BAYESIAN NONPARAMETRIC PROBABILISTIC METHODS IN MACHINE LEARNING

by

Justin C. Sahs

APPROVED BY SUPERVISORY COMMITTEE:

Bhavani Thuraisingham, Co-Chair

Latifur Khan, Co-Chair

Zhiqiang Lin

Weili Wu

BAYESIAN NONPARAMETRIC PROBABILISTIC METHODS IN MACHINE LEARNING

by

JUSTIN C. SAHS, BS, MS

DISSERTATION

Presented to the Faculty of

The University of Texas at Dallas

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY IN

COMPUTER SCIENCE

THE UNIVERSITY OF TEXAS AT DALLAS

December 2018

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Latifur Khan, for supporting my research while leaving me free to explore my interests. Without him, this dissertation—and the abundant learning it has required—would not have been possible. Additionally, I would like to thank Dr. Bhavani Thuraisingham, Dr. Zhiqiang Lin, and Dr. Weili Wu for forming the rest of my committee.

September 2018

BAYESIAN NONPARAMETRIC PROBABILISTIC METHODS IN MACHINE LEARNING

Justin C. Sahs, PhD
The University of Texas at Dallas, 2018

Supervising Professors: Bhavani Thuraisingham, Co-Chair
Latifur Khan, Co-Chair

Many aspects of modern science, business and engineering have become data-centric, relying on tools from Artificial Intelligence and Machine Learning. Practitioners and researchers in these fields need tools that can incorporate observed data into rich models of uncertainty to make discoveries and predictions. One area of study that provides such models is the field of Bayesian Nonparametrics. This dissertation is focused on furthering the development of this field.

After reviewing the relevant background and surveying the field, we consider two areas of structured data:

- We first consider relational data that takes the form of a 2-dimensional array—such as social network data. We introduce a novel nonparametric model that takes advantage of a representation theorem about arrays whose column and row order is unimportant. We then develop an inference algorithm for this model and evaluate it experimentally.

- Second, we consider the classification of streaming data whose distribution evolves over time. We introduce a novel nonparametric model that finds and exploits a dynamic hierarchical structure underlying the data. We present an algorithm for inference in

this model and show experimental results. We then extend our streaming model to handle the emergence of novel and recurrent classes, and evaluate the extended model experimentally.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In many fields—including science, business, government, and medicine—there is a constant need to make decisions under uncertainty. For example, a business making decisions about what products to produce and market necessarily must deal with the uncertainty of predicting customer needs and desires. However, there is usually data available—often in large amounts—to inform such decisions. In recent years, the field of machine learning has supplied many of the computational techniques used to turn these data sets into actionable intelligence. In practice, machine learning is essentially a field of applied statistics with an emphasis on computational aspects, with many of its techniques founded in statistical theory and probability.

An increasingly popular approach to statistics and machine learning is the *Bayesian* paradigm (see, e.g., (Gelman et al. 2004; Box and Tiao 2011; Bernardo and Smith 1994; Barber 2013; Thibaux 2008)), wherein probability is used to directly model uncertainty. The general pattern of a Bayesian technique involves:

1. Defining a probability distribution over the observed data, called the *likelihood*, which generally depends on unobserved parameters
2. Defining a probability distribution over these unobserved parameters, called the *prior*
3. Observing the data and updating the prior to produce the *posterior* distribution

The likelihood and prior should ideally encode whatever is known before the data is seen, i.e., the scientist's prior beliefs.

This approach is very general, and even methods that are not traditionally considered Bayesian can be re-cast into the Bayesian paradigm. For example, Neural Networks can be trained using Bayesian techniques (Neal 1993a, 1992; Andrieu, Freitas, and Doucet 1999; Solla

and Winther 1998); Support Vector Machines can be cast as a probabilistic model (Franc, Zien, and Schölkopf 2011), or used as part of a Bayesian model (Zhu, Chen, and Xing 2014); k-nearest neighbor models can also be made more robust via Bayesian techniques (Guo and Chakraborty 2009).

Historically, the process of computing posterior distributions—also called *inference*—has been prohibitively expensive (Green et al. 2015; Betancourt 2017b), such that only uselessly simple models were tractable. However, advances in inference techniques (Andrieu et al. 2003) have enabled much more expressive models to be used. Thus, the Bayesian paradigm has become a practical option worth considering. Indeed, there are many arguments in favor of the Bayesian approach. On the practical level, it is consistent with the "common-sense interpretation of statistical conclusions" and has "flexibility and generality [which] allow it to cope with very complex problems" (Gelman et al. 2004). Indeed, "[t]he only relevant thing is uncertanty" (de Finetti 1974): whether a process is deterministic, or whether more information about it is available to other people, any decision or conclusion a scientist must make can only depend on the information available to them, and their degree of certainty. Additionally, only the Bayesian approach permits the use of information known prior to the current experiment, and gives a formal method of combining such knowledge with the current observations (Ferguson 1983). See also (Bernardo and Smith 1994; Barber 2013; Gelman 2008; Gelman and Robert 2013; Morey et al. 2015; Hauer 2004; Carlin and Louis 2009) for further discussion on the merits of Bayesianism.

The remainder of this dissertation is organized as follows. In Chapter 2, we review the theoretical background underlying Bayesian methods. In Chapter 3, we review nonparametric techniques. In Chapter 4, we review the computational techniques for performing inference in our models. In Chapter 5, we present a Bayesian nonparametric model of relational data such as social networks. In Chapter 6, we present a Bayesian nonparametric model of streaming data whose distribution can change over time, and in Chapter 7, we extend this model to handle novel and recurrent classes. Finally, in Chapter 8, we present our conclusions.

CHAPTER 2

BAYESIAN PROBABILITY THEORY

To gain a deep understanding of Bayesian techniques, it is helpful to explore the theoretical underpinings of modern probability theory. Here we present a shallow review of the most relevant aspects of modern axiomatic probability. We assume familiarity with the standard concepts and notation of basic set theory. See, e.g., (Roitman 1990) or the first chapter of (Munkres 2000) for an introduction to this topic.

2.1 Notation and Preliminaries

We start with some preliminary definitions and notation.

We abbreviate the phrase "if and only if" by "iff." We denote by $\mathbb{R}_+$ the set $[0, \infty)$ of positive real numbers, and by $\overline{\mathbb{R}}$ the set $\mathbb{R} \cup \{-\infty, \infty\}$. For any two functions $f, g : A \to B$, we write $f \equiv g$ if $f(x) = g(x)$ for any $x \in A$. We may also denote a constant function by $f \equiv b$ for some $b \in B$. We denote by $a \propto b$ that $a = cb$ for some constant c, i.e., that a is "proportional to" b.

Given some ordered set A, the supremum, $\sup A$, is the smallest upper bound on A, and the infimum, $\inf A$, is the largest lower bound. For countable sets, these are the same as maximum and minimum, respectively, but are more general. For example, the open interval $(0, 1)$ has no maximum or minimum, but has supremum 1 and infimum 0.

We denote by $\Gamma(\cdot)$ the Gamma function

$$\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx,$$

which generalizes the factorial to real (or complex) inputs. Relatedly, we denote by $B(\cdot, \cdot)$ the Beta function

$$B(x, y) = \int_0^1 t^{x-1}(1-t)^{y-1} dt = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}.$$

2.2 Topological Spaces

Let Ω be some arbitrary set. A *topology* $\mathcal{T}$ on Ω is a collection of subsets of Ω such that:

- $\emptyset \in \mathcal{T}$ and $\Omega \in \mathcal{T}$
- Let $\mathcal{U}$ be an arbitrary subcollection of $\mathcal{T}$; then $\cup_{T \in \mathcal{U}} T \in \mathcal{T}$
- Let $\mathcal{U}$ be a finite subcollection of $\mathcal{T}$; then $\cap_{T \in \mathcal{U}} T \in \mathcal{T}$

That is, $\mathcal{T}$ is closed under arbitrary (even uncountable) unions and finite intersections. Then, we call $(\Omega, \mathcal{T})$ a *topological space*, and the sets $T \in \mathcal{T}$ are called the *open sets* of Ω. The complement of any open set is called *closed*; a set can be both open and closed. If $(S, \mathcal{S})$ and $(T, \mathcal{T})$ are topological spaces, then a mapping $f : S \to T$ is called *continuous* if for any $A \in \mathcal{T}$, $f^{-1}A \in \mathcal{S}$, i.e., it maps open sets to open sets. See (Munkres 2000) for a thorough treatment of topology.

2.3 Measure Theory

A *σ-algebra* $\mathcal{A}$ on Ω is a nonempty collection of subsets of Ω such that:

- If $A \in \mathcal{A}$, then the complement of A, $A^c \in \mathcal{A}$
- If $\{A_1, A_2, \ldots\}$ is a countable collection of elements of $\mathcal{A}$, then $\cup_i A_i \in \mathcal{A}$ and $\cap_i A_i \in \mathcal{A}$

That is, $\mathcal{A}$ is closed under complementation, countable unions and countable intersections. These properties imply that $\emptyset \in \mathcal{A}$, and $\Omega \in \mathcal{A}$: if $A \in \mathcal{A}$, then A^c, $A \cap A^c = \emptyset$ and $\cup A^c = \Omega$ are all in $\mathcal{A}$. The pair $(\Omega, \mathcal{A})$ is called a *measurable space*. A *sub-σ-algebra* of $\mathcal{A}$ is some σ-algebra $\mathcal{B}$ on Ω such that $\mathcal{B} \subseteq \mathcal{A}$.

If we have some arbitrary collection $\mathcal{C}$ of subsets of Ω, then there exists a smallest σ-algebra containing $\mathcal{C}$, which we call the σ-algebra *generated by* $\mathcal{C}$ and denote by $\sigma(\mathcal{C})$. In particular,

if $\mathcal{C}$ is a topology on Ω, we call $\sigma(\mathcal{C})$ the *Borel σ-algebra* on Ω, and denote it $\mathcal{B}(\Omega)$ when the topology is implicit. A *Borel space* is a measurable space S such that there exists a bijection f between S and some $T \in \mathcal{B}[0,1]$ such that both f and f^{-1} are measureable.

The *Lebesque measure* λ is a measure on $(\mathbb{R}, \mathcal{B}(\mathbb{R}))$ such that $\lambda[a,b] = b - a$.

If we have two measurable spaces, $(\Omega, \mathcal{A})$ and $(\Xi, \mathcal{B})$, then a mapping $f : \Omega \to \Xi$ is said to be *measurable* if for every $A \in \mathcal{B}$, $f^{-1}A \in \mathcal{A}$, so that measurable mappings are to measure theory what continuous mappings are to topology (indeed, any continuous mapping between topological spaces is measurable with respect to their Borel σ-algebras (Kallenberg 1997, Lemma 1.5)). The set $\{f^{-1}B | B \in \mathcal{B}\}$ is a σ-algebra on Ω, called the σ-algebra *generated by* f. Note that by the definition of measurability above, $\sigma(f) \subseteq \mathcal{A}$.

A *measure* on $(\Omega, \mathcal{A})$ is a function $\mu : \mathcal{A} \to [0, \infty]$ such that:

- $\mu\emptyset = 0$
- *countable additivity*: $\mu \cup_i A_i = \sum_i \mu A_i$, for any collection $\{A_i\}$ of disjoint elements of $\mathcal{A}$

Then, $(\Omega, \mathcal{A}, \mu)$ is called a *measure space*.

A set $A \in \mathcal{A}$ is called *null* if $\mu A = 0$. The *support* of μ, $\operatorname{supp} \mu$ is the smallest closed $A \in \mathcal{A}$ such that μA^c is null. If there exists some countable partition (A_n) of Ω such that $\mu A_n < \infty$ for all n, then μ is said to by *σ-finite*. If a property holds for all $\omega \in \operatorname{supp} \mu$ (but may fail to hold on some null set A), it is said to hold *almost everywhere* with respect to μ, or hold *μ-a.e.*. If $\operatorname{supp} \mu = \{\omega\}$ for some ω, then $\mu = c\delta_\omega$, where $\delta_\omega(A) = \mathbb{1}_A(\omega)$ is a *Dirac measure*, and $c > 0$. In general, if $\{\omega\} \in \mathcal{A}$ and $\mu\{\omega\} > 0$, then ω is called an *atom* of μ. If μ has no atoms, it is said to be *diffuse* or *nonatomic*.

Proposition 2.1. *Let $\mathcal{C} = C_1, C_2, \ldots$ be some partition of Ω, and let $f : \Omega \to \mathbb{R}$ be $\sigma(\mathcal{C})$-measurable. Then, f is constant over each C_i.*

Proof. By measurability, we have that $f^{-1}B \in \sigma(\mathcal{C})$ for any $B \in \mathcal{B}$. In particular, $f^{-1}\{x\} \in \sigma(\mathcal{C})$ for $x \in \mathbb{R}$, so $f^{-1}\{x\}$ is either some C_i or a countable union of C_is. □

For any $p > 0$, we denote by $L^p(\Omega, \mathcal{A}, \mu)$ the class of all measurable functions $f : \Omega \to \mathbb{R}$ such that

$$\|f\|_p = (\mu|f|^p)^{\frac{1}{p}} = \left(\int |f(\omega)|^p \mu(d\omega) \right)^{\frac{1}{p}} < \infty$$

$\|\cdot\|_p$ is called the *L^p-norm*. If $(\Omega, \mathcal{A}, \mu) = (\mathbb{R}, \mathcal{B}(\mathbb{R}), \lambda)$ or is obvious from context, we just write L^p.

We now define the integral of the measurable function $f : \Omega \to \mathbb{R}$:

$$\mu f = \int f d\mu = \int f(\omega) \mu(d\omega)$$

as follows. First, any non-negative real-valued measurable function f is the limit of a sequence (f_k) of *simple* measurable functions : $\Omega \to \mathbb{R}_+$ (Kallenberg 1997, Lemma 1.11), where a *simple* function is of the form

$$\sum_{i=1}^{n} c_i \mathbb{1}_{A_i},$$

where $c_i \in \mathbb{R}_+$, $A_i \in \mathcal{A}$, and

$$\mathbb{1}_{A_i}(x) = \begin{cases} 1 & x \in A_i \\ 0 & \text{otherwise} \end{cases}$$

is an *indicator function* for A_i. Then, we can define the integral of a simple non-negative function to be

$$\mu \left(\sum_{i=1}^{n} c_i \mathbb{1}_{A_i} \right) = \sum_{i=1}^{n} c_i \mu A_i.$$

Then, by (Kallenberg 1997, Theorem 1.19), if the sequence (f_k) converges to f, then μf_k converges to μf. Finally, Lebesque's dominated convergence theorem (Kallenberg 1997, Theorem 1.21) gives conditions under which this result extend to arbitrary real-valued measurable functions.

2.4 Probability Theory

If $\mu(\Omega) = 1$, then μ is a *probability measure*, and $(\Omega, \mathcal{A}, \mu)$ is a *probability space*, and the elements of $\mathcal{A}$ are called *events*. If a property holds μ-a.e., and μ is a probability measure, then the property is said to hold *almost surely (a.s.)* instead.

A function $k : S \times \mathcal{T} \to [0, \infty]$ is called a *kernel* from $(S, \mathcal{S})$ to $(T, \mathcal{T})$ if $k(\cdot, B)$ is $\mathcal{S}$-measurable for any $B \in \mathcal{T}$, and $k(s, \cdot)$ is a measure on $(T, \mathcal{T})$ for any $s \in S$. Equivalently, we can consider k to be a measurable function $k : S \to \mathcal{M}(T)$, where $\mathcal{M}(T)$ is the space of all σ-finite measures on $(T, \mathcal{T})$. If $k(s, T) = 1$ for all s, so that $k(s, \cdot)$ is a probability measure, we call k a *probability kernel*.

2.4.1 Random Elements, Distributions, and Densities

Rather than define a probability measure $(\Omega, \mathcal{A}, P)$ for each set we wish to sample from, we often fix a probability space, e.g., $([0, 1], \mathcal{B}([0, 1]), \lambda)$, and work with measurable functions from this space. A *random element* of some set S is a measurable function $\xi : \Omega \to S$. When $S = \mathbb{R}$, we use the term *variable* rather than *element*; when S is a space of functions, we call ξ a *stochastic process*. Then, for $B \in \mathcal{S}$, we may denote the event $\{\xi \in B\} = \xi^{-1}B$, and

$$P\{\xi \in B\} = P(\xi^{-1}B) = (P \circ \xi^{-1})B,$$

where we call $P \circ \xi^{-1}$ the *distribution* of ξ. We denote by $\xi \overset{d}{=} \eta$ the fact that $P \circ \xi^{-1} = P \circ \eta^{-1}$, i.e., ξ and η have the same distribution.

Two events $A, B \in \mathcal{A}$ are said to be *independent* (denoted $A \perp\!\!\!\perp B$) if $P(A \cap B) = PA \times PB$. Likewise, two random elements $\xi : \Omega \to S$ and $\eta : \Omega \to T$ are said to be independent, denoted $\xi \perp\!\!\!\perp \eta$ if, for any $A \in \sigma(\xi)$ and $B \in \sigma(\eta)$, $A \perp\!\!\!\perp B$.

The *expected value* of a random variable is defined as

$$\mathbb{E}\xi = \int \xi dP = \int x(P \circ \xi^{-1})(dx).$$

Likewise, for any measurable $f : S \to \mathbb{R}$,

$$\mathbb{E}f(\xi) = \int f(\xi)dP = \int f(s)(P \circ \xi^{-1})(ds) = \int x(P \circ (f \circ \xi)^{-1})(dx),$$

where ξ takes values in an arbitrary S. The *variance* of a random variable is defined as

$$\text{var}(\xi) = \mathbb{E}[\xi - \mathbb{E}[\xi]]^2 = \mathbb{E}[\xi^2] - (\mathbb{E}[\xi])^2.$$

Let $f : \Omega \to \mathbb{R}_+$ be measurable. Then, for any measure μ, we may define $\nu = f \cdot \mu$ by

$$\nu A = (f \cdot \mu)A = \mu(\mathbb{1}_A f) = \int_A f d\mu, \tag{2.1}$$

and call f the *μ-density* of ν. This is also denoted $f = d\nu/d\mu$.

Lemma 2.1. *Let $f, g : \Omega \to \mathbb{R}_+$ be measurable functions on some measure space $(\Omega, \mathcal{A}, \mu)$, and let ν be some other measure on $(\Omega, \mathcal{A})$. Then,*

$$\int g(\omega)\nu(d\omega) = \int f(\omega)g(\omega)\mu(d\omega)$$

iff $f = d\nu/d\mu$.

Proof. First, let $g = \mathbb{1}_A$ for some $A \in \mathcal{A}$. Then, the result is just Equation (2.1), so f must be a density. Using the linearity and dominated convergence techniques at the end of Section 2.3, we extend to arbitrary functions g. □

If $f = d(P \circ \xi^{-1})/dP$, for some $\xi : \Omega \to \Omega$, we refer to $f : S \to \mathbb{R}_+$ as just the *density* of ξ, and denote it $p(x)$ for $x \in \Omega$. If $p(x)$ has other parameters, we may indicate this by separating the other parameters with a semicolon, e.g., $p(x; a, b)$

We write $\xi \sim f$ or say that ξ is *drawn from* f to denote that f is the distribution or density of ξ. For a collection of random elements, we write $\xi_i \overset{ind}{\sim} f_i$ to give the distribution or density of each ξ_i and to declare that the ξ_i are independent of each other. Similarly, we write $\xi_i \overset{iid}{\sim} f$ to show that the ξ_i are independent and identically distributed with distribution or density f.

Some common distributions we will need:

- The *Uniform* distribution: $\xi \sim UA$ if $\xi : \Omega \to A$, A is some set, and

$$p(x) = \frac{1}{|A|}$$

- The *Beta* distribution: $\xi \sim Beta(\alpha, \beta)$ if $\xi : \Omega \to [0, 1]$, $\alpha, \beta > 0$, and

$$p(x; \alpha, \beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha, \beta)}$$

(if $\alpha = \beta = 1$, this reduces to $U(0, 1)$)

- The *Dirichlet* distribution: $\xi \sim Dirichlet(\vec{\alpha})$ if ξ is a random element of the d-simplex (the set of d-dimensional real vectors whose elements sum to 1), $\vec{\alpha} \in \mathbb{R}^d$, and

$$p(x; \vec{\alpha}) = \Gamma(\sum_{i=1}^{d} \alpha_i) \prod_{i=1}^{d} \frac{x_i^{\alpha_i - 1}}{\Gamma(\alpha_i)}$$

- The *Exponential* distribution: $\xi \sim Exp(\lambda)$ if $\xi : \Omega \to \mathbb{R}_+$, $\lambda > 0$, and

$$p(x; \lambda) = \lambda e^{-\lambda x}$$

- The *Gamma* distribution: $\xi \sim G(k, \theta)$ if $\xi : \Omega \to \mathbb{R}_+$, $k, \theta > 0$, and

$$p(x; k, \theta) = \frac{x^{k-1} e^{-\frac{x}{\theta}}}{\Gamma(k)\theta^k}$$

- The *Normal* distribution: $\xi \sim N(\mu, \sigma^2)$ if $\xi : \Omega \to \mathbb{R}$, $\mu \in \mathbb{R}$, $\sigma > 0$, and

$$p(x; \mu, \sigma^2) = \frac{\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}{\sqrt{2\pi\sigma^2}}$$

- The *Multivariate Normal* distribution: $\xi \sim N(\vec{\mu}, \vec{\Sigma})$ if $\xi : \Omega \to \mathbb{R}^d$, $\vec{\mu} \in \mathbb{R}^d$, $\vec{\Sigma} \in \mathbb{R}^{d \times d}$ and is positive semi-definite, and

$$p(\vec{x}; \vec{\mu}, \vec{\Sigma}) = \frac{\exp\left(-\frac{1}{2}(\vec{x} - \vec{\mu})^\top \vec{\Sigma}^{-1} (\vec{x} - \vec{\mu})\right)}{\sqrt{2\pi|\vec{\Sigma}|}}$$

- The *Poisson* distribution: $\xi \sim Poisson(\lambda)$ if $\xi : \Omega \to \mathbb{Z}_+$, $\lambda > 0$, and

$$p(n; \lambda) = e^{-\lambda} \frac{\lambda^n}{n!}$$

- The *Bernoulli* distribution: $\xi \sim Bernoulli(p)$ if $\xi : \Omega \to \{0, 1\}$, $p \in [0, 1]$, and

$$p(x; p) = \begin{cases} 1 - p & x = 0 \\ p & x = 1 \end{cases}$$

2.4.2 Product Measures and Conditional Distributions

Let $(S, \mathcal{S})$ and $(T, \mathcal{T})$ be measureable spaces. We can define the *product space* of S and T to be the measureable space $(S \times T, \mathcal{S} \otimes \mathcal{T})$, where $\mathcal{S} \otimes \mathcal{T}$ is the *product σ-algebra* generated by the set $\{A \times T | A \in \mathcal{S}\} \cup \{B \times S | B \in \mathcal{T}\}$. We now present an important result about measures on such product space:

Theorem 2.1 (Fubini)**.** *Let $(S, \mathcal{S}, \mu)$ and $(T, \mathcal{T}, \nu)$ be σ-finite measure spaces. Then, there exists a unique measure $\mu \otimes \nu$ on $(S \times T, \mathcal{S} \otimes \mathcal{T})$ such that*

$$(\mu \otimes \nu)(A \times B) = \mu A \cdot \nu B$$

for any $A \in \mathcal{S}$, $B \in \mathcal{T}$. Additionally, for any measurable function $f : S \times T \to \mathbb{R}$ with $(\mu \otimes \nu)|f| < \infty$,

$$(\mu \otimes \nu) f = \int \mu(ds) \int f(s,t)\nu(dt) = \int \nu(dt) \int f(s,t)\mu(ds)$$

For any sub-σ-algebra $\mathcal{F} \subseteq \mathcal{A}$, we define the *conditional expectation* $\mathbb{E}[\cdot|\mathcal{F}] : L^1 \to L^1(\mathcal{F})$ to be the a.e.-unique linear operator such that, for any random variable $\xi \in L^1$, and any $A \in \mathcal{F}$

$$\mathbb{E}[\mathbb{E}[\xi|\mathcal{F}]\mathbb{1}_A] = \mathbb{E}[\xi \mathbb{1}_A]. \tag{2.2}$$

See (Kallenberg 1997, Theorem 5.1) or (Billingsley 2012, Section 34) for proofs of the existance of this operator. To explore the meaning of this operator, consider some experiment that yields information about ξ. The possible outcomes of such an experiment induce some σ-algebra $\mathcal{F}$. At one extreme, if the experiment yields no information, then $\mathcal{F} = \{\emptyset, \Omega\}$, and we have $\mathbb{E}[\xi|\{\emptyset, \Omega\}] \equiv \mathbb{E}[\xi]$, i.e., we learn nothing and the conditional expectation is just the expectation. At the other extreme, $\mathcal{F} = \mathcal{A}$ and $\mathbb{E}[\xi|\mathcal{A}] = \xi$, i.e., the experiment tells us the exact value of ξ, so the conditional expection is naturally just ξ itself. Conditioning on a

random element η is defined as conditioning on $\sigma(\eta)$. Applying Proposition 2.1, there exists some constant $\alpha_i \in \mathbb{R}$ where

$$\mathbb{E}[\xi|\mathcal{F}](\omega) = \alpha_i$$

for all $\omega \in C_i$. Combining with Equation (2.2),

$$\mathbb{E}[\mathbb{E}[\xi|\mathcal{F}]\mathbb{1}_{C_i}] = \mathbb{E}[\alpha_i \mathbb{1}_{C_i}] = \alpha_i P(C_i) = \mathbb{E}[\xi \mathbb{1}_{C_i}],$$

so $\alpha_i = \mathbb{E}[\xi \mathbb{1}_{C_i}]/P(C_i)$.

Then, consider the random variable $\xi = \mathbb{1}_A$, which is 1 with probability $P(A)$. We define the *conditional probability* of A as $P[A|\mathcal{F}] = \mathbb{E}[\mathbb{1}_A|\mathcal{F}]$. Applying the above argument, when $\mathcal{F} = \sigma(\mathcal{C})$ for some partition $\mathcal{C}$ of Ω,

$$P[A|\mathcal{F}](\omega) = \frac{\mathbb{E}[\mathbb{1}_A \mathbb{1}_{C_i}]}{P(C_i)} = \frac{P(A \cap C_i)}{P(C_i)}$$

for $\omega \in C_i$.

The function $P[\xi \in \cdot|\eta]$, is called the *conditional distribution* of $\xi : \Omega \to S$, given $\eta : \Omega \to T$, which is a probability kernel from $(T, \mathcal{T})$ to $(S, \mathcal{S})$, i.e., for each $t \in T$, $P[\xi \in \cdot|\eta](t, \cdot)$ is a measure on $(S, \mathcal{S})$. In particular, $P[\xi \in \cdot|\eta](t, A) = P[\xi \in A|\eta](t) = \mathbb{E}[\mathbb{1}_{\xi^{-1}A}|\sigma(\eta)](\eta^{-1}t)$. See (Faden 1985) for necessary and sufficient conditions for this kernel to exist.

We can now define *conditional independence*: two events A and B are *conditionally independent given* $\mathcal{F}$ if

$$P(A \cap B|\mathcal{F}) = P(A|\mathcal{F})P(B|\mathcal{F}),$$

and the definition extends to random elements just as for unconditional independence.

Let $(\xi, \eta) : \Omega \to \Omega \times \Omega$ be a random vector with density $f(x, y)$. Equivalently, we say ξ and η are random elements with *joint density* $f(x, y)$. Then, by Theorem 2.1, $f(y) = \int_S f(x, y)(P \circ \xi^{-1})(dx)$ is the density of η alone, and $f(x) = \int_T f(x, y)(P \circ \eta^{-1})(dy)$ is the density of ξ. These are called the *marginal densities* of $f(x, y)$.

Theorem 2.2. *Let*

$$f(x|y) = \begin{cases} \frac{f(x,y)}{f(y)} & f(y) \neq 0 \\ 0 & \textit{otherwise.} \end{cases}$$

Then, $f(x|y)$ is the density of the conditional distribution of ξ given η, that is,

$$f(\cdot|y) = \frac{d(P[\xi \in \cdot|\eta](y))}{d(P \circ \xi^{-1})},$$

so that

$$\int_A f(x|y)(P \circ \xi^{-1})(dx) = P[\xi \in A|\eta](y)$$

for any $A \in \mathcal{S}$ and any $y \in T$.

Proof. Let $g : \Omega \to \mathbb{R}_+$ be some measureable function. We first prove that Equation (2.2) holds: that is, for any $B \in \mathcal{A}$,

$$\mathbb{E}\left[\int g(x)f(x|\eta)(P \circ \xi^{-1})(dx)\mathbb{1}_B(\eta)\right] = \mathbb{E}\left[g(\xi)\mathbb{1}_B(\eta)\right]$$

Note that the left expectation is over η alone, whereas the right expectation is over both ξ and η. Expanding both sides, and cancelling out the $f(y)$ factors on the left, we get

$$\int \left[\int g(x)f(x,y)(P \circ \xi^{-1})(dx)\mathbb{1}_B(y)\right] P(dy) = \int\int \left[g(x)\mathbb{1}_B(y)\right] f(x,y)P(dx,dy)$$

which follows directly from Theorem 2.1, with $\mu = P \circ \xi^{-1}$ and $\nu = P \circ \xi^{-1}$.

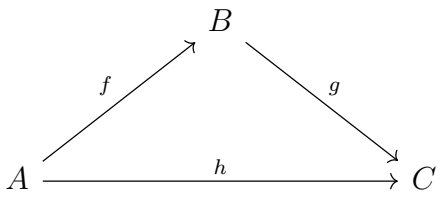
Thus, $\int g(x)f(x|\eta)(P \circ \xi^{-1})(dx)$ is the conditional expectation of $g(x)$ with respect to η. Applying $g(x) = \mathbb{1}_A(x)$ yields the result. □

Applying this, we get an immediate corollary:

Corollary 2.1 (Bayes' Theorem)**.** *For any random vector (ξ, η) with density $f(x, y)$, we have*

$$f(x|y) = \frac{f(y|x)f(x)}{f(y)}$$

The next result is best given in the visual language of category theory (see, e.g., (Adámek, Herrlich, and Strecker 1990; Baez and Stay 2010; Mac Lane 1978) for an introduction to category theory). A *category* is a class of *objects* together with *morphisms* or *arrows* between them. The objects represent "things," and the morphisms represent "ways to go between things" (Baez and Stay 2010). Importantly, morphisms can be composed: if f and g are arrows, so is the composition $g \circ f$. Often, category theorists use graphical *diagrams* to represent a category. For example,



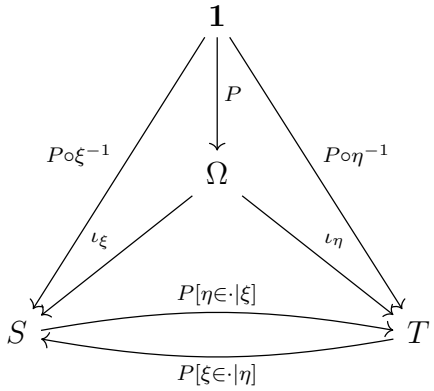
represents a simple category with three objects and three morphisms. A diagram is said to *commute* if every path between two objects is equal. In this example, the diagram commutes if $h = g \circ f$.

(Culbertson and Sturtz 2013) gives a categorical treatment of Bayesian probability, where the objects are certain measurable spaces which admit conditional probabilities, and the morphisms are certain probability kernels. The composition of kernels, denoted by $g \circ_k f$ is given by

$$(g \circ_k f)(x, A) = \int g(y, A) f(x, dy).$$

Finally, let us introduce the notation $\iota_f(\omega, A) = \mathbb{1}_A(f(\omega))$, which promotes the measurable function $f : \Omega \to S$ to a kernel from Ω to S. With this, we can present our next result:

Theorem 2.3. *The diagram*



commutes and

$$\int_A P[\eta \in \cdot|\xi](\cdot, B)d(P \circ \xi^{-1}) = P(\{\xi \in A\} \cap \{\eta \in B\}) = \int_B P[\xi \in \cdot|\eta](\cdot, A)d(P \circ \eta^{-1})$$

for any $A \in \mathcal{S}$ and $B \in \mathcal{T}$, provided the conditional probabilities exist.

Here, **1** represents the trivial measureable set with one element, so that a probability kernel from **1** to S is just a probability measure on S. (Culbertson and Sturtz 2013) proved a similar theorem where $\Omega = S \times T$.

Proof. The diagram encodes the following basic relations, plus their symmetrical counterpoints with ξ and η swapped:

(1) $(P \circ \xi^{-1})(A) = \int \iota_\xi(\omega, A)P(d\omega)$:

The right hand side expands to $\int \mathbb{1}_A(\xi(\omega))P(d\omega) = \mathbb{E}[\mathbb{1}_A\xi] = P(\xi \in A)$, which is defined as $(P \circ \xi^{-1})(A)$.

(2) $(P \circ \xi^{-1})(A) = \int P[\xi \in \cdot|\eta](t, A)(P \circ \eta^{-1})(dt)$:

The right hand side is $\mathbb{E}[P[\xi \in A|\eta]] = \mathbb{E}[\mathbb{E}[\mathbb{1}_{\xi^{-1}A}|\eta]]$. By Equation (2.2), this equals $\mathbb{E}[\mathbb{1}_{\xi^{-1}A}] = P(\xi^{-1}A) = (P \circ \xi^{-1})(A)$

Finally,

$$\begin{aligned}
\int_A P[\eta \in \cdot|\xi](\cdot, B)d(P \circ \xi^{-1}) &= \int_{\xi^{-1}A} P[\eta \in \cdot|\xi](\xi(\omega), B)P(d\omega) \\
&= \int_{\xi^{-1}A} P[\eta \in B|\xi](\xi(\omega))P(d\omega) \\
&= \mathbb{E}[P[\eta \in B|\xi](\xi(\omega))\mathbb{1}_{\xi^{-1}A}] \\
&= \mathbb{E}[\mathbb{E}[\mathbb{1}_{\eta^{-1}B}|\xi]\mathbb{1}_{\xi^{-1}A}] \\
&= \mathbb{E}[\mathbb{1}_{\eta^{-1}B}\mathbb{1}_{\xi^{-1}A}] \\
&= P(\eta^{-1}B \cap \xi^{-1}A) \\
&= P(\{\xi \in A\} \cap \{\eta \in B\})
\end{aligned}$$

The other equality follows symmetrically. □

We can interpret the composition of kernels as a sequential sampling operation: a kernel f from S to T gives a way of sampling from T for each $s \in S$. A kernel g from T to U likewise samples U given $t \in T$. The composition $g \circ_k f$ gives a way of sampling U given a $s \in S$: first sample $t \in T$ according to $f(s, \cdot)$, then sample from $g(t, \cdot)$. Using this language, (1) is just a restatement of the definition of the random element ξ: ω is drawn from P, then $\xi(\omega)$ is output. (2) says that we can sample the random element ξ by first sampling η then sampling from the conditional distribution $P[\xi \in \cdot|\eta]$. Then, the final statement says that if we sample ξ and then η from the conditional (or vice versa), and put the two together, this is the same as sampling from both unconditional distributions at the same time.

2.4.3 Martingales, Markov Processes, and Random Series

A *filtration* on some index set $T \subseteq \overline{\mathbb{R}}$ is a sequence of σ-algebras $\mathcal{F}_t \subseteq \mathcal{A}$, $t \in T$, with $\mathcal{F}_s \subseteq \mathcal{F}_t$ for $s < t$. A stochastic process on T (i.e., a random element of the set of functions $\{f : T \to U\}$) is *adapted* to $(\mathcal{F}_t)$ if it is $\mathcal{F}_t$-measurable for every t. The smallest filtration that X is adapted to is said to be *generated* by X, and is given by $\mathcal{F}_t = \sigma\{X_s|s \leq t\}$.

A real-valued process X is a *martingale* (with respect to $(\mathcal{F}_t)$) if

$$\mathbb{E}[M_t|\mathcal{F}_s] = X_s \text{ a.s.}$$

for all $s \leq t$.

Lemma 2.2. *Let (M_t) be a martingale such that M_t is finite for all t. Then,* $\text{var}(M_t) \geq \text{var}(M_s)$ *for all $t \geq s$.*

Proof. We begin by expanding $\text{var}(M_t)$:

$$\begin{aligned}
\text{var}(M_t) &= \mathbb{E}[M_t^2] - (\mathbb{E}[M_t])^2 \\
&= \mathbb{E}[\mathbb{E}[M_t^2|\mathcal{F}_s]] - (\mathbb{E}[\mathbb{E}[M_t|F_s]])^2 && \text{by Equation (2.2)} \\
&= \mathbb{E}[\text{var}[M_t|\mathcal{F}_s]] + \mathbb{E}[(\mathbb{E}[M_t|\mathcal{F}_s])^2] - (\mathbb{E}[\mathbb{E}[M_t|F_s]])^2 \\
&= \mathbb{E}[\text{var}[M_t|\mathcal{F}_s]] + \text{var}(\mathbb{E}[M_t|\mathcal{F}_s]) \\
&= \mathbb{E}[\text{var}[M_t|\mathcal{F}_s]] + \text{var}(M_s) && \text{by the martingale property.}
\end{aligned}$$

By definition, $\text{var}[M_t|\mathcal{F}_s] \geq 0$, so $\text{var}(M_t) \geq \text{var}(M_s)$. □

A S-valued process X on T is a *Markov process* if it is adapted to some filtration $(\mathcal{F}_t)$, and X_t is independent of $\mathcal{F}_s$, given X_s for $s \leq t$. If T starts at 0, then X is uniquely determined by an *initial distribution* ν and a set of kernels $\mu_{s,t}$. When $\mu_{s,t}$ is fixed, we write P_ν for the distribution of the paths (X_t). If $\mu_{s,t} = \mu_{t-s}$ depends only on the length of time $t - s$, X is said to be *time-homogeneous*.

A process X is called *stationary* if $\theta_t X \stackrel{d}{=} X$ for all t, where θ_t maps $\omega_s \mapsto \omega_{s+t}$. A measure ν is called *invariant* for (μ_t) if $\int \nu(dx)\mu_t(x, B) = \nu B$ for all $B \in \mathcal{S}$

Lemma 2.3. *A time-homogeneous Markov process with initial distribution ν and transition kernels μ_t is stationary iff ν is* invariant *for (μ_t).*

If S is countable, we call X a *Markov chain*; if $T = \mathbb{Z}_+$, we call it a *discrete-time* process. If $P_{\delta_i}\{\inf\{n > 0 | X_n = j\} < \infty\} > 0$ for all $i, j \in S$—so that every state is reachable with positive probability from every other state—we call X *irreducible*.

Let $\mu_1(i,j) = \mu(i,j)$, and $\mu_n(i,j) = \sum_{k \in S} \mu_{n-1}(i,k)\mu_{n-1}(k,j)$, so that $\mu_n(i,j)$ is the probability of reaching state j from state i in n steps. Then, consider $\mu_n(i, \{i\})$, for any state $i \in S$. Then, the set $R = \{n \in \mathbb{N} | \mu_n(i,\{i\}) > 0\}$ is the set of time-lags where it is possible for i to recur. If R has a greatest common denominator $d_i > 1$, so that i can only recur at some multiple of d_i, we call d_i the *period* of i. If there are no states i with periods, we call X *aperiodic*.

Theorem 2.4. *For an irreducible, aperiodic discrete-time Markov chain X, either*

1. *There exists a unique invariant distribution ν, such that $\nu(\{i\}) > 0$ for all $i \in S$ and*

$$\lim_{n \to \infty} \sup_A |(P_\mu \circ \theta_n^{-1})A - P_\nu A| = 0,$$

or

2. *No invariant distribution exists, and*

$$\lim_{n \to \infty} \mu_n(i, \{j\}) = 0, \text{ for all } i, j \in S$$

Thus, either X converges to a stationary chain P_ν, "forgetting" the initial distribution μ, or the multi-step transition probability $\mu_n(i,\{j\})$ spreads out without bound.

A measurable transformation T on $(\Omega, \mathcal{A}, \mu)$ is called *measure-preserving* if $\mu \circ T^{-1} = \mu$; equivalently, μ is called *invariant* relative to T. Then, $T \circ \xi \stackrel{d}{=} \xi$ for any random element ξ.

Lemma 2.4. *Let ξ be a random element of S, and let T be a measurable transformation on S. Then $T \circ \xi \stackrel{d}{=} \xi$ iff the discrete-time process $(T^n \xi), n \in \mathbb{Z}_+$ is stationary. Additionally, if f is measurable, $(f \circ T^n \xi)$ is also stationary. Any stationary process can be represented this way.*

Combining Theorem 2.4 and Lemma 2.4, we see that an irreducible, aperiodic discrete-time Markov chain X with invariant distribution ν, converges to a chain X' where $X'_n \stackrel{d}{=} \xi$ for some random variable ξ. Furthermore,

$$\begin{aligned}\nu(\{j\}) &= \sum_i \nu\{i\}\mu'_n(i,\{j\}) \\ &= \sum_i \nu\{i\}(P \circ \xi^{-1})\{j\} \\ &= (P \circ \xi^{-1})\{j\},\end{aligned}$$

so $X'_n \overset{iid}{\sim} \nu$. These results extend to analogous results with continuous time and state; see, e.g., (Kallenberg 1997, Theorem 20.15).

We end this section with a result on random series, i.e., sums over random sequences:

Theorem 2.5 (the law of large numbers)**.** *Let $\xi_1, \xi_2, \ldots$ be a sequence of i.i.d. random variables with $\mathbb{E}|\xi| < \infty$. Then,*

$$\lim_{n\to\infty} \frac{\sum_{i=1}^n \xi_i}{n} = \mathbb{E}\xi \ \ a.s.$$

See, e.g., (Kallenberg 1997, Theorem 3.23) for a proof.

2.4.4 Exchangeability

A random sequence (ξ_k) is *exchangeable* if

$$(\xi_{\pi_1}, \xi_{\pi_2}, \ldots) \stackrel{d}{=} (\xi_1, \xi_2, \ldots)$$

where (π_i) is a *finite permutation*, a permutation such that $\pi_i \neq i$ for only finitely many i. Such sequences are abundant in practice, to the extent that data that are not exchangeable are the exception, and are typically called things like "time-series data" or "streaming data."

Theorem 2.6 (de Finetti)**.** *Let* (ξ_k) *be some infinite random sequence on some Borel space* S*. Then,* (ξ_k) *is exchangeable iff*

$$\eta \sim \lambda$$
$$\xi_k \overset{iid}{\sim} \eta$$

for some measure λ *on the space of probability measures on* S*.*

This is a very useful—and very *Bayesian*—tool, as it says that any exchangeable data (which, as discussed above, is most data) can be treated as conditionally i.i.d., so that we may compute posteriors $P[\eta \in \cdot|\xi_1, \ldots, \xi_n]$ after n data points. λ is sometimes called the *mixing distribution* for (ξ_k). See (Aldous 1985; Diaconis 1988; de Finetti 1931, 1938) for more discussion of exchangeability.

2.5 Bayesian Methods

The formal Bayesian approach to statistics begins by modeling observed data as a random element ξ and unobserved parameters as a random element η, by defining the conditional distribution $P[\xi \in \cdot|\eta]$ (the likelihood), and the distribution $P \circ \eta^{-1}$ (the prior). Typically, this is done by defining the densities $p(x|\theta)$ and $p(\theta)$. Then, the posterior $p(\theta|x)$ is computed using Corollary 2.1:

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{\int p(x|\theta)p(d\theta)}. \tag{2.3}$$

In some cases, the parameter θ is the object of interest, but often the real interest is in making predictions about future observations. In these cases, we model the observed data as a vector $(x_1, \ldots, x_n)$, and compute the *posterior predictive distribution*

$$p(x_{n+1}|x_1, \ldots, x_n) = \int p(x_{n+1}|\theta)p(d\theta|x_1, \ldots, x_n)$$

Often, we assume the order of the data is irrelevant, representing a partial observation of an infinite exchangeable sequence, in which case we can apply Theorem 2.6.

In principle, the output of a Bayesian procedure is a full posterior distribution, but in practice this is hard to communicate. Often a summary of the posterior, such as the posterior *mean* (expectation) or *mode* (maximum) is used instead. Another summary is the *credible interval*, which is an interval, typically containing the mode which contain some set proportion (e.g., 95%) of the posterior probability, i.e., a parameter has 95% posterior probability of lying in a 95% credible interval.

2.6 Prior Selection

When specifying a model, the goal is to encode, as best as possible, the scientist's actual prior beliefs about the parameters θ and the relationship between those parameters and the observed data. For an illustrative example of this process involving human speech and birdsong, see (Yildiz, Kriegstein, and Kiebel 2013). Accordingly, there is a need for a large catalog of expressive, flexible priors and likelihoods. So far, we have only seen the small list of parametric distributions in Section 2.4.1, which is hardly a comprehensive toolkit: they are all *unimodal*, meaning that they have no local maxima other than the global maximum (except for the uniform, which of course has no maximum at all). However, these and similar simple distributions can be used as building blocks in constructing more complex priors. One such construction method is that of the *hierarchical* model: the basic two-level model of prior-plus-likelihood can be extended into a larger structure where the prior has parameters which themselves get so-called *hyperpriors*, which themselves may have parameters. Another construction method is a *mixture model* such as

$$\begin{aligned}
\vec{\pi} &\sim Dirichlet(\vec{\alpha}) \\
\theta_k &\overset{iid}{\sim} p(\theta_k) \\
x_i &\sim \sum_{k=1}^{K} \pi_i p(x_i|\theta_k).
\end{aligned}$$

Here, the likelihood is a weighted sum (mixture) of differing versions of the simpler distribution $p(x_i|\theta_k)$. For example, the $p(x_i|\theta_k)$ could be Normal distributions with means and standard deviations varying according to θ_k. In this case, the likelihood $p(x_i|\vec{\pi}, \vec{\theta})$ may be multimodal, in contrast to the unimodal Normal distribution it is built from.

When the scientist has a high degree of certainty about a parameter, the prior for that parameter will be highly concentrated around the nearly-known value. Conversely, if there they have a lot of uncertainty, the prior will be broad and diffuse. Such priors are also called *vague*. In the extreme case, where there is no information at all, the goal is to use an *objective* or *noninformative* prior. Over discrete spaces, and especially finite spaces, the objective prior is usually constant (i.e., uniform), but in some cases there may be underlying structure in the space or the likelihood that lead to other priors; see (Berger, Bernardo, and Sun 2012). In infinite cases, there is usually no such thing as a uniform probability distribution, but sometimes a prior based on a non-finite measure can yield a posterior distribution that does sum to one; such priors are called *improper*. If the likelihood can be expressed as $p(x|\theta) = f(x - \theta)$, then θ is called a *location parameter*, it is generally accepted that the uninformative prior is the improper uniform prior on $\mathbb{R}$ (Gelman et al. 2004). If $p(x|\theta) = f(y/\theta)$, then θ is a *scale parameter*, and the uninformative prior is the improper prior $p(\theta) \propto 1/\theta$ (Gelman et al. 2004).

(Kass and Wasserman 1996) reviews a number of proposed systems for defining noniformative priors. These include:

- Jeffreys' prior, $p(\theta) \propto \sqrt{\det(I(\theta))}$, where I is the Fisher information matrix,

$$I(\theta)_{ij} = \mathbb{E}\left[-\frac{\partial^2 \log p(x|\theta)}{\partial\theta_i \partial\theta_j}\right].$$

  If there are any location parameters, assume they are independent of other parameters.

- Laplace's priors based on the "principle of insufficient reason," which yields (often improper) uniform distributions. However, these are not invariant to reparameterization:

if the likelihood is expressed differently (for example, replacing the parameter σ with the parameter $\tau = 1/\sigma$ in the Normal distribution), the resulting prior is often far from uniform.

- Maximum entropy: find the prior that maximizes the entropy ($-\int p(\theta) \log p(\theta) \mu(d\theta)$ for some base measure μ) subject to certain restraints on expected values. Here, choosing μ is itself a challenge.

- Berger-Bernardo: find prior that maximizes expected *information gain* in the limit of infinite data, that is, maximize

$$\lim_{n \to \infty} \mathbb{E} \left[\int p(\theta | x_1, \ldots, x_n) \log \left(\frac{p(\theta | x_1, \ldots, x_n)}{p(\theta)} \right) d\theta \right]$$

  where the expectation is taken with respect to $p(x_1, \ldots, x_n) = \int p(x_1, \ldots, x_n | \theta) p(\theta) d\theta$.

- Zellner's method, the Maximal Data Information Prior (MDIP): let

$$Z(\theta) = -\int p(x|\theta) \log p(dx|\theta)$$

  then find the prior that maximizes

$$G = \int Z(\theta) p(d\theta) - \int p(\theta) \log p(d\theta),$$

  i.e., maximize the difference in expected entropy of the likelihood and the entropy of the prior. The solution is always

$$p(\theta) \propto e^{Z(\theta)}$$

  This prior is not invariant to reparameterization, but adding certain constraints in the maximization can allow certain reparameterizations.

Over certain topological spaces, (Dembski 1990) defines a general notion of uniform probability. In the simple case of the unit interval, their methods lead to the Lebesque measure, i.e., the usual uniform probability, as expected.

For an (incomplete) catalog of noninformative priors, see (Yang and Berger 1998).

Another approach to prior selection is to seek priors such that the Equation (2.3) is analytically tractable. One such approach is to use *conjugate priors*, which are chosen to be "closed under sampling" (Kadane 2011, Chapter 8), meaning that the posterior has the same form as the prior, with parameters updated based on the observed data. In addition to computational convenience, (Agarwal and Daumé III 2010) argues that there are geometric arguments that justify the use of conjugate priors. A popular family of distributions with abundant conjugacy properties is the exponential family (Diaconis and Ylvisaker 1979; Wainwright and Jordan 2008), which includes many of the most commonly used distributions, including all of the distributions listed in Section 2.4.1. A large number of conjugate pairs (i.e., prior-likelihood pairs that are conjugate) are given in (Fink 1997) (including examples that are not exponential family distributions).

CHAPTER 3

BAYESIAN NONPARAMETRICS

The usual formulation of the Bayesian paradigm of placing a prior on a parameter θ and defining a likelihood $p(x|\theta)$ can be reformulated as placing a prior $p(\eta)$ on some probability distribution η, and then drawing $x \sim \eta$, as seen in Theorem 2.6. Then, $p(\eta)$ places probability 1 on the set of measures of the *parametric* form $p(x|\theta)$. Viewed this way, such priors seem very restrictive. For example, if $p(x|\theta) \equiv N(x; \mu, \sigma^2)$, then all of the mass of $p(\eta)$ is concentrated on identically shaped symmetric, unimodal distributions that only vary in their location and scale. The field of Bayesian nonparametrics (Ghosh and Ramamoorthi 2003; Hjort et al. 2010) is concerned with identifying less restrictive priors using stochastic processes to define priors on the infinite-dimensional space of measures (or, more generally, on any infinite-dimensional space, such as spaces of densities or other functions).

The use of nonparametric priors can allow for fewer assumptions. For example, when using mixture models, one must choose the number of mixture components. Nonparametrics offer one way to let this factor be random as well.

We are left with the problem of defining priors on infinite-dimensional spaces. The following theorem gives one method (Ferguson 1973; Walker et al. 1999):

Theorem 3.1. *Let $(\Omega, \mathcal{A})$ be a measurable space, and let $\mathbb{D}(\mathcal{A}) = \{(A_{1,1}, \ldots, A_{m,k})\}$ be the set of all finite sequences of pairwise disjoint sets $A_{i,j} \in \mathcal{A}$. Then, let $\{F(A)\}_{A \in \mathcal{A}}$ be a collection of random variables with values in $(0, 1)$ such that $F(\Omega) = 1$ almost surely, and*

$$(F(\cup_i A_{1,i}), \ldots, F(\cup_i A_{m,i})) \stackrel{d}{=} \left(\sum_i F(A_{1,i}), \ldots, \sum_i F(A_{m,i}) \right)$$

for each element of $\mathbb{D}(\mathcal{A})$, then there exists a unique measure λ on the space of probability measures on $(\Omega, \mathcal{A})$ yielding these finite dimensional distributions.

3.1 The Dirichlet Process

Among the oldest and most popular nonparametric priors is the *Dirichlet Process* (Ferguson 1973). Let α be a finite, non-null measure on $(S, \mathcal{S})$. Then, the random measure P is a *Dirichlet Process*, $P \sim DP(\alpha)$ if, for any measurable partition $(A_1, \ldots, A_n)$ of S,

$$(P(A_1), \ldots, P(A_n)) \stackrel{d}{=} Dirichlet(\alpha(A_1), \ldots, \alpha(A_n))$$

By Theorem 3.1, this defines a probability measure on the space of probability measures on S. Moreover, $DP(\alpha)$ places probability 1 on the subspace of discrete probability measures.

Additionally, (Ferguson 1973) showed that the Dirichlet Process has a conjugacy property:

Theorem 3.2. *Let $P \sim DP(\alpha)$, and let $x_i \sim P$, $i \in \{1, \ldots, n\}$ be samples from P. Then,*

$$P|x_1, \ldots, x_n \sim DP(\alpha + \sum_{i=1}^{n} \delta_{x_i})$$

(Blackwell and MacQueen 1973) gives what is known as the *Pólya urn* representation of the Dirichlet Process:

$$x_i|x_1, \ldots, x_{i-1} \sim \frac{\alpha + \sum_{j=1}^{i-1} \delta_{x_j}}{\left(\alpha + \sum_{j=1}^{i-1} \delta_{x_j}\right)(S)}$$

To explain the name, assume S is finite, and $\alpha\{s\} \in \mathbb{Z}_+$ for all $s \in S$. Then, for each s, place $\alpha\{s\}$ balls of color s into an urn. Generate each x_i by drawing a ball at random from the urn, then return it to the urn along with another ball of the same color. (Blackwell and MacQueen 1973) showed that this process converges (as $i \to \infty$) to the Dirichlet Process, even when S and α are unrestricted.

(Sethuraman 1994) developed the following constructive definition of the Dirichlet Process:

$$\begin{aligned}
\theta_i &\stackrel{iid}{\sim} Beta(1, \alpha(S)) \\
p_i &= \theta_i \prod_{j=1}^{i-1}(1 - \theta_j) \\
y_i &\stackrel{iid}{\sim} \frac{\alpha}{\alpha(S)} \\
P &= \sum_{i=1}^{\infty} p_i \delta_{y_i}
\end{aligned}$$

This process has been called "stick-breaking:" we start with a unit-length stick, then break of a piece of length $p_1 = \theta_1$, and use it as the weight of y_1. Then, θ_2 gives the proportion of the remaining stick that we break off for y_2, and p_2 is the absolute length of this second piece. This process continues indefinitely. Using this representation, (Sethuraman 1994) presented a simpler proof of Theorem 3.2. Furthermore, this representation leads to better inference techniques (Walker 2007).

Consider the simple model

$$\begin{aligned} P &\sim DP(\alpha) \\ x_i &\overset{iid}{\sim} P. \end{aligned}$$

Thus, the Dirichlet process is the de Finetti mixing distribution for the exchangeable sequence (x_i). The distribution $P[(x_i)]$, appearing first in (Aldous 1985) (but attributed to Jim Pitman), is the *Chinese Restaurant Process* (CRP):

$$\begin{aligned} z_i | z_1, \ldots, z_{i-1} &\sim \sum_{j=1}^{i-1} \left(\frac{1}{i - 1 + \alpha(S)} \delta_{z_j} \right) + \frac{\alpha(S)}{i - 1 + \alpha(s)} \delta_{z^*} \\ \phi_i &\overset{iid}{\sim} \frac{\alpha}{\alpha(S)} \\ x_i &= \phi_{z_i}, \end{aligned}$$

where $z^* = \max\{z_1, \ldots, z_{i-1}\} + 1$ is the lowest value not yet assigned to any z_j. Equivalently, letting $n_z = |\{z_j | j < i, z_j = z\}|$,

$$\begin{aligned} z_i | z_1, \ldots, z_{i-1} &\sim \sum_{z=1}^{z^*-1} \left(\frac{n_z}{i - 1 + \alpha(S)} \delta_z \right) + \frac{\alpha(S)}{i - 1 + \alpha(s)} \delta_{z^*} \\ \phi_i &\overset{iid}{\sim} \frac{\alpha}{\alpha(S)} \\ x_i &= \phi_{z_i}, \end{aligned}$$

The analogy which gives this process its name is as follows: a sequence of customers arrive at a (Chinese) restaurant. The first customer sits at the first table ($z_1 \equiv 1$), and orders a

dish ($\phi_1 \sim \frac{\alpha}{\alpha(S)}$). Subsequently, the i^{th} customer sits at an occupied table z with probability $n_z/(i-1+\alpha(s))$, or chooses a new table with probability $\alpha(S)/(i-1+\alpha(S))$. If the chosen table is already occupied, the new customer orders the same dish as the other customers at that table. Otherwise, they order a new dish from $\frac{\alpha}{\alpha(S)}$. This representation is computationally convenient, as we need only keep track of the observations x_i and their tables z_i; the (infinite-dimensional) P has been integrated out.

This representation also gives insight into the meaning of the parameter $\alpha(S)$: if $\alpha(S)$ is very large, then the relative probability of sitting at a new table (and thus ordering a new dish i.i.d. from $\frac{\alpha}{\alpha(S)}$) grows, so that samples $x_i \overset{iid}{\sim} P$ resemble samples $x_i^* \overset{iid}{\sim} \frac{\alpha}{\alpha(S)}$. Conversely, if $\alpha(S)$ is small, then P will be further from $\frac{\alpha}{\alpha(S)}$.

The theoretical properties of the Dirichlet Process have been extensively studied; see, e.g., (Ferguson 1974; Kingman 1974; Green and Richardson 2001; Lo 1984; Ghosal, Ghosh, and Ramamoorthi 1999; Gnedin and Kerov 2001; Pitman 1996).

Often, it is undesireable to be restricted to discrete distributions. (Ferguson 1983) presents a simple workaround:

$$\begin{aligned} P &\sim DP(M\alpha) \\ (\mu_i, \sigma_i) &\overset{iid}{\sim} P \\ x_i &\overset{ind}{\sim} N(\mu_i, \sigma_i^2), \end{aligned}$$

where α is the Normal-Inverse Gamma prior conjugate to the Normal distribution, i.e., each unique $\rho_i = \frac{1}{\sigma_i^2} \sim G(\alpha, \frac{2}{\beta})$ with the corresponding $\mu_i \sim N(\mu_0, \frac{1}{(\sigma_i \tau)^2})$. This approach is called a Dirichlet Process Mixture (DPM) of Normals. (Escobar and West 1995) discusses this model further, and (Görür and Rasmussen 2010) extends this model to the multivariate Normal case, and discusses options for selecting α.

Of course, DPMs of other distributions are possible. (Canale and Scarpa 2015) discusses DPMs of skew-Normal distributions. (West 1992) discusses setting M in DPMs of arbitrary

exponential family distributions. Additionally, there are other techniques for smoothing a Dirichlet Process draw; see (Petrone 1999b, 1999a).

(Antoniak 1974) presents the converse, mixtures of Dirichlet Processes, where the parameter α is given a (hyper)prior; equivalently:

$$u \sim H$$
$$P \sim DP(\alpha(u, \cdot))$$

for some kernel $\alpha(\cdot, \cdot)$.

Dirichlet Processes have been applied to traditional machine learning tasks such as classification (Shahbaba and Neal 2009), clustering (Kulis and Jordan 2012), and computer vision (Sudderth 2006). (Stimberg, Ruttor, and Opper 2014) applies Dirichlet Processes to neurology in modelling EEG data, and (Navarro et al. 2006) provides an application in psychology. (Zhang, Pati, and Srivastava 2015) uses DPMs with a model of shapes to cluster curves including protein structures and cell shapes. (Yuan et al. 2015) develops a DPM with a rank-based likelihood for each mixture component, and (Kottas, Müller, and Quintana 2005) applies DPMs to discrete data.

Consider the problem of modeling data in the form of a collection of documents, i.e., a collection of lists or sets of words. In (Blei, Ng, and Jordan 2003), the following parametric model, called *Latent Dirichlet Allocation* (LDA) is proposed:

$$N_i \sim Poisson(\xi)$$
$$\theta_i \sim Dirichlet(\vec{\alpha})$$
$$z_{ij} \sim \theta_i$$
$$w_{ij} \sim p(\cdot|z_{ij}, \beta)$$

where i varies over documents, and j varies over words in each document. Thus, each document consists of a collection of words w_{ij}, each drawn from a distribution associated with

the "topic" z_{ij}, which itself is drawn from a document-specific topic distribution θ_i (based on this terminology, LDA is known as a *topic model*). As a parametric model, it is assumed that the number of topics is known *a priori*. The naïve approach to a nonparametric version of this model is to replace the Dirichlet distribution with a Dirichlet Process. However, this leads to a situation where every document has its own Dirichlet Process, and therefor its own a.s. unique topics; no two documents will share any topics.

(Teh et al. 2006) produces a solution to this problem with the Hierarchical Dirichlet Process (HDP):

$$G_0 \sim DP(\alpha)$$
$$G_j \overset{ind}{\sim} DP(\gamma G_0)$$

where $\gamma \in \mathbb{R}_+$ is a parameter that controls the total mass of the measure γG_0, since G_0 will a.s. have total mass 1. Because G_0 is a.s. discrete, the G_j will have positive probability of sharing atoms with each other. Teh *et al.* develop a stick-breaking representation and a CRP-like representation of the HDP. Put together, the nonparametric LDA is as follows:

$$G_0 \sim DP(\alpha)$$
$$G_j \overset{ind}{\sim} DP(\gamma G_0)$$
$$\theta_{ij} \sim G_j$$
$$w_{ij} \sim \theta_{ij}$$

where α is a base measure on a space of probability distributions over the (fixed) vocabulary of the corpus.

(Kim, Kim, and Oh 2012) presents a similar model where the documents can have multiple labels. Each label gets a Dirichlet Process G_0^k, and the base measure of each G_j is a finite mixture $\sum_k \lambda_{jk} G_0^k$, where k varies over the labels of document j. In addition to topic modeling, the HDP has been in neural activity modeling (Knudson and Pillow 2013; Kim

and Smyth 2006), and computer vision and image processing (Kivinen, Sudderth, and Jordan 2007b, 2007a; Sudderth et al. 2007).

In some cases, it is useful to model additional structure beyond i.i.d. topics. (Blei et al. 2004) introduces the *nested CRP* (nCRP) to build a model where topics are organized into a hierarchy. In the nCRP, each table of the restaurant is associated with another restaurant, whose tables are associated with other restaurants and so on, to L levels. Each customer visits the root restaurant and chooses a table according to the CRP. They subsequently visit the restaurant associated with the chosen table, and so on. At each level, they still sample dishes, so that overall each customer samples a path of restaurants with associated dishes. Applying this to create the hierarchical LDA (hLDA), each document is a customer, and each dish is a topic, so that each document has exactly L topics. Completing the model, we have

$$\theta_i \sim Dirichlet(\alpha)$$
$$z_{ij} \sim \theta_i$$
$$w_{ij} \sim topic(z_{ij})$$

where θ_i is a distribution over document i's L topics. This induces a hierarchy over the topics, as topics associated with restaurants near the root are shared by more documents and therefor should be more general. (Wang and Blei 2009) shows that the nCRP has a stick-breaking construction. This model is extended to eliminate the need for L in (Blei, Griffiths, and Jordan 2010) by making the tree of restaurants infinitely deep and essentially replacing the Dirichlet-distributed θ_i with a Dirichlet Process draw.

(Paisley et al. 2015) generalizes the nCRP by allowing each word to be associated with its own path in the tree, so that each document has a subtree rather than a single path. (Li and McCallum 2006) adds inter-topic dependencies to the LDA, and (Li, Blei, and McCallum 2007) generalizes this work to the nonparametric setting.

3.2 The Pitman-Yor Process

(Pitman and Yor 1995) generalizes the Dirichlet Process $DP(\alpha)$ to what is now known as the Pitman-Yor Process $PYP(\theta, \alpha)$. This process can be defined in terms of a stick-breaking representation:

$$\begin{aligned}
\theta_i &\overset{ind}{\sim} Beta(1-\theta, \alpha(S) + i\theta) \\
p_i &= \theta_i \prod_{j=1}^{i-1}(1-\theta_j) \\
y_i &\overset{iid}{\sim} \frac{\alpha}{\alpha(S)} \\
P &= \sum_{i=1}^{\infty} p_i \delta_{y_i}
\end{aligned}$$

It also admits a Chinese Restaurant representation:

$$\begin{aligned}
z_i | z_1, \ldots, z_{i-1} &\sim \sum_{z=1}^{z^*-1}\left(\frac{n_z - \theta}{i - 1 + \alpha(S)}\delta_z\right) + \frac{\alpha(S) + (z^*-1)\theta}{i-1+\alpha(S)}\delta_{z^*} \\
\phi_i &\overset{iid}{\sim} \frac{\alpha}{\alpha(S)} \\
x_i &= \phi_{z_i},
\end{aligned}$$

Distributions drawn from a Pitman-Yor Process have *power-law* behaviour (Broderick, Jordan, and Pitman 2012), meaning that there exists constants $c > 0$ and $a \in (0, 1)$ such that

$$\lim_{n\to\infty} \frac{K_n}{cn^a} = 1 \text{ a.s.},$$

where K_n is the number of mixture components that have been sampled after n draws (i.e., the number of occupied tables $z^* - 1$ after n customers), and that there exists constants $d > 0$ and $b \in (0, 1)$ such that

$$\lim_{n\to\infty} \frac{K_{n,i}}{dn^b} \frac{i!\Gamma(1-b)}{b\Gamma(i-b)} = 1 \text{ a.s.},$$

where $K_{n,i}$ is the number of samples from the i^{th} mixture component (i.e., the number of customers at table i after n total customers). Distributions with such properties have been observed in a number of real-world situations (Zipf 1949; Gnedin, Hansen, and Pitman 2007).

Like the Dirichlet Process, models based on hierarchies of Pitman-Yor Processes have been used for language modeling (Wood et al. 2011; Teh 2006; Wood et al. 2009) and image processing (Sudderth and Jordan 2008; Shyr et al. 2011).

3.3 Pólya Trees

Another generalization of the Dirichlet Process is the *Pólya Tree*, introduced by (Mauldin, Sudderth, and Williams 1992), and further studied by (Lavine 1992, 1994); see also (Ferguson 1974). Let $E = \{0, \ldots, k-1\}$ be some finite set, let E^m denote the set of sequences of elements of E of length m, and let $E^* = \cup_m E^m$ denote the set of all finite sequences of elements of E (including the empty sequence $\emptyset$). Next, let $\mathcal{A} = \{\alpha_\epsilon | \epsilon \in E^*\}$ be a set of k-dimensional vectors, and let $\Pi = \{\pi_0, \pi_1, \ldots\}$ be an infinite tree of partitions of Ω such that $\pi_0 = \{\Omega\}$ and π_m is constructed by partitioning each element of π_{m-1} into k pieces. Finally, for any $\epsilon \in E^*$, let B_ϵ be obtained by traversing Π: $B_\emptyset = \Omega$, $B_{(0)}$ is the first partition of π_1, $B_{(0,0)}$ is the first part of $B_{(0)}$ as defined by π_2, and so on. Then, the random probability measure P on Ω has a Pólya tree distribution, $P \sim PT(\Pi, \mathcal{A})$, if there exists a set of random k-dimensional vectors $\mathcal{Y} = \{Y_\epsilon | \epsilon \in E^*\}$ such that

1. $Y_\epsilon \sim Dirichlet(\alpha_\epsilon)$ and
2. for any $\epsilon \in E^*$ of length m, $P(B_\epsilon) = \prod_{j=1}^{m} Y_{(\epsilon_1, \ldots, \epsilon_{j-1})}(\epsilon_j)$

where $Y(e)$ is the e^{th} element of the vector Y. In words, $x \sim P$ is drawn by randomly traversing the tree Π according to random variables with Dirichlet distributions whose parameters come from $\mathcal{A}$.

Dirichlet Processes are a special case of Pólya trees where $\alpha_\epsilon = \sum_e \alpha_{\epsilon e}$, and like the Dirichlet Process, Pólya trees are conjugate under sampling: if $P \sim PT(\Pi, \mathcal{A})$ and $x \sim P$, then $P|x$ also has a Pólya tree distribution. With certain choices of Π and $\mathcal{A}$, P can be an a.s. continuous distribution.

3.4 The Indian Buffet Process

The Dirichlet Process and its variants are sometimes referred to as "latent class" models (see, e.g., (Ghahramani, Griffiths, and Sollich 2006))—the z_i in the CRP representation serve as class assignments. An alternative to latent classes are latent "features". The Indian Buffet Process (IBP) (Ghahramani, Griffiths, and Sollich 2006; Griffiths and Ghahramani 2005; Ghahramani and Griffiths 2006) is a model that assigns latent binary features to each data point. As in the CRP, in a draw $Z \sim IBP(\alpha)$, customers arrive sequentially, but now they walk down an (infinitely long) buffet line. The first customer samples the first $Poisson(\alpha)$ dishes; subsequently, the i^{th} customer samples previously-sampled dishes with probability $\frac{m_k}{i}$, where m_k is the number of times the k^{th} dish has previously been sampled. They then sample $Poisson(\frac{\alpha}{i})$ new dishes. Then, Z is a matrix with one row per customer and an infinite number of columns, of which only a finite number are non-zero, where z_{ik} is 1 if the i^{th} customer sampled the k^{th} dish.

Like the Dirichlet Process, the IBP has a stick-breaking representation (Teh, Görür, and Ghahramani 2007): let μ_k denote the prior probability of any customer sampling the k^{th} dish, and let $\mu_{(1)} > \mu_{(2)} > \cdots$ denote the μ_ks in decreasing order. Then,

$$\begin{aligned} \nu_{(k)} &\overset{iid}{\sim} Beta(\alpha, 1) \\ \mu_{(k)} &= \prod_{i=1}^{k} \nu_{(i)} \\ z_{ik} &\sim Bernoulli(\mu_{(k)}) \end{aligned}$$

Just as the CRP is derived from the Dirichlet Process by integrating out the random measure P, (Thibaux and Jordan 2007) considered the model

$$\begin{aligned} B &\sim BP(H) \\ Z_i &\overset{iid}{\sim} BeP(B), \end{aligned}$$

where $BP(H)$ is the *Beta process* with base measure H (Hjort 1990), and $BeP(B)$ is called the *Bernoulli process* with *hazard measure* B. Then, if we integrate out B, Z is distributed according to the IBP.

There is a rich body of theoretical work on the IBP and the Beta process. (Wang and Carin 2012) develops a representation of the Beta process as a Lévy process (Applebaum 2004). (Broderick, Jordan, and Pitman 2012) develops a 3-parameter generalization of the Beta process that exhibits power-law behaviour. (Broderick, Jordan, and Pitman 2013) and (Broderick, Pitman, and Jordan 2013) discuss the combinatorial properties of the IBP and the Beta process. (Paisley et al. 2010) develops a stick-breaking representation for the Beta process.

There has also been much work on applications using the IBP and the Beta process. (Hu et al. 2012) uses the IBP for image representation, for visual feature extraction and image reconstruction. (Fox et al. 2014) uses the Beta process to model multiple related time series in the context of motion capture segmentation. (Wood, Griffiths, and Ghahramani 2006) and (Adams, Wallach, and Ghahramani 2010) use the IBP (and a recursive variant called the cascading IBP) to learn the structure of Neural Networks. (Knowles and Ghahramani 2007) and (Knowles and Ghahramani 2011) present models based on the IBP to perform signal separation, wherein an observed dataset is assumed to be a linear combination of multiple signals, which these models attempt to recover. In particular, they apply their models to the problem of finding gene transcription factors based on gene expression data. (Polatkan et al. 2015) uses the Beta-Bernoulli process for image superresolution, wherein a high-resolution image is constructed based on a low-resolution image. (Ruiz et al. 2014) applies the IBP to the problem of detecting patterns of psychiatric disorders appearing together.

3.5 The Gaussian Process

Another popular nonparametric prior is the *Gaussian Process* (Rasmussen and Williams 2006). Like the Dirichlet Process, the Gaussian process is defined by its finite-dimensional distributions: if X is a Gaussian Process on some index set T, $X \sim GP(m, k)$, then

$$\mu = (m_{t_1}, \ldots, m_{t_n})$$

$$\Sigma_{ij} = k_{t_i, t_j}$$

$$(X_{t_1}, \ldots, X_{t_n}) \sim N(\mu, \Sigma)$$

for any $(t_1, \ldots, t_n) \in T^n$. Typically, X is thought of as a random real-valued function $X(t)$. Then, $m : T \to \mathbb{R}$ is called the *mean function*, and $k : T \times T \to \mathbb{R}$ is called the *covariance function*.

As a special case, let $m \equiv 0$ and $k(s, t) = \min\{s, t\}$. Then, $X(t) \sim N(0, t)$, and X is called *Brownian motion* (Ghosh and Ramamoorthi 2003; Zanten 2007).

Gaussian Processes have been used in many areas, ranging from regression (Rasmussen and Williams 2006), classification (Rasmussen and Williams 2006; Barber and Williams 1997; Mackay and Gibbs 2000), extrapolation (Wilson and Adams 2013), and density estimation (Murray, MacKay, and Adams 2009).

3.6 Other Nonparametric Models

It is often possible and useful to combine nonparametric priors. (Williamson et al. 2010) combines the HDP and the IBP in a topic model that allows documents to focus on rare topics. Similarly, (Paisley, Wang, and Blei 2012) combines the HDP with a Gaussian Process for a correlated-topic model.

(Iwata, Duvenaud, and Ghahramani 2013) present a model that combines the Dirichlet Process with the Gaussian Process: each data point is drawn from a DPM of multivariate

Normals, with each dimension then "warped" by a function drawn from a Gaussian Process, giving much more free-form cluster shapes:

$$\begin{aligned} P &\sim DP(M\alpha) \\ f_d &\sim GP(m(\cdot), k(\cdot,\cdot)) \\ \mu_i, \sigma_i &\overset{iid}{\sim} P \\ y_i &\overset{ind}{\sim} N(\mu_i, \sigma_i^2) \\ x_{id} &\sim N(f_d(x_{nd}), \beta^{-1}). \end{aligned}$$

Relatedly, (Jackson et al. 2007) presents a Dirichlet Process mixture of Gaussian Processes, where the parameters $m(\cdot), k(\cdot,\cdot)$ of the Gaussian Process are drawn from a Dirichlet Process. (Rai and Daumé III 2008) develops a sparse variant of the IBP and combines it with a nonparametric prior over trees called Kingman's Coalescent (Kingman 1982).

(MacEachern 1999) adapts the Dirichlet Process to a regression setting: in cases where a standard linear regression with parametric noise is a poor fit, a better approach is to use a nonparametric distribution for the errors that smoothly evolves depending on the covariate. The Dependent Dirichlet Process (DDP) is proposed to achieve this goal. Consider the stick-breaking construction of the Dirichlet Process above. To adapt to the regression context, we first replace the random variables $y_i \overset{iid}{\sim} \frac{\alpha}{\alpha(S)}$ with stochastic processes $y_{i\mathcal{X}} = (y_{ix})_{x\in\mathcal{X}}$, where $\mathcal{X}$ is the covariate space, essentially specializing to the case where α is a measure over a space of functions $\mathcal{X} \to S$ for some S. Next, we similarly replace the stick length proportions θ_i with processes $\theta_{i\mathcal{X}} = (\theta_{ix})_{x\in\mathcal{X}}$, likewise replacing p_i with $p_{i\mathcal{X}}$. This effectively replaces the parameter $\alpha(S)$ with a parameter $M_{\mathcal{X}}$, which allows the degree to which samples from the DDP resemble parametric regression to vary along $\mathcal{X}$. See (Lin, Grimson, and Fisher 2010; Barrientos, Jara, and Quintana 2012) for more theoretical discussion of DDPs; (Campbell et al. 2013) generalizes the clustering algorithm of (Kulis and Jordan 2012) to the dependent setting using DDPs.

Hidden Markov models (HMMs) are a popular model of sequential data such as speech, consisting of an unobserved (hidden) state which evolves among a finite number of states according to a Markov chain. At each time step, an observed variable is generated based on the current state. (Beal, Ghahramani, and Rasmussen 2002) introduce a nonparametric technique for this setting by incorporating a HDP into the transition probability that governs the unobserved Markov chain, allowing the state space to be infinite. (Bratières et al. 2010) and (Van Gael et al. 2008) discuss inference in this model. (Palla, Knowles, and Ghahramani 2014) develops a similar model where the Markov chain is guaranteed to be reversible, whereas (Stepleton et al. 2009) enforces additional structure on the transition probability that produces partitions (i.e. clusters) of the hidden state space. (Blunsom and Cohn 2011) discusses an infinite HMM using hierarchical Pitman-Yor Processes.

Other works develop completely new models. (Titsias 2008) develops nonparametric distribution over non-negative integer-valued matrices called the Infinite Gamma-Poisson Process (IGPP). Similarly, (Broderick et al. 2015) produces the Beta Negative Binomial Process (BNBP), which is similar to the Beta-Bernoulli Process underlying the IBP; they also investigate the Hierarchical BNBP. (Zhou et al. 2012) shows that these two processes are closely related, and (Zhou 2014) further develops the theory of the BNBP. Finally, (Heaukulani and Roy 2016) derives the negative binomial version of the IBP.

Sometimes it is useful to develop new models by incorporating additional constraints into existing models. (Dalal 1979) present a modified Dirichlet Process that produces distributions that are invariant to certain transformations. (Gershman, Frazier, and Blei 2015) modifies the IBP to incorporate a distance metric between customers such that similar customers are more likely to sample the same dishes; (Miller, Griffiths, and Jordan 2008) presents a similar modification that uses a phylogenetic structure among customers. (Williamson, Maceachern, and Xing 2013) presents a method for modifying the IBP and other nonparametric processes on infinite matrices, including the BNBP and IGPP, to change the distribution of the

number of non-zero entries per row. In the case of the IBP, this corresponds to changing the distribution for the number of dishes each customer samples.

Often, connections are found between seemingly disparate processes. For example, (Heaukulani and Roy 2015) investigates connections between the Dirichlet Process and the IBP, and a similar connection between the PYP and the three-parameter IBP of (Teh and Görür 2009). (James, Orbanz, and Teh 2015) explores further connections between the CRP and the IBP and other processes. (Roy 2014) explores connections between the Dirichlet Process and the Beta Process and uses this to generalize the IBP. (Jordan 2010) reviews a class of models called Completely Random Measures (CRMs) (Kingman 1967), which includes the Dirichlet Process and the Beta and Bernoulli Processes. See (Broderick, Wilson, and Jordan 2018) for recent work in this area. (Broderick, Jordan, and Pitman 2013) develops a theory of the combinatorial structure of the CRP and IBP.

The stick-breaking representations of the Dirichlet Process and IBP have lead to other generalizations. (Dunson and Park 2008) incorporates random locations and a distance measure into the stick-breaking weights of the Dirichlet Process. (Ghahramani, Jordan, and Adams 2010) generalizes to a tree-structured stick breaking wherein sticks can be broken into more than two pieces. (Nalisnick and Smyth 2016) develops a stick-breaking-based deep Neural Network model. See also (Perman, Pitman, and Yor 1992).

The Wishart distribution, $W_n(V, \nu)$, is a distribution over positive definite $n \times n$ matrices (i.e., $n \times n$ matrices M for which $z^\top M z$ is strictly positive for all non-zero n-vectors z) which is commonly used as a prior for covariance matrices. If $u_i \overset{iid}{\sim} N(0, V)$, then $\sum_{i=1}^{\nu} u_i u_i^\top \sim W(V, \nu)$ (although the Wishart distribution is also defined for non-integer ν); (Wilson and Ghahramani 2009) defines the Generalized Wishart Process analagously by replacing the Gaussian vectors u_i with Gaussian Processes, arriving at a distribution over infinite collections of positive definite matrices.

CHAPTER 4

INFERENCE

When applying Bayesian techniques, we wish to calculate posterior distributions or posterior expectations (such as posterior predictive distributions). However, these calculations often involve intractable integrals. Accordingly, there has been a great deal of effort in finding efficient methods to accurately approximate these quantities. We will mostly focus on *Monte Carlo* techniques, also called simulation techniques.

Suppose we wish to calculate $\mathbb{E}[f(\theta)|x] = \int f(\theta)p(d\theta|x)$. If we can sample

$$\theta^{(i)} \sim p(\theta|x),$$

then

$$\widehat{I} = \frac{1}{K}\sum_{i=1}^{K} f(\theta^{(i)})$$

is a good estimator. In particular, a good estimator of this form should have two properties: it should be *unbiased*, meaning that the expected value of the estimator should equal the true value, i.e., $\mathbb{E}[\widehat{I}] = \mathbb{E}[f(\theta)|x]$, and it should have decreasing variance as the number of samples K grows. The second property follows from the law of large numbers (Theorem 2.5); we prove the first:

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{K}\sum_{i=1}^{K} f(\theta^{(i)})\right] &= \frac{1}{K}\int\left[\sum_{i=1}^{K} f(\theta^{(i)})\right]\prod_{i=1}^{K} p(d\theta^{(i)}|x) \\
&= \frac{1}{K}\sum_{i=1}^{K}\int f(\theta^{(i)})\prod_{j=1}^{K} p(d\theta^{(j)}|x)
\end{aligned}$$

$\int f(\theta^{(i)})p(d\theta^{(j)}|x) = f(\theta^{(i)})$ for $i \neq j$, so we have

$$\begin{aligned}
&= \frac{1}{K}\sum_{i=1}^{K}\int f(\theta^{(i)})p(d\theta^{(i)}|x) \\
&= \frac{1}{K}\sum_{i=1}^{K}\mathbb{E}[f(\theta)|x] = \mathbb{E}[f(\theta)|x]
\end{aligned}$$

Unfortunately, it is often the case that one cannot sample $p(\theta|x)$ directly, so that another layer of approximation must be used. We focus on two classes of such methods: *Markov Chain Monte Carlo* (MCMC) and *Sequential Monte Carlo* (SMC).

4.1 Markov Chain Monte Carlo

MCMC is based around the construction of a Markov chain with transition kernel $P(\theta, B)$ which has $p(\theta|x)$ as its stationary distribution.

Any Markov chain satisfying the global balance condition

$$\int p(\theta|x)P(\theta, d\theta') = \int p(d\theta'|x)P(d\theta', \theta)$$

will have stationary distribution $p(\theta|x)$. A stricter condition that implies global balance is *detailed balance:*

$$p(\theta|x)P(\theta, \theta') = p(\theta'|x)P(\theta', \theta)$$

A generic form of such a kernel, known as the Metropolis-Hastings algorithm (Metropolis et al. 1953; Hastings 1970), is

$$P(\theta, B) = \sum_m \int_B \underbrace{q_m(\theta, d\theta')}_{\substack{\text{propose a} \\ \text{move } m \\ \text{from } \theta \\ \text{to } \theta'}} \underbrace{\alpha_m(\theta, \theta')}_{\substack{\text{accept the} \\ \text{move}}} + \underbrace{s(\theta)\mathbb{1}_{\{\theta \in B\}}}_{\substack{\text{no move} \\ \text{(rejected or} \\ \text{none} \\ \text{proposed)}}}$$

where m varies over some set of available "moves," which identify proposal distributions $q_m(\cdot, \cdot)$. Sampling proceeds by first selecting a move m, then proposing a new state $\theta' \sim q_m(\theta, \cdot)$. Then the new state θ' is either accepted (with probability $\alpha_m(\theta, \theta')$), or rejected, in which case the chain stays in state θ.

We may enforce detailed balance if we let

$$\alpha_m(\theta, \theta') = \min\left\{1, \frac{f_{m^*}(\theta', \theta)}{f_m(\theta, \theta')}\right\},$$

where $f_m(\theta, \theta')$ is the (finite) density of $p(d\theta|x)q_m(\theta, d\theta')$ with respect to some symmetric measure μ_m, and m^* is the move that can take state θ' to θ (in many samplers, $m^* = m$).

Multiple kernels may be combined in a mixture or in a cycle. *Gibbs sampling* is a special case where $\theta = (\theta_1, \ldots, \theta_d)$, and each cycle samples $\theta_i \sim p(\theta_i|\theta_{\neg i})$ exactly. In this case, $\alpha(\theta, \theta') = 1$.

See (Andrieu et al. 2003; Neal 1993b; Robert and Casella 1999; Brooks et al. 2011) for reviews of MCMC techniques. (Gilks, Richardson, and Spiegelhalter 1996) and (Besag et al. 1995) contain numerous examples of applied MCMC algorithms. (Green et al. 2015) is a review of more recent advances.

There has been a considerable work on MCMC methods. In some cases, the acceptance probability $\alpha(\theta, \theta')$ is difficult to compute, in which case it can be estimated (Sherlock et al. 2015) or decomposed into multiple ratios that are tested (so that if the sample is rejected, the whole ratio is never computed) (Banterle, Grazian, and Robert 2014; Banterle et al. 2015; Sherlock, Golightly, and Henderson 2016). The random proposals of a Metropolis-Hastings kernel can sometimes be slow to explore the full support of the target distribution, leading to low quality estimates. (Neal 1995) addresses this problem for Gibbs sampling by "ordered overrelaxation:" when sampling $\theta_i' \sim p(\cdot|\{\theta_j\}_{j\neq i})$, generate K samples, then order them $\theta_i^{(1)} \leq \ldots \leq \theta_i^{(r)} = \theta_i \leq \ldots \leq \theta_i^{(K)}$ (r is the index of the current value), $\theta_i' = \theta_i^{(K-r)}$; this tends to make θ_i' more different from θ_i than plain Gibbs, while preserving invariance. If a CDF is available, more efficient versions are available. (Tak, Meng, and Dyk 2017) approaches the same problem from another direction by incorporating the gradient of the target density into the proposal and alternating between "attractive" and "repulsive" cycles which move toward and away from local modes, respectively.

Another approach to this problem is *slice sampling* (Mira 1998; Neal 2003): let $f(\theta)$ be the (possibly unnormalized) density we wish to sample from, and let y be an auxiliary

variable such that the joint density is

$$p(\theta, y) = \begin{cases} \frac{1}{\int f(\theta) d\theta} & 0 < y < f(\theta) \\ 0 & \text{otherwise;} \end{cases}$$

then,

$$p(\theta) = \int_0^{f(\theta)} \left(\frac{1}{\int f(\theta) d\theta} \right) dy \propto f(\theta),$$

so sampling from this joint distribution, then discarding y values yields the samples we desire. The basic idea is to define a Markov chain that will converge to this joint distribution. (Walker 2007) applies this technique to Dirichlet Process Mixtures.

Other methods of improving efficiency focus on the computational aspect, specifically enabling parallel and distributed implementations (Neal 2012; Angelino et al. 2014; Neiswanger, Wang, and Xing 2014), and data subsampling techniques (Maire, Friel, and Alquier 2015; Bardenet, Doucet, and Holmes 2017); see also (Angelino, Johnson, and Adams 2016).

Finally, there has been substantial work in the field of *exact sampling* (Green and Murdoch 1998), which is focused on using MCMC methods to generate samples that are exactly distributed as $p(\theta|x)$. See also (Murdoch and Green 1998; Wang, Schwing, and Urtasun 2014; Fill and Huber 2010). (Casella et al. 2002) develops an exact slice sampler.

4.1.1 Reversible Jump MCMC

In many cases, the states θ and θ' may be of different dimensions. In such cases, special care must be taken to maintain detailed balance while the parameter space may change dimension between moves. Reversible Jump MCMC (RJMCMC) (Green 1995; Green and Hastie 2009) allows us to do this.

Let $\theta = (k, \theta^{(k)})$, where k is the "model index," which identifies the parameter space $\mathbb{R}^{n_k}$, and $\theta^{(k)}$ denotes the parameters of this model. Then, let $\mathscr{C}_k = \{k\} \times \mathbb{R}^{n_k}$ and $\mathscr{C} = \cup_k \mathscr{C}_k$, so that $\mathscr{C}$ is the space we wish to sample from.

To construct the new state $\theta' = (k', \theta^{(k')})$, because $\theta^{(k)} \in \mathbb{R}^{n_k}$ and $\theta^{(k')} \in \mathbb{R}^{n_{k'}}$ with $n_k \neq n_{k'}$ in general, it is often necessary to sample new real values. That is, when moving from θ to θ', we set $\theta^{(k')}$ to some deterministic function of $\theta^{(k)}$ and $u^{(k)}$, where $u^{(k)} \sim q_k(u^{(k)})$ is a real-valued random vector. Similarly, moving from θ' to θ involves sampling $u^{(k')} \sim q_{k'}(u^{(k')})$. $u^{(k)}$ and $u^{(k')}$ are of dimensions m_k and $m_{k'}$, respectively, such that $n_k + m_k = n_{k'} + m_{k'}$ (noting that one or both of m_k and $m_{k'}$ may be zero).

Finally, to propose a new state θ', we must first pick a "move" m. Let $j_m(\theta, \theta')$ denote the probability of proposing the move m from the state θ. This includes both the probability of choosing the m^{th} "type" of move, and the probabilities of any random variables sampled in constructing the new state θ' according to m (including the sampling of $u^{(k')}$).

Let us revisit the requirement that $p(d\theta|x)q_m(\theta, d\theta')$ have a finite density $f_m(x, x')$ with respect to some symmetric measure μ_m. To maintain symmetry, for any $A \subset \mathscr{C}_k$, $B \subset \mathscr{C}_{k'}$, we must have

$$\mu_m(A \times B) = \lambda\{(\theta^{(k)}, u^{(k)}) : \theta^{(k)} \in A, \theta^{(k')}(\theta^{(k)}, u^{(k)}) \in B\},$$

where λ is the Lebesque measure. Moreover, consider the inverse,

$$\mu_m(B \times A) = \lambda\{(\theta^{(k')}, u^{(k')}) : \theta^{(k')} \in B, \theta^{(k)}(\theta^{(k')}, u^{(k')}) \in A\}.$$

By symmetry, $\mu_m(A \times B) = \mu_m(B \times A)$, which requires that we have a bijection between $(\theta^{(k')}, u^{(k')})$ and $(\theta^{(k)}, u^{(k)})$, so we can move back and forth freely and deterministically.

Then, applied to our trans-dimensional model, the density $f_m(\theta, \theta')$ is

$$\begin{aligned} f_m(\theta, \theta') &= p(k, \theta^{(k)}|x) j_m(\theta, \theta') \\ f_m(\theta', \theta) &= p(k', \theta^{(k')}|x) j_m(\theta', \theta) \times \left| \frac{\partial(\theta^{(k')}, u^{(k')})}{\partial(\theta^{(k)}, u^{(k)})} \right|, \end{aligned}$$

where the Jacobian term comes from the fact that $dv_1 \dots dv_n = \left| \frac{\partial(v_1, \dots, v_n)}{\partial(u_1, \dots, u_n)} \right| du_1 \dots du_n$, as used in the general substitution rule in calculus.

Then, $\alpha_m(\theta, \theta') = \min\{1, \alpha_m^*(\theta, \theta')\}$, where

$$\begin{aligned}\alpha_m^*(\theta, \theta') &= \frac{p(k', \theta^{(k')}|x) j_{m^*}(\theta', \theta)}{p(k, \theta^{(k)}|x) j_m(\theta, \theta')} \left|\frac{\partial(\theta^{(k')}, u^{(k')})}{\partial(\theta^{(k)}, u^{(k)})}\right| \\ &= \underbrace{\frac{p(x|k', \theta^{(k')})}{p(x|k, \theta^{(k)})}}_{\substack{\text{likelihood} \\ \text{ratio}}} \underbrace{\frac{p(k', \theta^{(k')})}{p(k, \theta^{(k)})}}_{\substack{\text{prior} \\ \text{ratio}}} \underbrace{\frac{j_{m^*}(\theta', \theta)}{j_m(\theta, \theta')}}_{\substack{\text{proposal} \\ \text{ratio}}} \underbrace{\left|\frac{\partial(\theta^{(k')}, u^{(k')})}{\partial(\theta^{(k)}, u^{(k)})}\right|}_{\text{Jacobian}}\end{aligned}$$

(Hastie and Green 2012) and (Andrieu and Doucet 1999) apply RJMCMC to the problem of selecting a model, i.e., where $p(k, \theta^{(k)})$ and $p(k', \theta^{(k')})$ are entirely different models. (Green and Mira 2001) proposes a *delayed rejection* approach to RJMCMC in which rejected samples are instead given a second chance by re-applying a move. (Al-Awadhi, Hurn, and Jennison 2004) discusses the development of efficient RJMCMC moves (i.e., moves with a high acceptance rate).

4.2 Sequential Monte Carlo

Consider a state-space model or HMM, where an unobserved parameter evolves according to a Markovian kernel $p(\theta_i|\theta_{i-1})$, and at each i, an observation is generated from $p(x_i|\theta_i)$. An MCMC approach to inference in such a model would have to repeatedly sample θ_i and x_i for every i. Sequential Monte Carlo (SMC) methods work recursively, each step producing an approximation of $p(\theta_i|x_{1:i-1})$ by modifying the previous step's approximation of $p(\theta_{i-1}|x_{1:i-2})$, taking advantage of a the Markovian structure.

A simple SMC method samples

$$\begin{aligned}\theta_1^{(k)} &\overset{iid}{\sim} p(\theta_1) \\ \theta_i^{(k)} &\overset{ind}{\sim} p(\theta_i|\theta_{i-1}^{(k)}, x_i)\end{aligned}$$

and estimates $p(\theta_i|x_{1:i})$ by

$$\widehat{p}(\theta_i|x_{1:i}) \propto p(x_i|\theta_i) \sum_{k=1}^{K} p(\theta_i|\theta_{i-1}^{(k)})$$

As with all Monte Carlo estimators, we wish our estimators to be unbiased:

$$\begin{aligned}
\mathbb{E}[\widehat{p}(\theta_i|x_{1:i})] &\propto p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(k)},x_{i-2})}\mathbb{E}_{p(\theta_{i-1}^{(k)}|\theta_{i-2}^{(k)},x_{i-1})}\left(\sum_{k=1}^{K}p(\theta_i|\theta_{i-1}^{(k)})\right)\\
&= p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(k)},x_{i-2})}\left(\int\left(\sum_{k=1}^{K}p(\theta_i|\theta_{i-1}^{(k)})\right)\prod_{k=1}^{K}p(d\theta_{i-1}^{(k)}|\theta_{i-2}^{(k)},x_{i-1})\right)\\
&= p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(k)},x_{i-2})}\left(\sum_{k=1}^{K}\int p(\theta_i|\theta_{i-1}^{(k)})\prod_{j=1}^{K}p(d\theta_{i-1}^{(j)}|\theta_{i-2}^{(j)},x_{i-1})\right)\\
&= p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(k)},x_{i-2})}\left(\sum_{k=1}^{K}\int p(\theta_i|\theta_{i-1}^{(k)},\theta_{i-2}^{(k)},x_{i-1})p(d\theta_{i-1}^{(k)})\right)\\
&= p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(k)},x_{i-2})}\left(\sum_{k=1}^{K}p(\theta_i|\theta_{i-2}^{(k)},x_{i-1})\right)
\end{aligned}$$

Thus, the rightmost expectation replaces the term $\theta_{i-1}^{(k)}$ with $\theta_{i-2}^{(k)}$ and adds a dependency on x_{i-1}. Each successive expectation proceeds likewise until we get

$$\begin{aligned}
&= p(x_i|\theta_i)\sum_{k=1}^{K}p(\theta_i|x_{1:i-1})\\
&\propto p(x_i|\theta_i)p(\theta_i|x_{1:i-1})\\
&= p(\theta_i|x_{1:i})p(x_i)\\
&\propto p(\theta_i|x_{1:i})
\end{aligned}$$

However, this algorithm is not applicable in most applications: we cannot typically sample from $p(\theta_i|\theta_{i-1}^{(k)},x_i)$.

Working around this problem leads to one of the most basic SMC methods, the Sequential Importance Sampling (SIS) algorithm, which is the basis for most SMC methods (Handschin and Mayne 1969; Handschin 1970). We begin with samples from a *proposal distribution*,

$$\theta_1^{(k)} \sim q(\cdot|x_1),$$

and compute the *importance weights*

$$w_1^{(k)} \propto p(x_1|\theta_1^{(k)})\frac{p(\theta_1^{(k)})}{q(\theta_1^{(k)}|x_1)},$$

At each subsequent step,

$$\theta_i^{(k)} \sim q(\cdot|\theta_{i-1}^{(k)}, x_i)$$
$$w_i^{(k)} \propto w_{i-1}^{(k)} p(x_i|\theta_i^{(k)}) \frac{p(\theta_i^{(k)}|\theta_{i-1}^{(k)})}{q(\theta_i^{(k)}|\theta_{i-1}^{(k)}, x_i)},$$

which depend on the previous step's weights. Then, at each time step, we have the approximation

$$\widehat{p}(\theta_i|x_{1:i}) = p(x_i|\theta_i) \sum_k w_{i-1}^{(k)} p(\theta_i|\theta_i^{(k)}).$$

As before, we show that this estimator is unbiased:

$$\begin{aligned}
\mathbb{E}[\widehat{p}(\theta_i|x_{1:i})] &\propto p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})} \cdots \mathbb{E}_{p(\theta_{i-1}^{(k)}|\theta_{i-2}^{(k)})} \left(\sum_{k=1}^{K} w_{i-1}^{(k)} p(\theta_i|\theta_{i-1}^{(k)}) \right) \\
&= p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})} \cdots \mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(k)})} \\
&\qquad \left(\sum_{k=1}^{K} \int w_{i-2}^{(k)} p(x_{i-1}|\theta_{i-1}^{(k)}) p(\theta_i|\theta_{i-1}^{(k)}) \frac{p(\theta_{i-1}^{(k)}|\theta_{i-2}^{(k)})}{q(\theta_{i-1}^{(k)}|\theta_{i-2}^{(k)}, x_{i-1})} q(d\theta_{i-1}^{(k)}|\theta_{i-2}^{(k)}, x_{i-1}) \right) \\
&= p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})} \cdots \mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(k)})} \left(\sum_{k=1}^{K} w_{i-2}^{(k)} p(x_{i-1}, \theta_i|\theta_{i-2}^{(k)}) \right) \\
&\propto p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})} \cdots \mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(k)})} \left(\sum_{k=1}^{K} w_{i-2}^{(k)} p(\theta_i|\theta_{i-2}^{(k)}, x_{i-1}) \right)
\end{aligned}$$

The rest of the proof mirrors the previous proof.

However, this simple algorithm leads to degenerate estimates: the normalized weights tend towards a state where one sample has weight 1 and all other samples have weight 0. To prove this, we first show that $(w_i^{(k)})$ is a martingale, following (Kong, Liu, and Wong 1994). First, we re-express $w_i^{(k)}$ to match their expression:

$$
\begin{aligned}
w_i^{(k)} &\propto w_{i-1}^{(k)} p(x_i|\theta_i^{(k)}) \\
&\propto w_{i-1}^{(k)} \frac{p(\theta_{1:i}^{(k)}|x_{1:i})}{p(\theta_{1:i}^{(k)}|x_{1:i-1})} \\
\mathbb{E}_{p(x_i|x_{1:i-1})}[w_i^{(k)}|\theta_{1:i-1}^{(k)}, x_{1:i-1}] &\propto w_{i-1}^{(k)} \mathbb{E}_{p(x_i|x_{1:i-1})} \left(\frac{p(\theta_{1:i}^{(k)}|x_{1:i})}{p(\theta_{1:i}^{(k)}|x_{1:i-1})} \right) \\
&= w_{i-1}^{(k)} \int \frac{p(\theta_{1:i}^{(k)}|x_{1:i})}{p(\theta_{1:i}^{(k)}|x_{1:i-1})} p(dx_i|x_{1:i-1}) \\
&= w_{i-1}^{(k)} \int p(dx_i|\theta_{1:i}^{(k)}, x_{1:i-1}) \\
&= w_{i-1}^{(k)}
\end{aligned}
$$

Since the expected value of the i^{th} item is the $(i-1)^{\text{th}}$ item, we have a martingale, as desired. From Lemma 2.2, we have $\text{var}(w_i^{(k)}) > \text{var}(w_{i-1}^{(k)})$.

Then, since $\sum_k w_i^{(k)} = 1$, as variance increases, the set $\{w_i^{(k)}\}_k$ must "spread out" to 0 and 1. As at most one element may be 1, all others must be 0, completing our proof.

To correct this problem, we turn to the Sampling/Importance Resampling (SIR) (Rubin 1987; Smith and Gelfand 1992) algorithm (see also (Gordon, Salmond, and Smith 1993; West 1993)). At each time step, we sample (with replacement) from the previous step's population of samples instead of including their weight recursively:

$$
\begin{aligned}
a_i^{(k)} &\sim \sum_{j=1}^{K} w_{i-1}^{(j)} \delta_j \\
\theta_i^{(k)} &\sim q(\cdot|\theta_{i-1}^{(a_i^{(k)})}, x_i) \\
w_i^{(k)} &\propto p(x_i|\theta_i^{(k)}) \frac{p(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})})}{q(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})}, x_i)}.
\end{aligned}
$$

Thus, samples with low weight tend to die out, leaving only samples whose history had higher weight, i.e., samples that better match the data.

This estimator is also unbiased:

$$\begin{aligned}
\mathbb{E}[\widehat{p}(\theta_i|x_{1:i})] &\propto p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-1}^{(k)}|\theta_{i-2}^{(a_{i-1}^{(k)})})p(a_{i-1}^{(k)}|\{w_{i-2}^{(k)}\}_k)}\left(\sum_{k=1}^{K} w_{i-1}^{(k)} p(\theta_i|\theta_{i-1}^{(k)})\right) \\
&= p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(a_{i-2}^{(k)})})p(a_{i-2}^{(k)}|\{w_{i-3}^{(k)}\}_k)} \\
&\qquad \left(\sum_{k=1}^{K}\sum_{j=1}^{K} w_{i-2}^{(j)} \int p(x_{i-1}|\theta_{i-1}^{(k)})p(\theta_i|\theta_{i-1}^{(k)})p(d\theta_{i-1}^{(k)}|\theta_{i-2}^{(j)})\right) \\
&= p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(a_{i-2}^{(k)})})p(a_{i-2}^{(k)}|\{w_{i-3}^{(k)}\}_k)}\left(\sum_{k=1}^{K}\sum_{j=1}^{K} w_{i-2}^{(j)} p(x_{i-1},\theta_i|\theta_{i-2}^{(j)})\right) \\
&\propto p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(a_{i-2}^{(k)})})p(a_{i-2}^{(k)}|\{w_{i-3}^{(k)}\}_k)}\left(\sum_{k=1}^{K}\sum_{j=1}^{K} w_{i-2}^{(j)} p(\theta_i|\theta_{i-2}^{(j)},x_{i-1})\right) \\
&\propto p(x_i|\theta_i)\mathbb{E}_{p(\theta_1^{(k)})}\cdots\mathbb{E}_{p(\theta_{i-2}^{(k)}|\theta_{i-3}^{(a_{i-2}^{(k)})})p(a_{i-2}^{(k)}|\{w_{i-3}^{(k)}\}_k)}\left(\sum_{k=1}^{K} w_{i-2}^{(k)} p(\theta_i|\theta_{i-2}^{(k)},x_{i-1})\right)
\end{aligned}$$

Again the rest of the proof mirrors the original.

A critical weakness remains in this algorithm: because the history indicators $a_i^{(k)}$ do not depend on x_i, $\theta_i^{(k)}$ depends on x_i only through $q(\cdot|\cdot)$, which tends to be close to the prior $p(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})})$. In cases where $p(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})}, x_i)$ differs significantly from $q(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})})$ (as may be the case if x_i is an outlier), the $\theta_i^{(k)}$ will be poor samples, the weights $w_i^{(k)}$ will be highly skewed, and $\widehat{p}(\theta_i|x_{1:i})$ will be inaccurate. The *Auxiliary Particle Filter* of Pitt and Shephard (Pitt and Shephard 2001) addresses this problem by incorporating a forward-looking estimator $g(x_i|\theta_{i-1}^{(k)})$ into the history indicator probabilities:

$$\begin{aligned}
a_i^{(k)} &\sim \sum_{j=1}^{K} w_{i-1}^{(j)} g(x_i|\theta_{i-1}^{(j)})\delta_j \\
w_i^{(k)} &\propto \frac{p(x_i|\theta_{i-1}^{(a_i^{(k)})})}{g(x_i|\theta_{i-1}^{(a_i^{(k)})})}\frac{p(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})},x_i)}{q(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})},x_i)}. \\
&= \frac{p(x_i|\theta_i^{(k)})}{g(x_i|\theta_{i-1}^{(a_i^{(k)})})}\frac{p(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})})}{q(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})},x_i)}.
\end{aligned}$$

If $g(x_i|\theta_{i-1}^{(a_i^{(k)})})$ approximates $p(x_i|\theta_{i-1}^{(k)}) = \int p(x_i|\theta_i^{(k)})p(d\theta_i^{(k)}|\theta_{i-1}^{(k)})$ well, the terms (approximately) cancel out, and weights will be (approximately) equal. Because $p(x_i|\theta_i^{(k)})$ is now represented in the weights $w_i^{(k)}$, we change our estimator to

$$\widehat{p}(\theta_i|x_{1:i}) = \sum_{k=1}^{K} w_i^{(k)} \delta_{\theta_i^{(k)}}.$$

This estimator is unbiased:

$$\begin{aligned}
&\mathbb{E}[\widehat{p}(\theta_i\ |x_{1:i})] \\
&\quad = \mathbb{E}_{q(\theta_1^{(k)}|x_1)} \cdots \mathbb{E}_{q(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})},x_i)p(a_i^{(k)}|\{w_{i-1}^{(k)}\}_k)} \left(\sum_{k=1}^{K} w_i^{(k)} \delta_{\theta_i^{(k)}}(\theta_i) \right) \\
&\quad \propto \mathbb{E}_{q(\theta_1^{(k)}|x_1)} \cdots \mathbb{E}_{q(\theta_{i-1}^{(k)}|\theta_{i-2}^{(a_{i-1}^{(k)})},x_{i-1})p(a_{i-1}^{(k)}|\{w_{i-2}^{(k)}\}_k)} \\
&\qquad \left(\sum_{k=1}^{K} \sum_{j=1}^{K} w_{i-1}^{(j)} g(x_i|\theta_{i-1}^{(j)}) \int \frac{p(x_i|\theta_i^{(k)})}{g(x_i|\theta_{i-1}^{(j)})} \frac{p(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})})}{q(\theta_i^{(k)}|\theta_{i-1}^{(a_i^{(k)})},x_i)} \delta_{\theta_i^{(k)}}(\theta_i) q(d\theta_i^{(k)}|\theta_{i-1}^{(j)}, x_i) \right) \\
&\quad = p(x_i|\theta_i)\mathbb{E}_{q(\theta_1^{(k)}|x_1)} \cdots \mathbb{E}_{q(\theta_{i-1}^{(k)}|\theta_{i-2}^{(a_{i-1}^{(k)})},x_{i-1})p(a_{i-1}^{(k)}|\{w_{i-2}^{(k)}\}_k)} \left(\sum_{k=1}^{K} \sum_{j=1}^{K} w_{i-1}^{(j)} p(\theta_i|\theta_{i-1}^{(j)}) \right) \\
&\quad \propto p(x_i|\theta_i)\mathbb{E}_{q(\theta_1^{(k)}|x_1)} \cdots \mathbb{E}_{q(\theta_{i-1}^{(k)}|\theta_{i-2}^{(a_{i-1}^{(k)})},x_{i-1})p(a_{i-1}^{(k)}|\{w_{i-2}^{(k)}\}_k)} \left(\sum_{k=1}^{K} w_{i-1}^{(k)} p(\theta_i|\theta_{i-1}^{(k)}) \right)
\end{aligned}$$

As before, the rest of the proof is the same.

In addition to the dynamic state θ_i, many models have a static parameter vector ϕ. To handle static parameters, we must maintain a collection of $\phi_i^{(k)}$ drawn approximatly from $p(\phi|x_{1:i})$, which presents two challenges:

1. the target $p(\phi|x_{1:i})$ is intractable, and
2. there is no $p(\phi_i|\phi_{i-1})$ to base a proposal distribution on

However, we can recover a recursive estimator by observing that

$$p(\theta_{1:i}, \phi|x_{1:i}) \propto p(\theta_i|\theta_{i-1}, \phi, x_i)p(\phi|\theta_{1:i-1}, x_{1:i-1})p(\theta_{1:i-1}|x_{1:i-1}).$$

so we can estimate $p(\theta_{1:i-1}|x_{1:i-1})$ by generating an estimate of $p(\theta_{1:i-1}, \phi|x_{1:i-1})$ and discarding the $\phi_{i-1}^{(k)}$. Such methods must sample from $p(\phi|\theta_{1:i-1}, x_{1:i-1})$, which typically has complexity that is at least linear in i. In models where we can take advantage of conjugacy, $p(\phi|\theta_{1:i-1}, x_{1:i-1}) = p(\phi|T_{i-1})$, for some sufficient statistics T_{i-1}; this gives the algorithm from (Storvik 2002). Most models do not enjoy such conjugacy properties, however.

(Liu and West 2001) proposes using a Gaussian Kernel density estimate to approximate $p(\phi|\theta_{1:i}, x_{1:i})$:

$$\begin{aligned}
\bar{\phi}_{i-1} &= \sum_{j=1}^{K} w_{1:i-1}^{(j)} \phi_{i-1}^{(j)} \\
\vec{V}_{i-1} &= \sum_{j=1}^{K} w_{1:i-1}^{(j)} (\phi_{i-1}^{(j)} - \bar{\phi}_{i-1})(\phi_{i-1}^{(j)} - \bar{\phi}_{i-1})^{\top} \\
\vec{m}_{i-1}^{(j)} &= \sqrt{1-h^2}\phi_{i-1}^{(j)} + (1 - \sqrt{1-h^2})\bar{\phi}_{i-1} \\
q(\phi_i^{(k)}|x_{1:i}) &= N(\vec{m}_{i-1}^{(a_i^{(k)})}, h^2 \vec{V}_{i-1}),
\end{aligned}$$

with smoothing parameter h. This is then mixed into the Auxiliary Particle Filter. That is, we approximate $p(\phi|\theta_{1:i}, x_{1:i})$ with a Normal distribution whose mean and standard deviation are derived from the previous iteration's collection of $\phi_{i-1}^{(k)}$ values. The weights $w_i^{(k)}$ and history indicators $a_i^{(k)}$ ensure that each iterations $\{\phi_i^{(k)}\}_k$ are approximate samples of $p(\phi|\theta_{1:i}, x_{1:i})$.

4.2.1 Other SMC Works

Here, we survey other works related to Sequential Monte Carlo techniques. See (Doucet, Freitas, and Gordon 2001; Kantas et al. 2015) for other general surveys. (Cappe, Godsill, and Moulines 2007) surveys recent advances in SMC, and (Liu and Chen 1998) gives guidelines for using SMC techniques. (Doucet, Godsill, and Andrieu 2000) reviews SMC techniques from a specifically Bayesian perspective. (Creal 2012) reviews SMC for use in economics and finance. Some works use the term "Particle Filter" instead of SMC; (Arulampalam et al. 2002; Künsch 2013) review such works. (Fearnhead 1998) reviews SMC and Particle Filters.

There has been a large volume of work specializing SMC methods to certain problems. (Doucett et al. 2000) adapts SMC methods to the setting of Dynamic Bayesian Networks. More generally, (Naesseth, Lindsten, and Schön 2014) develops SMC for Probabilistic Graphical Models. (Andrieu, De Freitas, and Doucet 1999) uses SMC for Bayesian Model Selection. (Naesseth, Lindsten, and Schön 2015) proposes using an SMC estimate as a proposal in a higher-level SMC algorithm for use with high-dimensional data. (Del Moral and Murray 2015) develops a specialized SMC algorithm for use with "highly informative" data, wherein the posterior distribution is very different from the prior. (Wang, Bouchard-Côté, and Doucet 2015; Everitt et al. 2016; Dinh, Darling, and Matsen IV 2017) develop SMC algorithms for infering phylogenetic trees. In the setting of models with both static and dynamic parameters, (Carvalho et al. 2010) uses sufficient statistics in a manner similar to that of (Storvik 2002) discussed above, and (Nemeth, Fearnhead, and Mihaylova 2014) presents an SMC algorithm that adapts to abruptly changing dynamic parameters. Certain models (especially based on physical processes) are defined using partial differential equations. In such models, discretization can lead to growing errors; (Beskos et al. 2017b) adapts an SMC algorithm to this setting.

Another vein of research concentrates on the computational aspects of SMC. In particular, there is interest in using SMC methods on parallel and distributed systems, as discussed in (Cotter, Cotter, and Russell 2015; Vergé et al. 2013; Chen et al. 2011). (Paige et al. 2014) presents a parallel and distributed SMC method that is "anytime" in nature, meaning that it continuosly improves its accuracy (by generating more samples), but can be stopped at any time to give an unbiased estimate.

There has also been more theoretical work on SMC methods. (Kitagawa 1996; Douc and Cappe 2005; Whiteley, Lee, and Heine 2016) discuss the effect of resampling on SMC algorithm, including different resampling distributions and methods to avoid resampling at every iteration. Similarly, (Whiteley and Lee 2014) develops a resampling distribution that

minimizes the growth of variance of the estimator. (Bengtsson, Bickel, and Li 2008; Beskos, Crisan, and Jasra 2014) explore properties of SMC when the data are high dimensional. (Gilks and Berzuini 2001) incorporates MCMC transitions (on the entire history of a sample) into an SMC algorithm. Similarly, (Fearnhead 2002) incorporates MCMC transitions that take advantage of conjugacy and sufficient statistics to remain efficient.

SMC methods can be used for problems other than state-space models. (Del Moral, Doucet, and Jasra 2006) discusses using SMC for sampling from arbitrary sequences of distributions. (Cappé et al. 2004) introduces Population Monte Carlo, which is essentially an SMC algorithm where the target distribution is the same at every iteration. (Jasra, Stephens, and Holmes 2007b) reviews similar algorithms, including ones that incorporate MCMC techniques. (Wraith et al. 2009; Kilbinger et al. 2010) discuss the application of Population Monte Carlo and other algorithms to cosmology. (Jasra, Stephens, and Holmes 2007a) combines these concepts with RJMCMC for trans-dimensional targets.

Similarly, (Andrieu, Doucet, and Holenstein 2010) introduces Particle MCMC, which uses SMC algorithms for proposal distributions within an MCMC algorithm. (Pitt et al. 2012) and (Chopin and Singh 2015) explore the theoretical properties of Particle MCMC, and (Lindsten, Jordan, and Schön 2014) adds a kind of resampling between iterations of the MCMC algorithm. (Bouchard-Côté, Doucet, and Roth 2017) incorporates split-merge moves into Particle MCMC for mixture models, and (Golightly, Henderson, and Sherlock 2014) incorporates delayed acceptance into Particle MCMC. (Meent et al. 2015) applies the algorithm of (Lindsten, Jordan, and Schön 2014) to probabilistic programming.

4.3 Other Inference Techniques

4.3.1 Hamiltonian Monte Carlo

Originally introduced as "Hybrid Monte Carlo" (Duane et al. 1987; Neal 1992), Hamiltonian Monte Carlo (HMC) is a simulation technique based in physics. In physics, Hamiltonian

dynamics represent the time-evolution of systems wherein energy is conserved (i.e., there is no input of energy from outside of the system). The total energy of such systems can be represented as the sum of potential energy (such as from a gravitational or electromagnetic field) and kinetic energy. The basic idea of HMC is to generate samples from some target probability distribution $\pi(x)$ by simulating a dynamical system with potential energy given by $-\log \pi(x)$: first a momentum p (and thereby a kinetic energy) is sampled from some arbitrary distribution $\pi(p|x)$. Then, the Hamiltonian system is simulated for some time t (e.g., using any of the techniques in (Lindsten, Jordan, and Schön 2014)), yielding (x', p'). Because Hamiltonian systems conserve total energy $H = -\log \pi(p|x) - \log \pi(x)$, we have $(x', p') \sim \pi(x, p) = \pi(x)\pi(p|x)$. Discarding p' leaves us with a sample $x' \sim \pi(x)$, as desired (technically, because the simulation is of finite precision, there is a Metropolis-Hastings step to occasionally reject bad samples). See (Betancourt 2017a) for an in-depth and intuitive introduction to HMC methods that explains their efficacy, and see (Betancourt 2017b) for a historical review of MCMC and HMC.

HMC methods have proven to be quite effective when they are applicable, and have thus engendered a lot of study. (Betancourt et al. 2017) and (Barp et al. 2017) provide a theoretical foundation of HMC based in differential geometry; (Betancourt 2014) uses some of these properties to generalize HMC methods. More theoretical work can be found in (Seiler, Rubinstein-Salzedo, and Holmes 2014; Livingstone et al. 2016; Zhang et al. 2016; Betancourt 2015).

Another area of study is the setting of the parameters of the HMC algorithm. (Betancourt 2016b) identifies the optimal integration time (i.e. length of the simulation). Hamiltonian simulations typically proceed by making small "steps;" (Hoffman and Gelman 2014) develops a variant of HMC without the need to tune the size of such steps. (Betancourt 2016a) and (Livingstone, Faulkner, and Roberts 2017) discuss the kinetic energy of the system, i.e., the distribution $\pi(p|x)$.

(Strathmann et al. 2015) HMC for problems w/out tractable gradients: fit a kernellized exponential family model and use it's gradients (or estimate the gradient with the model?) (Sohl-Dickstein, Mudigonda, and DeWeese 2014) HMC w/out rejection (instead, longer paths are simulated) (Tripuraneni et al. 2017) develops a variant of HMC with different dynamics which still preserve energy. (Lan, Streets, and Shahbaba 2014) modifies HMC with a periodic search for modes which are then linked via "wormholes" so that the samples can more effectively explore a multimodal distribution.

The original HMC algorithm is only applicable to simulating real-valued variables with smooth densities with efficiently computable gradients. There has been work (Beskos et al. 2011; Beskos et al. 2017a; Byrne and Girolami 2013; Beskos 2014) on generalizing to non-real spaces, as well as to discrete variables (Pakman and Paninski 2013; Nishimura, Dunson, and Lu 2017) and densities with discontinuities (Afshar and Domke 2015). (Strathmann et al. 2015) and (Stoehr, Benson, and Friel 2017) develop variants that do not require exact gradients.

Finally, (Sohl-Dickstein, Mudigonda, and DeWeese 2014) develops a variant of HMC without rejection (instead, longer paths are simulated), and (Betancourt and Girolami 2015) specializes HMC for hierarchical models.

4.3.2 Simulated Annealing

Plain MCMC is useful for computing posterior means and other summary statistics, but it is not efficient when the goal is to find the *maximum a posteriori* (MAP) estimate (i.e., the mode of the posterior distribution), since it will generate many samples far from the mode, unless the distribution is sharply concentrated around the mode (which it rarely is). A useful alternative is Simulated Annealing (Laarhoven and Aarts 1987; Andrieu, Freitas, and Doucet 2000), wherein instead of sampling from a homogeneous Markov chain that (assymptotically) generates samples from some $\pi(x)$ one samples from an inhomogeneous Markov chain that

generates samples from $\pi_i(x) \propto \pi^{T_i}(x)$, where T_i increases as the simulation continues. As T_i increases, $\pi_i(x)$ becomes more and more concentrated around the mode(s) of $\pi(x)$. By introducing the increasing T_i into the proposals and acceptance ratios of an MCMC sampler, we can therefor produce better MAP estimates. The rate at which T_i increases is itself a subject of study (Hajek 1988; Kiwaki 2002). Alternatively, one can adapt SMC algorithms to target $\pi_i(x)$ (Iba 2003).

If the model being used has so-called "nuisance parameters" α—parameters whose value is not interesting—an alternative is to replace $\pi^{T_i}(x, \alpha)$ with $\prod_{k=1}^{T_i} \pi(x, \alpha^{(k)})$ (Doucet, Godsill, and Robert 2002; Zhao et al. 2015). This is better at integrating out α.

(Andrieu, Freitas, and Doucet 2000) and (Bandyopadhyay 2005) combine the RJMCMC techniques of Section 4.1.1 with simulated annealing.

4.3.3 Approximate Bayesian Computation

For sufficiently complex models, even evaluating the likelihood can be prohibitively expensive. Approximate Bayesian Computation (ABC) (Sunnåker et al. 2013) is a class of algorithms to handle inference in even the case that the prior and likelihood can be simulated but the likelihood cannot be computed. The simplest version is to simulate an artificial set of parameters and data $(\widehat{\theta}, \widehat{x})$, then compare the generated data with the true observed data. If they are similar enough, i.e., $\rho(x, \widehat{x}) < \epsilon$ for some distance metric $\rho(\cdot, \cdot)$, then we accept $\widehat{\theta}$; otherwise, we reject. Thus, the set of accepted $\widehat{\theta}$s, being associated with virtual data similar to the real data, are approximately distributed according to the posterior $p(\theta|x)$.

ABC has been adapted to state-space models (Vakilzadeh, Beck, and Abrahamsson 2018) as well as classification using the popular random forest model (Marin et al. 2016). Additionally, it can be incorporated into other inference mechanisms, such as SMC (Bonassi and West 2015).

4.3.4 Variational Methods

So far, all of the inference techniques we have discussed have been based on simulation and sampling. An alternative approach, *variational inference* (Jordan et al. 1999; Blei, Kucukelbir, and McAuliffe 2017; Wainwright and Jordan 2008) is instead based on optimization. The posterior $p(\theta|x)$ is approximated by the distribution $q(\theta;\phi)$ by solving the optimization problem

$$\arg\min_{\phi} D_{KL}(q(\theta|\phi)\|p(\theta|x)),$$

where $D_{KL}(\cdot\|\cdot)$ is the Kullback-Leibler (KL) divergence (Kullback and Leibler 1951),

$$D_{KL}(f(x)\|g(x)) = \int f(dx)\log f(x) - \int f(x)\log g(x).$$

In general, $D_{KL}(q(\theta|\phi)\|p(\theta|x))$ is not available; instead, an equivalent optimization maximizes the *evidence lower bound*:

$$\arg\max_{\phi} \int q(d\theta|\phi)\log p(\theta,x) - \int q(d\theta|\phi)\log q(\theta|\phi). \tag{4.1}$$

Additionally, other divergence measures may be used (Minka 2005; Ranganath et al. 2016)

Optimizing Equation (4.1) requires calculating integrals and gradients which are not always available; see (Paisley, Blei, and Jordan 2012; Ruiz, Titsias, and Blei 2016) for methods to handle such situations. Even in cases where these gradients are available, it can be time-consuming to derive them. (Kucukelbir et al. 2017) presents a system which uses *automatic differentiation* (Baydin et al. 2018) to allow users to perform inference in a broad class of differentiable models without needing to derive any gradients themselves.

Variational inference has been applied to many problem areas, including applications involving "big data" and scalability (Armagan and Dunson 2011; Ko and Seeger 2012; Dai et al. 2015) as well as streaming data (Broderick et al. 2013; Tank, Foti, and Fox 2015). It has been applied to popular models such as neural networks (Jaakkola and Jordan 1996), Markov Decision Processes (Cheng et al. 2013), and logistic regression (Jaakkola and Jordan

1997; Jaakkola and Jordan 2000)—including a nonparametric multinomial version involving Gaussian Processes (Chai 2012).

4.3.5 The Gumbel-max Trick

Consider the problem of sampling a random variable $x \in \{1, \ldots, m\}$ with $p(x = i) = f(i)/z$ for some unknown normalizing factor z. The "Gumbel-max trick" is as follows: let $G_i, i \in \{1, \ldots, m\}$ be i.i.d. standard Gumbel random variables, meaning that they have density $e^{-(G_i + e^{-G_i})}$. Then, let

$$x^* = \arg\max_i \log f(i) + G_i.$$

Proposition 4.1. $p(x^* = i) = p(x = i)$

Proof. First, note that the cumulative distribution function of G_i is given by $p(G_i < g) = \int_0^g p(dG_i) = e^{-e^{-g}}$. Then,

$$\begin{aligned}
p(x^* = i) &= \int \prod_{j \neq i} p(G_j < \log f(i) - \log f(j) + G_i) p(dG_i) \\
&= \int \prod_{j \neq i} \exp\left(-\exp\left(-\log f(i) + \log f(j) - G_i\right)\right) p(dG_i)
\end{aligned}$$

Letting $G_i' = G_i - \log f(i)$ and expanding its density, we get

$$\begin{aligned}
&= \int \exp\left(-G_i' + \log f(i) - \exp\left(-G_i' + \log f(i)\right)\right) \\
&\quad \times \prod_{j \neq i} \exp\left(-\exp\left(\log f(j)\right) \exp\left(-G_i'\right)\right) dG_i' \\
&= \int \exp\left(-G_i' + \log f(i) - \exp\left(-G_i' + \log f(i)\right)\right) \\
&\quad \times \exp\left(-\exp\left(-G_i'\right) \sum_{j \neq i} f(j)\right) dG_i' \\
&= \int \exp\left(-G_i' + \log f(i) - \exp\left(-G_i' + \log f(i)\right) - \exp\left(-G_i'\right) \sum_{j \neq i} f(j)\right) dG_i' \\
&= f(i) \int \exp\left(-G_i' - f(i) \exp\left(-G_i'\right) - \exp\left(-G_i'\right) \sum_{j \neq i} f(j)\right) dG_i'
\end{aligned}$$

$$= f(i) \int \exp\left(-G'_i - \exp\left(-G'_i\right) \sum_j f(j)\right) dG'_i$$
$$= \frac{f(i)}{\sum_j f(j)}$$

□

Using this technique to sample from $p(x)$ is no more efficient than computing z, but this leads novel approximation algorithms (Papandreou and Yuille 2011; Jaakkola and Hazan 2012; Hazan, Maji, and Jaakkola 2013). These techniques have been generalized to continuous spaces (Maddison, Tarlow, and Minka 2014), and distributions other than the Gumbel (Balog et al. 2017). Additionally, theoretical connections between these techniques and Monte Carlo techniques have been established (Maddison 2016); (Orabona et al. 2014) develops more general theory.

CHAPTER 5

BAYESIAN NONPARAMETRIC RELATIONAL LEARNING WITH THE BROKEN TREE PROCESS[1]

Recently, an increase in the availability and importance of relational datasets—such as social network data or protein interaction data—has lead to increased interest in modelling and learning from such data. Such data are often modelled as *exchangeable arrays*, yielding a particular representation due to Aldous and Hoover. We present a Bayesian nonparametric model based on this representation, which uses a novel process to generate a partition of the data. We present a Reversible Jump MCMC algorithm for inference in this model, and demonstrate the effectiveness of this approach on real-world data.

5.1 Introduction

Relational data—which appear in many areas of science, ranging from social networks to protein interaction networks—are observations of relationships between sets of objects. A useful representation of such data is in the form of arrays of random variables[2], e.g., $R = (R_{ij})$, where i and j index objects x_i. A basic challenge that arises in the study of relational data is that of link prediction, where some entries of R are missing and the goal is to predict these unobserved links based on the observed structure. That is, let R_+ denote the observed part of R, and R_- the unobserved part. The goal is to calculate the posterior predictive distribution $p(R_-|R_+)$. In order to address such challenges, it is necessary to develop rich, flexible models of relational data.

---

[2]We restrict our discussion to binary relations here, but higher-dimensional analogs of our methods are possible.

In this chapter, we propose a probabilistic model of relational arrays based on a novel Bayesian nonparametric prior on partitions of the unit square. This prior allows our model to find irregular co-clusters of the relation entries, leading to a flexible model that can find rich latent structures in the data.

The rest of this chapter is organized as follows: in section 5.2, we review some technical background, and present a representation theorem due to Aldous (Aldous 1981) and Hoover (Hoover 1979). In section 5.3, we review related work and present these works in the light of the Aldous-Hoover theorem. In section 5.4, we introduce our novel model. In section 5.5 we present inference in this model. In section 5.6, we present our experimental results. Finally, in section 5.7, we conclude the chapter.

5.2 Background

A *random array* is a collection X of random variables $(X_{ij})_{i,j\in\mathbb{N}}$ taking values in some set $\mathbf{X}$. Relational data such as graphs can be represented as arrays (in the case of unweighted graphs, these arrays have binary values). An array is called *jointly exchangeable* when $(X_{ij}) \stackrel{d}{=} (X_{\pi(i),\pi(j)})$ for any permutation π of $\mathbb{N}$; X is called *separately exchangeable* when $(X_{ij}) \stackrel{d}{=} (X_{\pi_1(i),\pi_2(j)})$ for any pair (π_1, π_2) of permutations of $\mathbb{N}$.

Theorem 5.1 (Aldous, Hoover)**.** *Let X be a random array. Then, X is separately exchangeable iff*

$$(X_{ij}) \stackrel{d}{=} (F(U_i, U_j, U_{ij})),$$

where

- *$F : [0,1]^3 \to \mathbf{X}$ is a random function*
- *$(U_i)_{i\in\mathbb{N}}$ and $(U_{ij})_{i,j\in\mathbb{N}}$ are collections of i.i.d. $U[0,1]$ random variables.*

We will use this representation to frame our discussion of our work and the works discussed in the next section.

5.3 Related Work

A popular approach to modelling relations is the Stochastic Blockmodel (Wang and Wong 1987), which assumes that the relationships between objects can be explained as class-class interactions: the objects are assigned to classes (either observed classes or through clustering), and each pair of classes is assigned a link probability. In the Aldous-Hoover representation, each cluster gets a column (or row) in a grid. Each grid-block is then associated with a height. Then, each object i is assigned a U_i, which will fall in one of the columns/rows, corresponding to a cluster assignment. $F(U_i, U_j, U_{ij})$ is a threshold function: if U_{ij} is lower than the "height" of the block containing (U_i, U_j), then there is a link between objects i and j.

The Infinite Relational Model (IRM) (Kemp et al. 2006) (and the closely related Infinite Hidden Relational Model (IHRM) (Xu et al. 2006)) first applied the techniques of Bayesian nonparametrics (Hjort et al. 2010; Ghosh and Ramamoorthi 2003) to the Stochastic Blockmodel, using the Dirichlet Process (Antoniak 1974) to assign objects to an unknown number of clusters. In (Ishiguro, Ueda, and Sawada 2012), Ishiguro *et al.* add "relevance" indicators to the IRM allowing objects to be marked as irrelevant noise, and therefore have no affect on the clustering or other parameters of the model. In the Aldous-Hoover representation, there is a fixed L-shaped block that contains all of the "irrelevant" points. Similarly, in (Ohama et al. 2013), Ohama *et al.* modify the IRM to allow particular pairs of objects to be "irrelevant." This corresponds to setting a random subset of the height-map to a particular amount, such that the "irrelevant" subset is independent of the underlying grid structure, and such that every point (U_i, U_j) has the same probability of being "irrelevant."

Another vein of research has been to replace the IRM's Dirichlet Process-based latent clustering with the latent feature model of the Indian Buffet Process (Ghahramani, Griffiths, and Sollich 2006). In (Miller, Griffiths, and Jordan 2009), the IBP assigns latent binary features to each object and a Gaussian-distributed weight matrix and sigmoid function are used to assign link probabilities. Additionally, the model provides the ability to include

Algorithm 5.1: $MP(\lambda, \Theta)$

input : $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_n$, the space to be partitioned (e.g., a square), λ, the budget parameter
output : M, a partition of Θ

begin
  $\tau \leftarrow \sum_i |\Theta_i|$
  $E \sim Exp(\tau)$
  $\lambda' \leftarrow \lambda - E$
  if $\lambda' < 0$ **then**
    return $\{\Theta\}$
  else
    $c \sim U[0, \tau]$
    $d \leftarrow d$ such that $\sum_{i=1}^{d} |\Theta_i| < c < \sum_{i=1}^{d+1} |\Theta_i|$
    $(\Theta_{d<}, \Theta_{d>}) \leftarrow$ split Θ_d at $c - \sum_{i=1}^{d} |\Theta_i|$
    Define $\Theta_<$ and $\Theta_>$ appropriately
    $m_< \sim MP(\lambda', \Theta_<)$
    $m_> \sim MP(\lambda', \Theta_>)$
    return $m_< \cup m_>$

(observed) side-information when available. In (Mørup, Schmidt, and Hansen 2011), the weight matrix and sigmoid function are replaced with Beta-distributed link probabilities combined through a noisy-or function. Like the IRM, these correspond to grid-based threshold functions F. In (Palla and Knowles 2012), the IBP is used to assign latent features to each object, and each feature is further clustered with the Dirichlet Process. Each cluster is assigned a Gaussian-distributed weight, and link probabilities are recovered via a sigmoid function, similar to (Miller, Griffiths, and Jordan 2009).

The Mondrian Process (Roy and Teh 2008) is a Bayesian nonparametric prior over partitions of an n-dimensional unit hypercube[3]. The process is defined recursively, as shown in Algorithm 5.1, wherein the hypercube is split in half by a random hyperplane, and the

[3]In general, the Mondrian Process is defined over any product space $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_n$ where each Θ_i is a one-dimensional simply-connected space, so that one can randomly select a point in Θ_i to split the space into two spaces, and repeat this on the new spaces. We use $\Theta_i = [0, 1]$ for all i, but other spaces—such as trees—can be used.

two halves are split in the next recursion, and so on, with a recursion depth controlled by a "budget" parameter λ and a cost variable E that is exponentially distributed with inverse mean equal to the sum of the lengths of the sides of the hypercube being split, so that costs (tend to) rise as the recursion depth increases. This leads to a more flexible partition of the hypercube than the grids associated with the previous works. (Roy et al. 2006) presents a similar model based on random binary trees. (Blundell and Teh 2013) presents a model based on random rose trees.

Finally, in (Lloyd et al. 2012), Lloyd *et al.* present a model based on the Aldous-Hoover representation, where the random function $F(U_i, U_j, U_{ij}) = H(U_{ij}, \Theta(U_i, U_j))$, where Θ is a random function drawn from a Gaussian Process (Rasmussen and Williams 2006), and $H(a, b)$ returns 1 if $b > a$ and 0 otherwise (in the case of graph data).

See (Orbanz and Roy 2015) for a more thorough review of the theory and existing work in modelling exchangeable relations.

5.4 The Broken Tree Process

5.4.1 Definition

A draw $\mathcal{M} \sim BTP(\lambda, \alpha)$ is as follows:

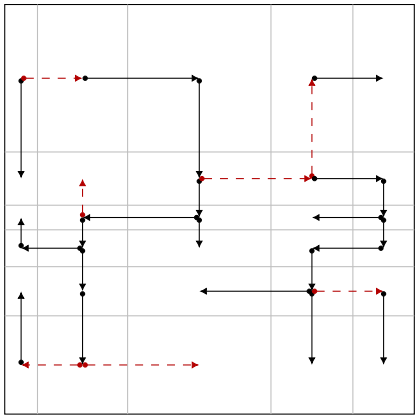
1. Draw

$$m_x \sim Poisson(\lambda) \qquad m_y \sim Poisson(\lambda)$$

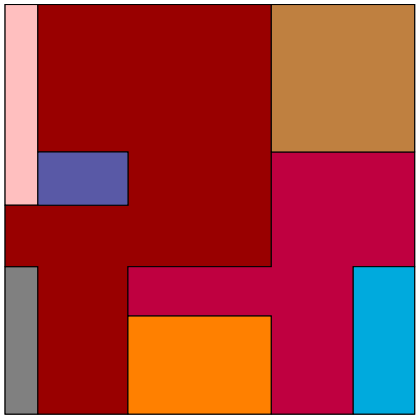
2. Draw

$$\zeta_x^i \sim U(0,1) \text{ for } i \in [1, m_x]$$
$$\zeta_y^i \sim U(0,1) \text{ for } i \in [1, m_y],$$

defining m_x vertical and m_y horizontal lines on the unit square, with offsets given by the corresponding ζs, and thus a grid with $(m_x + 1)(m_y + 1)$ blocks



(a) The structure underlying $\mathcal{M}$, with $m_x = 4, m_y = 5$; shows ζ (as gray lines), $\mathcal{T}$ (as arrows), and $\mathfrak{B}$ (as red dashed arrows)



(b) The partition of $\mathcal{M}$

Figure 5.1: A draw $\mathcal{M} \sim BTP(\lambda, \alpha)$

3. Let $\mathcal{G}$ be the graph whose nodes correspond to the blocks of this grid, with edges between adjacent blocks. Let $\mathcal{T}$ be a uniform random spanning tree of $\mathcal{G}$

4. Each node v (except the root) of $\mathcal{T}$ is independently added to the "break set" $\mathfrak{B}$ with probability α. Then, these "broken" nodes induce a forest by removing the edges from their parents (i.e., each broken node becomes the root of a tree in this forest).

5. For each tree in the forest, we take the union of the grid-blocks associated with its nodes; the collection of such unions defines a partition of the unit square

This process is depicted in Figure 5.1.

The density $p(\mathcal{M}|\lambda, \alpha)$ is given by

$$p(\mathcal{M}|\lambda,\alpha) = \frac{\lambda^{m_x}}{m_x!}e^{-\lambda}\frac{\lambda^{m_y}}{m_y!}e^{-\lambda}\binom{(m_x+1)(m_y+1)-1}{k} \times \frac{\alpha^k(1-\alpha)^{(m_x+1)(m_y+1)-k-1}}{\tau(\mathcal{G}(m_x+1,m_y+1))}$$

where $k = |\mathfrak{B}|$ and $\tau(\mathcal{G}(x,y))$ is the number of possible spanning trees of the grid graph $\mathcal{G}$, given by (Kreweras 1978)

$$\tau(\mathcal{G}(x,y)) = \prod_{i=1}^{m_y-1} \prod_{j=1}^{m_x-1} 4\sin^2(\frac{j\pi}{2m_x}) + 4\sin^2(\frac{i\pi}{2m_y})$$

5.4.2 The BTP Relational Model

To complete our model of relational data, we draw a threshold ϕ_s for each partition $s \in \mathcal{M}$. Then, our Aldous-Hoover F is a threshold function: $F(U_i, U_j, U_{ij}) = 1$ if $U_{ij} > \phi_{s_{ij}}$ where s_{ij} is the partition that contains the point (U_i, U_j). Equivalently, our model is

$$\begin{aligned}
\mathcal{M} &\sim BTP(\lambda, \alpha) \\
\xi_i &\overset{iid}{\sim} U[0,1] \text{ for each object } i \\
\phi_s &\overset{iid}{\sim} Beta(a,b) \text{ for each partition } s \text{ in } \mathcal{M} \\
R_{ij} &\overset{ind}{\sim} Bernoulli(\phi_{s_{ij}})
\end{aligned}$$

Finally, we assign hyperpriors to the parameters (λ, α, a, b):

$$\begin{aligned}
p(\lambda) &\propto \lambda^{-\frac{1}{2}} \\
p(\alpha) &= Beta(\alpha; \frac{1}{2}, \frac{1}{2}) \\
p(a,b) &= e^{-(a+b)}
\end{aligned}$$

Here, we have placed an (improper) uninformative Jeffreys prior on λ, and a (proper) Jeffreys prior on α. The prior $p(a,b)$ is equivalent to placing a uniform prior on the mean $\frac{a}{a+b}$, and a vague exponential prior on the "sample size" $a+b$.

5.5 Inference

We are interested in computing

$$p(R_-|R_+) = \int p(R_-|\psi)p(\psi|R_+)d\psi$$

where $\psi = (\mathcal{M}, \xi, \phi, \lambda, \alpha, a, b)$. Because this integral is intractable, we use the Reversible Jump Markov Chain Monte Carlo techniques of Section 4.1.1.

5.5.1 Inference in the BTPRM

For our model, $p(k, \theta^{(k)}|x) \equiv p(\psi|R_+)$, so we have

$$\alpha_m^*(\psi, \psi') = \underbrace{\frac{p(R|\psi')}{p(R|\psi)}}_{\substack{\text{likelihood} \\ \text{ratio}}} \underbrace{\frac{p(\psi')}{p(\psi)}}_{\substack{\text{prior} \\ \text{ratio}}} \underbrace{\frac{j_{m^*}(\psi', \psi)}{j_m(\psi, \psi')}}_{\substack{\text{proposal} \\ \text{ratio}}} \underbrace{\left|\frac{\partial(\theta^{(k')}, u^{(k')})}{\partial(\theta^{(k)}, u^{(k)})}\right|}_{\text{Jacobian}}$$

where $\theta^{(k)}$ contains all of the real-valued variables in ψ.

Our sampler uses a Gibbs sampling structure, cycling through each component of ψ and sampling (sometimes approximately) from $p(\psi_i|R_+, \psi_{-i})$.

Sampling $\mathcal{M}$ To sample $\mathcal{M}$ from $p(\mathcal{M}|R_+, \lambda, \alpha, \xi)$, we use a RJMCMC sampler with six moves:

1. *Tree:* Sample a new tree $\mathcal{T}$ that is consistent with the current partition by uniformly sampling subtrees in each block. (This is its own inverse.)

2. *Split:* First select a dimension, then uniformly select an interval (i.e., a pair of adjacent ζ), then add a new ζ at uniform between them and increment the appropriate m_x or m_y. Sample a new tree on the expanded grid. (This is the inverse of *Merge*.)

3. *Merge:* First select a dimension, then uniformly select a ζ and remove it. Decrement m_x or m_y as appropriate. Sample a new tree on the reduced grid. (This is the inverse of *Split*.)

4. *Warp:* First select a dimension, then uniformly pick a ζ, and move it to a uniform location between its neighbors. The tree does not change. (This is its own inverse.)

5. *Fall:* Choose a broken node or the root at random, then either replace it with one of its children (with probability $1-\alpha$) or add the child (and keep the parent). If the child is already broken, adding it has no effect. If the root is chosen, it is neither added nor removed from $\mathfrak{B}$. (This is the inverse of *Rise*.)

6. *Rise:* Choose a broken node at random, then either replace it with its parent (with probability $1-\alpha$) or add the parent (and keep the child). If the parent is already broken or the root, adding it has no effect. (This is the inverse of *Fall*.)

To choose a move m, we first choose a category among $\{\mathit{Tree}\}$, $\{\mathit{Split}_x, \mathit{Merge}_x\}$, $\{\mathit{Split}_y, \mathit{Merge}_y\}$, $\{\mathit{Warp}_x, \mathit{Warp}_y\}$, and $\{\mathit{Fall}, \mathit{Rise}\}$. Let $\mathbb{M}$ be the number of valid categories; in general, $\mathbb{M} = 5$, but there are special cases: if $m_x = 0$ or $m_y = 0$ then there can be no *Tree* move; if both are zero, there can be no *Merge*, *Warp*, *Rise*, or *Fall* moves. Thus, we choose among the valid categories at uniform (with probability $1/\mathbb{M}$), then a move is chosen uniformly from the chosen category.

After selecting a move type m, we generate the proposal $\mathcal{M}'$, and then accept the proposal with acceptance ratio $\alpha^*_m(\mathcal{M}, \mathcal{M}') = \mathcal{L}r(\mathcal{M}, \mathcal{M}')\mathcal{A}(m)$, where $\mathcal{L}r(\mathcal{M}, \mathcal{M}')$ is the likelihood ratio,

$$\mathcal{L}r(\mathcal{M}, \mathcal{M}') = \frac{\prod_{s\in\mathcal{M}'} B(a + n'^{+}_s, b + n'^{-}_s)}{\prod_{s\in\mathcal{M}} B(a + n^{+}_s, b + n^{-}_s)}$$

where s varies over the blocks of $\mathcal{M}$, and n^+_s (resp. n^-_s) are the number of positive (resp. negative) entries in s, i.e., the number of points $(\xi_i, \xi_j) \in s$ such that $R_{ij} = 1$ (resp. $R_{ij} = -1$). The $\mathcal{A}(m)$ are given by:

$$\begin{aligned}
\mathcal{A}(\mathit{Tree}) &= 1 \\
\mathcal{A}(\mathit{Split}_x) &= \frac{\lambda}{m_x + 1}(1-\alpha)^{m_y+1}\frac{((m_x+2)(m_y+1)-1)^{\underline{k}}}{((m_x+1)(m_y+1)-1)^{\underline{k}}}(\zeta^{\rightarrow} - \zeta^{\leftarrow}) \\
\mathcal{A}(\mathit{Merge}_x) &= \frac{m_x}{\lambda}\alpha^{k'-k}(1-\alpha)^{-(m_y+1)+k'-k}k^{\underline{k-k'}}\frac{(m_x(m_y+1)-1)^{\underline{k'}}}{((m_x+1)(m_y+1)-1)^{\underline{k}}}(\zeta^{\rightarrow} - \zeta^{\leftarrow})^{-1} \\
\mathcal{A}(\mathit{Warp}_x) &= 1
\end{aligned}$$

$$\mathcal{A}(\textit{Fall}) = |C(\beta^\dagger)| \begin{cases} \frac{k+1}{k} & \beta^\dagger \notin \mathfrak{B}' \wedge \beta^* \notin \mathfrak{B} \\ \frac{k+1}{k-1}\frac{k}{(m_x+1)(m_y+1)-k} & \beta^\dagger \notin \mathfrak{B}' \wedge \beta^* \in \mathfrak{B} \\ \frac{(m_x+1)(m_y+1)-k-1}{k+1} & \beta^\dagger \in \mathfrak{B}' \wedge \beta^* \notin \mathfrak{B} \\ \frac{k+1}{k} & \beta^\dagger \in \mathfrak{B}' \wedge \beta^* \in \mathfrak{B} \end{cases}$$

$$\mathcal{A}(\textit{Rise}) = |C(\beta^*)|^{-1} \begin{cases} \frac{k}{k+1} & \beta^\dagger \notin \mathfrak{B}' \wedge \beta^* \notin \mathfrak{B} \\ \frac{k}{(m_x+1)(m_y+1)-k} & \beta^\dagger \notin \mathfrak{B}' \wedge \beta^* \in \mathfrak{B} \\ \frac{k}{k+2}\frac{(m_x+1)(m_y+1)-k-1}{k+1} & \beta^\dagger \in \mathfrak{B}' \wedge \beta^* \notin \mathfrak{B} \\ \frac{k}{k+1} & \beta^\dagger \in \mathfrak{B}' \wedge \beta^* \in \mathfrak{B} \end{cases}$$

where

- $\beta^\dagger$ is the randomly selected broken node, and β^* is the newly generated broken node;
- $\zeta^{\leftarrow}$ and $\zeta^{\rightarrow}$ are the ζs to the left and right of the added or removed ζ;
- $x^{\underline{y}}$ denotes the falling factorial $x(x-1)\cdots(x-y+1)$
- $\mathcal{A}(\textit{Split}_y), \mathcal{A}(\textit{Merge}_y)$, and $\mathcal{A}(\textit{Warp}_y)$ are the same as their x-counterpoint, with m_x and m_y exchanged.

Note that for the *Tree* move, $\mathcal{L}r(\mathcal{M}, \mathcal{M}') = 1$ as the partitioning does not change. Thus, *Tree* moves are always accepted.

Sampling ξ Next, we sample ξ from $p(\xi|\mathcal{M}, R_+)$. We sample each ξ_i uniformly, then accept with $\alpha^*(\xi, \xi') = \mathcal{L}r(\xi, \xi')$ (since the prior and proposal ratios are both one). The marginal posterior $p(\xi_i|R_+, \mathcal{M}, \xi^-)$ (where ξ^- denotes all of ξ except ξ_i) is a piecewise-constant function, as the probability of ξ_i only changes when some corresponding points ξ_j move from one partition to another. We can therefore sample from the constant segments, then sample at uniform within that segment. Let χ_i denote the sampled segment. Then,

$$p(\chi_i|R_+, \mathcal{M}, \xi^-) \propto |\chi_i| \prod_s B(a + n_s^+(\chi_i), b + n_s^-(\chi_i))$$

Because each χ_i can only take on a relatively small number of values, we can sample from this distribution directly.

Sampling λ Given the prior in Section 5.4.2, the posterior on λ is given by

$$p(\lambda|\mathcal{M}) = G(m_x + m_y + \frac{1}{2}, 2)$$

which we can sample exactly.

Sampling α Given the prior in Section 5.4.2, the posterior on α is given by

$$p(\alpha|\mathcal{M}) = Beta(k + \frac{1}{2}, (m_x + 1)(m_y + 1) - k + \frac{1}{2})$$

which can also be exactly sampled.

Sampling (a, b) The prior for (a, b) yields the posterior

$$p(a, b|R_+) \propto e^{-(a+b)} \prod_s \frac{B(a + n_s^+, b + n_s^-)}{B(a, b)}$$

which is not directly sampleable, so we use a Metropolis-Hastings proposal

$$a' \sim N(a, \sigma^2)$$

$$b' \sim N(b, \sigma^2)$$

and accept the proposal with

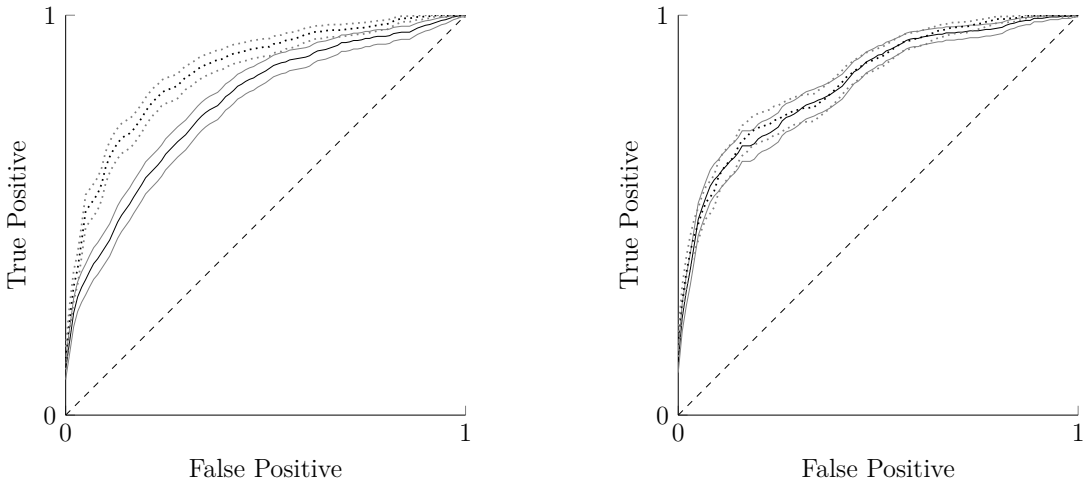
$$\alpha^*((a, b), (a', b')) = \mathcal{L}r((a, b), (a', b'))e^{(a+b)-(a'+b')}$$

(because the proposal distribution is symmetric, the proposal ratio is 1).

5.6 Experiments

We compare the performance of our model against the IRM (Kemp et al. 2006) on real-world data. For each dataset, we ran 5-fold cross validation, running both the BTPRM and IRM on the same folds, generating 10,000 samples per fold per method.

We use two datasets from the classic *Countries* collection (Wasserman and Faust 1994). Both are relations on 24 countries regarding trade. The first reports the trade of food and live animals, and the second reports the trade of minerals and fuel.



(a) Results on the Food and Live Animals Trade Data

(b) Results on the Mineral and Fuel Trade Data

Figure 5.2: ROC curves with 95% credible intervals. ≡ BTPRM; ⁞⁞ IRM

The results of our experiments are shown in Figure 5.2, which shows Receiver Operating Characteristics (ROC) curves for the BTPRM and IRM, with a 95% credible interval computed by treating the True Positive rate as a Binomial random sample with a Jeffreys' $Beta(1/2, 1/2)$ prior for each False Positive rate. Thus, the central line is the posterior mean, and the upper and lower bounds of the shaded region are the 97.5^{th} and 2.5^{th} percentiles, respectively. We can then compute the area under the curve (AUC) for each curve (using AUC of the upper and lower curves as upper and lower bounds on a credible interval for the AUC). For the food and live animal trade dataset, the IRM outperforms the BTPRM slightly: the IRM achieves an AUC of 0.86 (credible interval $(0.84, 0.88)$), compared to the BTPRM's 0.78 (credible interval $(0.75, 0.80)$). For the mineral and fuel trade dataset, the two methods are essentially indistinguishable: the IRM achieves and AUC of 0.84 (credible interval $(0.81, 0.86)$), and the BTPRM achieves and AUC of 0.83 (credible interval $(0.80, 0.85)$).

5.7 Conclusion

We have presented a novel Bayesian nonparametric approach, the BTPRM, to the problem of relational modeling, and link prediction in particular. The BTPRM is based on a novel distribution over partitions of the unit square and the Aldous-Hoover representation theorem. The partitions used by the BTPRM are more flexible than previous work such as the IRM (Kemp et al. 2006) or MPRM (Roy and Teh 2008), thus leading to a more flexible model.

Experimental results show that the BTPRM performs comparably to the IRM, which has the advantage of a large body of work leading to very efficient inference machinery. We expect that future work to develop similarly sophisticated and specialized inference techniques for the BTP would lead to improved performance relative to the IRM. For example, there may be better RJMCMC moves than those used in this chapter, or it may be possible to use other techniques such as variational inference.

We believe that the development of highly expressive nonparametric priors and associated inference techniques will lead to a strong theoretical foundation for future techniques in machine learning and artificial intelligence. Accordingly, future work should also include developing extensions to and variations of the BTP. Additionally, applications to tasks other than relational modelling should be investigated; for example, the Mondrian process (which was also originally used for relational modeling) has recently been applied to classification and regression tasks (Lakshminarayanan, Roy, and Teh 2014, 2016).

CHAPTER 6

ONLINE CLASSIFICATION OF NONSTATIONARY STREAMING DATA WITH DYNAMIC PITMAN-YOR DIFFUSION TREES[1]

In Artificial Intelligence and Machine Learning, there is a need for flexible, expressive models of uncertainty. In the case of online classification, such models should be able to adapt to the dynamics of the data-generating system, i.e., they should be nonstationary. We introduce the Dynamic Pitman-Yor Diffusion Tree (DPYDT), a generalization of the Pitman-Yor Diffusion Tree (PYDT) (Knowles and Ghahramani 2010) to nonstationary streaming data. These Bayesian nonparametric priors model hierarchical structure in the data, providing interpretable structural information about patterns in the data. Our model allows this structure to evolve over time in response to changes in the data distribution. We give a description of the generative process and derive closed form expressions for the joint density of a sequence of trees, and the predictive density of successive trees. We also discuss generalizations of the diffusion underlying the PYDT to discrete variables. Finally, we describe a Sequential Monte Carlo algorithm for inference in our model, and discuss its efficiency.

6.1 Introduction

In online classification, data points arrive sequentially, and the learning algorithm attempts to predict the labels of arriving points. After the prediction, the label of the new data point is provided, and the model is updated to reflect the new information. One framework for online learning is the Bayesian probabilistic paradigm, in which the posterior distribution of model parameters is continually updated as data points arrive.

[1]

Tree structures in particular have great value in machine learning, as they can capture complex dependencies between data points while remaining interpretable. One of the most popular nonparametric methods is the Dirichlet Process Mixture Model (see Section 3.1), which assumes a flat structure in which different mixture components (i.e., clusters) are independent. The Dirichlet Diffusion Tree (Neal 2001) and its generalization the Pitman-Yor Diffusion Tree (Knowles and Ghahramani 2010) instead incorporate hierarchical structure.

These models are applied in the setting of stationary, exchangeable distributions, where it is assumed that all data is available at once, and the order of the data is irrelevant. We describe a modification of the Pitman-Yor Diffusion Tree that extends it to the case where the data are nonstationary, i.e., their distribution may change over time, as the underlying system evolves.

The rest of this chapter is organized as follows: in Section 6.2, we review prior work culminating in the PYDT and its properties. In Section 6.3, we present our dynamic process. In Section 6.4, we present details of inference in our model. In Section 6.5, we present some experimental evaluation. Finally, in Section 6.6, we give some example directions for future work.

6.2 Background

We begin in the offline setting, where data is assumed to be available all-at-once, and the distribution is assumed to be stationary and exchangeable; that is, the order of the data does not change the joint distribution of the data. In this setting, the Dirichlet Process and Pitman-Yor Process Mixture models of Sections 3.1 and 3.2 are popular, computationally tractable choices.

A limitation of the DPMM and PYPMM is that each mixture component has independent parameters. A more flexible alternative is to use a prior that produces dependence between components through a hierarchical structure. The Dirichlet Diffusion Tree (Neal 2001) is a

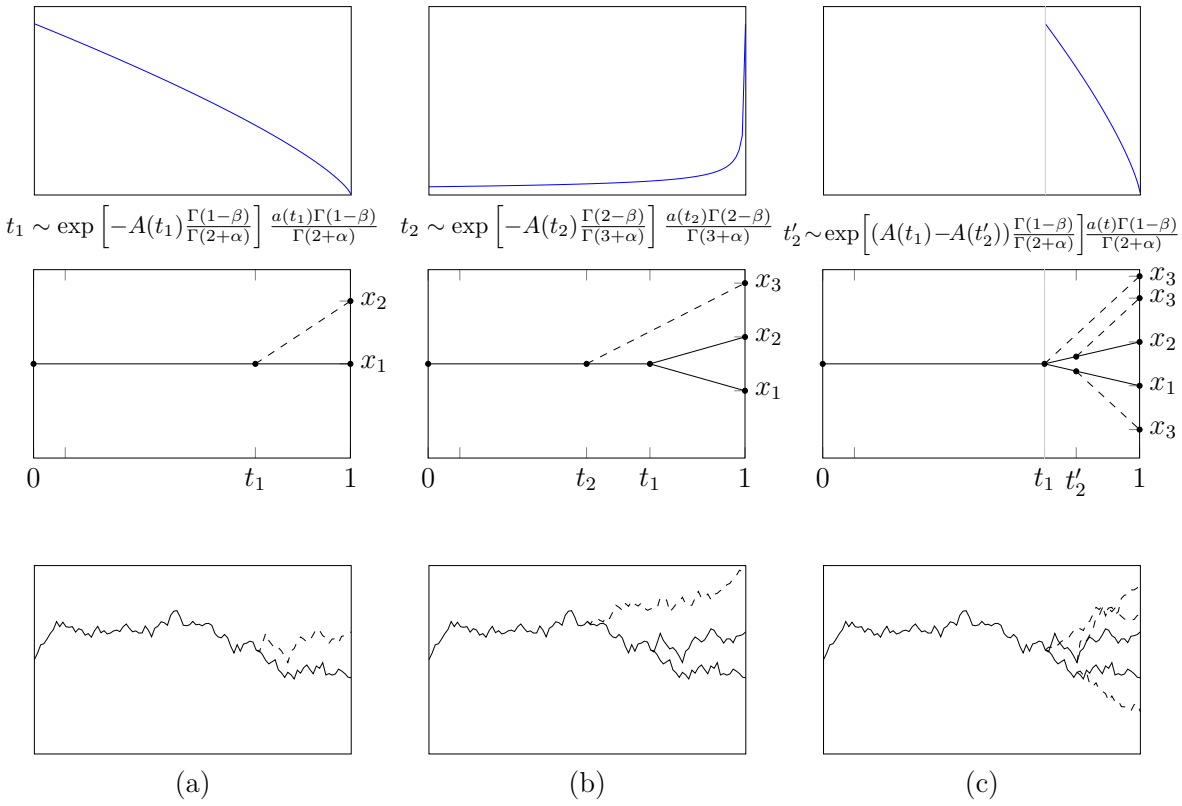


Figure 6.1: The Pitman-Yor Diffusion Tree. *Top*: divergence densities. *Middle*: tree structure. *Bottom*: diffusion paths. (a) x_1 diffuses from $t = 0$ to $t = 1$, and x_2 diverges from the path to x_1 at t_1, drawn from the shown density. (b) If $t_2 < t_1$, x_3 diverges. (c) If $t_2 > t_1$, x_3 may branch or follow an existing branch then diverge at t_2'.

prior that generalizes the Dirichlet process to hierarchical structures, and the Pitman-Yor Diffusion Tree (Knowles and Ghahramani 2010) is likewise a tree-structured generalization of the Pitman-Yor Process.

We can describe the data-generating distribution of the PYDT sequentially, as with the Chinese Restaurant Process description of the Dirichlet Process. This is illustrated in Figure 6.1. Consider particles undergoing random diffusion from the root at $t = 0$ to leaves at $t = 1$. The first particle starts at some origin, then follows Brownian motion[2] until $t = 1$. Then, the i^{th} particle initially follows the path of the previous particles. At each infinitesimal

[2]We generalize to other continuous-time Markov processes in Section 6.3.2

time interval $[t, t+dt]$, the i^{th} particle will diverge from this path with probability

$$\frac{a(t)\Gamma(m_{i-1}^{(t)} - \beta)dt}{\Gamma(m_{i-1}^{(t)} + 1 + \alpha)},$$

where $a(t)$ is a parameter called the "divergence function," $\Gamma(\cdot)$ is the Gamma function, and $m_{i-1}^{(t)}$ is the number of particles that have followed the path up to this divergence point before the current particle. We assume that $\int_0^1 a(t)dt = \infty$, as this guarantees that each particle will eventually diverge, so that the values at the leaves will be drawn from a nonatomic distribution for any continuous feature. If particle i reaches the point where previous particles have diverged, it follows existing branch k with probability

$$\frac{b_{i-1}^{(t,k)} - \beta}{m_{i-1}^{(t)} + \alpha}$$

or creates a new branch with probability

$$\frac{\alpha + \beta K_{i-1}^{(t)}}{m_{i-1}^{(t)} + \alpha},$$

where $K_{i-1}^{(t)}$ is the number of existing branches, and $b_{i-1}^{(t,k)}$ is the number of particles that have followed branch k before. When a particle diverges or creates a new branch, it independently diffuses until $t = 1$; otherwise, it follows the path previous particles have taken.

Next, we turn to the density of divergence times, $p(t)$:

$$\begin{aligned}
&p(\text{go from } s \text{ to } t \text{ without diverging}) \\
&= \lim_{k\to\infty} \prod_{i=0}^{k-1} p(\text{go from } s_i = s + (i-1)(t-s)/k \text{ to } t_i = s + i(t-s)/k \text{ without diverging}) \\
&= \lim_{k\to\infty} \prod_{i=0}^{k-1} \left(1 - \frac{a(t_i)\Gamma(m-\beta)}{\Gamma(m+1+\alpha)}(t_i - s_i)\right) \\
&= \lim_{k\to\infty} \prod_{i=0}^{k-1} \left(1 - \frac{a(s_{i+1})\Gamma(m-\beta)}{\Gamma(m+1+\alpha)}(s_{i+1} - s_i)\right)
\end{aligned}$$

$$= \exp\left(\lim_{k\to\infty}\sum_{i=0}^{k-1}\log\left(1-\frac{a(s_{i+1})\Gamma(m-\beta)}{\Gamma(m+1+\alpha)}(s_{i+1}-s_i)\right)\right)$$

Using the fact that $\lim_{x\to 0}\frac{\log(1-ax)}{-ax}=1$, i.e. $\log(1-ax)\sim_0 -ax$, we can substitute:

$$= \exp\left[\lim_{k\to\infty}-\sum_{i=0}^{k-1}\frac{a(s_{i+1})\Gamma(m-\beta)}{\Gamma(m+1+\alpha)}(s_{i+1}-s_i)\right]$$

$$= \exp\left[-\int_s^t a(t)dt\frac{\Gamma(m-\beta)}{\Gamma(m+1+\alpha)}\right]$$

$$= \exp\left[(A(s)-A(t))\frac{\Gamma(m-\beta)}{\Gamma(m+1+\alpha)}\right]$$

Thus,

$$p(t) = \exp\left[(A(s)-A(t))\frac{\Gamma(m_{i-1}^{(t)}-\beta)}{\Gamma(m_{i-1}^{(t)}+1+\alpha)}\right]\frac{a(t)\Gamma(m_{i-1}^{(t)}-\beta)}{\Gamma(m_{i-1}^{(t)}+1+\alpha)}, \tag{6.1}$$

where $A(t)=\int_0^t a(t)dt$, and s is the time of the last branch (or 0). (If the sampled t is beyond an existing branch point, then the particle follows or branches as above, and has a new chance to diverge if it follows an existing path.)

6.3 The Dynamic Pitman-Yor Diffusion Tree

Turning to the online setting, we drop the assumption that the data distribution is stationary. In the sequential picture of the PYDT, as each data point arrives, the underlying tree grows by one leaf. This presents two problems: first, as $m_{i-1}^{(t)}$ (the number of particles that have traversed a path) grows, the factor $\Gamma(m_{i-1}^{(t)}-\beta)/\Gamma(m_{i-1}^{(t)}+1+\alpha)$ in the divergence density approaches zero, leading to later and later divergence times, so that new data points will be more and more likely to be distributed like previous data points. Second, to make predictions about the next data point, the entire tree must be known, leading to slower inference as time goes on. To adapt the PYDT to the non-stationary setting, we allow subtrees to be removed as new leaves are added. This means that the distribution tends to "forget" older leaves, leading to both gradual and sudden changes in distribution.

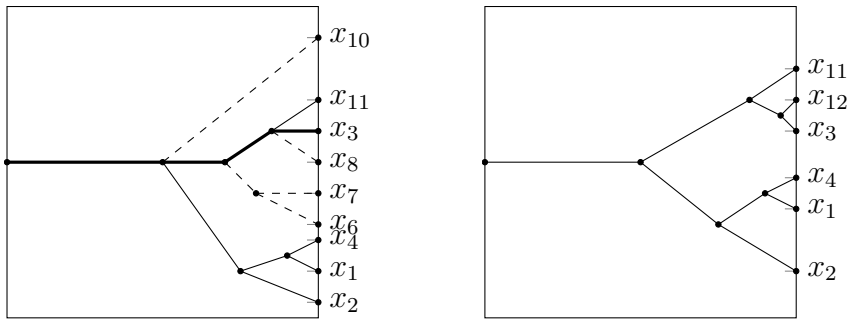


Figure 6.2: Deletion in the Dynamic Pitman-Yor Diffusion Tree. (a) As x_{12} is added to the tree, it follows the bold path to x_3 before diverging. At the intervening branches, some subtrees are deleted (dashed lines). (b) The new tree. Diffusion values are not shown.

The deletion process (illustrated in Figure 6.2) is as follows: each time a particle reaches a branch point at t, each subtree $\mathbb{T}_{i-1}^{(t,k)}$ rooted at that point is removed with probability $\rho(\mathbb{T}_{i-1}^{(t,k)})$, where $\rho(\cdot)$ is a parameter and is bounded from above by 1. Let $\widetilde{m}_{i-1}^{(t)}$ and $\widetilde{K}_{i-1}^{(t)}$ denote the counts and branching factors after this deletion, but before the new leaf is added, so that $m_i^{(t)} = \widetilde{m}_{i-1}^{(t)} + 1$, and $K_i^{(t)} = \widetilde{K}_{i-1}^{(t)} + 1$ if the new data point creates a new branch at t, and $K_i^{(t)} = \widetilde{K}_{i-1}^{(t)}$ otherwise. Particle i's probability of choosing an existing branch versus creating a new branch depends on these post-deletion numbers.

6.3.1 Probability Densities

Let $\mathbb{T}_{1:i}$ denote the sequence of trees $\mathbb{T}_i$ that generate the observations x_1 through x_i as their leaves. We cannot represent the (uncountably infinite) paths between nodes on these trees, but our requirement that the diffusion be a Markov process allows us to just sample the diffusion at the nodes. We are interested in deriving $p(\mathbb{T}_{1:i}|\theta)$, where $\theta = (\alpha, \beta, a(\cdot), \rho(\cdot), \psi)$ are the parameters of the model, with ψ being the parameters of the diffusion (discussed in Section 6.3.2).

To achieve a succinct, readable result, we require some additional notation:

- In a slight abuse of notation, let t identify a unique point on the tree, with the understanding that the application $a(t)$ applies $a(\cdot)$ only to the time component of t.

- Let ξ_t denote the diffusion value at t.

- Let $\mathbb{T}_i$ be considered a collection of intervals $[s,t]$, one for each segment of the tree. Likewise, $\mathbb{T}_{1:i}$ is the union of these collections.

- For each t, let $\mathfrak{m}_i^{(t)}$ be the list of $m_{j-1}^{(t)}$ values for every particle $j \leq i$ that traverses t.

- Let $m_i^{*(t)}$ denote the $m_{j-1}^{(t)}$ for the $j \leq i$ that first diverged at t.

- Let $\mathfrak{r}_i^{(t)}$ denote the list of subtrees $\mathbb{T}_{j-1}^{(t,k)}$ for every branch k that is retained by particle $j \leq i$ at the branch point t, and let $\mathfrak{r}'^{(t)}_i$ denote the subtrees for deleted branches.

- Let $\mathfrak{b}_i^{(t)}$ denote the list of pairs (m,k), where $m = \widetilde{m}_{j-1}^{(t)}$ and $k = \widetilde{K}_{j-1}^{(t)}$ for each particle $j \leq i$ that creates a new branch at t.

- Let $\mathfrak{f}_i^{(t)}$ denote the list of pairs (m,b), where $m = \widetilde{m}_{j-1}^{(t)}$ and $b = b_{j-1}^{(t,k)}$ for each particle $j \leq i$ that follows an existing branch at t.

Using this notation, the joint distribution of a sequence of trees is given by

$$
\begin{aligned}
p(\mathbb{T}_{1:i}|\theta) = & \prod_{\mathbb{T}\in\mathfrak{r}_i^{(0)}} (1-\rho(\mathbb{T})) \prod_{\mathbb{T}\in\mathfrak{r}'^{(0)}_i} \rho(\mathbb{T}) \\
& \times \prod_{[s,t]\in\mathbb{T}_{1:i}} a(t) \frac{\Gamma(m^{*(t)}-\beta)}{\Gamma(m^{*(t)}+1+\alpha)} \\
& \quad \times \prod_{m\in\mathfrak{m}_i^{(t)}} \exp\left[(A(s)-A(t)) \frac{\Gamma(m-\beta)}{\Gamma(m+1+\alpha)}\right] \\
& \quad \times \prod_{\mathbb{T}\in\mathfrak{r}_i^{(t)}} (1-\rho(\mathbb{T})) \prod_{\mathbb{T}\in\mathfrak{r}'^{(t)}_i} \rho(\mathbb{T}) \\
& \quad \times \prod_{(m,k)\in\mathfrak{b}_i^{(t)}} \frac{\alpha+\beta k}{m+\alpha} \prod_{(m,b)\in\mathfrak{f}_i^{(t)}} \frac{b-\beta}{m+\alpha} \\
& \quad \times \; p(\xi_t|\xi_s,\psi,t-s)
\end{aligned}
$$

Additionally, the predictive distribution is given in terms of $\mathcal{P}_i$, which is the path from the root to the last node *before* the new leaf in the new tree (as all changes to the tree take place along this path):

$$
\begin{aligned}
&p(\mathbb{T}_i|\mathbb{T}_{i-1},\theta) \\
&\quad= \begin{cases} 1-\rho(\mathbb{T}_{i-1}) & \mathbb{T}_{i-1}\in \mathfrak{r}_i^{(0)} \\ \rho(\mathbb{T}_{i-1}) & \mathbb{T}_{i-1}\in \mathfrak{r}'^{(0)}_i \end{cases} \\
&\quad\times \prod_{[s,t]\in\mathcal{P}_i} \exp\left[(A(s)-A(t))\frac{\Gamma(m_{i-1}^{(t)}-\beta)}{\Gamma(m_{i-1}^{(t)}+1+\alpha)}\right] \\
&\qquad\qquad \times \begin{cases} p(\xi_t|\xi_s,\xi_{s^+},\psi,t-s,s^+-t) & \\ \quad\times a(t)\frac{\Gamma(m_{i-1}^{(t)}-\beta)}{\Gamma(m_{i-1}^{(t)}+1+\alpha)}D & \mathcal{P}_i \text{ diverges at } t \\ R\frac{\alpha+\beta\widetilde{K}_{i-1}^{(t)}}{m_{i-1}^{(t)}+\alpha}D & \mathcal{P}_i \text{ branches at } t \\ R\frac{b_{i-1}^{(t,k)}-\beta}{m_{i-1}^{(t)}+\alpha} & \mathcal{P}_i \text{ follows } k \text{ at } t, \end{cases}
\end{aligned}
$$

where $R=\prod_{\mathbb{T}\in\mathfrak{r}_i^{(t)}}(1-\rho(\mathbb{T}))\prod_{\mathbb{T}\in\mathfrak{r}'^{(t)}_i}\rho(\mathbb{T})$ is the joint probability of all branch deletions and retentions at t, and $D=p(\xi_1|\xi_t,\psi,1-t)$ is the probability of diffusing from the final branch or diverge to the leaf.

6.3.2 Diffusions

In the original formulation of the DDT and PYDT, the particle diffusion was defined to be Brownian motion, so that $\xi_{t+dt}\sim N(\xi_t,\sigma^2 dt)$. We are interested in performing inference on data that may have discrete elements, and are in particular interested in predicting a discrete classification label. For any diffusion, we assume that the diffusion is a continuous-time Markov process, since we will need its value at arbitrary times, and we cannot simulate or store the full path. Note that the diffusion need not remain stationary between data points: it is acceptable for the diffusion distribution to update based on (sufficient statistics of) the data.

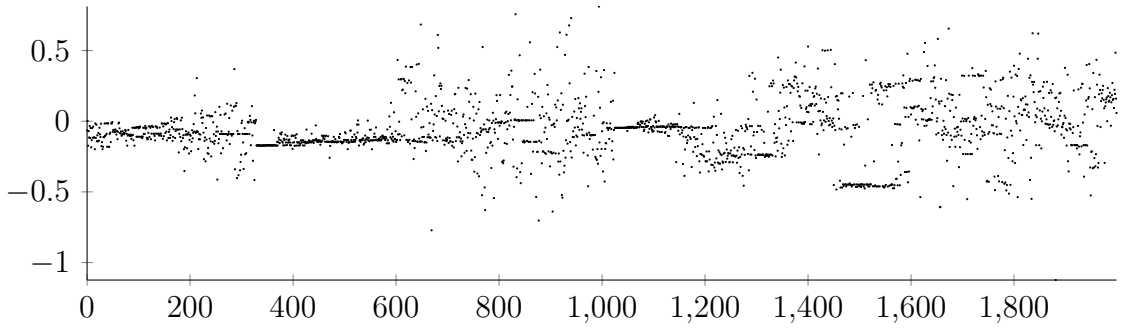


Figure 6.3: A sample from the DPYDT of length 2000 using Brownian diffusion. This sample displays the nonstationary nature of the DPYDT; it can produce both sudden changes in distribution (e.g., the sudden appearance of a tight cluster around sample 1,500) and smooth changes (e.g., the gradual drifting near the end of the sample).

We will need two distributions for each diffusion: the diffusion itself $p(\xi_t|\xi_s, t-s)$, and the bridging distribution $p(\xi_t|\xi_s, \xi_u, t-s, u-t)$ for sampling a point ξ_t between existing points ξ_s and ξ_u.

Continuous Variables

For continuous variables, we use Brownian motion:

$$\xi_t|\xi_s, t-s \sim N(\xi_s, \sigma^2(t-s))$$

If there are multiple continuous variables that may be correlated, they can share a multivariate Brownian motion with covariance matrix Σ.

The bridging distribution is given by

$$p(\xi_t|\xi_s, \xi_u, t-s, u-t) = N\left(\xi_t; \xi_s + \tfrac{t-s}{u-s}(\xi_u - \xi_s), \sigma^2 \tfrac{(u-t)(t-s)}{u-s}\right)$$

A 1-dimensional dataset sampled with Brownian diffusion is shown in Figure 6.3.

Discrete Variables

For discrete variables we use pure jump-type Markov processes (Kallenberg 1997); more specifically, we use what Kallenberg calls pseudo-Poisson processes. Such a process is

represented as the composition of a discrete-time Markov chain and a Poisson point process. That is, at random times governed by the Poisson point process, the diffusion jumps according to the kernel $K(\xi_s, \xi_t)$, i.e., $\xi_t \sim K(\xi_s, \cdot)$. Then, we have diffusion probabilities given by

$$p(\xi_t|\xi_s, t-s) = \sum_{n=0}^{\infty} e^{-\lambda(t-s)} \frac{(\lambda(t-s))^n}{n!} K^n(\xi_s, \xi_t).$$

For variables from a finite sample space $X = \{1, 2, \ldots, k\}$, we follow Kemp *et al.* 2004 and use a uniform kernel, $K(\xi_s, \xi_t) = 1/k$. Then,

$$K^n(\xi_s, \xi_t) = \begin{cases} 1/k & n > 0 \\ \mathbb{1}(\xi_t = \xi_s) & \text{otherwise,} \end{cases}$$

and

$$p(\xi_t|\xi_s, t-s) = \frac{1}{k}\left((k\mathbb{1}(\xi_t = \xi_s) - 1)\, e^{-\lambda(t-s)} + 1\right),$$

and

$$p(\xi_t|\xi_s, \xi_u, t-s, u-t) = \frac{\left((k\mathbb{1}(\xi_u = \xi_t) - 1)\, e^{-\lambda(u-t)} + 1\right)\left((k\mathbb{1}(\xi_t = \xi_s) - 1)\, e^{-\lambda(t-s)} + 1\right)}{k\left((k\mathbb{1}(\xi_u = \xi_s) - 1)\, e^{-\lambda(u-s)} + 1\right)}$$

6.3.3 Hyperpriors

Finally, we must define the hyperprior $p(\theta)$. We recommend the following: first, we have the parameters α and β, where $\beta \in (0, 1)$, and $\alpha \in (-2\beta, \infty)$. A fairly vague prior for this is

$$\beta \sim Beta(1, 1) \quad \delta \sim G(20, \frac{1}{20}) \quad \alpha = \delta - 2\beta,$$

where $Beta(\cdot, \cdot)$ is the Beta distribution, and $G(\cdot, \cdot)$ is the Gamma distribution. Following (Knowles and Ghahramani 2010), we choose $a(t) = \frac{c}{1-t}$ with $c \sim G(4, \frac{1}{2})$. We then choose

$$\rho(\mathbb{T}_{i-1}^{(t,k)}) = \min\{1, t \tanh b((i - i^*) - |\mathbb{T}_{i-1}^{(t,k)}|)\},$$

with $b \sim G(2, \frac{1}{50})$, where i^* is the i corresponding to the newest leaf of $\mathbb{T}_{i-1}^{(t,k)}$, so that $i - i^*$ is the "age" of the newest leaf. Using this choice of $\rho(\cdot)$, we have increased probability of deleting small trees, trees that diverge closer to the leaves, and trees that have not been updated recently.

To complete the model, we define the hyperpriors for the diffusion parameter ψ. For the Brownian diffusion (in one dimension), we have $\psi = \sigma^2$, $\frac{1}{\sigma^2} \sim G(\frac{3}{2}, 1)$. For finite discrete variables, we have $\psi = \lambda$, $\lambda \sim G(2, \frac{1}{2})$.

6.4 Inference

We are interested in performing classification; that is, each data point is of the form $y_i = (v_i, l_i)$, where l_i is a discrete label. We want to compute $p(l_i|v_i, y_{1:i-1}) = \int p(l_i|\mathbb{T}_i)p(\mathbb{T}_i|v_i, y_{1:i-1})d\mathbb{T}_i$, which—because of the high-dimensional, mixed discrete and continuous nature of $\mathbb{T}_{i-1}$—is highly intractable. We will therefore resort to Monte Carlo estimates, which estimate the integral by

$$p(l_i|v_i, y_{1:i-1}) \approx \frac{1}{K}\sum_k p(l_i|\mathbb{T}_i^{(k)})$$

where the $\mathbb{T}_i^{(k)}$ are i.i.d. samples from $p(\mathbb{T}_i|v_i, y_{1:i-1})$. Generating exact samples from this distribution is also intractable, so we will rely on the Sequential Monte Carlo methods from Section 4.2.

To complete the specification of our inference algorithm, we must specify the proposal distribution for $a_i^{(k)}$. Liu and West use

$$g(y_i|x_{i-1}^{(k)}, \vec{m}_{i-1}^{(k)}) = p(y_i|\mathbb{E}[x_i|x_{i-1}^{(k)}, \theta_{i-1}^{(k)}], \vec{m}_{i-1}^{(k)})$$

but this expectation is not analytically available or even necessarily defined in general. Indeed, in our model, the x_i are not real numbers but trees, and thus do not have expectation values. Therefore, we use

$$\widehat{x}_i^{(m,k)} \overset{iid}{\sim} p(\cdot|x_{i-1}^{(k)}, \theta_{i-1}^{(k)})$$
$$g(y_i|x_{i-1}^{(k)}, \vec{m}_{i-1}^{(k)}) = \frac{1}{M}\sum_{m=1}^{M} p(y_i|\widehat{x}_i^{(m,k)}, \vec{m}_{i-1}^{(k)})$$

6.4.1 Inference in Our Model

Specializing to our model, we note that $(x_{1:i}, y_{1:i}) = \mathbb{T}_{1:i}$, where each y_i is the value of newest leaf in $\mathbb{T}_i$, and x_i corresponds to the rest of the tree, including the leaf with value y_i, but excluding its value. In a slight abuse of notation, we will use $\mathbb{T}_i$ in place of x_i, with the

Algorithm 6.1: SMC for our model

$\theta_1^{(k)} \overset{iid}{\sim} p(\theta)$
$\mathbb{T}_1^{(k)} \leftarrow$ The (deterministic) tree with one leaf at y_1
$w_1^{(k)} \propto 1$
foreach $i > 1$ **do**
    /* sample $a_i^{(k)}$ */
    $\widehat{\mathbb{T}}_i^{(m,k)} \overset{iid}{\sim} p(\cdot|\mathbb{T}_{i-1}^{(k)}, \theta_{i-1}^{(k)})$
    $g(v_i|\mathbb{T}_{i-1}^{(k)}, \theta_{i-1}^{(k)}) \leftarrow \frac{1}{M}\sum_{m=1}^{M} p(v_i|\widehat{\mathbb{T}}_i^{(m,k)}, \vec{m}_{i-1}^{(k)})$
    $a_i^{(k)} \overset{iid}{\sim} \sum_{j=1}^{K} w_{i-1}^{(j)} g(v_i|\mathbb{T}_{i-1}^{(j)}, \theta_{i-1}^{(j)})\delta_j$
    /* Sample θ and $\mathbb{T}_i$ */
    $\theta_i^{(k)} \sim N(\vec{m}_{i-1}^{(k)}, h^2\vec{V}_{i-1})$
    $\mathbb{T}_i^{(k)} \sim q(\cdot|\mathbb{T}_{i-1}^{(a_i^{(k)})}, \theta_i^{(k)}, y_i)$
    /* Calculate new weights, classifier */
    $w_{v,i}^{(k)} \propto \frac{p(v_i|\mathbb{T}_i^{(k)}, \theta_i^{(k)}) p(\mathbb{T}_i^{(k)}|\mathbb{T}_{i-1}^{(a_i^{(k)})}, \theta_i^{(k)})}{g(v_i|\mathbb{T}_{i-1}^{(a_i^{(k)})}, \theta_{i-1}^{(a_i^{(k)})}) q(\mathbb{T}_i^{(k)}|\mathbb{T}_{i-1}^{(a_i^{(k)})}, \theta_i^{(k)}, v_i)}$
    $\widehat{p}(l_i|v_i, y_{1:i-1}) \leftarrow \sum_{j=1}^{K} w_{v,i}^{(k)} p(l_i|\mathbb{T}_i^{(j)}, \theta_i^{(j)})$
    $w_i^{(k)} \propto w_{v,i}^{(k)} p(l_i|\mathbb{T}_i^{(k)}, \theta_i^{(k)})$

understanding that the newest leaf has a fixed value. Then, we have

$$p(y_i|\mathbb{T}_i, \theta) = p(\xi_1^*|\xi_{s^*}^*, \theta, 1 - s^*),$$

where $\xi_1^* = y_i$ is the value of the newest leaf, and $(\xi_{s^*}^*, s^*)$ is the value and time of its parent node, so that the probability of y_i is just the diffusion probability from its attachment to the tree. To adapt the algorithm to the task of classification, we note that

$$p(y_i|\mathbb{T}_i^{(k)}, \theta_i^{(k)}) = p(l_i|\mathbb{T}_i^{(k)}, \theta_i^{(k)}) p(v_i|\mathbb{T}_i^{(k)}, \theta_i^{(k)}),$$

(and likewise for $g(y_i|\mathbb{T}_i^{(a_i^{(k)})}, \theta_i^{(a_i^{(k)})})$), so we can compute $\widetilde{w}_{1:i}^{(k)}$ and $g(y_i|\mathbb{T}_i^{(a_i^{(k)})})$ using only v_i, yielding samples from $p(\mathbb{T}_i|v_i, y_{1:i-1})$, as desired. We then update the weights by multiplying by $p(l_i|\mathbb{T}_i^{(k)}, \theta_i^{(k)})$ and renormalizing before propagating the particles. The whole inference process is given in Algorithm 6.1, and depicted graphically in Figure 6.4.

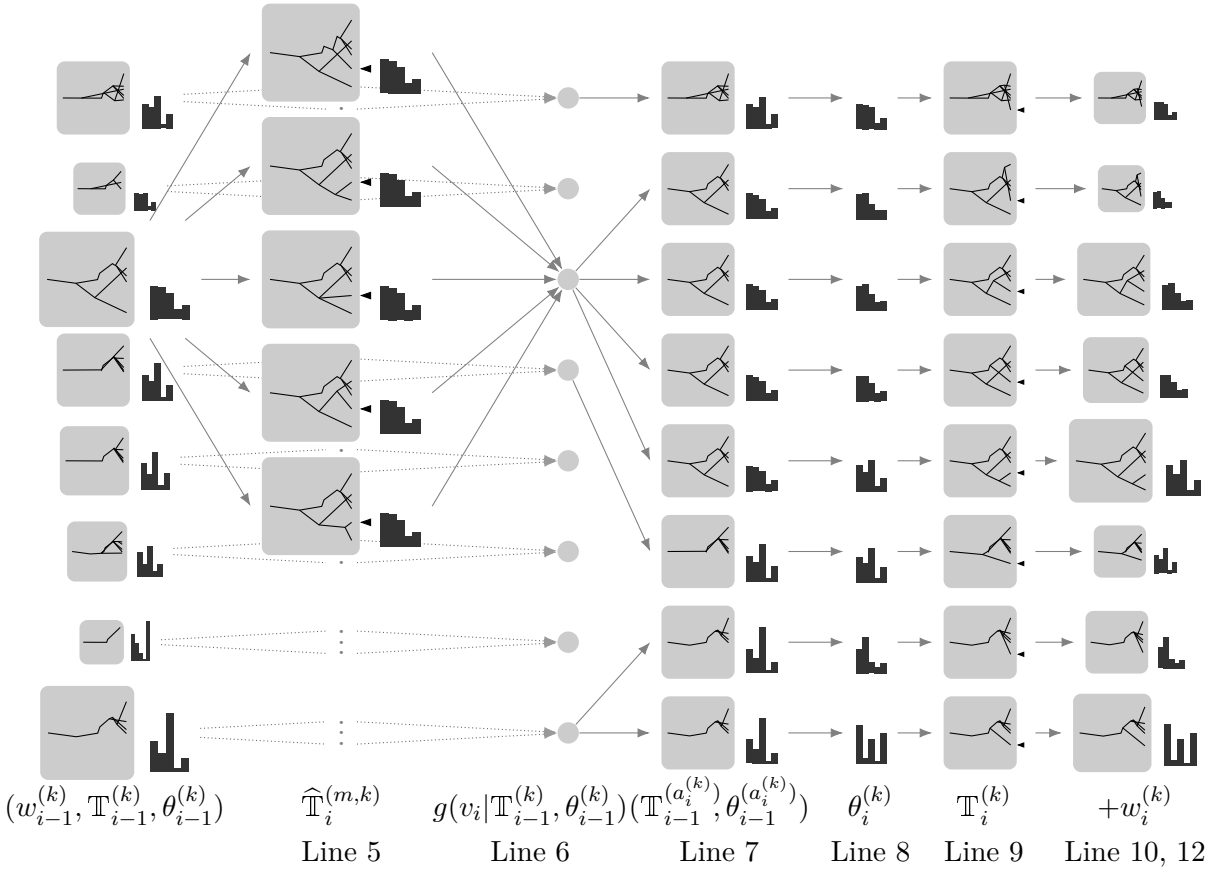


Figure 6.4: The inner loop of our Sequential Monte Carlo algorithm.[3] In the first and last columns, the size of each element is proportional to its weight. The values of the components of θ are represented as bar graphs. Small arrowheads mark new leaves.

For the proposal $q(\cdot|\mathbb{T}_{i-1}^{(a_i^{(k)})}, \theta_i^{(k)}, y_i)$, we can simply choose $p(\cdot|\mathbb{T}_{i-1}^{(a_i^{(k)})}, \theta_i^{(k)})$, given in Section 6.3.1, which is easy to draw from and cancels out of the expression for $\widetilde{w}_{1:i}^{(k)}$, but lacks dependence on y_i. To sample from this proposal, we must generate divergence times t according to the distribution in Equation (6.1). To do this, sample $U \sim Uniform[0,1]$, then take

$$t = A^{-1}\left(A(s) - \frac{\Gamma(m+1+\alpha)}{\Gamma(m-\beta)}\log(1-U)\right)$$

with the appropriate m.

[3]Figure format heavily inspired by Figure 1 in (Dinh, Darling, and Matsen IV 2017)

For Algorithm 6.1 to be applicable to streaming data, the amount of work per data point should be approximately constant. Thus, sampling $\widehat{\mathbb{T}}_i^{(m,k)}$ and $\mathbb{T}_i^{(k)}$, and calculating $p(\mathbb{T}_i^{(k)}|\mathbb{T}_{i-1}^{(a_i^{(k)})}, \theta_{i-1}^{(a_i^{(k)})})$ and $q(\mathbb{T}_i^{(k)}|\mathbb{T}_{i-1}^{(a_i^{(k)})}, \theta_{i-1}^{(a_i^{(k)})}, y_i)$ should require a constant amount of work. Each of these computations involve traversing $\mathbb{T}_{i-1}^{(k)}$ or $\mathbb{T}_i^{(k)}$, so a bound on the size of $\mathbb{T}_i$ would also bound the work-per-data-point.

In the worst case, $|\mathbb{T}_i| = |\mathbb{T}_{i-1}| + 1$. This happens either because the new leaf diverges before any removal can happen, or because no branches were removed at branch points the new leaf traversed. In the first case, we incur a multiplicative penalty of $a(t)\Gamma(m-\beta)/\Gamma(m+1+\alpha)$, where m is the total number of leaves in the tree. As the tree grows, m grows, and t must get smaller to remain before all other branch points. $\Gamma(m-\beta)/\Gamma(m+1+\alpha)$ falls off rapidly as shown in Figure 6.5:

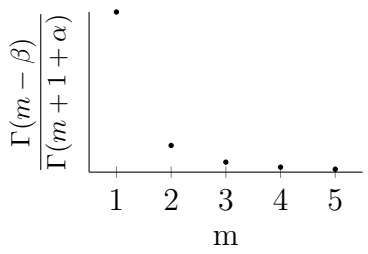


Figure 6.5: $\Gamma(m-\beta)/\Gamma(m+1+\alpha)$, with $\beta = \frac{1}{2}$ and $\alpha = 1$

So, as long as $a(t)$ does not exponentially prefer early divergence, the probability of early divergence falls rapidly as the tree grows. In the second case, every branch retained incurs a multiplicative penalty of the form $(1-\rho(\mathbb{T})) \leq 1$, and the number of branches encountered increases as the tree grows. So long as $\rho(\mathbb{T})$ is nonvanishing, this also rapidly penalizes tree growth. Thus, with high probability, $\mathbb{T}_i$ will quickly stop growing, and the work required at each data point is bounded.

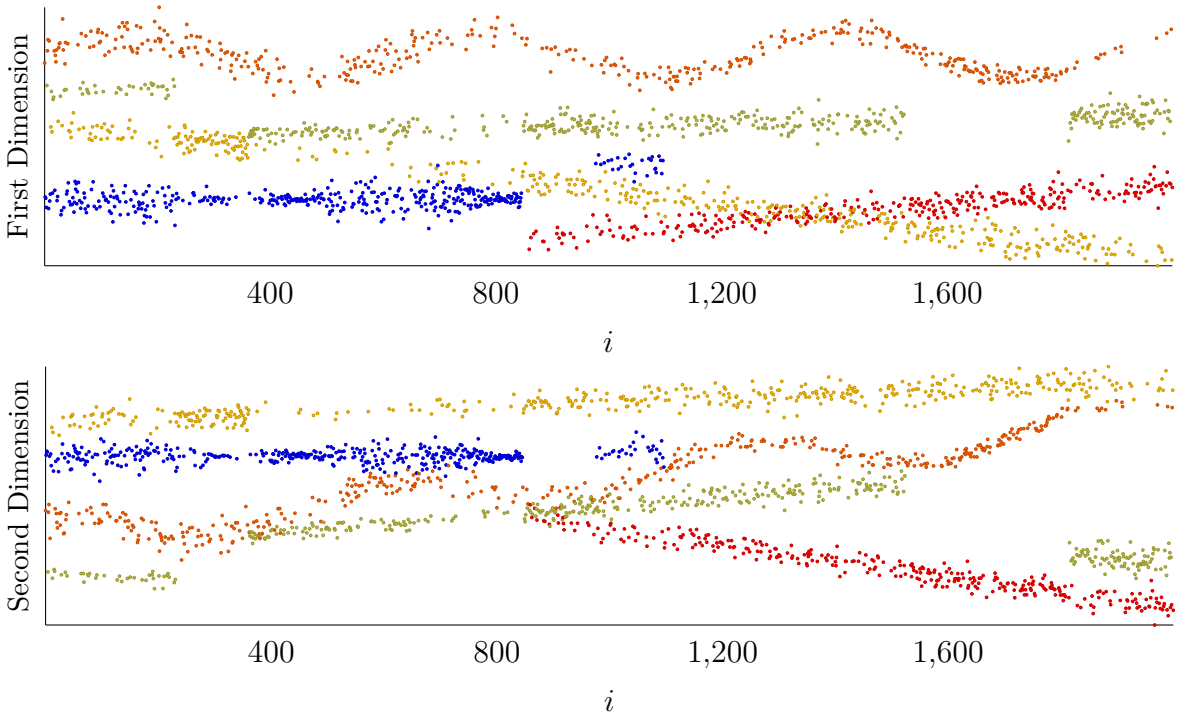


Figure 6.6: The synthetic dataset

6.5 Experimental Evaluation

We experimentally validate our model and inference methods on a two-dimensional synthetic dataset intended to showcase the nonstationarity of the DPYDT. The dataset is shown in Figure 6.6. The dataset contains both gradual and sudden changes, including classes that disappear and reappear at various times.

Figure 6.7 shows the negative log predictive probability (NLPP), $-\log \widehat{p}(l_i|v_i, y_{1:i-1})$. A value of 0 indicates that the model assigns probability 1 to the correct label. Figure 6.7 shows that the NLPP is very close to 0 most of the time, with higher spikes mostly found at points where the classes get close or overlap in at least one dimension (e.g., the higher spikes just after $i = 800$, where there is a lot of overlap in the first dimension, as shown in Figure 6.6), or when a class (re)appears (e.g., at around $i = 1700$). Finally, Figure 6.8 provides evidence supporting our argument that $|\mathbb{T}_i|$ is bounded with high probability: after a steep climb in the beginning, the average tree size does not deviate far from 100.

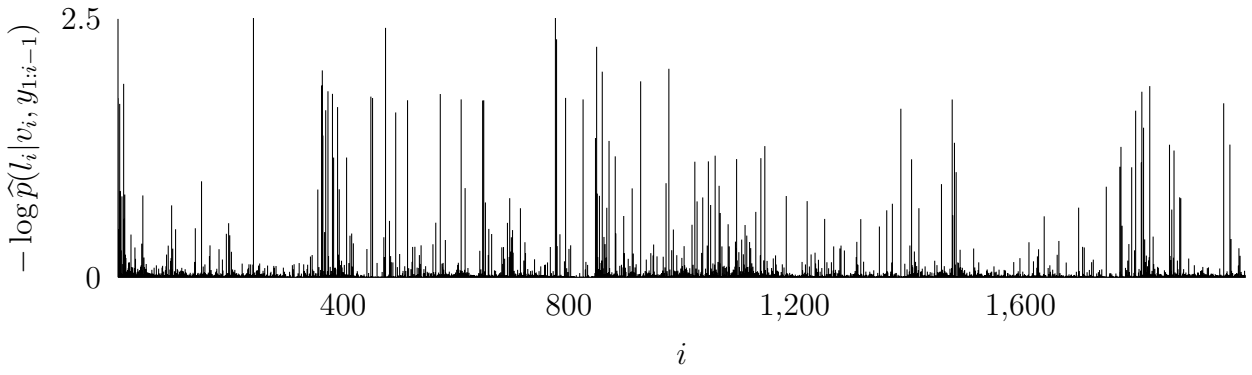


Figure 6.7: Negative Log Predictive Probability over time

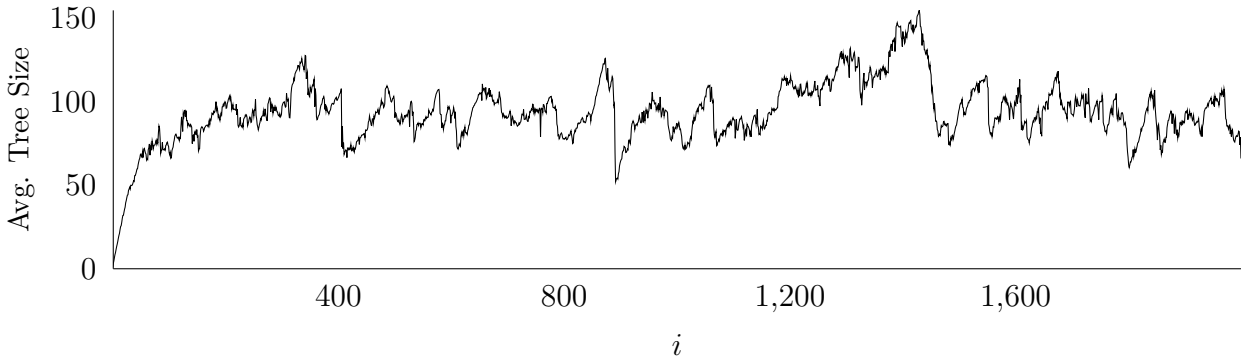


Figure 6.8: The average tree size over time, $\frac{1}{K}\sum_k |\mathbb{T}_i^{(k)}|$

6.6 Future Work

Some directions for future work include:

- Other diffusions. Examples: continuous value diffusions with fatter tails; diffusions over structured spaces such as random-walk diffusions on graphs; diffusions over non-numeric spaces

- Other divergence rate and deletion functions

- Alternative inference methods such as adapting the variational inference methods from (Knowles, Van Gael, and Ghahramani 2011; Knowles 2012)

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Numbers SBE-SMA-1539302 and CNS-1229652, and the Air Force Office of Scientific Research under Award No. FA-9550-12-1-0077.

CHAPTER 7

THE ANCHORED DPYDT FOR MODELING STREAMS WITH NOVEL AND RECURRENT CLASSES

7.1 Introduction

As in the previous chapter, we are interested in the task of online classification. In many cases, the set of possible classes may vary over time. Previously-unseen, or *novel* classes may appear long after the start of the stream. Classes may disappear for long stretches, only to reappear as a *recurrent* class.

These possibilities present a number of challenges. First, we must distinguish a class that has not recently been seen (whether novel or recurrent) from the classes that are currently "active" in the data. Second, we must determine whether the differing class is one that has ever been seen before. Additionally, all of this takes place in a setting of nonstationary class distributions, so that old data may be unreliable.

In this chapter, we seek to modify the DPYDT of the previous chapter to handle these new challenges. To handle novel classes, we introduce the idea of time-nonstationary diffusions and develop a diffusion based on the CRP. To handle recurrent classes, we first introduce the idea of *anchors*: subtrees that are removed from the tree, but which leave behind a pseudo-node that remembers the size and divergence point of the removed tree. We also modify the CRP-based diffusion to allow recurrence.

The remainder of this chapter is organized as follows. In Section 7.2, we introduce anchoring and derive a new density expression. In Section 7.3, we develop our new diffusions that handle novel and recurrent values. In Section 7.4, we discuss the hyperpriors on the various parameters of our model. In Section 7.5, we discuss inference in our model. In Section 7.6, we prove that the size of the tree cannot grow without bound. In Section 7.7, we present experimental evaluation of our model. Finally, in Section 7.8, we conclude our paper and discuss future work.

7.2 Anchoring the DPYDT

The problem of recurrent classes requires some form of longer-term memory mechanism. When a class disappears, the leaves labeled with that class will eventually be deleted from the tree. When that class later becomes recurrent, under the DPYDT, the new members of that class will be completely independent of the now-deleted previous members. However, we would like to remember where the previous clusters associated with the recurrent class were, so that the new members may be similarly distributed.

To address this problem, we introduce the concept of *anchoring*: when traversing the tree to add a new leaf, in addition to the existing deletion mechanism, we add the ability to remove a subtree but retain its size and the location of its root. Subsequent traversals have a small probability—controlled by the new parameter γ—of turning towards an anchor instead of turning towards an existing branch or creating a new one. Each anchor, denoted $\mathbb{A}_{i-1}^{(t,j)}$, where j varies only over the anchors attached at t has a probability of being chosen proportional to $(i-l)|\mathbb{A}_{i-1}^{(t,j)}|$, where l is the iteration when $\mathbb{A}_{i-1}^{(t,j)}$ was created, i.e., $(i-l)$ is the age of the anchor. As with existing branches, there are two possibilities:

- **Partial Resurrection:** the new data point diverges before the anchor point is reached, in which case the anchor remains, attached to the new divergence point; and

- **Full Resurrection:** the new data point does not diverge, in which case the anchor becomes a regular internal node, except that the associated count $m^{(t)}$ remains higher.

These new mechanisms are illustrated in Figure 7.1.

7.2.1 Probability Densities

As in Section 6.3.1, we wish to derive an expression for $p(\mathbb{T}_i|\mathbb{T}_{i-1},\theta)$. To achieve a succinct, readable result, we require some additional notation:

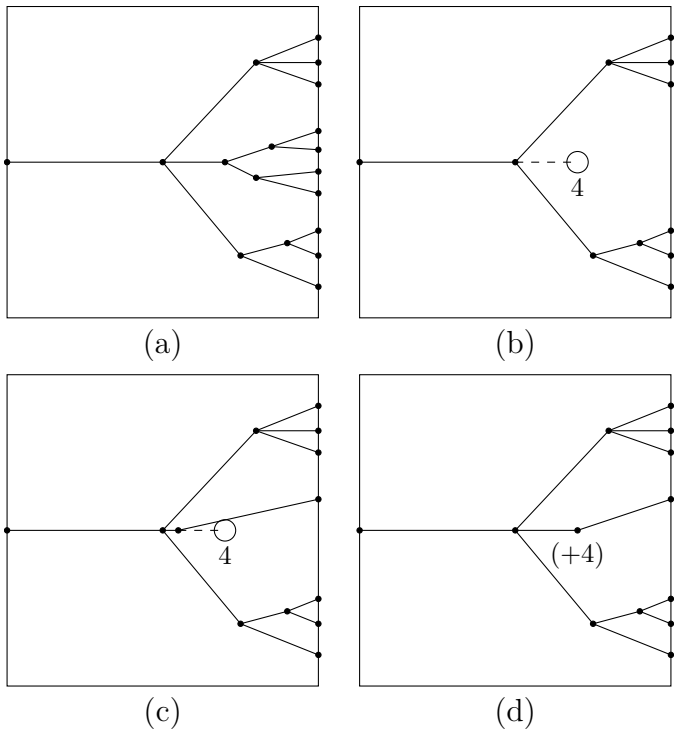


Figure 7.1: Anchoring. (a) Before any anchoring. (b) The middle subtree is anchored, retaining only its size and the location of its root. (c) A new node diverges between choosing the anchor and reaching it; this is called "partial resurrection." (d) A new node reaches the anchor; this is called "full resurrection." Diffusion is not shown.

- In a slight abuse of notation, let t identify a unique point on the tree, with the understanding that the application $a(t)$ applies $a(\cdot)$ only to the time component of t.

- Let ξ_t denote the diffusion value at t.

- Let $\mathcal{P}_t$ denote the path from the root to the point t, represented as a collection of intervals $[s, t]$ whose endpoints are divergence or branch points.

- Let $\varepsilon_{[s,t]} = \exp\left[(A(s) - A(t))\frac{\Gamma(m_{i-1}^{(t)}-\beta)}{\Gamma(m_{i-1}^{(t)}+1+\alpha)}\right]$ be the probability of not diverging in $[s, t]$.

- Let

$$R_t = \prod_{\mathbb{T}_{i-1}^{(t,k)}} \begin{cases} \left(1-\rho(\mathbb{T}_{i-1}^{(t,k)})\right) & \mathbb{T}_{i-1}^{(t,k)} \text{ is not deleted} \\ \times \left(1-\tau(\mathbb{T}_{i-1}^{(t)}, \mathbb{T}_{i-1}^{(t,k)})\right) & \text{or anchored} \\ \left(1-\rho(\mathbb{T}_{i-1}^{(t,k)})\right) & \mathbb{T}_{i-1}^{(t,k)} \text{ is anchored} \\ \times\tau(\mathbb{T}_{i-1}^{(t)}, \mathbb{T}_{i-1}^{(t,k)}) & \\ \rho(\mathbb{T}_{i-1}^{(t,k)}) & \mathbb{T}_{i-1}^{(t,k)} \text{ is deleted} \end{cases}$$
$$\times \prod_{\mathbb{A}_{i-1}^{(t,j)}} \begin{cases} \left(1-\rho(\mathbb{A}_{i-1}^{(t,j)})\right) & \mathbb{A}_{i-1}^{(t,j)} \text{ is not deleted} \\ \rho(\mathbb{A}_{i-1}^{(t,j)}) & \mathbb{A}_{i-1}^{(t,j)} \text{ is deleted} \end{cases}$$

  be the joint probability of all deletion/anchoring decisions made at the branch point t.

- Let $z_t = \sum_j (i - l(\mathbb{A}_{i-1}^{(t,j)}))|\mathbb{A}_{i-1}^{(t,j)}|$ be the normalizing constant for (partial) resurrection at t

- Let $[s_0, t_0]$ denote the final interval traversed by the new leaf before it diverges or creates a new branch

- Let $D = p(\xi_1|\xi_{t_0}, \psi, 1-t_0)$ be the probability of diffusion from the point where the new leaf diverges or creates a new branch

Using this notation, the predictive distribution is given by:

$$p(\mathbb{T}_i|\mathbb{T}_{i-1}, \theta) = DR_0 \left(\prod_{[s,t]\in\mathcal{P}_{s_0}} \varepsilon_{[s,t]} R_t \frac{b_{i-1}^{(t,k)} - \beta}{m_{i-1}^{(t)} + \gamma + \alpha} \right) \varepsilon_{[s_0,t_0]}$$
$$\times \begin{cases} \frac{\gamma(i-l(\mathbb{A}_{i-1}^{(s_0,j)}))|\mathbb{A}_{i-1}^{(s_0,j)}|}{z_{s_0}\left(m_{i-1}^{(s_0)}+\gamma+\alpha\right)} & \mathbb{A}_{i-1}^{(s_0,j)} \text{ is fully resurrected} \\ \frac{\gamma(i-l(\mathbb{A}_{i-1}^{(s_0,j)}))|\mathbb{A}_{i-1}^{(s_0,j)}|}{z_{s_0}\left(m_{i-1}^{(s_0)}+\gamma+\alpha\right)} a(t_0) \frac{\Gamma(m_{i-1}^{(t_0)}-\beta)}{\Gamma(m_{i-1}^{(t_0)}+1+\alpha)} & \mathbb{A}_{i-1}^{(s_0,j)} \text{ is partially resurrected} \\ R_{t_0} \frac{\alpha+\beta\widetilde{K}_{i-1}^{(t_0)}}{m_{i-1}^{(t_0)}+\gamma+\alpha} & \text{branch} \\ a(t_0) \frac{\Gamma(m_{i-1}^{(t_0)}-\beta)}{\Gamma(m_{i-1}^{(t_0)}+1+\alpha)} & \text{diverge} \end{cases} \tag{7.1}$$

7.3 Nonstationary Diffusions

In addition to the diffusions discussed in Section 6.3.2, we introduce here diffusions that are adapted to handling novel and recurrent classes. First we note that the diffusion process need not remain stationary between data points; i.e., the distribution used to generate one data point could change over time, depending on arbitrary properties of the tree being modified. In particular, we present diffusions that depend on the class counts of current leaves.

As in Section 6.3.2, we will use pseudo-Poisson processes as our diffusions over discrete spaces. However, in place of the time-stationary kernel $K(\xi_s, \xi_t)$, we will have a varying kernel $K_i(\xi_s, \xi_t)$.

Unbounded Discrete Variables: For variables from a discrete space without known bound (in particular, for class labels), we introduce a diffusion based on the CRP, so that novel classes may be discovered:

$$K_i(\xi_s, \xi_t) = \begin{cases} \frac{n_k}{i-1+\eta} & \xi_t = k \leq M_i \\ \frac{\eta}{i-1+\eta} & \xi_t = M_i + 1, \end{cases}$$

where n_k is the number of leaves in the tree with value k, and M_i is the number of different values at the leaves. This diffusion has a fundamental flaw, however: if every leaf with value k is removed from the tree, then M_i will decrease, reusing an existing value inappropriately. If M_i is simply changed to be the number of distinct values *ever* taken, we still assign probability 0 of ever assigning the value k to a leaf.

Unbounded Discrete Variables with Recurrence: To address this, we artificially inflate the count associated with each k so that even if $n_k = 0$ there is positive probability of assigning the value k to a leaf:

$$K_i(\xi_s, \xi_t) = \begin{cases} \frac{n_k+\zeta}{i-1+\eta+M_i\zeta} & \xi_t = k \leq M_i \\ \frac{\eta}{i-1+\eta} & \xi_t = M_i + 1, \end{cases}$$

where M_i is the number of distinct values ever assigned.

7.4 Hyperpriors

Lastly, we give a hyperprior for the parameter vector θ. First, we have the parameters $\beta \in (0, 1)$ and $\alpha \in (-2\beta, \infty)$. We use the vague prior

$$\begin{aligned}\beta &\sim Beta(1, 1)\\ \delta &\sim G(20, \frac{1}{4})\\ \alpha &= \delta - 2\beta,\end{aligned}$$

where $Beta(\cdot, \cdot)$ is the Beta distribution, and $G(\cdot, \cdot)$ is the Gamma distribution. We choose $a(t) = \frac{c}{(1-t)^2}$ with $c \sim G(4, \frac{1}{2})$. Previous works (Knowles and Ghahramani 2010) and the previous chapter used $a(t) = \frac{c}{1-t}$, but squaring the denominator favors slightly earlier divergence tendencies, leading to more samples with multiple discrete clusters. As in the previous chapter, we choose

$$\rho(\mathbb{T}_{i-1}^{(t,k)}) = \min\{1, t^* \tanh b((i - i^*) - |\mathbb{T}_{i-1}^{(t,k)}|)\},$$

where $b \sim G(2, \frac{1}{50})$, t^* is the t at the root of $\mathbb{T}_{i-1}^{(t,k)}$, and $i - i^*$ is the "staleness" of $\mathbb{T}_{i-1}^{(t,k)}$: the number of iterations since the newest leaf was added (i^* is the iteration where that leaf was added). Choosing this $\rho(\cdot)$ gives higher probability of deleting smaller subtrees, subtrees whose root is closer to the leaves, and subtrees that have not been updated recently. The use of tanh gives $\rho(\cdot)$ a sigmoidal curve whose steepness is controlled by b. Next, we choose

$$\tau(\mathbb{T}_{i-1}^{(t)}, \mathbb{T}_{i-1}^{(t,k)}) = \min\left\{1, \frac{(1 - t^*)^{\frac{|\mathbb{T}_{i-1}^{(t,k)}|}{|\mathbb{T}_{i-1}^{(t)}|}}}{1 + \exp\left[-b'\left((i - i^*) - (|\mathbb{T}_{i-1}^{(t)}| + |\xi_t - \xi_{t,k}^*|)\right)\right]}\right\},$$

which assigns higher probability to anchoring

- subtrees rooted closer to the root of the overall tree, as we want to anchor discrete clusters and not individual leaves

- subtrees that are medium-sized: larger trees should be retained longer, and smaller trees should just be deleted

- subtrees that are more stale, as in $\rho(\cdot)$

- subtrees whose root value, $\xi^*_{t,k}$, is further from its parent node's value, ξ_t, as this also favors discrete clusters

Finally, we complete the model by specifying hyperpriors for the diffusion parameter ψ. For Brownian diffusion and finite discrete variables, we use the same hyperpriors as in the previous chapter (see Section 6.3.3), and for unbounded discrete variables, $\psi = (\lambda, \alpha)$ with $\lambda \sim G(2, \frac{1}{2})$ again and $\alpha \sim G(2, \frac{1}{2})$ as well.

7.5 Inference

We use the same basic inference algorithm of Section 6.4, except we change the proposal $q(\cdot|\mathbb{T}^{(a_i^{(k)})}_{i-1}, \theta_i^{(k)}, v_i)$. In designing such a proposal, we must balance the desire to closely match the prior Equation (7.1) with the desire to depend on v_i, as any deviation from the prior will increase the computational complexity of the acceptance ratio, but independence from v_i will lead to higher variance in weights and thus a skewed estimator. Towards this end, we mostly follow the prior Equation (7.1), except when choosing among branches or anchors to follow, we use

$$p(k) \propto \begin{cases} \frac{b_{i-1}^{(t,k)}}{|\xi^*_{t,k} - v_i|} & \text{branch } k \\ \frac{\gamma |\mathbb{A}_l^{(s_0,j)}}{|\xi^*_{t,k} - v_i|} & \text{anchor } k \end{cases}$$

instead of

$$p(k) \propto \begin{cases} b_{i-1}^{(t,k)} - \beta & \text{branch } k \\ \frac{\gamma(i-l)|\mathbb{A}_l^{(s_0,j)}|}{z_{s_0}} & \text{anchor } k \end{cases},$$

where $\xi^*_{t,k}$ is the feature vector of the next node along branch k. That is, we bias towards choosing branches or anchors that are closer to v_i while still prefering more popular routes.

7.6 Bounding the Size of the Tree

To efficiently perform inference in this model, the amount of work per data point must be approximately constant. To show this is the case, we bound the size of the tree $|\mathbb{T}_i|$: if the tree does not grow unbounded, neither can the work-per-datapoint. Let $\|\mathbb{T}_i\|_L$ denote the number of leaves (not counting anchors) of $\mathbb{T}_i$, and note that $|\mathbb{T}_i| \leq \|\mathbb{T}_i\|_L^2 - 1$.

Theorem 7.1. $|\mathbb{T}_i|$ *does not grow without bound.*

Proof. Let us represent $\mathbb{T}_i$ as a collection of triples $([s,t], m_i^{(t)}, \widetilde{K}_i^{(t)})$. Then, if $\|\mathbb{T}_{i+1}\|_L > \|\mathbb{T}_i\|_L$, then no deletion or anchoring events have happened, a triple $([u,1],1,0)$ is added, and some subcollection of triples will change as follows:

- Any intervals that are only traversed or wherein a full resurrection event occurs will have $m_{i+1}^{(t)} = m_i^{(t)} + 1$.
- Any intervals wherein a diverge event or partial resurrection event occurs will be split into smaller intervals: $[s,t] \mapsto [s,u],[u,t]$ for some $u \in (s,t)$, with $m_{i+1}^{(u)} = m_i^{(t)} + 1$.
- Any intervals wherein a branch event occurs will have $\widetilde{K}_{i+1}^{(t)} = \widetilde{K}_i^{(t)} + 1$.

Then, note that for fixed $s < 1$ and $m^{(t)}$,

$$\lim_{t-s\to 0} \exp\left[(A(s) - A(t))\frac{\Gamma(m^{(t)} - \beta)}{\Gamma(m^{(t)} + 1 + \alpha)}\right] = 1,$$

and for fixed $s < t < 1$,

$$\lim_{m^{(t)}\to\infty} \exp\left[(A(s) - A(t))\frac{\Gamma(m^{(t)} - \beta)}{\Gamma(m^{(t)} + 1 + \alpha)}\right] = 1,$$

so that the probability of diverging in any interval $[s,t], t < 1$ goes to 0. Then, as $\mathbb{T}_i$ grows, the growth of the $m_i^{(t)}$ leads to later divergences, and the growth of the $\widetilde{K}_i^{(t)}$ leads to more branches. In either case, the number of subtrees rooted along the path $\mathcal{P}_{s_0}$ to the newest leaf

increases, adding more $(1 - \rho(\mathbb{T}_{i-1}^{(t,k)}))$ and $(1 - \tau(\mathbb{T}_{i-1}^{(t)}, \mathbb{T}_{i-1}^{(t,k)}))$ terms to the distribution of $\mathbb{T}_{i+1}$.

In other words, as $\mathbb{T}_i$ grows, the probability of not deleting any leaves gets smaller, unless $\rho(\cdot)$ shrinks faster. However, the subtrees rooted along $\mathcal{P}_{s_0}$ can be new, small trees corresponding to recent branch and diverge events or old, large trees that are frequently traversed. Thus, there is no one kind of tree that predominates, so that $\rho(\cdot)$, as a function only of the subtrees, cannot uniformly shrink. Similarly, $\tau(\cdot, \cdot)$ cannot uniformly shrink as $\mathbb{T}_i$ grows, so that the probability of deleting or anchoring grows as $\mathbb{T}_i$ grows. In other words, as $\mathbb{T}_i$ grows, the probability of growing decreases, eventually leading to a maximum size. □

7.7 Experimental Evaluation

We experimentally test our proposed model on three datasets. The first—to compare against the DPYDT without anchoring—is the same synthetic dataset from Chapter 6. The second is synthetic dataset with 50 dimensions which exhibits novel and recurrent classes with sharp boundaries that drift. The third is a real-world network intrusion dataset with 41 dimensions from the KDD-Cup 1999 competition (Stolfo et al. 1999), obtained from the UCI Machine Learning Repository (Dheeru and Karra Taniskidou 2017). Figure 7.2 shows the NLPP results of all three datasets. We see from the histograms in particular that the large majority of data points are predicted well (NLPP near 0, so predictive probability near 1). We can convert these numbers to typical classifier accuracies by declaring a "correct classification" when the model predicts the correct label with probability at least 0.5. Using this rule, we get 92.5%, 66.1%, and 75.9% accuracy on the three respective datasets.

7.8 Conclusion and Future Work

We have presented a novel Bayesian nonparametric method for handling nonstationary data with novel and recurrent classes, by adding an anchoring mechanism to the DPYDT and

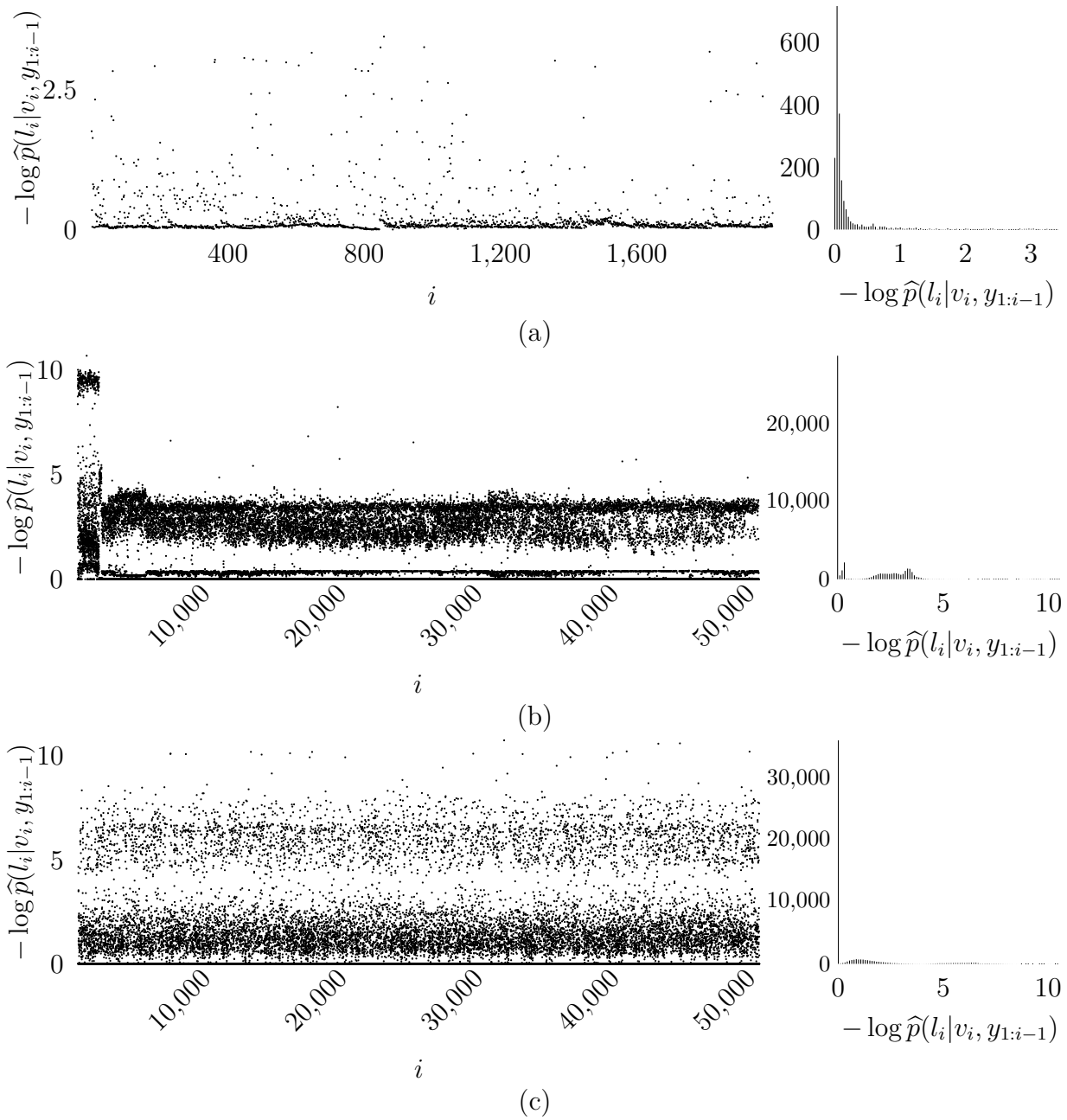


Figure 7.2: Experimental results: (a) The synthetic dataset from Chapter 6. (b) The new synthetic dataset. (c) The KDD-Cup 1999 dataset. *Left:* Negative Log Predictive Probability over time. *Right:* Histograms of Negative Log Predictive Probability.

considering diffusion distributions that vary over time. Experimental results demonstrate that the proposed method handles these challenges well. Possible subjects for future work include exploration of other divergence rate, deletion, and anchoring functions, as well as improved inference methods. Additionally, other applications such as regression could be explored.

CHAPTER 8

CONCLUSION AND FUTURE WORK

In this dissertation, we explored the field of Bayesian nonparametrics. We presented novel models for relational and streaming data, and developed inference algorithms for them. Finally, we showed that these models perform well in experimental evaluation.

There is still a lot of room for future work. The development of rich, flexible nonparametric priors should continue, as such techniques can form the foundation of powerful learning systems. To support this endeavor, there is still much work to be done on both the theoretical underpinnings of nonparametric methods and on powerful, efficient inference in such models. These two directions will support eachother as theoretical results enable new inference techniques and exploration of inference algorithms leads to the discovery of theoretical properties.

Additionally, these powerful techniques should see broader application, especially as the inference algorithms speed up. Again, such research will feed back insights and ideas, improving the theoretical and practical sides as well.

REFERENCES

Adámek, J., H. Herrlich, and G.E. Strecker. 1990. *Abstract and concrete categories: the joy of cats*. Pure and applied mathematics. Wiley. ISBN: 9780471609223. https://books.google.com/books?id=KwTvAAAAMAAJ.

Adams, Ryan Prescott, Hanna M. Wallach, and Zoubin Ghahramani. 2010. "Learning the Structure of Deep Sparse Graphical Models". In *International Conference on Artificial Intelligence and Statistics*.

Afshar, Hadi Mohasel, and Justin Domke. 2015. "Reflection, Refraction, and Hamiltonian Monte Carlo". In *Advances in Neural Information Processing Systems*, 3007–3015.

Agarwal, Arvind, and Hal Daumé III. 2010. "A geometric view of conjugate priors". *Machine Learning* 81, no. 1: 99–113. ISSN: 0885-6125. doi:10.1007/s10994-010-5203-x.

Aldous, David J. 1981. "Representations for partially exchangeable arrays of random variables". *Journal of Multivariate Analysis* 11, no. 4: 581–598. ISSN: 0047-259X. doi:10.1016/0047-259x(81)90099-3.

— . 1985. "Exchangeability and related topics". In *École d'Été de Probabilités de Saint-Flour XIII — 1983*, ed. by P.L. Hennequin, 1–198. Lecture Notes in Mathematics. Springer Berlin Heidelberg. ISBN: 9783540393160. doi:10.1007/bfb0099421.

Andrieu, C., N. De Freitas, and A. Doucet. 1999. "Sequential MCMC for Bayesian model selection". In *Proceedings of the IEEE Signal Processing Workshop on Higher-Order Statistics. SPW-HOS '99*. Institute of Electrical / Electronics Engineers (IEEE). ISBN: 0-7695-0140-0. doi:10.1109/host.1999.778709.

Andrieu, C., and A. Doucet. 1999. "Joint Bayesian model selection and estimation of noisy sinusoids via reversible jump MCMC". *IEEE Transactions on Signal Processing* 47: 2667–2676. ISSN: 1053587X. doi:10.1109/78.790649. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=790649.

Andrieu, Christophe, Nando De Freitas, Arnaud Doucet, and Michael I. Jordan. 2003. "An introduction to MCMC for machine learning". *Machine learning* 50: 5–43.

Andrieu, Christophe, Arnaud Doucet, and Roman Holenstein. 2010. "Particle Markov chain Monte Carlo methods". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 72, no. 3: 269–342. ISSN: 13697412. doi:10.1111/j.1467-9868.2009.00736.x.

Andrieu, Christophe, J.F.G. de Freitas, and Arnaud Doucet. 1999. *Robust Full Bayesian Learning for Neural Networks*. Tech. rep. Cambridge University.

Andrieu, Christophe, Nando de Freitas, and Arnaud Doucet. 2000. "Reversible Jump MCMC Simulated Annealing for Neural Networks". In *Uncertainty in Artificial Intelligence*, 11–18.

Angelino, Elaine, Matthew James Johnson, and Ryan P. Adams. 2016. "Patterns of Scalable Bayesian Inference". *Foundations and Trends® in Machine Learning* 9: 119–247. ISSN: 1935-8245. doi:10.1561/2200000052.

Angelino, Elaine, Eddie Kohler, Amos Waterland, Margo Seltzer, and Ryan P. Adams. 2014. "Accelerating MCMC via parallel predictive prefetching". In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, 22–31. AUAI Press. arXiv: 1403.7265.

Antoniak, Charles E. 1974. "Mixtures of Dirichlet Processes with Applications to Bayesian Nonparametric Problems". *The Annals of Statistics* 2: 1152–1174.

Applebaum, David. 2004. *Lévy Processes and Stochastic Calculus*. Cambridge university press.

Armagan, Artin, and David Dunson. 2011. "Sparse variational analysis of linear mixed models for large data sets". *Statistics & Probability Letters* 81, no. 8: 1056–1062. ISSN: 0167-7152. doi:10.1016/j.spl.2011.02.029.

Arulampalam, M.S., S. Maskell, N. Gordon, and T. Clapp. 2002. "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking". *IEEE Transactions on Signal Processing* 50: 174–188. ISSN: 1053587X. doi:10.1109/78.978374. http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=978374.

Al-Awadhi, Fahimah, Merrilee Hurn, and Christopher Jennison. 2004. "Improving the acceptance rate of reversible jump MCMC proposals". *Statistics & Probability Letters* 69, no. 2: 189–198. ISSN: 0167-7152. doi:10.1016/j.spl.2004.06.025.

Baez, John, and Mike Stay. 2010. "Physics, topology, logic and computation: a Rosetta Stone". In *New structures for physics*, 95–172. Springer.

Balog, Matej, Nilesh Tripuraneni, Zoubin Ghahramani, and Adrian Weller. 2017. "Lost Relatives of the Gumbel Trick". In *International Conference on Machine Learning*, 371–379.

Bandyopadhyay, Sanghamitra. 2005. "Simulated Annealing Using a Reversible Jump Markov Chain Monte Carlo Algorithm for Fuzzy Clustering". *IEEE Transactions on Knowledge and Data Engineering* 17.

Banterle, Marco, Clara Grazian, Anthony Lee, and Christian P. Robert. 2015. "Accelerating Metropolis – Hastings algorithms by Delayed Acceptance". arXiv: 1503.00996.

Banterle, Marco, Clara Grazian, and Christian P. Robert. 2014. "Accelerating Metropolis-Hastings algorithms: Delayed acceptance with prefetching". arXiv: 1406.2660.

Barber, David. 2013. *Bayesian Reasoning and Machine Learning*. Cambridge University Press. http://www.cs.ucl.ac.uk/staff/d.barber/brml/.

Barber, David, and Christopher K.I. Williams. 1997. "Gaussian processes for Bayesian classification via hybrid Monte Carlo". In *Advances in neural information processing systems*, 340–346.

Bardenet, Rémi, Arnaud Doucet, and Chris Holmes. 2017. "On Markov chain Monte Carlo methods for tall data". *The Journal of Machine Learning Research* 18: 1515–1557.

Barp, Alessandro, François-Xavier Briol, Anthony D. Kennedy, and Mark Girolami. 2017. "Geometry and Dynamics for Markov Chain Monte Carlo". *Annual Review of Statistics and Its Application* 5, no. 1. ISSN: 2326-831X. doi:10.1146/annurev-statistics-031017-100141.

Barrientos, Andrés F., Alejandro Jara, and Fernando a. Quintana. 2012. "On the Support of MacEachern's Dependent Dirichlet Processes and Extensions". *Bayesian Analysis* 7: 277–310. ISSN: 1931-6690. doi:10.1214/12-BA709. http://ba.stat.cmu.edu/journal/2012/vol07/issue02/barrientos.pdf.

Baydin, Atılım Günes, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. 2018. "Automatic Differentiation in Machine Learning: a Survey". *Journal of Machine Learning Research* 18:1–43.

Beal, Matthew J., Zoubin Ghahramani, and Carl E. Rasmussen. 2002. "The infinite hidden Markov model". In *Advances in neural information processing systems*, 577–584.

Bengtsson, Thomas, Peter Bickel, and Bo Li. 2008. "Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems". *Probability and Statistics: Essays in Honor of David A. Freedman*: 316–334. doi:10.1214/193940307000000518.

Berger, James O., José M. Bernardo, and Dongchu Sun. 2012. "Objective Priors for Discrete Parameter Spaces". *Journal of the American Statistical Association* 107, no. 498: 636–648. ISSN: 1537-274X. doi:10.1080/01621459.2012.682538.

Bernardo, José M., and Adrian F.M. Smith. 1994. *Bayesian Theory*. John Wiley & Sons.

Besag, Julian, Peter Green, David Higdon, and Kerrie Mengersen. 1995. "Bayesian Computation and Stochastic Systems". *Statistical Science* 10, no. 1: 3–41. ISSN: 0883-4237. doi:10.1214/ss/1177010123.

Beskos, Alexandros. 2014. "A stable manifold MCMC method for high dimensions". *Statistics & Probability Letters* 90: 46–52. ISSN: 0167-7152. doi:10.1016/j.spl.2014.03.016.

Beskos, Alexandros, Dan Crisan, and Ajay Jasra. 2014. "On the stability of sequential Monte Carlo methods in high dimensions". *The Annals of Applied Probability* 24, no. 4: 1396–1445. ISSN: 1050-5164. doi:10.1214/13-aap951.

Beskos, Alexandros, Mark Girolami, Shiwei Lan, Patrick E. Farrell, and Andrew M. Stuart. 2017a. "Geometric MCMC for infinite-dimensional inverse problems". *Journal of Computational Physics* 335:327–351.

Beskos, Alexandros, Ajay Jasra, Kody Law, Raul Tempone, and Yan Zhou. 2017b. "Multilevel sequential monte carlo samplers". *Stochastic Processes and their Applications* 127: 1417–1440.

Beskos, Alexandros, Frank J. Pinski, Jesús Marıa Sanz-Serna, and Andrew M. Stuart. 2011. "Hybrid monte carlo on hilbert spaces". *Stochastic Processes and their Applications* 121: 2201–2230.

Betancourt, Michael. 2015. "The fundamental incompatibility of scalable Hamiltonian Monte Carlo and naive data subsampling". In *International Conference on Machine Learning*, 533–540.

— . 2016a. "Diagnosing Suboptimal Cotangent Disintegrations in Hamiltonian Monte Carlo". arXiv: 1604.00695.

— . 2016b. "Identifying the Optimal Integration Time in Hamiltonian Monte Carlo". arXiv: 1601.00225.

— . 2017a. "A conceptual introduction to Hamiltonian Monte Carlo". arXiv: 1701.02434.

— . 2017b. "The Convergence of Markov chain Monte Carlo Methods: From the Metropolis method to Hamiltonian Monte Carlo". arXiv: 1706.01520.

Betancourt, Michael, Simon Byrne, Sam Livingstone, and Mark Girolami. 2017. "The geometric foundations of Hamiltonian Monte Carlo". *Bernoulli* 23, no. 4A: 2257–2298. ISSN: 1350-7265. doi:10.3150/16-bej810.

Betancourt, Michael, and Mark Girolami. 2015. "Hamiltonian Monte Carlo for Hierarchical Models". *Current Trends in Bayesian Methodology with Applications*: 79–101. doi:10.1201/b18502-5.

Betancourt, M.J. 2014. "Adiabatic Monte Carlo". arXiv: 1405.3489.

Billingsley, Patrick. 2012. *Probability and measure*. Wiley Series in Probability and Statistics. Wiley. ISBN: 9781118341919.

Blackwell, David, and James B. MacQueen. 1973. "Ferguson Distributions Via Polya Urn Schemes". *The Annals of Statistics* 1: 353–355.

Blei, David M., Thomas L. Griffiths, and Michael I. Jordan. 2010. "The Nested Chinese Restaurant Process and Bayesian Nonparametric Inference of Topic Hierarchies". *Journal of the ACM* 57, no. 2: 1–30. ISSN: 00045411. doi:10.1145/1667053.1667056. http://portal.acm.org/citation.cfm?doid=1667053.1667056.

Blei, David M., Thomas L. Griffiths, Michael I. Jordan, and Joshua B. Tenenbaum. 2004. "Hierarchical topic models and the nested chinese restaurant process". In *Advances in neural information processing systems*, 17–24.

Blei, David M., Alp Kucukelbir, and Jon D. McAuliffe. 2017. "Variational inference: A review for statisticians". *Journal of the American Statistical Association* 112: 859–877. arXiv: 1601.00670.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. "Latent dirichlet allocation". *Journal of machine Learning research* 3: 993–1022.

Blundell, Charles, and Yee Whye Teh. 2013. "Bayesian Hierarchical Community Discovery". In *Proceedings of Neural Information Processing Systems*, 1–9.

Blunsom, Phil, and Trevor Cohn. 2011. "A hierarchical Pitman-Yor process HMM for unsupervised part of speech induction". In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, 865–874. Association for Computational Linguistics.

Bonassi, Fernando V., and Mike West. 2015. "Sequential Monte Carlo with Adaptive Weights for Approximate Bayesian Computation". *Bayesian Analysis* 10, no. 1: 171–187. ISSN: 1936-0975. doi:10.1214/14-ba891.

Bouchard-Côté, Alexandre, Arnaud Doucet, and Andrew Roth. 2017. "Particle Gibbs Split-Merge Sampling for Bayesian Inference in Mixture Models". *Journal of Machine Learning Research* 18: 1–39. http://jmlr.org/papers/v18/15-397.html.

Box, George E.P., and George C. Tiao. 2011. *Bayesian inference in statistical analysis*. Vol. 40. John Wiley & Sons.

Bratières, Sébastien, Jurgen Van Gael, Andreas Vlachos, and Zoubin Ghahramani. 2010. "Scaling the iHMM: parallelization versus Hadoop". In *2010 10th IEEE International Conference on Computer and Information Technology (CIT 2010)*, 1235–1240. IEEE. doi:10.1109/cit.2010.223.

Broderick, Tamara, Nicholas Boyd, Andre Wibisono, Ashia C. Wilson, and Michael I. Jordan. 2013. "Streaming variational bayes". In *Advances in Neural Information Processing Systems*, 1727–1735.

Broderick, Tamara, Michael I. Jordan, and Jim Pitman. 2012. "Beta Processes, Stick-Breaking, and Power Laws". *Bayesian Analysis* 7: 439–476. ISSN: 1931-6690. doi:10.1214/12-BA715. http://ba.stat.cmu.edu/journal/2012/vol07/issue02/broderick.pdf.

— . 2013. "Cluster and Feature Modeling from Combinatorial Stochastic Processes". *Statistical Science* 28, no. 3: 289–312. ISSN: 0883-4237. doi:10.1214/13-STS434. arXiv: 1206.5862. http://projecteuclid.org/euclid.ss/1377696938.

Broderick, Tamara, Lester Mackey, John Paisley, and Michael I. Jordan. 2015. "Combinatorial clustering and the beta negative binomial process". *IEEE transactions on pattern analysis and machine intelligence* 37: 290–306.

Broderick, Tamara, Jim Pitman, and Michael I. Jordan. 2013. "Feature Allocations, Probability Functions, and Paintboxes". *Bayesian Analysis* 8, no. 4: 801–836. ISSN: 1936-0975. doi:10.1214/13-ba823.

Broderick, Tamara, Ashia C. Wilson, and Michael I. Jordan. 2018. "Posteriors, conjugacy, and exponential families for completely random measures". *Bernoulli* 24: 3181–3221. arXiv: 1410.6843.

Brooks, Steve, Andrew Gelman, Galin Jones, and Xiao-Li Meng. 2011. *Handbook of Markov Chain Monte Carlo*. CRC press.

Byrne, Simon, and Mark Girolami. 2013. "Geodesic Monte Carlo on embedded manifolds". *Scandinavian Journal of Statistics* 40: 825–845.

Campbell, Trevor, Miao Liu, Brian Kulis, Jonathan P. How, and Lawrence Carin. 2013. "Dynamic Clustering via Asymptotics of the Dependent Dirichlet Process Mixture". In *Proceedings of Neural Information Processing Systems*.

Canale, Antonio, and Bruno Scarpa. 2015. "Bayesian nonparametric location–scale–shape mixtures". *TEST* 25, no. 1: 113–130. ISSN: 1863-8260. doi:10.1007/s11749-015-0446-2.

Cappé, O., A. Guillin, J.M. Marin, and C.P. Robert. 2004. "Population Monte Carlo". *Journal of Computational and Graphical Statistics* 13, no. 4: 907–929. ISSN: 1537-2715. doi:10.1198/106186004x12803.

Cappe, Olivier, Simon J. Godsill, and Eric Moulines. 2007. "An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo". *Proceedings of the IEEE* 95, no. 5: 899–924. ISSN: 0018-9219. doi:10.1109/jproc.2007.893250.

Carlin, Bradley P., and Thomas A. Louis. 2009. *Bayesian Methods for Data Analysis*. Texts in Statistical Science. CRC Press. ISBN: 9781584886976.

Carvalho, Carlos M., Michael S. Johannes, Hedibert F. Lopes, and Nicholas G. Polson. 2010. "Particle Learning and Smoothing". *Statistical Science* 25, no. 1: 88–106. ISSN: 0883-4237. doi:10.1214/10-sts325.

Casella, G., K.L. Mengersen, C.P. Robert, and D.M. Titterington. 2002. "Perfect slice samplers for mixtures of distributions". *J Royal Statistical Soc B* 64, no. 4: 777–790. ISSN: 1467-9868. doi:10.1111/1467-9868.00360.

Chai, Kian Ming A. 2012. "Variational Multinomial Logit Gaussian Process". *Journal of Machine Learning Research* 13:1745–1808.

Chen, Tianshi, Thomas B. Schon, Henrik Ohlsson, and Lennart Ljung. 2011. "Decentralized Particle Filter With Arbitrary State Decomposition". *IEEE Transactions on Signal Processing* 59, no. 2: 465–478. ISSN: 1053-587X. doi:10.1109/tsp.2010.2091639.

Cheng, Qiang, Qiang Liu, Feng Chen, and Alexander Ihler. 2013. "Variational Planning for Graph-based MDPs". In *Proceedings of Neural Information Processing Systems*, 1–9.

Chopin, Nicolas, and Sumeetpal S. Singh. 2015. "On particle Gibbs sampling". *Bernoulli* 21, no. 3: 1855–1883. ISSN: 1350-7265. doi:10.3150/14-bej629.

Cotter, Colin, Simon Cotter, and Paul Russell. 2015. "Parallel Adaptive Importance Sampling". arXiv: 1508.01132.

Creal, Drew. 2012. "A Survey of Sequential Monte Carlo Methods for Economics and Finance". *Econometric Reviews* 31, no. 3: 245–296. ISSN: 1532-4168. doi:10.1080/07474938.2011.607333.

Culbertson, Jared, and Kirk Sturtz. 2013. "A Categorical Foundation for Bayesian Probability". *Applied Categorical Structures* 22, no. 4: 647–662. ISSN: 1572-9095. doi:10.1007/s10485-013-9324-9.

Dai, Bo, Niao He, Hanjun Dai, and Le Song. 2015. "Scalable Bayesian inference via particle mirror descent". In *Artificial Intelligence and Statistics*. arXiv: 1506.03101.

Dalal, S.R. 1979. "Dirichlet Invariant Processes and Applications to Nonparametric Estimation of Symmetric Distribution". *Stochastic Processes and their Applications* 9:99–107.

Del Moral, Pierre, Arnaud Doucet, and Ajay Jasra. 2006. "Sequential Monte Carlo samplers". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68, no. 3: 411–436. ISSN: 1467-9868. doi:10.1111/j.1467-9868.2006.00553.x.

Del Moral, Pierre, and Lawrence M. Murray. 2015. "Sequential Monte Carlo with Highly Informative Observations". *SIAM/ASA J. Uncertainty Quantification* 3, no. 1: 969–997. ISSN: 2166-2525. doi:10.1137/15m1011214.

Dembski, William a. 1990. "Uniform probability". *Journal of Theoretical Probability* 3: 611–626. ISSN: 08949840. doi:10.1007/BF01046100.

Dheeru, Dua, and Efi Karra Taniskidou. 2017. *UCI Machine Learning Repository*. University of California, Irvine, School of Information and Computer Sciences. http://archive.ics.uci.edu/ml.

Diaconis, Persi. 1988. "Recent progress on de Finetti's notions of exchangeability". *Bayesian statistics* 3:111–125.

Diaconis, Persi, and Donald Ylvisaker. 1979. "Conjugate Priors for Exponential Families". *The Annals of Statistics* 7.

Dinh, Vu, Aaron E. Darling, and Frederick A. Matsen IV. 2017. "Online Bayesian phylogenetic inference: theoretical foundations via Sequential Monte Carlo". *Systematic biology* 67: 503–517. arXiv: 1610.08148.

Douc, R., and O. Cappe. 2005. "Comparison of resampling schemes for particle filtering". In *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005.* IEEE. ISBN: 953-184-089-X. doi:10.1109/ispa.2005.195385.

Doucet, Arnaud, Nando de Freitas, and Neil Gordon. 2001. *Sequential Monte Carlo Methods in Practice*. Springer Nature. ISBN: 978-1-4757-3437-9. doi:10.1007/978-1-4757-3437-9.

Doucet, Arnaud, Simon J. Godsill, and Christian P. Robert. 2002. "Marginal maximum a posteriori estimation using Markov chain Monte Carlo". *Statistics and Computing* 12: 77–84.

Doucet, Arnaud, Simon Godsill, and Christophe Andrieu. 2000. "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering". *Statistics and Computing* 10:197–208.

Doucett, Arnaud, Nando De Freitast, Kevin Murphy, and Stuart Russell. 2000. "Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks". In *Uncertainty in Artificial Intelligence*.

Duane, Simon, Anthony D. Kennedy, Brian J. Pendleton, and Duncan Roweth. 1987. "Hybrid monte carlo". *Physics letters B* 195: 216–222.

Dunson, D.B., and J.-H. Park. 2008. "Kernel stick-breaking processes". *Biometrika* 95, no. 2: 307–323. ISSN: 1464-3510. doi:10.1093/biomet/asn012.

Escobar, Michael D., and Mike West. 1995. "Bayesian Density Estimation and Inference Using Mixtures". *Journal of the American Statistical Association* 90: 577–588.

Everitt, Richard G., Richard Culliford, Felipe Medina-Aguayo, and Daniel J. Wilson. 2016. "Sequential Bayesian inference for mixture models and the coalescent using sequential Monte Carlo samplers with transformations". arXiv: 1612.06468.

Faden, Arnold M. 1985. "The Existence of Regular Conditional Probabilities: Necessary and Sufficient Conditions". *The Annals of Probability* 13, no. 1: 288–298. ISSN: 0091-1798. doi:10.1214/aop/1176993081.

Fearnhead, Paul. 1998. "Sequential Monte Carlo methods in filter theory". PhD thesis, University of Oxford Oxford.

— . 2002. "Markov chain Monte Carlo, Sufficient Statistics, and Particle Filters". *Journal of Computational and Graphical Statistics* 11, no. 4: 848–862. ISSN: 1537-2715. doi:10.1198/106186002835.

Ferguson, Thomas S. 1973. "A Bayesian Analysis of Some Nonparametric Problems". *The Annals of Statistics* 1, no. 2: 209–230. ISSN: 0090-5364. doi:10.1214/aos/1176342360.

— . 1974. "Prior Distributions on Spaces of Probability Measures". *The Annals of Statistics* 2, no. 4: 615–629. ISSN: 0090-5364. doi:10.1214/aos/1176342752.

— . 1983. "Bayesian Density Estimation by Mixtures of Normal Distributions". *Recent Advances in Statistics*.

Fill, James Allen, and Mark L. Huber. 2010. "Perfect Simulation of Vervaat Perpetuities". *Electronic Journal of Probability* 15, no. 0. ISSN: 1083-6489. doi:10.1214/ejp.v15-734.

de Finetti, Bruno. 1974. "Theory of Probability". Ed. by Antonio Machí and Adrian Smith. *Wiley Series in Probability and Statistics*.

— . 1931. "Funzione caratteristica di un fenomeno aleatorio". In *Atti della R. Accademia Nazionale dei Lincii Ser. 6, Memorie, classe di Scienze, Fisiche, Matamatiche e Naturali*, 251–299. 4.

— . 1938. "Sur la condition d'equivalence partielle". *Actualites Scientifiques et Industrielles*, no. 739.

Fink, Daniel. 1997. *A Compendium of Conjugate Priors*. Tech. rep. Montana State University.

Fox, Emily B., Michael C. Hughes, Erik B. Sudderth, and Michael I. Jordan. 2014. "Joint modeling of multiple time series via the beta process with application to motion capture segmentation". *The Annals of Applied Statistics* 8, no. 3: 1281–1313. ISSN: 1932-6157. doi:10.1214/14-AOAS742. arXiv: 1308.4747. http://projecteuclid.org/euclid.aoas/1414091214.

Franc, Vojtěch, Alexander Zien, and Bernhard Schölkopf. 2011. "Support vector machines as probabilistic models". In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 665–672.

Gelman, Andrew. 2008. "Objections to Bayesian statistics". *Bayesian Analysis* 3, no. 3: 445–449. ISSN: 1936-0975. doi:10.1214/08-ba318.

Gelman, Andrew, John B. Carlin, Hal S. Stern, and Donald B. Rubin. 2004. *Bayesian data analysis*. 2nd ed. Texts in Statistical Science. Chapman & Hall/CRC.

Gelman, Andrew, and Christian P. Robert. 2013. ""Not Only Defended But Also Applied": The Perceived Absurdity of Bayesian Inference". *The American Statistician* 67, no. 1: 1–5. ISSN: 0003-1305. doi:10.1080/00031305.2013.760987.

Gershman, Samuel J., Peter I. Frazier, and David M. Blei. 2015. "Distance dependent infinite latent feature models". *IEEE transactions on pattern analysis and machine intelligence* 37: 334–345.

Ghahramani, Zoubin, and Thomas L. Griffiths. 2006. "Infinite latent feature models and the Indian buffet process". In *Advances in Neural Information Processing Systems 18*, ed. by Y. Weiss, B. Schölkopf, and J.C. Platt, 475–482. MIT Press. http://papers.nips.cc/paper/2882-infinite-latent-feature-models-and-the-indian-buffet-process.pdf.

Ghahramani, Zoubin, Thomas L. Griffiths, and Peter Sollich. 2006. "Bayesian nonparametric latent feature models". In *8th World Meeting on Bayesian Statistics*, 1–19.

Ghahramani, Zoubin, Michael I. Jordan, and Ryan P. Adams. 2010. "Tree-Structured Stick Breaking for Hierarchical Data". In *Advances in Neural Information Processing Systems 23*, ed. by J.D. Lafferty, C.K.I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, 19–27. Curran Associates, Inc. http://papers.nips.cc/paper/4108-tree-structured-stick-breaking-for-hierarchical-data.pdf.

Ghosal, S., J.K. Ghosh, and R.V. Ramamoorthi. 1999. "Posterior Consistency of Dirichlet Mixtures in Density Estimation". *The Annals of Statistics* 27: 143–158.

Ghosh, J.K., and R.V. Ramamoorthi. 2003. *Bayesian Nonparametrics*. Springer Series in Statistics. Springer. ISBN: 0387955372.

Gilks, Walter R., and Carlo Berzuini. 2001. "Following a moving target-Monte Carlo inference for dynamic Bayesian models". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 63, no. 1: 127–146. ISSN: 1467-9868. doi:10.1111/1467-9868.00280.

Gilks, W.R., S. Richardson, and D.J. Spiegelhalter. 1996. *Markov Chain Monte Carlo in Practice*. Chapman & Hall. ISBN: 9780412055515.

Gnedin, Alexander, Ben Hansen, and Jim Pitman. 2007. "Notes on the occupancy problem with infinitely many boxes: general asymptotics and power laws". *Probability surveys* 4:146–171.

Gnedin, Alexander, and Sergei Kerov. 2001. "A Characterization of GEM Distributions". *Combinatorics, Probability and Computing* 10, no. 03. ISSN: 1469-2163. doi:10.1017/s0963548301004692.

Golightly, Andrew, Daniel A. Henderson, and Chris Sherlock. 2014. "Delayed acceptance particle MCMC for exact inference in stochastic kinetic models". *Stat Comput* 25, no. 5: 1039–1055. ISSN: 1573-1375. doi:10.1007/s11222-014-9469-x.

Gordon, N.J., D.J. Salmond, and A.F.M. Smith. 1993. "Novel approach to nonlinear/non-Gaussian Bayesian state estimation". *IEE Proceedings F Radar and Signal Processing* 140: 107. ISSN: 0956-375X. doi:10.1049/ip-f-2.1993.0015.

Görür, Dilan, and Carl Edward Rasmussen. 2010. "Dirichlet Process Gaussian Mixture Models: Choice of the Base Distribution". *Journal of Computer Science and Technology* 25: 615–626. doi:10.1007/s11390-010-1051-1.

Green, Peter J. 1995. "Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination". *Biometrika* 82, no. 4: 711. ISSN: 00063444. doi:10.2307/2337340. http://www.jstor.org/stable/2337340?origin=crossref.

Green, Peter J., and David I. Hastie. 2009. "Reversible jump MCMC". *Genetics* 155: 1391–1403.

Green, Peter J., Krzysztof Łatuszyński, Marcelo Pereyra, and Christian P. Robert. 2015. "Bayesian computation: a summary of the current state, and samples backwards and forwards". *Statistics and Computing*: 835–862. ISSN: 0960-3174. doi:10.1007/s11222-015-9574-5.

Green, Peter J., and Duncan J. Murdoch. 1998. "Exact Sampling for Bayesian Inference: Towards General Purpose Algorithms". *Bayesian Statistics* 6.

Green, Peter J., and Sylvia Richardson. 2001. "Modelling Heterogeneity With and Without the Dirichlet Process". *The Scandinavian Journal of Statistics* 28:355–375.

Green, P.J., and A. Mira. 2001. "Delayed rejection in reversible jump Metropolis-Hastings". *Biometrika* 88, no. 4: 1035–1053. ISSN: 1464-3510. doi:10.1093/biomet/88.4.1035.

Griffiths, Thomas L., and Zoubin Ghahramani. 2005. *Infinite Latent Feature Models and the Indian Buffet Process*. Tech. rep. University College London.

Guo, Ruixin, and Sounak Chakraborty. 2009. "Bayesian Adaptive Nearest Neighbor". *Static Analysis and Data Mining*. doi:10.1002/sam.

Hajek, Bruce. 1988. "Cooling Schedules for Optimal Annealing". *Mathematics of Operations Research* 13: 311–330.

Handschin, J.E. 1970. "Monte Carlo techniques for prediction and filtering of non-linear stochastic processes". *Automatica* 6: 555–563.

Handschin, Johannes Edmund, and David Q. Mayne. 1969. "Monte Carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering". *International journal of control* 9: 547–559.

Hastie, David I., and Peter J. Green. 2012. "Model choice using reversible jump Markov chain Monte Carlo". *Statistica Neerlandica* 66, no. 3: 309–338. ISSN: 0039-0402. doi:10.1111/j.1467-9574.2012.00516.x.

Hastings, W.K. 1970. "Monte Carlo sampling methods using Markov chains and their applications". *Biometrika* 57: 97–109.

Hauer, Ezra. 2004. "The harm done by tests of significance". *Accident Analysis and Prevention* 36:495–500. ISSN: 00014575. doi:10.1016/S0001-4575(03)00036-8.

Hazan, Tamir, Subhransu Maji, and Tommi Jaakkola. 2013. "On Sampling from the Gibbs Distribution with Random Maximum A-Posteriori Perturbations". In *Proceedings of Neural Information Processing Systems*, 1–9.

Heaukulani, Creighton, and Daniel M. Roy. 2015. "Gibbs-type Indian buffet processes". arXiv: 1512.02543.

— . 2016. "The combinatorial structure of beta negative binomial processes". *Bernoulli* 22, no. 4: 2301–2324. ISSN: 1350-7265. doi:10.3150/15-bej729.

Hjort, Nils Lid. 1990. "Nonparametric Bayes Estimators Based on Beta Processes in Models for Life History Data". *The Annals of Statistics* 18, no. 3: 1259–1294. ISSN: 0090-5364. doi:10.1214/aos/1176347749.

Hjort, Nils Lid, Chris Holmes, Peter Müller, and Stephen G. Walker. 2010. *Bayesian Nonparametrics*. Cambridge Series in Statistical and Probabilitistic Mathematics. Cambridge University Press. ISBN: 9780521513463.

Hoffman, Matthew D., and Andrew Gelman. 2014. "The No-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo." *Journal of Machine Learning Research* 15: 1593–1623.

Hoover, Douglas N. 1979. *Relations on probability spaces and arrays of random variables*. Tech. rep. Institute for Advanced Study, Princeton, NJ.

Hu, Yuening, Ke Zhai, Sinead Williamson, and Jordan Boyd-graber. 2012. "Modeling Images using Transformed Indian Buffet Processes". In *International Conference on Machine Learning 2012*.

Iba, Yukito. 2003. "Population Annealing: An approach to finite-temperature calculation". *Joint Workshop of Hayashibara Foundation and SMAPIP*: 1–7. http://www.smapip.is.tohoku.ac.jp/%7B~%7Dsmapip/2003/hayashibara/proceedings/YukitoIba.pdf.

Ishiguro, Katsuhiko, Naonori Ueda, and Hiroshi Sawada. 2012. "Subset Infinite Relational Models". In *International Conference on Artificial Intelligence and Statistics*, vol. XX.

Iwata, Tomoharu, David Duvenaud, and Zoubin Ghahramani. 2013. "Warped mixtures for nonparametric cluster shapes". In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, 311–320. AUAI Press.

Jaakkola, T., and M. Jordan. 1997. "A variational approach to Bayesian logistic regression models and their extensions". In *Sixth International Workshop on Artificial Intelligence and Statistics*, 82:4.

Jaakkola, Tommi S., and Michael I. Jordan. 1996. "Computing upper and lower bounds on likelihoods in intractable networks". In *Proceedings of the Twelfth international conference on Uncertainty in artificial intelligence*, 340–348. Morgan Kaufmann Publishers Inc.

— . 2000. "Bayesian Parameter Estimation via Variational Methods". *Statistics and Computing* 10:25–37.

Jaakkola, Tommi, and Tamir Hazan. 2012. "On the Partition Function and Random Maximum A-Posteriori Perturbations". In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, 991–998. ISBN: 978-1-4503-1285-1.

Jackson, Edmund, Manuel Davy, Arnaud Doucet, and William J. Fitzgerald. 2007. "Bayesian unsupervised signal classification by Dirichlet process mixtures of Gaussian processes". In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, 3:III–1077. IEEE. doi:10.1109/icassp.2007.366870.

James, Lancelot F., Peter Orbanz, and Yee Whye Teh. 2015. "Scaled subordinators and generalizations of the Indian buffet process". arXiv: 1510.07309.

Jasra, Ajay, David a. Stephens, and Chris C. Holmes. 2007a. "Population-Based Reversible Jump Markov Chain Monte Carlo". *Biometrika* 94:787–807. ISSN: 0006-3444. doi:10.1093/biomet/asm069. arXiv: 0711.0186.

Jasra, Ajay, David a. Stephens, and Christopher C. Holmes. 2007b. "On population-based simulation for static inference". *Statistics and Computing* 17:263–279. ISSN: 09603174. doi:10.1007/s11222-007-9028-9.

Jordan, Michael I. 2010. "Hierarchical models, nested models and completely random measures". *Frontiers of statistical decision making and Bayesian analysis: In honor of James O. Berger. New York: Springer*: 207–218.

Jordan, Michael I., Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. 1999. "An Introduction to Variational Methods for Graphical Models". *Machine Learning* 37:183–233.

Kadane, Joseph. 2011. *Principles of Uncertainty*. Chapman & Hall/CRC Texts in Statistical Science. Chapman / Hall/CRC. ISBN: 9781439861622. doi:10.1201/b11322.

Kallenberg, Olav. 1997. *Foundations of Modern Probability*. 535. Probability and its Applications. Springer.

Kantas, Nikolas, Arnaud Doucet, Sumeetpal S. Singh, Jan Maciejowski, and Nicolas Chopin. 2015. "On Particle Methods for Parameter Estimation in State-Space Models". *Statist. Sci.* 30, no. 3: 328–351. ISSN: 0883-4237. doi:10.1214/14-sts511.

Kass, Robert E., and Larry Wasserman. 1996. "The Selection of Prior Distributions by Formal Rules". *Journal of the American Statistical Association* 91: 1343–1370.

Kemp, Charles, Thomas L. Griffiths, Sean Stromsten, and Joshua B. Tenenbaum. 2004. "Semi-supervised learning with trees". In *Advances in neural information processing systems*, 257–264.

Kemp, Charles, Joshua B. Tenenbaum, Thomas L. Griffiths, Takeshi Yamada, and Naonori Ueda. 2006. "Learning Systems of Concepts with an Infinite Relational Model". In *National Conference on Artificial Intelligence*.

Kilbinger, Martin, Darren Wraith, Christian P. Robert, Karim Benabed, Olivier Cappé, Jean-François Cardoso, Gersende Fort, Simon Prunet, and François R. Bouchet. 2010. "Bayesian model comparison in cosmology with Population Monte Carlo". *Monthly Notices of the Royal Astronomical Society*. ISSN: 1365-2966. doi:10.1111/j.1365-2966.2010.16605.x.

Kim, Dongwoo, Suin Kim, and Alice Oh. 2012. "Dirichlet Process with Mixed Random Measures: A Nonparametric Topic Model for Labeled Data". In *International Conference on Machine Learning 2012*.

Kim, Seyoung, and P. Smyth. 2006. "Hierarchical Dirichlet processes with random effects". *Nips*. https://papers.nips.cc/paper/2975-hierarchical-dirichlet-processes-with-random-effects.pdf.

Kingman, J.F.C. 1974. "Random Discrete Distributions". In *Royal Statistical Society*, vol. 1. 2.

— . 1982. "The coalescent". *Stochastic Processes and their Applications* 13, no. 3: 235–248. ISSN: 03044149. doi:10.1016/0304-4149(82)90011-4. http://linkinghub.elsevier.com/retrieve/pii/0304414982900114.

Kingman, John. 1967. "Completely random measures". *Pacific Journal of Mathematics* 21, no. 1: 59–78. ISSN: 0030-8730. doi:10.2140/pjm.1967.21.59. http://msp.org/pjm/1967/21-1/p06.xhtml.

Kitagawa, Genshiro. 1996. "Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models". *Journal of Computational and Graphical Statistics* 5, no. 1: 1–25. ISSN: 1537-2715. doi:10.1080/10618600.1996.10474692.

Kivinen, Jyri J., Erik B. Sudderth, and Michael I. Jordan. 2007a. "Image denoising with nonparametric hidden Markov trees". In *Image Processing, 2007. IEEE International Conference on*, 3:III–121. IEEE.

— . 2007b. "Learning multiscale representations of natural scenes using Dirichlet processes". In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 1–8. IEEE. doi:10.1109/ICCV.2007.4408870.

Kiwaki, Taichi. 2002. "Variational Optimization of Annealing Schedules". arXiv: 1502.05313.

Knowles, D.A., J. Van Gael, and Z. Ghahramani. 2011. "Message Passing Algorithms for Dirichlet Diffusion Trees". In *International Conference on Machine Learning*.

Knowles, David A., and Zoubin Ghahramani. 2010. "Pitman-Yor Diffusion Trees". In *Uncertainty in artificial intelligence*.

Knowles, David Arthur. 2012. "Bayesian non-parametric models and inference for sparse and hierarchical latent structure". PhD thesis, University of Cambridge.

Knowles, David, and Zoubin Ghahramani. 2007. "Infinite Sparse Factor Analysis and Infinite Independent Components Analysis". In *International Conference on Independent Component Analysis and Signal Separation*. 1.

— . 2011. "Nonparametric Bayesian Sparse Factor Models with Application to Gene Expression Modelling". *Annals of Applied Statistics*: 1–21. arXiv: 1011.6293 [math.PR].

Knudson, Karin, and Jonathan W. Pillow. 2013. "Spike train entropy-rate estimation using hierarchical Dirichlet process priors". In *Proceedings of Neural Information Processing Systems*, 1–9. 1.

Ko, Young Jun, and Matthias Seeger. 2012. "Large Scale Variational Bayesian Inference for Structured Scale Mixture Models". In *International Conference on Machine Learning 2012*.

Kong, Augustine, Jun S. Liu, and Wing Hung Wong. 1994. "Sequential Imputations and Bayesian Missing Data Problems". *Journal of the American Statistical Association* 89, no. 425: 278–288. ISSN: 1537-274X. doi:10.1080/01621459.1994.10476469.

Kottas, Athanasios, Peter Müller, and Fernando Quintana. 2005. "Nonparametric Bayesian Modeling for Multivariate Ordinal Data". *Journal of Computational and Graphical Statistics* 14, no. 3: 610–625. ISSN: 1537-2715. doi:10.1198/106186005x63185.

Kreweras, Germain. 1978. "Complexité et circuits eulériens dans les sommes tensorielles de graphes". *Journal of Combinatorial Theory, Series B* 24: 202–212.

Kucukelbir, Alp, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M. Blei. 2017. "Automatic differentiation variational inference". *The Journal of Machine Learning Research* 18: 430–474.

Kulis, Brian, and Michael Jordan. 2012. "Revisiting k-means: New Algorithms via Bayesian Nonparametrics". In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, ed. by John Langford and Joelle Pineau, 513–520. ICML '12. Edinburgh, Scotland, GB: Omnipress. ISBN: 978-1-4503-1285-1.

Kullback, S., and R.A. Leibler. 1951. "On Information and Sufficiency". *The Annals of Mathematical Statistics* 22, no. 1: 79–86. ISSN: 0003-4851. doi:10.1214/aoms/1177729694.

Künsch, Hans R. 2013. "Particle filters". *Bernoulli* 19, no. 4: 1391–1403. ISSN: 1350-7265. doi:10.3150/12-bejsp07.

Laarhoven, Peter J.M. Van, and Emile H.L. Aarts. 1987. *Simulated Annealing, Theory with Applications*. 187. Mathematics and Its Applications. Springer-Science+Business Media, B.V. ISBN: 978-94-015-7744-1. doi:10.1007/978-94-015-7744-1.

Lakshminarayanan, Balaji, Daniel M. Roy, and Yee Whye Teh. 2014. "Mondrian forests: Efficient online random forests". In *Advances in neural information processing systems*, 3140–3148.

— . 2016. "Mondrian Forests for Large-Scale Regression when Uncertainty Matters". In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, 1478–1487. http://www.jmlr.org/proceedings/papers/v51/lakshminarayanan16.pdf.

Lan, Shiwei, Jeffrey Streets, and Babak Shahbaba. 2014. "Wormhole Hamiltonian Monte Carlo". In *AAAI*, 1953–1959.

Lavine, Michael. 1992. "Some Aspects of Polya Tree Distributions for Statistical Modelling". *The Annals of Statistics* 20, no. 3: 1222–1235. ISSN: 0090-5364. doi:10.1214/aos/1176348767.

— . 1994. "More aspects of Polya tree distributions for statistical modelling". *The Annals of Statistics*: 1161–1176.

Li, Wei, David M. Blei, and Andrew McCallum. 2007. "Nonparametric Bayes Pachinko Allocation". In *UAI 2007, Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*, 243–250.

Li, Wei, and Andrew McCallum. 2006. "Pachinko allocation: DAG-structured mixture models of topic correlations". In *Proceedings of the 23rd international conference on Machine learning*, 577–584. ACM.

Lin, Dahua, Eric Grimson, and John W. Fisher. 2010. "Construction of dependent Dirichlet processes based on Poisson processes". In *Advances in neural information processing systems*, 1396–1404.

Lindsten, Fredrik, Michael I. Jordan, and Thomas B. Schön. 2014. "Particle Gibbs with Ancestor Sampling". *Journal of Machine Learning Research* 15:2145–2184.

Liu, Jane, and Mike West. 2001. "Combined Parameter and State Estimation in Simulation-Based Filtering". In *Sequential Monte Carlo Methods in Practice*, ed. by Arnaud Doucet, Nando de Freitas, and Neil Gordon, 197–223. Springer New York. ISBN: 978-1-4757-3437-9. doi:10.1007/978-1-4757-3437-9_10.

Liu, Jun S., and Rong Chen. 1998. "Sequential Monte Carlo Methods for Dynamic Systems". *Journal of the American Statistical Association* 93, no. 443: 1032–1044. ISSN: 1537-274X. doi:10.1080/01621459.1998.10473765.

Livingstone, Samuel, Michael Betancourt, Simon Byrne, and Mark Girolami. 2016. "On the geometric ergodicity of Hamiltonian Monte Carlo". arXiv: 1601.08057.

Livingstone, Samuel, Michael F. Faulkner, and Gareth O. Roberts. 2017. "Kinetic energy choice in Hamiltonian/hybrid Monte Carlo". arXiv: 1706.02649.

Lloyd, James Robert, Peter Orbanz, Zoubin Ghahramani, and Daniel M. Roy. 2012. "Random function priors for exchangeable arrays with applications to graphs and relational data". In *Advances in Neural Information Processing Systems*, 1–9.

Lo, Albert Y. 1984. "On a Class of Bayesian Nonparametric Estimates: I. Density Estimates". *The Annals of Statistics* 12: 351–357.

Mac Lane, Saunders. 1978. *Categories for the working mathematician*. Springer Science & Business Media.

MacEachern, Steven N. 1999. "Dependent nonparametric processes". *ASA proceedings of the section on bayesian statistical science*: 50–55. http://aima.eecs.berkeley.edu/%7B~%7Drussell/classes/cs294/f05/papers/maceachern-1999.pdf.

Mackay, D.J.C., and M.N. Gibbs. 2000. "Variational Gaussian process classifiers". *IEEE Transactions on Neural Networks* 11: 1458–1464. ISSN: 1045-9227. doi:10.1109/72.883477.

Maddison, Chris J. 2016. "A Poisson process model for Monte Carlo". In *Perturbation, Optimization, and Statistics*, ed. by Tamir Hazan, George Papandreou, and Daniel Tarlow, 193–232. MIT Press. arXiv: 1602.05986.

Maddison, Chris J., Daniel Tarlow, and Tom Minka. 2014. "A* sampling". In *Advances in Neural Information Processing Systems*, 3086–3094.

Maire, Florian, Nial Friel, and Pierre Alquier. 2015. "Light and Widely Applicable MCMC : Approximate Bayesian Inference for Large Datasets". arXiv: 1503.04178.

Marin, Jean-Michel, Louis Raynal, Pierre Pudlo, Mathieu Ribatet, and Christian P. Robert. 2016. "ABC random forests for Bayesian parameter inference". arXiv: 1605.05537.

Mauldin, R. Daniel, William D. Sudderth, and S.C. Williams. 1992. "Polya Trees and Random Distributions". *The Annals of Statistics* 20: 1203–1221.

Meent, Jan-Willem van de, Hongseok Yang, Vikash Mansinghka, and Frank Wood. 2015. "Particle Gibbs with Ancestor Sampling for Probabilistic Programs". In *AISTATS*.

Metropolis, Nicholas, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. 1953. "Equation of state calculations by fast computing machines". *The journal of chemical physics* 21: 1087–1092.

Miller, Kurt T., Thomas L. Griffiths, and Michael I. Jordan. 2008. "The Phylogenetic Indian Buffet Process: A Non-Exchangeable Nonparametric Prior for Latent Features". In *UAI 2008, Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence, Helsinki, Finland, July 9-12, 2008*, 403–410.

— . 2009. "Nonparametric Latent Feature Models for Link Prediction". In *Advances in Neural Information Processing Systems*.

Minka, Tom. 2005. *Divergence measures and message passing*. Tech. rep. Technical report, Microsoft Research.

Mira, Antonietta. 1998. "Ordering, slicing and splitting Monte Carlo Markov chains". PhD thesis, University of Minnesota.

Morey, Richard D., Rink Hoekstra, Jeffrey N. Rouder, Michael D. Lee, and Eric-Jan Wagenmakers. 2015. "The fallacy of placing confidence in confidence intervals". *Psychonomic Bulletin & Review* 23, no. 1: 103–123. ISSN: 1531-5320. doi:10.3758/s13423-015-0947-8.

Mørup, Morten, Mikkel N. Schmidt, and Lars Kai Hansen. 2011. "Infinite multiple membership relational modeling for complex networks". In *2011 IEEE International Workshop on Machine Learning for Signal Processing*. IEEE. ISBN: 9781457716218. doi:10.1109/mlsp.2011.6064546.

Munkres, James R. 2000. *Topology*. 2nd ed. Prentice Hall.

Murdoch, D.J., and P.J. Green. 1998. "Exact Sampling from a Continuous State Space". *Scandinavian Journal of Statistics* 25, no. 3: 483–502. ISSN: 0303-6898. doi:10.1111/1467-9469.00116.

Murray, Iain, David MacKay, and Ryan P. Adams. 2009. "The Gaussian Process Density Sampler". In *Advances in Neural Information Processing Systems 21*, ed. by D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, 9–16. Curran Associates, Inc. http://papers.nips.cc/paper/3410-the-gaussian-process-density-sampler.pdf.

Naesseth, Christian Andersson, Fredrik Lindsten, and Thomas B. Schön. 2014. "Sequential Monte Carlo for graphical models". In *Advances in Neural Information Processing Systems*, 1862–1870.

Naesseth, Christian, Fredrik Lindsten, and Thomas Schön. 2015. "Nested sequential monte carlo methods". In *International Conference on Machine Learning*, 1292–1301.

Nalisnick, Eric, and Padhraic Smyth. 2016. "Deep Generative Models with Stick-Breaking Priors". arXiv: 1605.06197.

Navarro, Daniel J., Thomas L. Griffiths, Mark Steyvers, and Michael D. Lee. 2006. "Modeling individual differences using Dirichlet processes". *Journal of Mathematical Psychology* 50, no. 2: 101–122. ISSN: 0022-2496. doi:10.1016/j.jmp.2005.11.006.

Neal, Radford M. 1992. *Bayesian Training of Backpropagation Networks by the Hybrid Monte Carlo Method*. Tech. rep. University of Toronto.

— . 1993a. "Bayesian learning via stochastic dynamics". In *Advances in neural information processing systems*, 475–482.

— . 1993b. *Probabilistic Inference Using Markov Chain Monte Carlo Methods*. Tech. rep. University of Toronto.

— . 1995. *Suppressing Random Walks in Markov Chain Monte Carlo Using Ordered Overrelaxation*. Tech. rep. 9508. University of Toronto.

— . 2001. *Defining Priors for Distributions Using Dirichlet Diffusion Trees*. Tech. rep. University of Toronto.

— . 2003. "Slice sampling". *The Annals of Statistics* 31, no. 3: 705–767. ISSN: 0090-5364. doi:10.1214/aos/1056562461.

— . 2012. "How to view an MCMC simulation as a permutation, with applications to parallel simulation and improved importance sampling". arXiv: 1205.0070.

Neiswanger, Willie, Chong Wang, and Eric P. Xing. 2014. "Asymptotically exact, embarrassingly parallel MCMC". In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, 623–632. AUAI Press.

Nemeth, Christopher, Paul Fearnhead, and Lyudmila Mihaylova. 2014. "Sequential Monte Carlo Methods for State and Parameter Estimation in Abruptly Changing Environments". *IEEE Transactions on Signal Processing* 62, no. 5: 1245–1255. ISSN: 1941-0476. doi:10.1109/tsp.2013.2296278.

Nishimura, Akihiko, David Dunson, and Jianfeng Lu. 2017. "Discontinuous Hamiltonian Monte Carlo for sampling discrete parameters". arXiv: 1705.08510.

Ohama, Iku, Hiromi Iida, Takuya Kida, and Hiroki Arimura. 2013. "An Extension of the Infinite Relational Model Incorporating Interaction between Objects". In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 147–159.

Orabona, Francesco, Tamir Hazan, Anand Sarwate, and Tommi Jaakkola. 2014. "On measure concentration of random maximum a-posteriori perturbations". In *International Conference on Machine Learning*, 432–440.

Orbanz, Peter, and Daniel M. Roy. 2015. "Bayesian Models of Graphs, Arrays and Other Exchangeable Random Structures". *IEEE transactions on pattern analysis and machine intelligence* 37: 437–461. ISSN: 0162-8828. doi:10.1109/TPAMI.2014.2334607. arXiv: 1312.7857.

Paige, Brooks, Frank Wood, Arnaud Doucet, and Yee Whye Teh. 2014. "Asynchronous anytime sequential monte carlo". In *Advances in Neural Information Processing Systems*, 3410–3418.

Paisley, John, David M. Blei, and Michael I. Jordan. 2012. "Variational Bayesian Inference with Stochastic Search". In *International Conference on Machine Learning 2012*. 2000.

Paisley, John, Chong Wang, and David M. Blei. 2012. "The Discrete Infinite Logistic Normal Distribution". *Bayesian Analysis* 7: 235–272.

Paisley, John, Chong Wang, David M. Blei, and Michael I. Jordan. 2015. "Nested Hierarchical Dirichlet Processes". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, no. 2: 256–270. ISSN: 2160-9292. doi:10.1109/tpami.2014.2318728.

Paisley, John, Aimee Zaas, Christopher W. Woods, Geoffrey S. Ginsburg, and Lawrence Carin. 2010. "A Stick-Breaking Construction of the Beta Process". In *International Conference on Machine Learning*.

Pakman, Ari, and Liam Paninski. 2013. "Auxiliary-variable exact Hamiltonian Monte Carlo samplers for binary distributions". In *Advances in neural information processing systems*, 2490–2498.

Palla, Konstantina, and David A. Knowles. 2012. "An Infinite Latent Attribute Model for Network Data". In *International Conference on Machine Learning 2012*.

Palla, Konstantina, David A. Knowles, and Zoubin Ghahramani. 2014. "A reversible infinite HMM using normalised random measures". In *International Conference on Machine Learning*.

Papandreou, George, and Alan L. Yuille. 2011. "Perturb-and-map random fields: Using discrete optimization to learn and sample from energy models". In *Computer Vision (ICCV), 2011 IEEE International Conference on*, 193–200. IEEE. doi:10.1109/ICCV.2011.6126242.

Perman, Mihael, Jim Pitman, and Marc Yor. 1992. "Size-biased sampling of Poisson point processes and excursions". *Probab. Th. Rel. Fields* 92, no. 1: 21–39. ISSN: 1432-2064. doi:10.1007/bf01205234.

Petrone, Sonia. 1999a. "Bayesian density estimation using Bernstein polynomials". *The Canadian Journal of Statistics* 27: 105–126.

— . 1999b. "Random Bernstein Polynomials". *Scandinavian Journal of Statistics* 26:373–393.

Pitman, Jim. 1996. "Some Developments of the {Blackwell-Macqueen} {URN} Scheme". *Lecture Notes-Monograph Series* 30:245–268. ISSN: 07492170. doi:10.2307/4355949.

Pitman, Jim, and Marc Yor. 1995. *The Two-parameter Poisson-Dirichlet Distribution Derived from a Stable Subordinator*. Tech. rep. University of California.

Pitt, Michael K., and Neil Shephard. 2001. "Auxiliary Variable Based Particle Filters". In *Sequential Monte Carlo Methods in Practice*, ed. by Arnaud Doucet, Nando de Freitas, and Neil Gordon, 273–293. Springer New York. ISBN: 978-1-4757-3437-9. doi:10.1007/978-1-4757-3437-9_13.

Pitt, Michael K., Ralph dos Santos Silva, Paolo Giordani, and Robert Kohn. 2012. "On some properties of Markov chain Monte Carlo simulation methods based on the particle filter". *Journal of Econometrics* 171, no. 2: 134–151. ISSN: 0304-4076. doi:10.1016/j.jeconom.2012.06.004.

Polatkan, Gungor, Mingyuan Zhou, Lawrence Carin, David Blei, and Ingrid Daubechies. 2015. "A Bayesian Nonparametric Approach to Image Super-Resolution". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, no. 2: 346–358. ISSN: 2160-9292. doi:10.1109/tpami.2014.2321404.

Rai, Piyush, and Hal Daumé III. 2008. "The Infinite Hierarchical Factor Regression Model". In *Advances in Neural Information Processing Systems*, 1–8.

Ranganath, Rajesh, Dustin Tran, Jaan Altosaar, and David Blei. 2016. "Operator variational inference". In *Advances in Neural Information Processing Systems*, 496–504.

Rasmussen, C.E., and K.I. Williams. 2006. *Gaussian Processes for Machine Learning*. MIT Press. ISBN: 026218253X.

Robert, Christian P., and George Casella. 1999. *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer New York. ISBN: 9781475730715. doi:10.1007/978-1-4757-3071-5.

Roitman, Judith. 1990. *Introduction to modern set theory*. Vol. 8. John Wiley & Sons.

Roy, Daniel M. 2014. "The Continuum-of-Urns Scheme, Generalized Beta and Indian Buffet Processes, and Hierarchies Thereof". arXiv: 1501.0020.

Roy, Daniel M., Charles Kemp, Vikash K. Mansinghka, and Joshua B. Tenenbaum. 2006. "Learning annotated hierarchies from relational data". In *Advances in Neural Information Processing Systems*.

Roy, Daniel M., and Yee Whye Teh. 2008. "The Mondrian Process". In *Advances in Neural Information Processing Systems*.

Rubin, Donald B. 1987. "A Noniterative Sampling/Importance Resampling Alternative to the Data Augmentation Algorithm for Creating a Few Imputations When Fractions of Missing Information Are Modest: The SIR Algorithm". *Journal of the American Statistical Association* 82, no. 398: 543. ISSN: 0162-1459. doi:10.2307/2289460.

Ruiz, Francisco J.R., Michalis K. Titsias, and David M. Blei. 2016. "Overdispersed black-box variational inference". In *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, 647–656. AUAI Press.

Ruiz, Francisco J.R., Isabel Valera, Carlos Blanco, and Fernando Perez-cruz. 2014. "Bayesian Nonparametric Comorbidity Analysis of Psychiatric Disorders". *Journal of Machine Learning Research* 15:1215–1247.

Seiler, Christof, Simon Rubinstein-Salzedo, and Susan Holmes. 2014. "Positive Curvature and Hamiltonian Monte Carlo". In *Advances in Neural Information Processing Systems*, 586–594.

Sethuraman, Jayaram. 1994. "A constructive definition of Dirichlet priors". *Statistica sinica*: 639–650.

Shahbaba, Babak, and Radford Neal. 2009. "Nonlinear Models Using Dirichlet Process Mixtures". *Journal of Machine Learning Research* 10:1829–1850.

Sherlock, Chris, Andrew Golightly, and Daniel A. Henderson. 2016. "Adaptive, Delayed-Acceptance MCMC for Targets With Expensive Likelihoods". *Journal of Computational and Graphical Statistics* 26, no. 2: 434–444. ISSN: 1537-2715. doi:10.1080/10618600.2016.1231064.

Sherlock, Chris, Alexandre H. Thiery, Gareth O. Roberts, and Jeffrey S. Rosenthal. 2015. "On the efficiency of pseudo-marginal random walk Metropolis algorithms". *The Annals of Statistics* 43, no. 1: 238–275. ISSN: 0090-5364. doi:10.1214/14-AOS1278. arXiv: 1309.7209. http://projecteuclid.org/euclid.aos/1418135621.

Shyr, Alex, Trevor Darrell, Michael Jordan, and Raquel Urtasun. 2011. "Supervised hierarchical Pitman-Yor process for natural scene segmentation". *CVPR 2011*. doi:10.1109/cvpr.2011.5995647.

Smith, A.F.M., and A.E. Gelfand. 1992. "Bayesian Statistics without Tears: A Sampling-Resampling Perspective". *The American Statistician* 46, no. 2: 84. ISSN: 0003-1305. doi:10.2307/2684170.

Sohl-Dickstein, Jascha, Mayur Mudigonda, and Michael R. DeWeese. 2014. "Hamiltonian Monte Carlo without detailed balance". In *International Conference on Machine Learning*, I–719. JMLR. org.

Solla, Sara, and Ole Winther. 1998. "Optimal Perceptron Learning: an On-line Bayesian Approach". Ed. by DavidEditor Saad. *On-Line Learning in Neural Networks*: 379–398. doi:10.1017/cbo9780511569920.018.

Stepleton, Thomas, Zoubin Ghahramani, Geoffrey Gordon, and Tai-Sing Lee. 2009. "The block diagonal infinite hidden Markov model". In *Artificial Intelligence and Statistics*, 552–559.

Stimberg, Florian, Andreas Ruttor, and Manfred Opper. 2014. "Poisson process jumping between an unknown number of rates: application to neural spike data". In *Advances in Neural Information Processing Systems*, 730–738.

Stoehr, Julien, Alan Benson, and Nial Friel. 2017. "Noisy Hamiltonian Monte Carlo for doubly-intractable distributions". arXiv: 1706.10096.

Stolfo, J., Wei Fan, Wenke Lee, Andreas Prodromidis, and Philip K. Chan. 1999. "Cost-based modeling and evaluation for data mining with application to fraud and intrusion detection". In *Results from the JAM Project by Salvatore*, 1–15.

Storvik, G. 2002. "Particle filters for state-space models with the presence of unknown static parameters". *IEEE Transactions on Signal Processing* 50: 281–289. ISSN: 1053-587X. doi:10.1109/78.978383.

Strathmann, Heiko, Dino Sejdinovic, Samuel Livingstone, Zoltan Szabo, and Arthur Gretton. 2015. "Gradient-free Hamiltonian Monte Carlo with efficient kernel exponential families". In *Advances in Neural Information Processing Systems*, 955–963.

Sudderth, Erik B., and Michael I. Jordan. 2008. "Shared segmentation of natural scenes using dependent Pitman-Yor processes". In *Advances in neural information processing systems*, 1585–1592.

Sudderth, Erik B., Antonio Torralba, William T. Freeman, and Alan S. Willsky. 2007. "Describing Visual Scenes Using Transformed Objects and Parts". *Int J Comput Vis* 77, no. 1-3: 291–330. ISSN: 1573-1405. doi:10.1007/s11263-007-0069-5.

Sudderth, Erik Blaine. 2006. "Graphical models for visual object recognition and tracking". PhD thesis, Massachusetts Institute of Technology.

Sunnåker, Mikael, Alberto Giovanni Busetto, Elina Numminen, Jukka Corander, Matthieu Foll, and Christophe Dessimoz. 2013. "Approximate Bayesian Computation". Ed. by Shoshana Wodak. *PLoS Computational Biology* 9, no. 1: e1002803. ISSN: 1553-7358. doi:10.1371/journal.pcbi.1002803.

Tak, Hyungsuk, Xiao-Li Meng, and David A. van Dyk. 2017. "A Repelling-Attracting Metropolis Algorithm for Multimodality". *Journal of Computational and Graphical Statistics*. ISSN: 1537-2715. doi:10.1080/10618600.2017.1415911.

Tank, Alex, Nicholas Foti, and Emily Fox. 2015. "Streaming variational inference for Bayesian nonparametric mixture models". In *Artificial Intelligence and Statistics*, 968–976.

Teh, Yee W., and Dilan Görür. 2009. "Indian buffet processes with power-law behavior". In *Advances in neural information processing systems*, 1838–1846.

Teh, Yee Whye. 2006. "A hierarchical Bayesian language model based on Pitman-Yor processes". In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, 985–992. Association for Computational Linguistics. doi:10.3115/1220175.1220299.

Teh, Yee Whye, Dilan Görür, and Zoubin Ghahramani. 2007. "Stick-breaking Construction for the Indian Buffet Process". In *International Conference on Artificial Intelligence and Statistics*.

Teh, Yee Whye, Michael I. Jordan, Matthew J. Beal, and David M. Blei. 2006. "Hierarchical Dirichlet Processes". *Journal of the American Statistical Association* 101, no. 476: 1566–1581. ISSN: 1537-274X. doi:10.1198/016214506000000302.

Thibaux, Romain Jean. 2008. "Nonparametric Bayesian Models for Machine Learning". PhD thesis, University of California at Berkeley.

Thibaux, Romain, and Michael I. Jordan. 2007. "Hierarchical Beta Processes and the Indian Buffet Process". In *International Conference on Artificial Intelligence and Statistics*.

Titsias, Michalis K. 2008. "The infinite gamma-Poisson feature model". In *Advances in Neural Information Processing Systems*, 1513–1520.

Tripuraneni, Nilesh, Mark Rowland, Zoubin Ghahramani, and Richard Turner. 2017. "Magnetic Hamiltonian Monte Carlo". In *International Conference on Machine Learning*, 3453–3461.

Vakilzadeh, Majid K., James L. Beck, and Thomas Abrahamsson. 2018. "Using approximate Bayesian computation by Subset Simulation for efficient posterior assessment of dynamic state-space model classes". *SIAM Journal on Scientific Computing* 40: B168–B195.

Van Gael, Jurgen, Yunus Saatci, Yee Whye Teh, and Zoubin Ghahramani. 2008. "Beam sampling for the infinite hidden Markov model". In *Proceedings of the 25th international conference on Machine learning - ICML '08*. Association for Computing Machinery (ACM). ISBN: 9781605582054. doi:10.1145/1390156.1390293.

Vergé, Christelle, Cyrille Dubarry, Pierre Del Moral, and Eric Moulines. 2013. "On parallel implementation of sequential Monte Carlo methods: the island particle model". *Statistics and Computing* 25, no. 2: 243–260. ISSN: 1573-1375. doi:10.1007/s11222-013-9429-x.

Wainwright, Martin J., and Michael I. Jordan. 2008. *Graphical Models, Exponential Families, and Variational Inference*. 1:1–305. Foundations and Trends® in Machine Learning, 1–2. Now Publishers, Inc. doi:10.1561/2200000001.

Walker, Stephen G. 2007. "Sampling the Dirichlet Mixture Model with Slices". *Communications in Statistics: Simulation and Computation*, no. 16.

Walker, Stephen G., Paul Damien, PuruShottam W. Laud, and Adrian F.M. Smith. 1999. "Bayesian Nonparametric Inference for Random Distributions and Related Functions". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61, no. 3: 485–527. ISSN: 1467-9868. doi:10.1111/1467-9868.00190.

Wang, Chong, and David M. Blei. 2009. "Variational inference for the nested Chinese restaurant process". In *Advances in Neural Information Processing Systems*, 1990–1998.

Wang, Liangliang, Alexandre Bouchard-Côté, and Arnaud Doucet. 2015. "Bayesian Phylogenetic Inference Using a Combinatorial Sequential Monte Carlo Method". *Journal of the American Statistical Association* 110, no. 512: 1362–1374. ISSN: 1537-274X. doi:10.1080/01621459.2015.1054487.

Wang, Shenlong, Alex Schwing, and Raquel Urtasun. 2014. "Efficient inference of continuous markov random fields with polynomial potentials". In *Advances in neural information processing systems*, 936–944.

Wang, Yingjian, and Lawrence Carin. 2012. "Lévy Measure Decompositions for the Beta and Gamma Processes". In *International Conference on Machine Learning 2012*.

Wang, Yuchung J., and George Y. Wong. 1987. "Stochastic Blockmodels for Directed Graphs". *Journal of the American Statistical Association* 82: 8–19.

Wasserman, Stanley, and Katherine Faust. 1994. *Social network analysis: Methods and applications*. Vol. 8. Cambridge university press.

West. 1992. *Hyperparameter estimation in Dirichlet process mixture models*. Tech. rep. Duke University.

West, Mike. 1993. "Mixture models, Monte Carlo, Bayesian updating, and dynamic models". *Computing Science and Statistics*: 325–325.

Whiteley, Nick, and Anthony Lee. 2014. "Twisted particle filters". *The Annals of Statistics* 42, no. 1: 115–141. ISSN: 0090-5364. doi:10.1214/13-aos1167.

Whiteley, Nick, Anthony Lee, and Kari Heine. 2016. "On the role of interaction in sequential Monte Carlo algorithms". *Bernoulli* 22, no. 1: 494–529. ISSN: 1350-7265. doi:10.3150/14-bej666.

Williamson, Sinead A., Steven N. Maceachern, and Eric P. Xing. 2013. "Restricting exchangeable nonparametric distributions". In *Proceedings of Neural Information Processing Systems*, 1–9.

Williamson, Sinead, Chong Wang, Katherine A. Heller, and David M. Blei. 2010. "The IBP Compound Dirichlet Process and its Application to Focused Topic Modeling". In *Proceedings of the 27th International Conference on Machine Learning*.

Wilson, Andrew Gordon, and Ryan Prescott Adams. 2013. "Gaussian Process Kernels for Pattern Discovery and Extrapolation". In *International Conference on Machine Learning*, vol. 28.

Wilson, Andrew Gordon, and Zoubin Ghahramani. 2009. "Generalised Wishart Processes". In *Uncertainty in artificial intelligence*.

Wood, Frank, Cédric Archambeau, Jan Gasthaus, Lancelot James, and Yee Whye Teh. 2009. "A stochastic memoizer for sequence data". In *Proceedings of the 26th Annual International Conference on Machine Learning*, 1129–1136. ACM. doi:10.1145/1553374.1553518.

Wood, Frank, Jan Gasthaus, Cédric Archambeau, Lancelot James, and Yee Whye Teh. 2011. "The Sequence Memoizer". *Communications of the ACM* 54: 91–98. doi:10.1145/1897816.

Wood, Frank, Thomas L. Griffiths, and Zoubin Ghahramani. 2006. "A Non-Parametric Bayesian Method for Inferring Hidden Causes". In *Uncertainty in Artificial Intelligence*.

Wraith, Darren, Martin Kilbinger, Karim Benabed, Olivier Cappé, Jean-François Cardoso, Gersende Fort, Simon Prunet, and Christian P. Robert. 2009. "Estimation of cosmological parameters using adaptive importance sampling". *Phys. Rev. D* 80, no. 2. ISSN: 1550-2368. doi:10.1103/physrevd.80.023507.

Xu, Zhao, Volker Tresp, Kai Yu, and Hans-Peter Kriegel. 2006. "Infinite Hidden Relational Models". In *Uncertainty in Artificial Intelligence*.

Yang, Ruoyong, and James O. Berger. 1998. *A catalog of noninformative priors*. Institute of Statistics / Decision Sciences, Duke University.

Yildiz, Izzet B., Katharina von Kriegstein, and Stefan J. Kiebel. 2013. "From Birdsong to Human Speech Recognition: Bayesian Inference on a Hierarchy of Nonlinear Dynamical Systems". Ed. by Viktor K.Editor Jirsa. *PLoS Computational Biology* 9, no. 9: e1003219. ISSN: 1553-7358. doi:10.1371/journal.pcbi.1003219.

Yuan, Xin, Ricardo Henao, Ephraim Tsalik, Raymond Langley, and Lawrence Carin. 2015. "Non-Gaussian discriminative factor models via the max-margin rank-likelihood". In *International Conference on Machine Learning*, 1254–1263. arXiv: 1504.07468.

Zanten, Harry Van. 2007. *An Introduction to Stochastic Processes in Continuous Time*. http://www.few.vu.nl/~RWJ.Meester/onderwijs/stochastic_processes/sp_new.pdf.

Zhang, Yizhe, Xiangyu Wang, Changyou Chen, Ricardo Henao, Kai Fan, and Lawrence Carin. 2016. "Towards unifying Hamiltonian Monte Carlo and slice sampling". In *Advances in Neural Information Processing Systems*, 1741–1749.

Zhang, Zhengwu, Debdeep Pati, and Anuj Srivastava. 2015. "Bayesian clustering of shapes of curves". *Journal of Statistical Planning and Inference* 166:171–186. arXiv: 1504.00377.

Zhao, Huasha, Biye Jiang, John F. Canny, and Bobby Jaros. 2015. "SAME but different: Fast and high quality gibbs parameter estimation". In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1495–1502. ACM. arXiv: 1409.5402.

Zhou, Mingyuan. 2014. "Beta-Negative Binomial Process and Exchangeable Random Partitions for Mixed-Membership Modeling". In *Advances in Neural Information Processing Systems*, 3455–3463.

Zhou, Mingyuan, Lauren A. Hannah, David B. Dunson, and Lawrence Carin. 2012. "Beta-negative binomial process and Poisson factor analysis". *Journal of Machine Learning Research*.

Zhu, Jun, Ning Chen, and Eric P. Xing. 2014. "Bayesian Inference with Posterior Regularization and Applications to Infinite Latent SVMs". *Journal of Machine Learning Research* 15:1799–1847.

Zipf, George K. 1949. *Human behaviour and the principle of least-effort.* Addison-Wesley.

BIOGRAPHICAL SKETCH

Justin Sahs earned his B.S. in Computer Science from The University of Texas at Dallas (UTD) in Fall 2011. He began working as a Research Assistant in the Big Data Analytics and Management Lab under Dr. Latifur Khan the following Spring, and officially entered the Ph.D. program in Fall 2012. His research interests include Artificial Intelligence and Machine Learning, with a special focus on nonparametric Bayesian probabilistic models and techniques.

CURRICULUM VITAE

Justin C. Sahs

Contact Information:

Department of Computer Science
The University of Texas at Dallas
800 W. Campbell Rd.
Richardson, TX 75080-3021, U.S.A.

Voice: (214) 505-2488
Email: justin.sahs@utdallas.edu

Educational History:

B.S., Computer Science, University of Texas at Dallas, 2011
M.S., Computer Science, University of Texas at Dallas, 2017
Ph.D. candidate, Computer Science, University of Texas at Dallas, 2018

Publications:

Justin Sahs and Latifur Khan. 2012. "A machine learning approach to android malware detection". In *Intelligence and security informatics conference (EISIC), 2012 European*, 141–147. IEEE

Swarup Chandra, **Justin Sahs**, Latifur Khan, Bhavani Thuraisingham, and Charu Aggarwal. 2014. "Stream mining using statistical relational learning". In *Data Mining (ICDM), 2014 IEEE International Conference on*, 743–748. IEEE. doi:10.1109/icdm.2014.144

David Sounthiraraj, **Justin Sahs**, Garret Greenwood, Zhiqiang Lin, and Latifur Khan. 2014. "SMV-Hunter: Large scale, automated detection of SSL/TLS man-in-the-middle vulnerabilities in Android apps". In *Proceedings of the 21st Annual Network and Distributed System Security Symposium (NDSS'14)*. https://www.ndss-symposium.org/ndss2014/smv-hunter-large-scale-automated-detection-ssltls-man-middle-vulnerabilities-android-apps

Justin Sahs. 2016. "Bayesian nonparametric relational learning with the Broken Tree Process". In *Intelligence and Security Informatics (ISI), 2016 IEEE Conference on*, 49–54. IEEE. doi:10.1109/ISI.2016.7745442

Justin Sahs and Latifur Khan. 2017. "Online Classification of Nonstationary Streaming Data with Dynamic Pitman-Yor Diffusion Trees". In *Tools with Artificial Intelligence (ICTAI), 2017 IEEE 29th International Conference on*, 477–484. IEEE

Employment History:

Research Assistant, The University of Texas at Dallas, January 2012 – December 2018
Student Worker, The University of Texas at Dallas, October 2011 – December 2011
Software Tools developer, QuickOffice, Inc. 2007 –2008