and Mills, G. B. (2012). A time-series DDP for functional proteomics profiles. *Biometrics*.

Ongaro, A. and Cattaneo, C. (2004). Discrete random probability measures: a general framework for nonparametric Bayesian inference. *Statistics and Probability Letters*, **67**, 33–45.

Paddock, S., Ruggeri, F., Lavine, M., and West, M. (2003). Randomised Polya Tree Models for Nonparametric Bayesian Inference. *Statistica Sinica*, **13**, 443–460.

Petrone, S., Guindani, M., and Gelfand, A. E. (2009). Hybrid Dirichlet mixture models for functional data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **71**, 755–782.

Piantadosi, S. (1997). *Clinical trials: a methodologic perspective*. Wiley, New York.

Pitman, J. (1995). Exchangeable and partially exchangeable random partitions. *Probab. Th. Rel. Fields*, **102**, 145–158.

Pitman, J. (1996). Some Developments of the Blackwell–MacQueen Urn Scheme. In T. S. Ferguson, L. S. Shapeley, and J. B. MacQueen, editors, *Statistics, Probability and Game Theory. Papers in Honor of David Blackwell*, pages 245–268. IMS Lecture Notes–Monograph Series, Hayward, CA.

Pitman, J. and Yor, M. (1997). The Two-Parameter Poisson-Dirichlet Distribution Derived from a Stable Subordinator. *The Annals of Probability*, **25**, 855–900.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian processes for machine learning*. The MIT Press.

Regazzini, E., Lijoi, A., and Prünster, I. (2003). Distributional results for means of normalized random measures with independent increments. *The Annals of Statistics*, **31**, 560–585.

Richardson, S. and Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society. Series B (Methodological)*, **59**, 731–792.

Robert, C. P. and Casella, G. (2005). *Monte Carlo Statistical Methods*. Springer-Verlag, New York, second edition.

Roberts, G. and Papaspiliopoulos, O. (2008). Retrospective Markov chain Monte Carlo methods for Dirichlet process hierarchical models. *Biometrika*, **95**, 169–186.

Rodríguez, A. and Dunson, D. B. (2011). Nonparametric Bayesian models through probit stick-breaking processes. *Bayesian Analysis*, **6**, 145–178.

Rodríguez, A. and Ter Horst, E. (2008). Bayesian dynamic density estimation. *Bayesian Analysis*, **3**, 339–366.

Rodríguez, A. and ter Horst, E. (2010). Measuring expectations in options markets: An application to the S&P500 index. *Quantitative Finance*, page doi:10.1080/14697680903193397.

Rodríguez, A., Dunson, D. B., and Gelfand, A. E. (2008). The nested Dirichlet process, with discussion. *Journal of American Statistical Association*, **103**, 1131–1144.

Rodríguez, A., Dunson, D. B., and Taylor, J. (2009). Bayesian hierarchically weighted finite mixtures models for samples of distributions. *Biostatistics*, **10**, 155–171.

Rodríguez, A., Dunson, D. B., and Gelfand, A. E. (2010). Latent stick-breaking processes. *Journal of American Statistical Association*, **105**, 647–659.

Rossi, P., Allenby, G., and McCulloch, R. (2005). *Bayesian Statistics and Marketing*. Wiley, New York.

Sethurman, J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica*, **4**, 639–650.

Silverman, B. W. (1985). Some aspects of the spline smoothing approach to non-parametric curve fitting. *Journal of the Royal Statistical Society, Series B*, **47**, 1–52.

Somoza, J. L. (1980). Illustrative analysis: infant and child mortality in colombia. *World Fertility Survey Scientific Reports*, **10**.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). Sharing clusters among related groups: Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, **101**, 1566–1581.

Thibaux, R. and Jordan, M. (2007). Hierarchical beta processes and the indian buffet process. In *Proceedings of the 11th Conference on Artificial Intelligence and Statistics (AISTAT)*, Puerto Rico.

Trippa, L., Müller, P., and Johnson, W. (2011). The multivariate beta process and an extension of the Polya tree model. *Biometrika*, **98**, 17–34.

Vidakovic, B. (1998). Nonlinear wavelet shrinkage with Bayes rules and Bayes factors. *Journal of the American Statistical Association*, **93**, 173–179.

Vidakovic, B. (1999). *Statistical Modeling by Wavelets*. Wiley, New York.

Walker, S. G. (2007). Sampling the Dirichlet mixture model with slices. *Communnications in Statistics, Part B - Simulation and Computation*, **36**, 45–54.

West, M. and Harrison, J. (1997). *Bayesian Forecasting and Dynamic Models*. Springer-Verlag, New York, second edition.

Yao, Y.-C. (1984). Estimation of a noisy discrete-time step function: Bayes and empirical Bayes approaches. *The Annals of Statistics*, **12**, 1434–1447.

Zabell, S. L. (1982). W. E. Johnson's "sufficientness" postulate. *The Annals of Statistics*, **10**, 1091–1099.

Zhang, S., Müller, P., and Do, K.-A. (2010). A Bayesian semiparametric survival model with longitudinal markers. *Biometrics*, **66**, 435–443.