PhD Written Candidacy Examination Part I: Dirichlet Process Models and Applications

Rick Farouni

Tuesday 15th December, 2015

Examination Committee

Dr. Bob Cudeck Dr. Steve MacEachern Dr. Zhong-Lin Lu

Contents

List of Figures

Part I

Dr MacEachern: Dirichlet process models and applications

The Dirichlet process forms the core of many nonparametric Bayesian models. These models are used for a variety of purposes, ranging from density estimation, to latent class models, to the mixed model, and beyond. In this question, you are asked to describe the basics of these models, describe uses of the models, and to explore some of their properties.

1 Question Part A: Model Setup

A basic approach to the one-sample problem relies on a smoothed Dirichlet process. Formally write a model which involves (i) a distribution drawn from a Dirichlet process, (ii) a kernel to smooth the distribution so that it is continuous rather than discrete, and (iii) a distribution on the parameter that governs the Dirichlet process. Explain your model.

1.1 Model Specification: Multivariate Case

Consider data $\mathcal{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_t, , \ldots)$ such that each observation $\mathbf{y}_t = [y_{t1}, y_{t2}, \ldots, y_{tD}]$, is a D dimensional vector of measurements. A **Dirichlet Process Mixture of Normals (DPMN)** model for such data has the following hierarchical specification:

$$\begin{aligned} \alpha &\sim \text{Gamma}(a_\alpha, b_\alpha) \\ G \mid \alpha &\sim \text{DP}(\alpha G_0) \\ (\boldsymbol{\mu}_t, \Sigma_t) \mid G &\sim G \\ \mathbf{y}_t \mid \boldsymbol{\mu}_t, \Sigma_k &\sim \text{Normal}(\boldsymbol{\mu}_t, \Sigma_t) \end{aligned} \tag{1}$$

Things to note:

- G is a **random probability measure** whose prior distribution is the **Dirichlet Process** with a base distribution αG_0 that is the product of two parameters:

    1. A **centering distribution** providing an initial guess at the measure G. Can be thought of as the mean parameter of the DP.

$$G_0 = p\left(\boldsymbol{\mu}, \Sigma\right) \equiv \text{Normal-Inv-Wishart}(\boldsymbol{\mu}_0, \kappa_0, \nu_0, \Lambda_0)$$

2. A **concentration parameter** α that governs how far from G_0, the realizations of G tend to be. Can be thought of as the inverse-variance parameter of the DP.

- The random measure G is discrete, consisting of point masses. In order to obtain a continuous probability measure on the data, we convolve the point masses by the Normal$(. \mid \boldsymbol{\mu}, \Sigma)$ kernel to smear them out.

- The *Normal-Inverse-Wishart* centering distribution is the conjugate prior of a multivariate normal distribution with unknown mean and covariance. The *Normal-Inverse-Wishart* has the following four parameters:

  $\boldsymbol{\mu}_0$: *Prior belief (i.e. confidence) about mean vector*

  κ_0 : *Number of pseudo-observations, control variance of the means relative to variance*

  ν_0 : *Number of pseudo-observations, control relative variance of prior belief*

  Λ_0 : *Precision matrix, controls variation from the mean vector*

1.2 Model Specification: Univariate Case

To model univariate data $\mathcal{Y} = (y_1, y_2, \ldots, y_t, \ldots)$, we replace the joint conjugate multivariate centering distribution with its univariate special case

$$\left(\mu, \sigma^2\right) \sim \text{Normal-Inv–}\chi^2\left(\mu_0, \kappa_0, \nu_0, \lambda_0\right)$$

which can be expressed with a two-step sampling hierarchy

$$\begin{aligned} \sigma^2 &\sim \text{Scale–Inv–}\chi^2(\nu_0, \frac{1}{\lambda_0^2}) \\ \mu \mid \sigma^2 &\sim \text{Normal}(\mu_0, \frac{\sigma^2}{\kappa_0}) \end{aligned}$$

The univariate Dirichlet Process Mixture of Normals model can be written as

$$\begin{aligned} \alpha &\sim \text{Inv–Gamma}(a_\alpha, b_\alpha) \\ G \mid \alpha &\sim \text{DP}(\alpha G_0) \\ (\mu_t, \sigma_t) \mid G &\sim G \\ y_t \mid \mu_t, \sigma_t^2 &\sim \text{Normal}(\mu_t, \sigma_t^2) \end{aligned} \tag{2}$$

where

$$\begin{aligned} G_0 = p\left(\mu, \sigma^2\right) &\equiv \text{Normal-Inv–}\chi^2\left(\mu_0, \kappa_0, \nu_0, \lambda_0\right) \\ \theta_k = \left(\mu_k, \sigma_k^2\right) &\text{ is the mean and variance of the kth class} \end{aligned}$$

Note that the Scale-Inv–$\chi^2(\nu_0, \frac{1}{\lambda_0^2})$ distribution is equivalent to Inv–Gamma$(\frac{\nu_0}{2}, \frac{\nu_0}{2\lambda_0^2})$.

1.3 Joint Conjugate Priors

By choosing the Normal-Inv–χ^2 joint conjugate prior for (μ, σ^2), we induce a variance-dependent prior on μ_k such that when the sampling variance σ_k^2 of the observations is high, the uncertainty about μ_k is correspondingly large - yet calibrated by κ_0, the number of pseudo-observations we assign *a priori*. This form of dependence can be seen in Figure 1, which shows plots of Normal-Inv–χ^2 density for different values of κ_0, ν_0, and λ_0. The variance-dependent prior on μ is appropriate for many modelling situations since an increase in the unknown variance corresponds to a proportional increase in the variance of the mean, but the dependency will need to be removed when we consider random effects and semiparametric models.

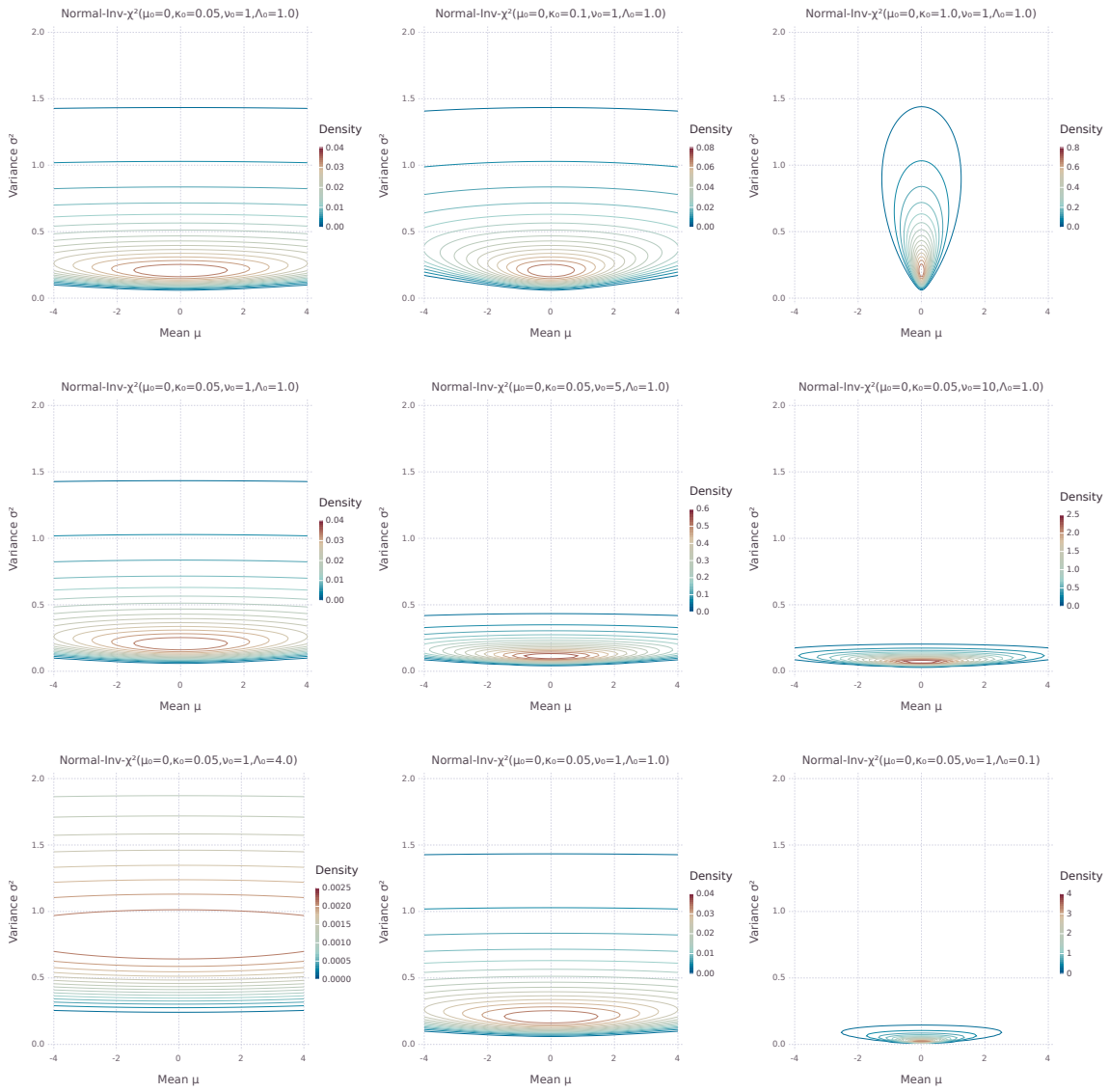

Figure 1: Probability Density of Normal-Inverse–χ^2

1.4 Model Description and Explanation

A Dirichlet Process Mixture of Normals is a *nonparametric Bayesian model*. To explicate the definition, we survey in the next few paragraphs, some basic concepts and *non-rigorous* definitions that are needed to clarify this class of models.

Probability Model We begin with the **sample space** $\mathcal{X}$ (e.g. $\mathcal{X} = \{0, 1\}$), a nonempty **set** whose elements are called **outcomes** $x \in \mathcal{X}$. From $\mathcal{X}$ we generate the **powerset** $\mathcal{P}(\mathcal{X})$ (e.g. $\mathcal{P}(\mathcal{X}) = \{\emptyset, \{0\}, \{1\}, \{0, 1\}\}$), the set of all **subsets** of $\mathcal{X}$. A set along with a collection of operations (e.g. $+, \times$) is called an **algebraic structure**. A powerset $(\mathcal{P}(\mathcal{X}), \cap, \cup, \neg, \emptyset, \mathcal{X})$ is an **algebra of sets**, a type of algebraic structure characterized by having only two binary operations $(\cap, \cup)$. Subsets of $\mathcal{X}$ are termed **events**. To equip a set with a structure, it should not be too large. For example, when the set is the real line (i.e. $\mathcal{X} = \mathbb{R}$), the powerset of $\mathbb{R}$, $\mathcal{P}(\mathbb{R})$, is simply too big to be manageable. More specifically, for a countably infinite set such as the natural numbers $\mathbb{N}$, for example, the cardinality of $\mathbb{N}$ is denoted by beth null, $\beth_0$. The cardinality of the set of all subsets of the natural numbers $P(\mathbb{N})$ is denoted by $\beth_1$, which is also equal to the cardinality of the **continuum** $\mathbb{R}$. The number of subsets of $\mathbb{R}$ (i.e. number of events) is then equal to the cardinality of $P(P(\mathbb{N}))$, the power set of the set of real numbers. The cardinality of this huge space is denoted by $\beth_2$. To stay within the limits of the continuum, we need to restrict ourselves to a subset of all possible events. This subset of interest is called a σ**-algebra** $\mathcal{B}(\mathcal{X})$. More specifically, a σ-algebra $\mathcal{B} \subset \mathcal{P}(\mathcal{X})$ is a sub-algebra of the powerset, completed to include countably infinite operations. The smallest possible σ-algebra is a collection of just two sets, $\{\mathcal{X}, \emptyset\}$. The largest possible σ-algebra is the collection of all the possible subsets of $\mathcal{X}$, the powerset. Now that we have found a way to enumerate all possible events, we need a function that assigns probabilities to events. That function $P : \mathcal{B} \to \mathbb{R}$, is called a **probability measure** and is defined on the **measurable space** $\Big(\mathcal{X}, \mathcal{B}(\mathcal{X})\Big)$. Finally, the combination of a measurable space and a probability measure gives us a **probability space** $(\mathcal{X}, \mathcal{B}, P)$, a space (i.e. a set equipped with some structure) in which the trinity of outcomes, events, and probabilities are rigorously defined.

Statistical Model There are infinitely many possible probability measures we can choose from. Indeed, let $\mathcal{M}(\mathcal{X})$ denote the **space of all probability measures** on $\Big(\mathcal{X}, \mathcal{B}(\mathcal{X})\Big)$. Since $\mathcal{M}(\mathcal{X})$ is vast, we might feel tempted to restrict ourselves to a subset of the space. We can do that by introducing a **parameter** $\theta \in \Theta$ that represents the pattern that explains the data. Also, we should not forget about the **parameter space** Θ, the set of all possible values of

θ. Here is an example, if we let $\Theta = \mathbb{R}^2$ represent the set of linear functions, then $\theta \in \mathbb{R}^2$ determines the linear trend in simple linear regression. As a result, the probability measures P_θ are now elements of $\mathcal{PM}(\mathcal{X})$, the **space of all probability measures** on Θ with elements $P_\theta \in \mathcal{PM}(\mathcal{X})$ indexed by a **parameter** $\theta \in \Theta$. A **statistical model** $\mathcal{P}$ therefore is a subset $\mathcal{P} \subset \mathcal{PM}(\mathcal{X})$ such that $\mathcal{P} = \{P_\theta \mid \theta \in \Theta\}$ where $\theta \to P_\theta$ is a bijective and measurable assignment. The model $\mathcal{P}$ is **parametric statistical model** if $\Theta \subset \mathbb{R}^d$ for some $d \in \mathbb{N}$. The subset $\mathcal{P} \subset \mathcal{PM}(\mathcal{X})$ can be restricted further by specifying a family of parametric models $\mathcal{G} = \{G_\theta \mid \theta \in \Theta\}$ where $\theta \to G_\theta$ is smooth. For example, $\mathcal{G} = \{N(\theta, 1) : \theta \in \Theta\}$ specifies the one-dimensional normal location family of models. If on the other hand, Θ is infinite dimensional, then $\mathcal{P}$ is **a nonparametric statistical model**. In this case Θ is equivalent to $\mathcal{M}(\mathcal{X})$, the space of all probability measures on $\Big(\mathcal{X}, \mathcal{B}(\mathcal{X})\Big)$.

Bayesian Model A **parameteric Bayesian statistical model** $(\mathcal{P}, \Pi)$ consists of a model $\mathcal{P}$, the **observation model**, and a **prior distribution** Π on Θ such that θ is a random variable taking values in Θ and $\Pi\Big(\{P_\theta : \theta \in \Theta\}\Big) = 1$. In a Bayesian model, data is generated hierarchically in two stages:

$$\theta \sim \Pi$$
$$X_1, X_2, \ldots \mid \theta \sim_{iid} P_\theta \qquad \theta \in \Theta \subset \mathbb{R}^d$$

After we observe data $(X_1, X_2, ..., X_T)$, the prior is updated to the posterior $\Pi(. \mid X_1, X_2, ..., X_T)$.

A **nonparametric Bayesian model** is a Bayesian model whose prior Π is defined on an infinite dimensional parameter space Θ. The corresponding two-stage hierarchical model is given as

$$P \sim \Pi$$
$$X_1, X_2, \ldots \mid P \sim_{iid} P \qquad P \in \mathcal{P}$$

A prior distribution on an infinite dimensional space is a **stochastic process**. Defining an infinite dimensional prior distributions is not straightforward, but one way to construct a prior distribution Π on Θ is through De Finetti's Theorem.

De Finetti's Theorem. *A sequence of random variables $\{X_t\}_{t=1}^{\infty}$ with values on $\mathcal{X}$ is* ***ex-***

changeable *if and only if there is a unique measure* Π *on* Θ *such that for all* T

$$P(X_1 = x_1, X_2 = x_2, ..., X_T = x_T) = \int_\Theta \left(\prod_{t=1}^{T} \theta(X_t = x_t) \right) \Pi(d\theta) \quad \textit{General Form}$$

$$\int_\Theta \left(\prod_{t=1}^{T} p(X_t = x_t \mid \theta) \right) p(\theta) d\theta \quad \textit{Specific Form}$$

The theorem gives us a infinite mixture representation of the joint probability of the observations. More importantly, it shows that exchangeability implies conditional independence of the observations given θ.

Dirichlet Distribution When $\mathcal{X}$ is finite, there is usually a natural unique measure Π we can obtain. More specifically, if $\mathcal{X} = \{1, 2, \ldots, K\}$, then $\mathcal{PM}(\mathcal{X}) = \{(p_1, \ldots, p_K) : 0 \leq p_k \leq 1, \sum p_k = 1\}$. That is, the space of probability measures corresponds to a simplex parametrized by a $K-1$ dimensional vector $\boldsymbol{p} = (p_1, \ldots, p_{K-1})$. A natural prior Π to specify on $\boldsymbol{p}$ is the Dirichet distribution. For example, consider the Bayesian model$(\mathcal{P}, \Pi)$ where the observation model $\mathcal{P}$ is the *Categorical distribution* defined on the sample space $\mathcal{X} = \{1, 2, \ldots, K\}$, and the prior Π is the *Dirichlet distribution* defined on the simplex $\Theta = \{(p_1, \ldots, p_K) : 0 \leq p_k \leq 1, \sum p_k = 1\}$

$$\begin{aligned} \boldsymbol{p} &\sim \text{Dirichlet}(\boldsymbol{\alpha}) \\ x_i \mid \boldsymbol{p} &\sim \text{Categorical}(\boldsymbol{p}) \\ \textit{where} \quad \boldsymbol{\alpha} &= \left(\frac{\alpha}{K}, \cdots, \frac{\alpha}{K} \right) \\ \boldsymbol{p} &= (p_1, \cdots, p_K) \end{aligned}$$

Here, *the Dirichlet distribution* is the conjugate prior of the Categorical distribution and the concentration hyperparameter vector $\boldsymbol{\alpha}$ represents the number of pseudo-observations, the *a priori* weights, for each of the K clusters. Figure 2 shows density and sampling plots corresponding to Dirichlet distributions for $K = 3$ and several choices of $\boldsymbol{\alpha}$.

Dirichlet Process The Dirichlet distribution is a conjugate prior that allows us to sample *iid* from $\mathcal{X} = \{1, 2, \ldots, K\}$. To generalize, we can make $\mathcal{X} = \mathbb{R}$ thus obtaining the space of all measures $\mathcal{M}(\mathbb{R})$ defined on the measurable space $(\mathbb{R}, \mathcal{B})$, which is the real line equipped with the **Borel σ-algebra**.The Borel σ-algebra is the σ-algebra generated by the open sets. An element of a Borel σ-algebra is a **Borel set**. It is a set that can be constructed from open or closed sets by repeatedly taking countable unions and intersections. For example: $\mathcal{B}(\mathbb{R}) := \sigma(\mathcal{C}); \mathcal{C} = \{(a, b], -\infty \leq a \leq b \leq \infty\}$.

It can be shown (Ghosh & Ramamoorthi, 2006) that for every partition $B_1, B_2, \ldots, B_k$ of the real line $\mathbb{R}$ by Borel sets, there exists a unique measure DP_α on $\mathcal{M}(\mathbb{R})$ called the Dirichlet Process with parameter α satisfying

$$(P(B_1), P(B_2), \ldots, P(B_k)) = DP(\alpha B_1, \alpha B_2, \ldots, \alpha B_k))$$

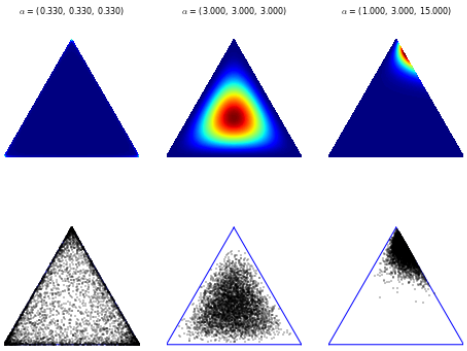

Figure 2: Dirichlet Distributions

That is, when $\mathcal{X} = \mathbb{R}$, then $\mathcal{M}(\mathbb{R})$, is the space of all probability measures on $\mathbb{R}$. If the sample space $\mathcal{X} = \mathbb{R}$ is partitioned into measurable subsets, then for every partition $(B_1, B_2, \ldots, B_k)$, the prior probability measure Π on $(p(B_1), p(B_2), \ldots, p(B_k))$ is a Dirichlet process prior.

As stated earlier, Bayesian models can be thought of as a **random mixture model** where we first sample from a mixing measure $\theta \sim \Pi$, then sample from a component $X_t \mid \theta \sim P_\theta$. In general a random mixing measure has the following form:

$$G(.) = \sum_{k=1}^{K} p_k \delta_{\theta_k}(.)$$

and the Dirichlet process is a distribution on **random probability measures** of the form

$$G(.) = \sum_{k=1}^{\infty} p_k \delta_{\theta_k}(.) \quad \textit{where} \sum_{k=1}^{\infty} p_k = 1$$

The Dirichlet process $DP(G_0, \alpha)$ is the simplest distribution - in terms of the extent of independence it assumes - that can have this form. That is probably why it has been referred to as *the normal distribution of Bayesian nonparametrics*. More specifically, the location of the atoms, the $\theta's$, are sampled *iid* but the weights are sampled as independent proportions. Now, the weights cannot be sampled *iid* because then they will not sum up to one. However, the next best thing to *iid* sampling is to obtain independent proportions, as is done using the stick-breaking representation.

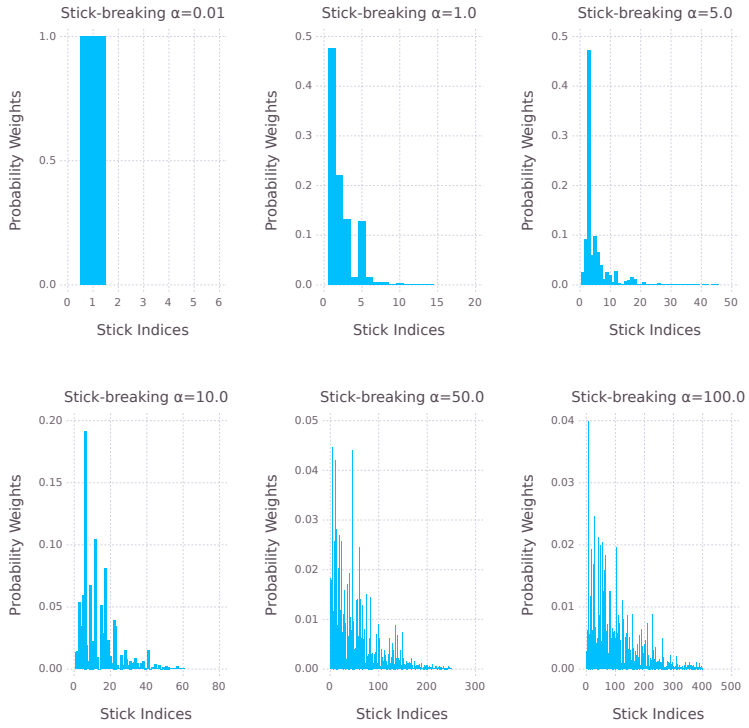

Figure 3: Realizations of the Dirichlet Process

Stick Breaking Representation The stick breaking construction, namely

$$G(.) = \sum_{k=1}^{\infty} p_k \delta_{\theta_k}(.)$$

$$\textit{where} \quad \theta \sim^{iid} G_0$$

$$V_k \sim Beta(1, \alpha)$$

$$p_k = V_k \prod_{i=1}^{k-1} (1 - V_i)$$

allows us to visualize the realizations of the **Dirichlet Process** $G \mid \alpha \sim \mathrm{DP}(\alpha G_0)$. Figure 3 shows plots of samples from the Dirichlet process for different settings of α by means of the *stick-breaking construction*

Dirichlet Process Mixtures As (Persi Diaconis, 1986) have shown, the Dirichlet process prior can behave pathologically and give inconsistent estimates even for a location parameter problem with known density. The Dirichlet Process is defined on discrete measure and is therefore not an appropriate prior for continuous distributions. Nonetheless, this drawback can be elevated by incorporating a continuous kernel density allowing it to be defined over continuous distributions.

Let $\mathcal{M}(\mathbb{R})$ denote the space of all probability measures on $(\mathbb{R}, \mathcal{B})$ and let Π be the prior over

the space of measures. For data $\mathbf{y} \mid G \sim F_G$ the **Dirichlet Process Mixtures of Normals (DPMN)** can be specified as

$$f_G(\mathbf{y}) = \int_\Theta \text{Normal}(\mathbf{y} \mid \theta) G(d\theta)$$

where the prior Π for the unknown mixing measure $G \sim \Pi$ is a Dirichlet Process prior $\text{DP}(\alpha G_0)$. Through the transform, the DP prior induces a prior on $f_G(\mathbf{y})$ called the **Dirichlet Process Mixture (DPM)**. The corresponding stick-breaking representation of the DPMN model is

$$\begin{aligned} f_G(\mathbf{y}) &= \sum_{k=1}^{\infty} p_k \,\text{Normal}(\mathbf{y} \mid (\boldsymbol{\mu}_k, \Sigma_k)) \\ \textit{where} \quad (\boldsymbol{\mu}_k, \Sigma_k) &\sim^{iid} G_0 \\ V_k &\sim Beta(1, \alpha) \\ p_k &= V_k \prod_{i=1}^{k-1} (1 - V_i) \\ G_0 &= p\,(\boldsymbol{\mu}, \Sigma) \equiv \text{Normal-Inv-Wishart}(\boldsymbol{\mu}_0, \kappa_0, \nu_0, \Lambda_0) \end{aligned}$$

2 Question Part B: Model Applications

Question Describe how the model in part A can be used (i) for density estimation, (ii) for latent class analysis, and (iii) as a random effects model. For each use of the model, describe the data that would be needed for that use and how output from the model could be interpreted. Include cautions on these interpretations, as appropriate.

Answer To see how the DPMN model can be used for density estimation, latent class analysis, and random effects modelling, we start with a finite mixture model to clarify the basic ideas.

Finite Mixtures Consider the data $\mathcal{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{500})$ depicted in the overlayed scatter plot of Figure 4. The data consists of $T = 500$ observations where each observation $\mathbf{y}_t = [y_{t1}, y_{t2}]$ is a $D = 2$ two-dimensional vector of measurements.

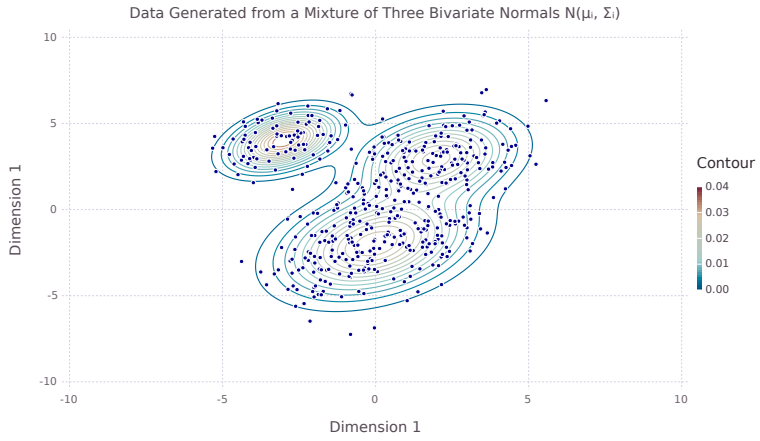

Figure 4: Mixture Data

If we observe, or rather assume, that the data exhibits some form of grouping pattern, we can then attempt to cluster the T observations into $K \leq T$ subsets. That is, we can give each observation $\mathbf{y}_i$ a *cluster label* $z_i \in \{1, \ldots, K\}$ such that each observation belongs to one cluster only. Of course, the clustering labels are unobserved and our goal, accordingly, is to uncover their values in order to determine the underlying statistical representation of the data.
We start by introducing the following quantities for given data $\mathcal{Y}$:

$$\mathcal{Z} = (z_1, \cdots, z_T) \text{ is the vector of unobserved class labels for all } T \text{ observations}$$

$$\mathcal{C} = \{c_1, c_2, \ldots, c_K : c_k \subseteq \{1, 2, \ldots, T\}\} \text{ is the clustering of the observations into } K \text{ clusters}$$

$$\mathcal{N} = \{N_1, N_2, \ldots, N_K : N_k = \sum_{i=1}^{T} \delta(z_i = k)\} \text{ is the number of observations in each cluster}$$

Besides the clustering pattern, we can also notice another statistical pattern. The observations within a cluster k are random yet seem to follow some distributional form. In other words, if we assume that we have observed the labels $\mathcal{Z}$ (by conditioning on $\mathcal{Z}$), then we can suppose that the clustered data $\mathcal{Y} \mid \mathcal{Z}$ is governed by a parametric distribution $\mathcal{K}(.)$. Accordingly, the data can be modelled as a mixture of K component distributions such that each component accounts for the dependencies observed within a cluster of observations. Now, if we associate with each component distribution $\mathcal{K}(.)$ a parameter vector θ_k, then the collection of all K parameter vectors can be denoted by

$$\Theta = (\theta_1, \cdots, \theta_K)$$

Since $z_t \in \{1, \ldots, K\}$ is a discrete random quantity with support on the positive integers $\mathbb{Z}^+$, a natural modelling choice to quantity our uncertainty is the Categorical distribution. As for the choice of the kernel, we require a real-valued distribution that has both a mean parameter to represent the centres of the components and a variance parameter that quantifies the spread of the observations around a particular center. According to **Jayne's principle of maximum entropy** (Jaynes, 2003), the normal distribution has maximum entropy among all distribution with a specified variance. The data can be modelled as follows:

$$\begin{aligned} z_t &\sim \text{Categorical}(\boldsymbol{p}) \\ \mathbf{y}_t &\sim \text{Normal}(\boldsymbol{\mu}_{z_t}, \Sigma_{z_t}) \end{aligned} \tag{3}$$

Where $\boldsymbol{p}$ is a probability vector. The joint distribution of the data and the two unobserved quantities we introduced so far is given by $p(\mathcal{Z}, \Theta, \mathcal{Y})$ which we can factor into

$$\mathcal{L}(\mathcal{Y} \mid \mathcal{Z}, \Theta) \Pr(\Theta) \Pr(\mathcal{Z})$$

Since the cluster assignment label z_i is unobserved, we can quantify our uncertainty about the relative proportions of clusters by assigning prior probabilities (e.g. $p_i = Pr(z_i = k)$) to the set of possible outcomes $z_i \in \{1, \ldots, K\}$. As have been mentioned earlier, the Dirichlet distribution is a natural prior for the vector of probabilities $\boldsymbol{p}$. Note that the joint distribution of the observable and unobservable quantities is now $p(\mathcal{Z}, \Theta, \boldsymbol{p}, \mathcal{Y})$. So far the probabilistic modelling framework we have been pursuing has naturally led us to the **Finite Dirichlet**

Mixture Model with the following hierarchical specification:

$$\begin{aligned} \boldsymbol{p} &\sim \text{Dirichlet}(\boldsymbol{\alpha}) \\ z_i \mid \boldsymbol{p} &\sim \text{Categorical}(\boldsymbol{p}) \\ \Sigma_k &\sim \text{Inv-Wishart}_{\nu_0}(\Lambda_0^{-1}) \\ \boldsymbol{\mu}_k \mid \Sigma_k &\sim \text{Normal}(\boldsymbol{\mu}_0, \frac{\Sigma_k}{\kappa_0}) \\ \mathbf{y}_i \mid z_i, \boldsymbol{\mu}_{z_i}, \Sigma_{z_i} &\sim \text{Normal}(\boldsymbol{\mu}_{z_i}, \Sigma_{z_i}) \end{aligned} \tag{4}$$

$$\begin{aligned} \textit{where} \quad \boldsymbol{\alpha} &= \left(\frac{\alpha}{K}, \cdots, \frac{\alpha}{K}\right) \ \textit{concentration hyperparameter} \\ \boldsymbol{p} &= (p_1, \cdots, p_K) \ \textit{probabilities of class labels} \\ \theta_k &= (\boldsymbol{\mu}_k, \Sigma_k) \ \textit{mean and covariance of observations in the kth class} \\ \Theta &= (\theta_1, \cdots, \theta_K) \ \textit{collection of all K parameter vectors} \end{aligned}$$

Choosing the concentration hyperparamter $\boldsymbol{\alpha}$ to be uniform across components gives us a symmetric Dirichlet distribution that allows us to obtain the Dirichlet Process Mixture Model as a limiting case of the finite Dirichlet Mixture Model. We outline how that can be done as we describe how the DDPM can be used as for clustering or latent class analysis.

2.1 Clustering Analysis

The posterior distribution of the parameters of the Finite Dirichlet Mixture Model after conditioning on the observed data is

$$f(\mathcal{Z}, \Theta, \boldsymbol{p} \mid \mathcal{Y}) \propto \mathcal{L}(\mathcal{Y} \mid \mathcal{Z}, \Theta) \Pr(\Theta) \Pr(\mathcal{Z} \mid \boldsymbol{p}) \Pr(\boldsymbol{p})$$

In clustering, we are mainly interested in $f(\mathcal{Z} \mid \mathcal{Y})$ the posterior distribution of the latent classes, or cluster labels. As a rule in Bayesian computation, we should not leave to numerical simulation that which we can obtain analytically! In our model, there are two simplifications we can make.

1. We can integrate $\boldsymbol{p}$ out to ease computation. That is

$$\begin{aligned} \mathcal{L}(\mathcal{Z}, \Theta \mid \mathcal{Y}) &\propto \mathcal{L}(\mathcal{Y} \mid \mathcal{Z}, \Theta) \Pr(\Theta) \Pr(\mathcal{Z}) \\ &\propto \mathcal{L}(\mathcal{Y} \mid \mathcal{Z}, \Theta) \Pr(\Theta) \int_{\mathbf{p}} \Pr(\mathcal{Z} \mid \boldsymbol{p})) \Pr(\boldsymbol{p}) \mathrm{d}\mathbf{p} \\ &\propto \mathcal{L}(\mathcal{Y} \mid \mathcal{Z}, \Theta) \Pr(\Theta) \int_{\mathbf{p}} \prod_{i=1}^{N} \Pr(z_i \mid \boldsymbol{p}) \Pr(\boldsymbol{p}) \mathrm{d}\mathbf{p} \end{aligned}$$

2. We can also integrate out Θ since the model employed conjugate priors.

$$f(\mathcal{Z} \mid \mathcal{Y}) \propto \prod_{k=1}^{K} \left[\int_{\theta} \prod_{i \in c_k} \mathcal{L}(y_i \mid \theta_k) \Pr(\theta_k) \mathrm{d}\theta \right] \Pr(\mathcal{Z})$$

Where

$$Pr(\mathcal{Z}) = \frac{\Gamma(\alpha)}{\Gamma(\alpha + T)} \times \prod_{k=1}^{K} \frac{\Gamma\left(N_k + \frac{\alpha}{K}\right)}{\Gamma\left(\frac{\alpha}{K}\right)} \tag{5}$$

is the prior probability of a particular cluster assignment (i.e. labelling) of the observations.

By keeping fixed a given cluster label z_t and assuming exchangibility of the T data observations, we can obtain the conditional prior of z_t given the remaining $T-1$ cluster labels $\mathcal{Z}^{(-t)}$

$$Pr(z_t = k \mid \mathcal{Z}^{(-t)}) = \frac{N_k^{(-t)} + \frac{\alpha}{K}}{(\alpha + T - 1)} \tag{6}$$

In the finite Dirichlet Mixture model, the number of clusters K is assumed to be known beforehand, thus necessitating fitting multiple models with different number of components. More importantly however, since the number of clusters K is a fixed quantity, the model's information capacity and complexity doesn't increase as more data comes in. Now if in Equation 6, we take the limit $K \to \infty$, we get

$$\Pr(z_t = k | \mathcal{Z}^{(-t)}) = \begin{cases} \frac{\alpha}{(\alpha+T-1)} & \text{if } k \in \text{new cluster}, \\ \\ \frac{N_k^{(-t)}+\alpha}{(\alpha+T-1)} & \text{if } k \in \text{ existing cluster}; \end{cases} \tag{7}$$

These are the prior probabilities of the **Chinese Restaurant Process**. The Chinese Restaurant Process is equivalent to the Polya Urn scheme of representing a Dirichlet Process mixture. Thus, by taking the limit of components to infinity, we obtain a Dirichlet Process Mixture from a finite Dirichlet Mixture (Rasmussen, 1999). Note that since the DP is discrete in nature, it permits the possibility of obtaining ties in the realized values of the latent cluster labels $\mathcal{Z}$. This in turn induces a probability model on clusters via the DPM model.

2.1.1 Application to Data

The DPMN model can be used with the objective of clustering data or classifying observations if the cluster labels are available. Although the model is flexible, it does make a few assumptions that are important to keep in mind.

- The most basic assumption is that number to of clusters K is countably infinite and for finite data $K \to \infty$ as $T \to \infty$. Moreover, the expected number of clusters is a function of the concentration parameter α. More precisely, Figure 5 shows plots of samples from the Dirichlet process for different settings of α by means of the *Polya Urn scheme* representation. As the plots show, the pattern of expected number of clusters tends to follow the theoretical relationship $\mathbb{E}(K) = \alpha \log(T)$. Figure 6 shows scatterplots of data generated from a bivariate DPMN model under different values of α.

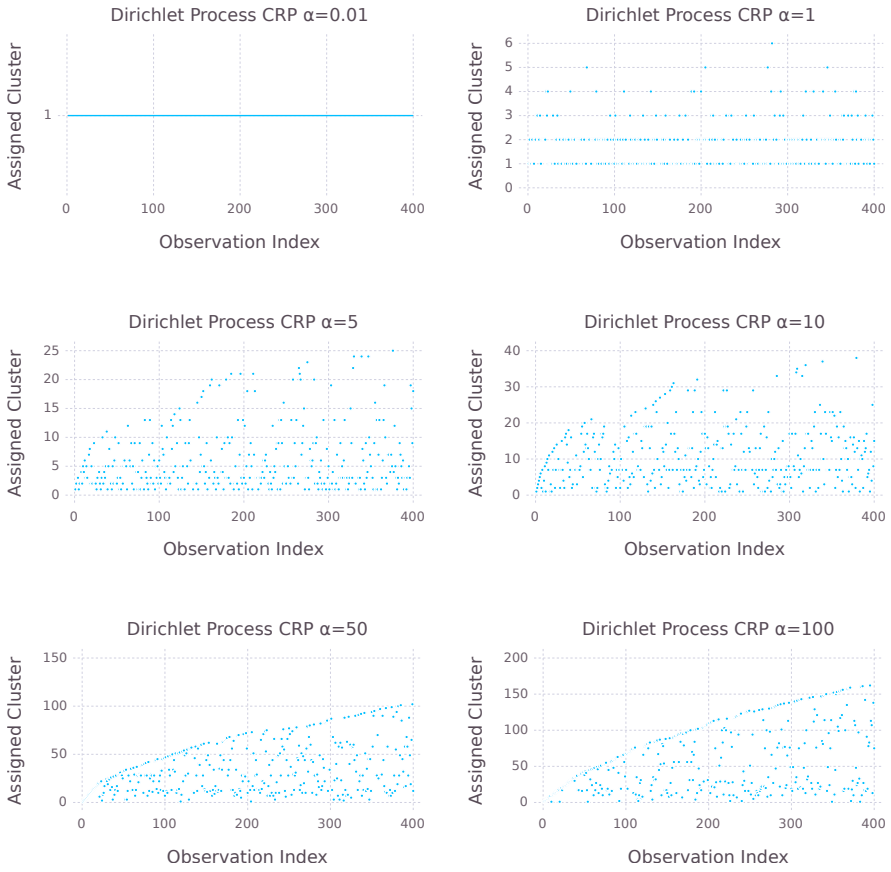

Figure 5: Realizations of the Dirichlet Process

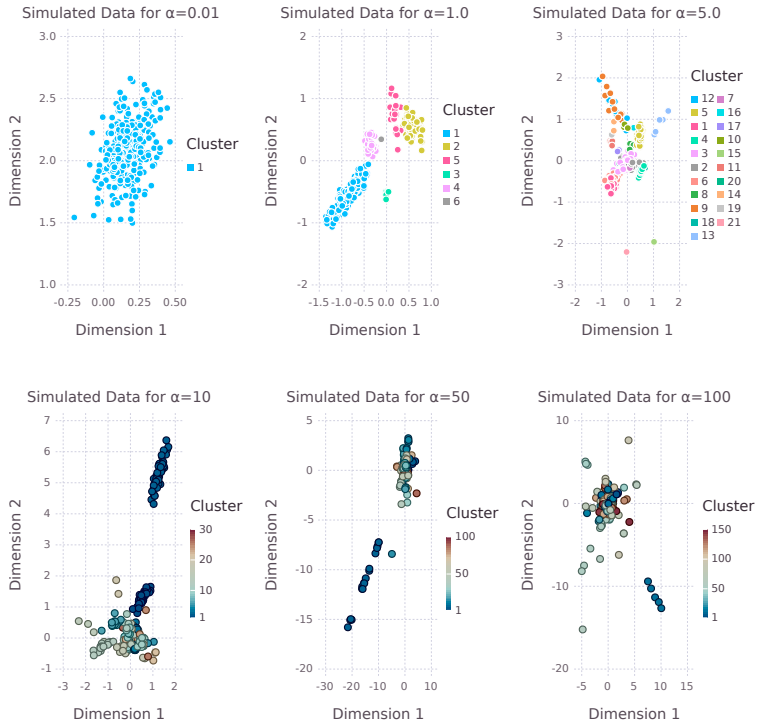

Figure 6: Scatterplots of DPMN Model for Bivariate Observations

- The model also assumes that the observations can be grouped into $K \leq T$ clusters such that each subject (i.e. observation) belongs to one single cluster only (i.e. *soft clustering*). If some of the observations belong to multiple clusters as is the case when data is characterized by multiple groupings and nested structure, then a more general model involving the **Dependent Dirichlet Process** or one of its special cases, the **Hierarchical Dirichlet Process**, would be more appropriate.

- A third feature of the model is the form of the kernel. Although the model is nonparametric and quite flexible, it is still a mixture of normals and does assume that the distribution of the observations in a given cluster is normally distributed. Clusters with non-Gaussian shapes might prove problematic for the model such as when the clusters tend to form a doughnut shape where the distance between clusters can be smaller than the distance within a cluster.

- A fourth aspect of the model that should be considered is the role of the variance parameter Λ_0 of the centering distribution G_0. The more diffuse the variance of the G_0 is, the lower the value of the marginal likelihood is and as a result fewer new clusters get introduced as more data comes in. In the limit, as variance goes to infinity, the model reduces to a parametric model with a single cluster. Interestingly, this limiting behaviour

is similar to what happens when the concentration parameter α goes to 0, namely, we get the following model

$$(\boldsymbol{\mu}, \Sigma) \sim G_0$$
$$\mathbf{y}_i \mid \boldsymbol{\mu}, \Sigma \sim \text{Normal}(\boldsymbol{\mu}, \Sigma)$$

Even if the data we would like to conduct inference on is appropriate for our model's assumptions, we still need to deal with one more issue that comes up in mixture models and clustering. When the objective of the analysis is inference on the parameters of specific components, then interpertablility of the model might be problematic due to identifiability issues, namely, the invariance of the posterior distribution with respect to the relabelling of the components. That is, we can switch the labels of the components and still obtain the same likelihood.

The multivariate Dirichlet Process Mixture of Normals model can be applied to clustering data in many applications. An identical model was used by (Wood & Black, 2008) for neural spike sorting in order to determine which spikes corresponds to a particular neuron. Similar model formulations were used for clustering activation patterns in fMRI data (Kim, Smyth, & Stern, 2006) and in verb clustering of linguistics data (Vlachos, 2008).
We have implemented a *sequential importance resampling* clustering algorithm in Appendix 2. The code was written in *Julia*, a fast high-level scientific programming language that has been developed recently with the aim of improving on some of the flaws found in current scientific programming languages such as *R, Python, and Matlab.* It is main advantage with respect to Bayesian computation lies in the language ability to compile code that reads like Matlab or R, into machine code runs as fast as *C*. The code was used to fit the model to a synthetic dataset with the goal of classifying the multivariate observations into clusters.

2.2 Density Estimation

In density estimation of mixture models, we are concerned with making inference on an unknown mixture distribution

$$f_G(\mathbf{y}) = \int_\Theta \text{Normal}(\mathbf{y} \mid \theta) G(d\theta)$$

given data $\mathcal{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_T)$ where $\mathbf{y}_t \sim f_G$ and $\theta = (\mu, \sigma^2)$. In our model G follows a Dirichlet Process prior $G \sim \text{DP}(\alpha G_0)$, a prior on the mixture densities.

We can generate posterior samples from the mixture model $f(G \mid \mathbf{y})$ based on the posterior distribution of G, which is a mixture of Dirichlet Processes

$$G \mid \mathcal{Y} \sim \int DP(\alpha + \sum_{t}^{T} \delta_{\theta_t}(.))dp(\Theta \mid \mathcal{Y})$$

Our object of interest however is $f_G \mid \mathcal{Y}$, which although can be expressed in an analytical form, the resulting expression is too complicated to be implemented in practice. Instead computation proceeds by simulation in which draws from the posterior are generated. The mean of the posterior predictive distribution

$$\mathbb{E}(f_G \mid \mathcal{Y}) = \frac{1}{|\alpha| + T} \int \text{Normal}(\mathbf{y} \mid \theta)\alpha(d\theta) + \frac{1}{|\alpha| + T} \mathbb{E}(\sum_{t}^{T} \text{Normal}(\mathbf{y} \mid \theta_t) \mid \mathcal{Y})$$

can be estimated by taking the mean of the posterior draws

2.2.1 Application to Data

Nonparametric density estimation using the DPMN model has been performed both on univariate datasets (Escobar & West, 1995) and multivariate datasets (Muller, Erkanli, & West, 1996). For example, the model was used to estimate and assess the multimodality of the distribution density of galaxy velocities. Support for multiple modes provides evidence for the existence of superclusters in the far universe.

With regards to interpreting posterior inference based on the DPMN model for density estimation, a few additional consideration to the ones we have discussed already are particularly relevant.

- One important consideration in density estimation is the choice of the kernel function. Since the Gaussian kernel that we have assigned belongs to the location-scale family of distribution functions, we are implicitly assuming that the sample space is defined on the entire real line $\mathbb{R}$. If the sample space was defined on an interval [0,1] instead, then a Beta kernel would have been an adequate choice. For other domains of support, we can choose the appropriate kernel on the basis of a Feller prior sampling scheme as detailed in the framework provided by (Petrone & Veronese, 2002) have provided.

- Once we have estimated the density, we should not confuse the number of modes with the number of components. The number of modes can serve as a lower bound on the number of components, but even then, the number of components should be secondary when the goal of the analysis is density estimation.

- In our model we have specified a gamma prior on α so that the data can inform the model of the correct number of components. This is important since the choice of *alpha* controls the number of components and we are therefore assuming that the data has enough signal to be informative about the concentration parameter. More important though is the choice of those hyperparameters that determine the variance of the kernel. This is so because high variance leads to less smoothing and an increase in the number of modes for any given number of components K.

2.3 Random Effects

The DPMN model can be used as random effects model on data consisting of several within-subject measurements taken from many subjects. The data usually has the same form as in the clustering application. That is, $\mathcal{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_t, , \ldots)$ is the data for T subjects where each vector $\mathbf{y}_t = [y_{t1}, y_{t2}, \ldots, y_{tD}]$ is the repeated measurement for subject t. The difference in the case of random effects modelling is that we model the mean and the sampling variance as independent. We also make two assumption: that the repeated measurements are not correlated and that the mean vector is actually a scalar that is equal across the repeated measurements. The second assumption implies that the different values that make up the mean vector $\boldsymbol{\mu}$ are in fact due to within subject error. In summary, whereas in the case of clustering of multivariate observations we considered the following conjugate model

$$\begin{aligned} \alpha &\sim \text{Inv–Gamma}(a_\alpha, b_\alpha) \\ G \mid \alpha &\sim \text{DP}(\alpha G_0) \\ (\boldsymbol{\mu}_t, \Sigma_t) \mid G &\sim G \\ \mathbf{y}_t \mid \boldsymbol{\mu}_t, \Sigma_k &\sim \text{Normal}(\boldsymbol{\mu}_t, \Sigma_t) \\ G_0 = p\left(\boldsymbol{\mu}, \Sigma\right) &\equiv \text{Normal-Inv-Wishart}(\boldsymbol{\mu}_0, \kappa_0, \nu_0, \Lambda_0) \end{aligned} \tag{8}$$

to model random effects, we modify the model slightly to obtain the semiparametric model

$$\begin{aligned} \alpha &\sim \text{Inv–Gamma}(a_\alpha, b_\alpha) \\ G \mid \alpha &\sim \text{DP}(\alpha G_0) \\ \sigma^2 &\sim \text{Inv–Gamma}(a_0, b_0) \\ \mu_t \mid G &\sim G \\ \mathbf{y}_t \mid \mu_t, \sigma^2 &\sim \text{Normal}(\mathbf{1}\mu_t, \sigma^2\mathbf{I}) \\ \textit{where } G_0 = p\left(\mu\right) &\equiv \text{Normal}\left(\mu_0, \tau_0^2\right) \end{aligned} \tag{9}$$

The above model is basically both a **random intercept model** that allows the random variability among subjects to be captured and a **latent class** model that allows the subjects to be clustered into groups. Note that the observation model can be reduced to

$$y_{td} \mid \mu_t, \sigma^2 \sim \text{Normal}(\mu_t, \sigma^2) \; for \; d = 1, \ldots, D$$

In a general linear random effects model the prior on the random effects is typically assumed to be normal. The normality assumption can be inappropriate in situations where we have subjects that tend to cluster together or where a few subjects tend to be very different from the rest. This is because the normal distribution has thin tails that prevents the subject effects from having very different values. In contrast, the nonparametric DP prior is more appropriate. The Dirichlet Process discrete nature allows the subjects to cluster. Furthermore, the nonparametric DPM prior is flexible enough to allow multimodality and skewness in the distribution of the random effects.

2.3.1 Application to Data

- As in the other two applications, we need to carefully consider the the choices of the hyperparameters. In the context of random effects, as $\alpha \to 0$ we have $G = \delta_\mu$, implying that all the subjects belong to a single cluster. That is we are effectively fitting a normal model that ignores the heterogenity of the subjects since the variance of the random effects distribution is zero. On the other hand, when $\alpha \to \infty$ we have $G = G_0$, implying that each subject will have has own cluster and we are effectively sampling the random effects from G_0.

- Moreover, the choice of variance hyperparameter τ_0^2 should be carefully considered. If we make it large and uninformative, we risk forcing all the subjects to be in the same cluster since a high variance parameter has the effect of assigning low probability to the introduction of new clusters. A proposed solution to deal with sensitivity of the cluster distribution to the variance hyperparmater is to standardize the data. The standardization can be viewed as an empirical Bayes approach to estimating G_0 as suggested by (McAuliffe, Blei, & Jordan, 2006).

- It should be noted that in the model specification, the DP prior is a discrete distribution on the random effects, implying that subjects that belongs to a given cluster have the same value of the random effect. This fact is important when interpreting the model's output since if we are interested in clustering subjects that have similar, not just identical,

random effects, we need to modify the model slightly by assigning the random effects, a DPM prior instead of a DP prior. More specifically,

$$
\begin{aligned}
\alpha &\sim \text{Gamma}(a_\alpha, b_\alpha) \\
G \mid \alpha &\sim \text{DP}(\alpha G_0) \\
\sigma^2 &\sim \text{Inv–Gamma}(a_0, b_0) \\
\left(\mu_{0t}, \tau_{0t}^2\right) \mid G &\sim G \\
\mu_t &\sim \text{Normal}\left(\mu_{0t}, \tau_{0t}^2\right) \\
\mathbf{y}_t \mid \mu_t, \sigma^2 &\sim \text{Normal}(\mathbf{1}\mu_t, \sigma^2\mathbf{I}) \\
\textit{where } G_0 = p\left(\mu_{0t}, \tau_{0t}^2\right) &\equiv \text{Normal-Inv–}\chi^2\left(\mu_0, \kappa_0, \nu_0, \lambda_0\right)
\end{aligned}
\tag{10}
$$

3 Question Part C: Sensitivity Analysis

Question The mass parameter is one portion of the parameter of the Dirichlet process. Consider the density estimation version of your model. Generate distributions from your model for a variety of mass parameters—say masses of 0.01, 1, 10, and 100. (Adjust this generation as appropriate for your model). Summarize the differences in the distributions as the mass parameter varies.

Answer Figure 7 shows density histograms of the univariate DPMN model for different settings of the concentration parameter α. As has been discussed earlier, as $\alpha \to 0$ we have $G = \delta_\mu$ and as a result all the observations belong to one component. In the other extreme, when $\alpha \to \infty$, we have $G = G_0$, and as a result each observation has its own component and the model reduces to G_0, a parametric model. Between these two extremes, the expected number of components is a determined by α according to the relation $\mathbb{E}(K) = \alpha \log(T)$, where T is the number of components. Also, the plots seem to indicate that the number of components is generally greater than the number of modes. The *Julia* code for generating the data and the plots is provided in the appendix.

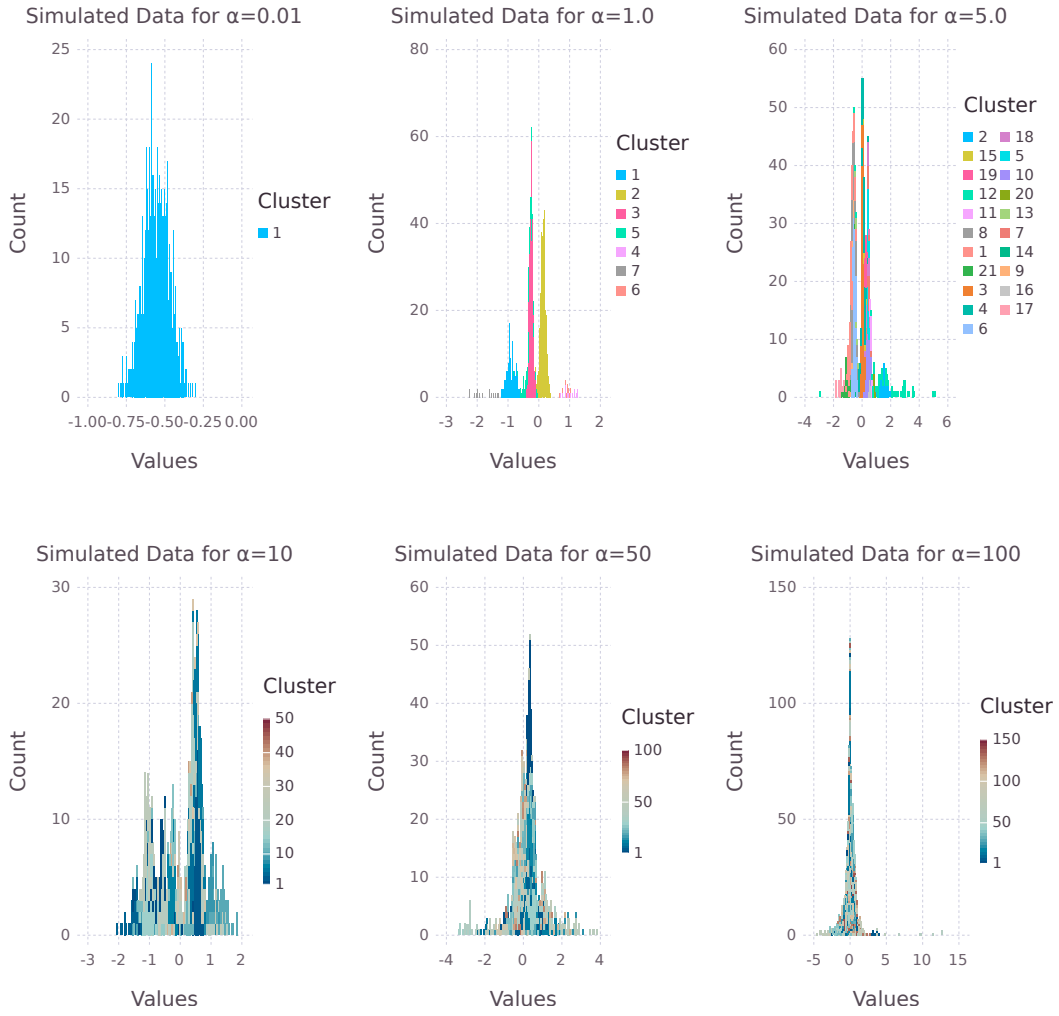

Figure 7: Histograms of Dirichlet Mixture Model for univariate Observations

4 Question Part D: Model Fit and Comparison

Question Provide a fit of your model in part A for a basic density estimation problem to a data set of your choice. You may find the R code in the Appendix of the book by Muller and Rodriguez helpful for fitting the model. Contrast this density estimate with a traditional (classical) kernel density estimate.

Answer The univariate DMPN model was fit to a neural dataset with the goal of estimating the distribution density of the measurements. The dataset contains 346 peak amplitude measurements of spontaneous currents flowing into individual brain cells of Guinea pigs (Paulsen & Heggelund, 1994). To goal of the experiment was to determine whether the current flow was due to a single burst or whether it was *quantal* in nature, consisting of several regularly spaced bursts that decrease in magnitude as the current amplitude increases.

Let $y_1, \ldots, y_i, \ldots\ldots, y_n$ be a sample from the density distribution f. The kernel density estimator $\hat{f}_h(y)$ of f based on the sample is

$$\hat{f}_h(y) = \frac{1}{n}\sum_{i=1}^{n} K_h(y - y_i) \quad = \frac{1}{nh}\sum_{i=1}^{n} K\Big(\frac{y - y_i}{h}\Big),$$

where h is the kernel's bandwidth. There are several choices of kernels but a commonly used kernel that we will considering is the Gaussian kernel

$$K\Big(x\Big) = \frac{e^{-\frac{1}{2}x^2}}{\sqrt{2\pi}}$$

Figure 8 shows three Gaussian kernel density estimates for varying values of h overlayed on the data histogram. As can be seen, as the value of h increases, the estimated density increases in smoothness. The choice of h has a noticeable effect on the density estimate, but if the true density being estimated is indeed normal an optimal choice for h is

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{\frac{1}{5}} \approx 1.06\hat{\sigma}n^{-1/5}$$

which in the case of our dataset is equal to 2.2.

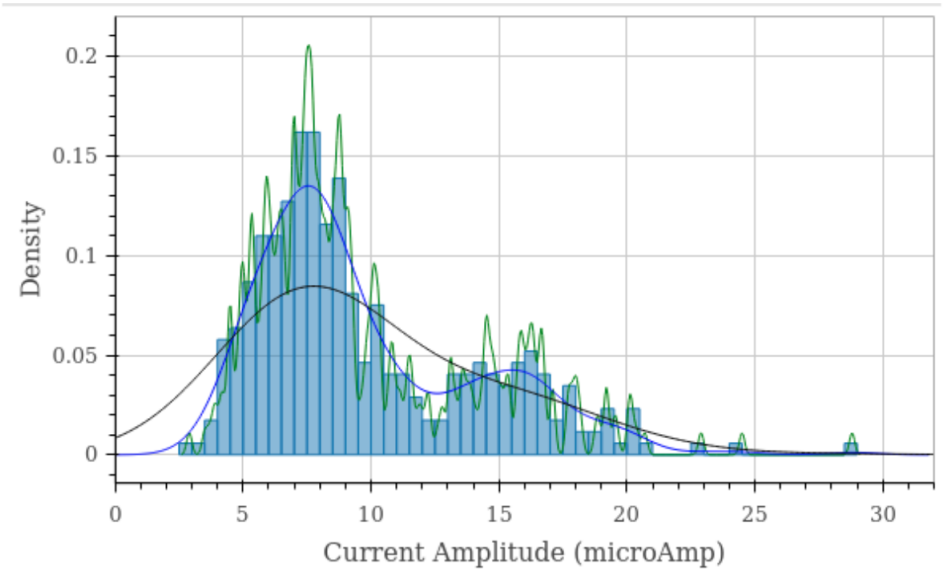

Figure 8: Kernel density estimates using bandwidth= 0.1, 1, 3

Now for each observation y_i, the Gaussian kernel estimator adds a normal distribution with mean y_i and standard deviation h to the model resulting in the addition of one parameter for each new observation, see Figure 9. Therefore, the number of parameters increases linearly with the number of data points n and, asymptotically, the number of parameters is infinite.

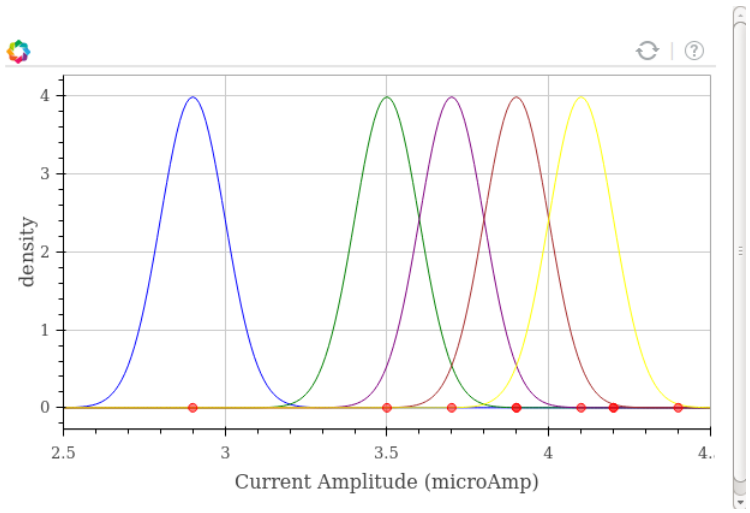

Figure 9: Gaussian ernels for the first 5 observations using bandwidth= 0.1

To determine whether the current was quantal or not, a bootstrap test of multimodality can be performed in which we try to see if large h is needed to make the estimated density unimodal. A test statistic suggested by (Davison & Hinkley, 1997) is to take

$$t = \min_{h>0}\{h : \hat{f}(y;h) \ \textit{is unimodal}\}$$

For the observed data, the minimum value of h that gives one mode is $h = 1.873$ and therefore

samples from the estimated density under the null distribution can be generated as

$$y_i^* \sim \text{Normal}(y_{I_i}, t^2) \quad i = 1, \ldots n$$

where I is random integer. The bootstrap statistics obtained from the conducting the test are $t_1* = 1$ with bias=0.02 and standard error=0.14. Thus, we can conclude that there is no evidence for multimodality under the null distribution.

We now contrast the kernel density approach with an alternative analysis that uses the univariate DPMN model with the following hyperparameters settings:

$$a_\alpha = 1 \quad b_\alpha = 1$$
$$\mu_0 = 10 \quad \kappa_0 = .05$$
$$\nu_0 = 4 \quad \lambda_0 = 1$$

The choice of hyperparameters basically specifies a DPMN model with a concentration parameter α that is random and a centering distribution parameter G_0 that is fixed.

Figure 10 summarizes the inference of the model fit. The density estimate of the model with random α and fixed G_0 is shown in yellow. Posterior inference on the parameters produced a mean of 8.72 for the number of clusters K and a mean of 1.49 for α. Plots of the posterior densities for the two parameters are shown side by side in Figure 11.

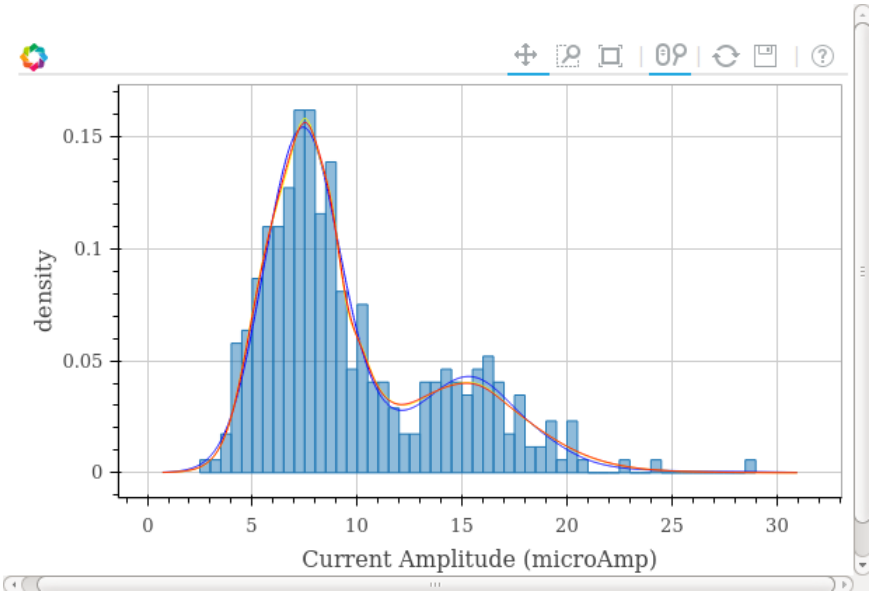

Figure 10: Density estimates using the DPMN model

Also shown in Figure 10 are two additional density estimates. One obtained from fitting an equivalent model but with fixed $\alpha = 1$, all-fixed model. The second (in blue) is obtained under a model in which both α and G_0 are random, all-random model. As can be seen, in all three

model specifications, the density estimates are very close to each other, all characterized by two modes.

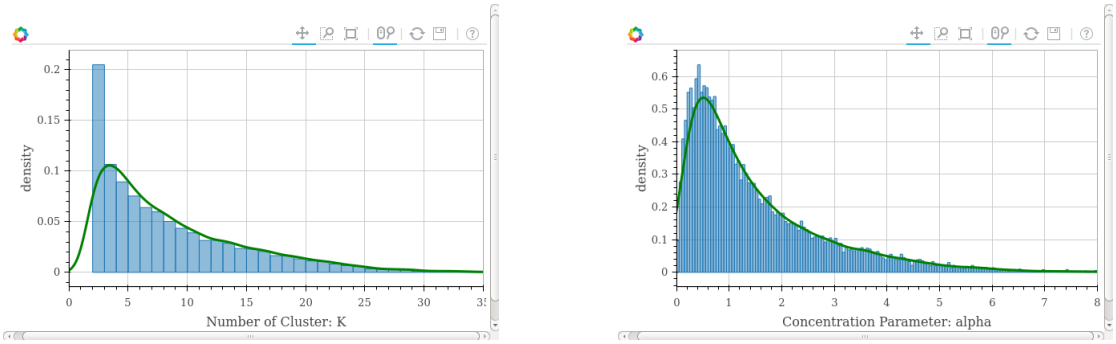

Figure 11: Posterior Densities of α and K

It is interesting to note that the distribution of the number of clusters parameter K is heavily skewed with a sharp mode at $K = 2$. The distribution gives us a better understanding of the possible number of clusters are and how likely they are. This a case where reliance of point estimates such maximum a posteriori probability, $K = 2$ or on point summaries such the posterior mean, 8.72, can be either insufficient or misleading. Instead, the entire distribution of the number of clusters provides an insightful representation of uncertainty in the analysis that can be updated as more data comes in.

Whereas a Gaussian parametric model for our dataset can have a single component whereas a non-parametric kernel density model can have as many components as a there are data points (i.e. 346), the DPMN model has an expected number of component $\mathbb{E}(K) = \alpha \log(n)$, which for the case of 346 observations and a $\alpha = 1$ is equal to $\mathbb{E}(K) = 5.85$. We see that by setting the value of α we can control the number of clusters that get introduced, all the while we are allowing for the possibility of an infinite number of components as more data comes in. At any rate, if the goal of the analysis is purely confined to density estimation, then the number of components should be of secondary concern since as can be seen from the model fits of the other two models, both had a lower posterior mean for K, namely, 6.3 for the all-random model, and 6.6 for the all-fixed model yet all three models considered had similar density estimates.

The model was fit using the DPpackage (Jara, Hanson, Quintana, Müller, & Rosner, 2011). The analysis script is provided in the appendix.

5 Question Part E: Model Extension

Question Many models rely on more than one distribution. Extend your model in part A to provide density estimates for a pair of distributions, as in the two-sample problem. Comment on the choices you have made in developing your model. What properties have you chosen to build into your model? What properties have you chosen to avoid building into your model?

The Two-Sample Problem Let $\mathcal{Y}_1 = (y_{11}, \ldots, y_{1i}, \ldots\ldots, y_{1I})$ be a random sample distribution F_{G_1} and $\mathcal{Y}_2 = (y_{21}, \ldots, y_{2j}, \ldots\ldots, y_{2J})$ be a random sample from distribution F_2. The two-sample problem is concerned with testing the hypothesis of the equality of two distribution $F_1 = F_2$ (Lehmann & Romano, 2006). When F_1 are F_2 normally distributed, the appropriate test for the the alternative hypothesis is a *Student's t test*. When no distributional assumptions are made several alternative hypotheses are usually considered. For example,

- $F_1 \neq F_2$ The two distributions are not equal
- $\forall y \quad F_1(y) = F_2(y - \Delta)$ The difference is due to an additive effect
- $\forall y \quad F_1(y) \leq F_2(y)$ The $y'_{2j}s$ are stochastically greater than the $y'_{1i}s$: The additive effect Δ is a non-negative function of y and the density $f_2(y)$ is shifted relative to $f_1(y)$

By the principle of invariance, the two-sample problem under the stochastic order restriction reduces to a rank test (e.g. Mann–Whitney U test or the normal-scores test). Now for the case of paired samples $\{(y_{1i}, y_{2i}\}_{i=1,\ldots,I}$ from bivariate distribution $F_{G_1,G_2}(y_1, y_2)$ we can have two additional alternative hypotheses

- $F_{G_1,G_2}(y_1, y_2) \neq F_{G_1,G_2}(y_2, y_1)$ $\quad F_{G_1,G_2}$ is not symmetric (i.e. there is an effect)
- $F_{G_1,G_2}(y_1, y_2) \neq F_1(y_1)F_2(y_2)$ $\quad$ The samples are dependent

Let F_1 and F_2 denote the distribution of particular biomarker for healthy and infected adults, respectively. If we do not make any assumption about nature of the effect or whether the samples are stochastically ordered and if we also do not model measurement errors due to the biomarker serology test, we can write the Dirichlet Process Mixture of Normals Model for the

two independent samples ($s = 1, 2$) as follows:

$$\begin{aligned} \alpha_s &\sim \text{Inv–Gamma}(a_{\alpha_s}, b_{\alpha_s}) \\ G_s \mid \alpha &\sim \text{DP}(\alpha_s G_{0s}) \\ \left(\mu_{st}, \sigma^2_{st}\right) \mid G_s &\sim G_s \\ y_{st} \mid \mu_{st}, \sigma^2_{st} &\sim \text{Normal}(\mu_{st}, \sigma^2_{st}) \\ G_{0s} = p\left(\mu_s, \sigma^2_s\right) &\equiv \text{Normal-Inv–}\chi^2\left(\mu_{0s}, \kappa_{0s}, \nu_{0s}, \lambda_{0s}\right) \end{aligned} \tag{11}$$

where

$$\theta_{ik} = \left(\mu_{ik}, \sigma^2_{ik}\right) \textit{ is the mean and variance of the kth component in the s'th group}$$

The model can also be written as

$$\begin{aligned} F_{G_s}(.) &= \int_{\Theta} \Phi(. \mid \theta) G_s(d\theta) \\ G_s \mid \alpha &\sim \text{DP}(\alpha_s G_{0s}) \end{aligned}$$

To incorporate the systematic error of measurement, we can assign the DPM prior on the mean components on similar to the semiparametric model of random effects discussed in the previous section. The specified model allows us assess the discriminatory ability of diagnostic marker and determine the amount of separation between the two distributions by means of ROC curve analysis.

However, the specified model assumes that the distributions of two populations are not constrained. The assumption implies, for example, that if the subpopulations that make up the mixture distribution of the infected population are in varying stages of disease severity, then it is possible that the early-stage cluster of infected individuals, as a whole, would have a lower serological score than a substantial portion of healthy adults. This can be fixed by incorporating a stochastic order restriction in the model specification as suggest by (Hanson, Kottas, &

Branscum, 2008).

$$
\begin{aligned}
\sigma^2 &\sim \text{Inv–Gamma}(a_0, b_0) \\
\alpha_H &\sim \text{Inv–Gamma}(a_{\alpha_G}, b_{\alpha_G}) \\
\alpha_G &\sim \text{Inv–Gamma}(a_{\alpha_H}, b_{\alpha_H}) \\
G, H \mid \alpha_H, \alpha_G &\sim \text{DP}(\alpha_H H_0)\,\text{DP}(\alpha_G G_0) \\
\phi_j \mid G &\sim G \quad j = 1, \ldots, n_2 \\
\mu_i \mid H &\sim H \quad i = 1, \ldots, n_1, \ldots, n_1 + n_2 \\
y_{2j} \mid \mu_{(n_1+j)}, \phi_j, \sigma^2 &\sim \text{Normal}(\max(\mu_{(n_1+j)}, \phi_j), \sigma^2) \quad j = 1, \ldots, n_2 \\
y_{1i} \mid \mu_i, \sigma^2 &\sim \text{Normal}(\mu_i, \sigma^2) \quad i = 1, \ldots, n_1 \\
G_0 &= p\left(\phi, \tau_G^2\right) \equiv \text{Normal-Inv–}\chi^2\left(\mu_G, \kappa_G, \nu_G, \lambda_G\right) \\
H_0 &= p\left(\mu, \tau_H^2\right) \equiv \text{Normal-Inv–}\chi^2\left(\mu_H, \kappa_H, \nu_H, \lambda_H\right)
\end{aligned}
\tag{12}
$$

The model can also be written as

$$
\begin{aligned}
F_1(.) &= \int \Phi(. \mid \mu, \sigma^2) H(d\mu) \\
F_2(.) &= \int \int \Phi(. \mid (\max(\mu, \phi), \sigma^2) G(d\phi) H(d\mu) \\
G, H \mid \alpha_H, \alpha_G &\sim \text{DP}(\alpha_H H_0)\,\text{DP}(\alpha_G G_0)
\end{aligned}
$$

Part II

Appendix: Julia and R Code

A Appendix: Part I

A.1 Forward Simulation Code in Julia

```
function sample_from_G₀!(μ,Σ,zᵢ)
            push!(Σ,rand(InverseWishart(ν₀, Λ₀)))
            push!(μ,rand(MultivariateNormal(μ₀,Σ[zᵢ]/κ₀)))
    return μ,Σ
end

function polya_urn(T,α,P,D)

    Z = Vector{Int64}(1) # intialize vector for cluster labels
    Z[1]=1 # assign first observation to first cluster
    # draw a set of parameters for the first observation
    Σ=[rand(InverseWishart(ν₀, Λ₀)) for j=1:1]
    μ=[rand(MultivariateNormal(μ₀,Σ[1]/κ₀)) for j=1:1]
    for t=2:T
        K=maximum(unique(Z)) # number of clusters
        C=[find(Z.==k) for k=1:K] # determine the clustering of  indices
        N=[size(C[k],1) for k=1:K] # compute the cardinality of each cluster
        pᵢ=float64([N,α]/ (α+t-1)) # compute all K+1 mixing proportions
        zᵢ=rand(Categorical(pᵢ),1)[1] # sample cluster label for observation t
        push!(Z,zᵢ) # append new label
        if zᵢ> K # if new cluster is created, draw parameters from G₀
            μ,Σ = sample_from_G₀!(μ,Σ,zᵢ)
        end
    end
    return Z,μ,Σ
end
```

Figure 12: Polya Urn Function

```
function generate_data_DPMG(T,α,P,D)

    Z,μ,Σ=polya_urn(T,α) # generate parameters and labels
    K=maximum(unique(Z)) # determine the number of clusters
    C=[find(Z.==k) for k=1:K] # determine the clustering indicies
    N=[size(C[k],1) for k=1:K] # compute the cardinality of each cluster
    y=[zeros(P,k) for k in N] #intialize data vector
    if D==1 # univariate case
        for k=1:K
            y[k]=rand(MultivariateNormal(repmat([μ[k]],P),Σ[k][1]*eye(P)),N[k])
        end
    else # multivariate case
        for k=1:K
            y[k]=rand(MultivariateNormal(μ[k],Σ[k]),N[k])
        end
    end
    # concatenate cluster label vector with data array
    data=vcat(vcat(hcat(y...)),Z[vcat(C...)]')
    data=data[:, randperm(T)]  #permute data
    return data,μ,Σ
end

using Gadfly, Distributions

T=300 # number of observations
P=3 # number of repeated measures
# D=1 # univariate case
D=1 # multivariate case
# set hyperparameters values
# note: when D=1, IW(ν,Ψ)=InvGamma(ν₀/2,Ψ/2 )
μ₀ = [0.  for i=1:D] # location: prior belief about mean vector
Λ₀ =0.02*eye(D) #inverse scale matrix: variation from the mean vector
ν₀= D+1 # degrees of freedom: confidence of prior belief
κ₀ = .04 # number of pseudo-observations ascribed to the prior
α=[0.01,1,10,100] # concentration parameters
```

Figure 13: Dirichlet Process Mixture: Data Simulation

```
p1=Dict()
for (index, value) in enumerate(α)
    data,μ,Σ=generate_data_DPMG(T,value,P,D)
    p1["$index"]=plot(x=data[1,:],y=data[2,:],color=data[P+1,:],
            Guide.xlabel("Dimension 1"),
            Guide.ylabel("Dimension 2"),
            Guide.title("Simulated Data for α=$value"),
            Guide.colorkey("Cluster"))
end

draw(PDF("clusters3.pdf", 8inch, 8inch),
        vstack(hstack(p1["1"],p1["2"]),hstack(p1["3"],p1["4"])))

p2=Dict()
for (index, value) in enumerate(α)
    data,μ,Σ=generate_data_DPMG(T,value,P,D)
    d =  DataFrame(stack(DataFrame(data[1:P,:])))
    d[:cluster]=vec(kron(data[P+1,:],repmat([1],P)')')
    p2["$index"]=plot(x=d[:value],y=d[:cluster],
            color=data[P+1,:],Geom.histogram,
            Guide.xlabel("Values"),
            Guide.ylabel("Count"),
            Guide.title("Simulated Data for α=$value"),
            Guide.colorkey("Cluster"))
end

draw(PDF("hist1.pdf", 8inch, 8inch),
        vstack(hstack(p2["1"],p2["2"]),hstack(p2["3"],p2["4"])))
```

Figure 14: Dirichlet Process Mixture: Plotting Functions

A.2 Kernel Density Estimation Scripts in R

```
library(rbokeh)
library(boot)
data(paulsen)
attach(paulsen)
y <- paulsen$y
f <-figure(
    width = 600, height = 400,xlim = c(0,5),tools = tools
  ) %>%
  ly_hist(y, breaks = 60, freq = FALSE) %>%
  x_axis(label = "Current Amplitude (microAmp)") %>%
  ly_density(y,color = "green",bw = 0.1) %>%
  ly_density(y,color = "blue",bw = 1) %>%
  ly_density(y,bw = 3)
f

xx <- seq(0,6,.01)
yy <- dnorm(xx,mean(paulsen$y),sd(paulsen$y))
dt <- sort(y)
dt <- dt[dt < 4.5]
yy1 <- dnorm(xx,dt[1],.1)
yy2 <- dnorm(xx,dt[2],.1)
yy3 <- dnorm(xx,dt[3],.1)
yy4 <- dnorm(xx,dt[4],.1)
yy5 <- dnorm(xx,dt[6],.1)

p1 <-figure(
    width = 600, height = 400,xlim = c(2.5,4.5)
  ) %>%
  ly_lines(xx, yy1,color = "blue") %>%
  ly_lines(xx, yy2,color = "green") %>%
  ly_lines(xx, yy3,color = "purple") %>%
  ly_lines(xx, yy4,color = "brown") %>%
  ly_lines(xx, yy5,color = "yellow") %>%
  ly_points(dt,0,color = "red",size = 7) %>%
  x_axis(label = "Current Amplitude (microAmp)") %>%
  y_axis(label = "density")
p1

n <- length(y)
sd(paulsen$y) * (4 / n * 3) ^ (.2)

#####################
#from Davison's Bootstrap methods and their application
peaks <- function(series,span = 3, ties.method = "first")
{
  if ((span <-
       as.integer(span)) %% 2 != 1)
    stop("'span' must be odd")
  z <- embed(series, span)
  s <- span %/% 2
  v <- max.col(z, ties.method = ties.method) == 1 + s
  pad <- rep(FALSE, s)
  result <- c(pad, v, pad)
  result
}

peak.test <- function(y, h) {
  dens <- density(y,bw = h,n = 100)
  sum(peaks(dens$y[(dens$x >= 0) & (dens$x <= 20)]))
}

peak.gen <- function(d,mle) {
  n <- mle[1] ;
  h <- mle [2]
  i <- sample(n,n,replace = T)
  d[i] + h * rnorm(n)
}
```

```
h = 1.873
peak.test(paulsen$y, h)

paulsen.boot <- boot(
  paulsen$y, peak.test, R = 999,
  sim = "parametric",
  ran.gen = peak.gen ,
  mle = c(nrow(paulsen),h),
  h = h
)
print(paulsen.boot)
plot(paulsen.boot)
```

A.3 DPMN Density Estimation Scripts in R

```
library(DPpackage)
library(rbokeh)
library(boot)
data(paulsen)
attach(paulsen)
current <- paulsen$y

state <- NULL
# MCMC parameters
nburn <- 1000
nsave <- 15000
nskip <- 10
ndisplay <- 100
mcmc <- list(
  nburn = nburn, nsave = nsave,nskip = nskip,ndisplay = ndisplay
)

prior1 <- list(
  a0 = 1, #alpha | a0, b0 ~ Gamma(a0,b0)
  b0 = 1,
  #G0 = N(mu| m1, (1/k0) Sigma) IW (Sigma | nu1, psi1
  m2 = 10, #m1 | m2, s2 ~ N(m2,s2)
  s2 = 100000,
  tau1 = 1, #k0 | tau1, tau2 ~ Gamma(tau1/2,tau2/2)
  tau2 = 100,
  nu1 = 4,
  nu2 = 4, #psi1 | nu2, psi2 ~ IW(nu2,psi2)
  psiinv2 = 1
)

prior2 <- list(
  a0 = 1, #alpha | a0, b0 ~ Gamma(a0,b0)
  b0 = 1,
  m1 = 10,
  k0 = .05,
  nu1 = 4,
  psiinv1 = 1
)

prior3 <- list(
  alpha = 1,
  m1 = 10,
  k0 = .05,
  nu1 = 4,
  psiinv1 = 1
)

fit1 <- DPdensity(
  y = current,prior = prior1,mcmc = mcmc,
  state = state,status = TRUE
)
fit2 <- DPdensity(
  y = current,prior = prior2,mcmc = mcmc,
  state = state,status = TRUE
)
fit3 <- DPdensity(y = current, prior = prior3, mcmc = mcmc,
  state = state,status = TRUE)

print(fit2)
plot(fit2,ask = FALSE,output = "param")
K <- fit2$save.state$thetasave[,4][nburn:nsave]
alpha <- fit2$save.state$thetasave[,5][nburn:nsave]

d1 <- figure(width = 600, height = 400) %>%
  ly_hist(current, breaks = 60, freq = FALSE) %>%
  ly_lines(fit1$x1, fit1$dens,color = "blue") %>%
  ly_lines(fit2$x1, fit2$dens,color = "yellow") %>%
```

```
  ly_lines( fit3$x1,  fit3$dens,color = "red") %>%
  x_axis(label = "Current Amplitude (microAmp)") %>%
  y_axis(label = "density")
d1

d2 <- figure(width = 600, height = 400,xlim = c(0,8)) %>%
  ly_hist(alpha, breaks = 200, freq = FALSE) %>%
  ly_density(
    alpha,color = "green",bw = .2,legend = "bandwidth=0.1",width = 3
  ) %>%
  x_axis(label = "Concentration Parameter: alpha") %>%
  y_axis(label = "density")
d2

d3 <- figure(width = 600, height = 400,xlim = c(0,35)) %>%
  ly_hist(K, breaks = 50, freq = FALSE) %>%
  ly_density(K,color = "green",legend = "bandwidth=0.1",width = 3) %>%
  x_axis(label = "Number of Cluster: K") %>%
  y_axis(label = "density")
d3
```

References

Davison, A. C., & Hinkley, D. V. (1997). *Bootstrap methods and their application* (Vol. 1). Cambridge university press.

Escobar, M. D., & West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the american statistical association*, *90*(430), 577–588.

Ghosh, J., & Ramamoorthi, R. (2006). *Bayesian Nonparametrics*. Springer Science & Business Media.

Hanson, T. E., Kottas, A., & Branscum, A. J. (2008). Modelling stochastic order in the analysis of receiver operating characteristic data: Bayesian non-parametric approaches. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *57*(2), 207–225.

Jara, A., Hanson, T., Quintana, F., Müller, P., & Rosner, G. (2011). DPpackage: Bayesian semi- and nonparametric modeling in R. *Journal of Statistical Software*, *40*(5), 1–30. Retrieved from `http://www.jstatsoft.org/v40/i05/`

Jaynes, E. T. (2003). *Probability Theory: The Logic of Science*. Cambridge University Press.

Kim, S., Smyth, P., & Stern, H. (2006). A nonparametric Bayesian approach to detecting spatial activation patterns in fMRI data. *Medical Image Computing and Computer*.

Lehmann, E. L., & Romano, J. P. (2006). *Testing statistical hypotheses*. Springer Science & Business Media.

McAuliffe, J. D., Blei, D. M., & Jordan, M. I. (2006). Nonparametric empirical bayes for the dirichlet process mixture model. *Statistics and Computing*, *16*(1), 5–14.

Muller, P., Erkanli, A., & West, M. (1996). Bayesian curve fitting using multivariate normal mixtures. *Biometrika*, 67–79.

Paulsen, O., & Heggelund, P. (1994). The quantal size at retinogeniculate synapses determined from spontaneous and evoked epscs in guinea-pig thalamic slices. *The Journal of physiology*, *480*(3), 505–511.

Persi Diaconis, D. F. (1986). On the consistency of bayes estimates. *The Annals of Statistics*, *14*(1), 1-26.

Petrone, S., & Veronese, P. (2002, feb). Non parametric mixture priors based on an exponential random scheme. *Statistical Methods & Applications*, *11*(1), 1–20.

Rasmussen, C. E. (1999). The infinite Gaussian mixture model. *NIPS*, *12*, 554–560.

Vlachos, A. (2008). Dirichlet process mixture models for verb clustering. *Proceedings of the ICML*.

Wood, F., & Black, M. J. (2008, aug). A nonparametric Bayesian alternative to spike sorting.

Journal of neuroscience methods, *173*(1), 1–12.