Introduction to Bayesian Nonparametric Statistics

Vanda Inácio

University of Edinburgh

December 2023

Outline

↪ Dependent Dirichlet process mixtures.

↪ Polya trees, mixtures of Polya trees, and dependent mixtures of Polya trees.

Dependent Dirichlet process mixtures

Density regression

- ↪ There is nowadays a vast literature on Bayesian regression that employs nonparametric priors.
- ↪ Most of this work rests on the traditional formulation

$$y_i = h(x_i) + \varepsilon_i,$$

where $h(x_i)$ is the regression function and ε_i arise, typically in an iid fashion, from some error distribution.

- ↪ The two main trends involve semiparametric modelling either by specifying
  1. A flexible mean function (e.g., penalised splines, gaussian processes, trees) and a parametric error distribution.
  2. A nonparametric error distribution (e.g., modelled through a DPM) and specifying the regression function parametrically (e.g., a linear regression term $h(x_i) = x_i'\beta$).

Dependent Dirichlet process mixtures

Density regression

- To achieve flexible density regression, mixture models are attractive tools.
- As we have seen, mixture models are very popularly used for density estimation due to their ability to approximate a large class of densities and their attractive balance between smoothness and flexibility in modelling local features.
- When additional covariate information is present, mixture models can be extended for density regression in one of two ways:
  1. Modelling the joint density of the response and covariates with a mixture model (known as the *joint approach*).
  2. Directly modelling the conditional density by allowing the mixing distribution, namely the mixture weights and atoms, to depend on the covariates (known as the *conditional approach*).
- Conditional models are often referred to as dependent mixture models in statistics and are also known as mixtures of experts in machine learning.

Dependent Dirichlet process mixtures

Density regression

- A simple extension of mixture models for density estimation to covariate-dependent density estimation augments the observations to include the covariates.

- The approach to density regression, within the BNP framework, dates back to Muller, Erkanli, and West (1996).

- These authors have considered a DPM model for the joint density of the response y and the vector of covariates $\mathbf{x}$

$$f(y, \mathbf{x}) = \int k(y, \mathbf{x} \mid \boldsymbol{\theta}) \mathrm{d}G(\boldsymbol{\theta}), \quad G \sim \mathrm{DP}(\alpha, G_0).$$

- For instance, for the case of a single covariate, and if both y and x are continuous and real-valued, one possibility for the kernel would be a bivariate normal distribution, i.e., $k(y, x \mid \boldsymbol{\theta}) = \mathrm{N}_2(y, x \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad \boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

Dependent Dirichlet process mixtures

Density regression

↪ Using Sethuraman's representation, the conditional density can be written as

$$f(y \mid \mathbf{x}) = \frac{f(y, \mathbf{x})}{f(\mathbf{x})} = \frac{\sum_{l=1}^{\infty} \omega_l k(y, \mathbf{x} \mid \boldsymbol{\theta}_l)}{\sum_{l'=1}^{\infty} \omega'_l k(\mathbf{x} \mid \boldsymbol{\theta}^x_{l'})} = \sum_{l=1}^{\infty} \omega^*_l(\mathbf{x}) k(y \mid \mathbf{x}, \boldsymbol{\theta}^{y|x}_l),$$

where $\omega^*_l(\mathbf{x}) = \frac{\omega_l k(\mathbf{x}|\boldsymbol{\theta}^x_l)}{\sum_{l'=1}^{\infty} \omega'_l k(\mathbf{x}|\boldsymbol{\theta}^x_{l'})}$.

↪ The mean regression function implied by the joint model is given by

$$E(Y \mid \mathbf{x}) = \sum_{l=1}^{\infty} \omega^*_l(\mathbf{x}) \mathbb{E}(Y \mid \mathbf{x}, \theta_l)$$

where $E(Y \mid \mathbf{x}, \theta_l)$ is the conditional mean of the jth component.

↪ Note that in this case, the conditional mean of each component, $E(Y \mid \mathbf{x}, \theta_l)$, is a linear regression function and the fact that the weights are covariate dependent, leading to a locally weighted mixture of linear regressions, is key to allow estimation of nonlinear regression relationships and general density shapes for the conditional response distribution.

Dependent Dirichlet process mixtures

Density regression

- ↪ We note that this approach is more meaningful if the covariates can be considered as random variables, and can be problematic for fixed covariates, for instance, binary treatment allocation variables in clinical trial studies.

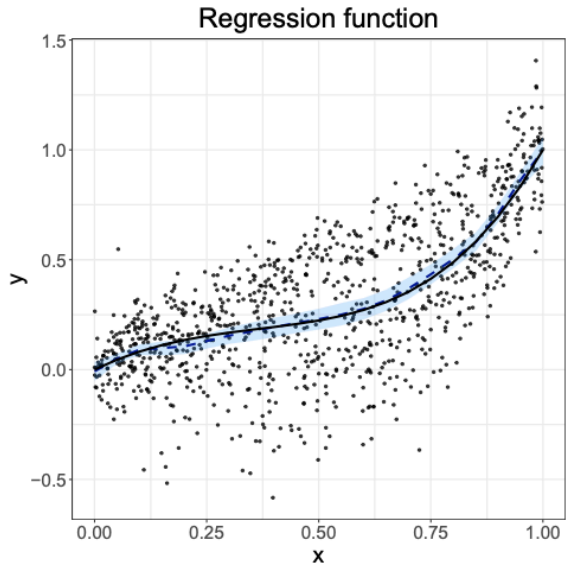
- ↪ Quoting Muller (2018): “The approach only works easily when x and y are both continuous.”

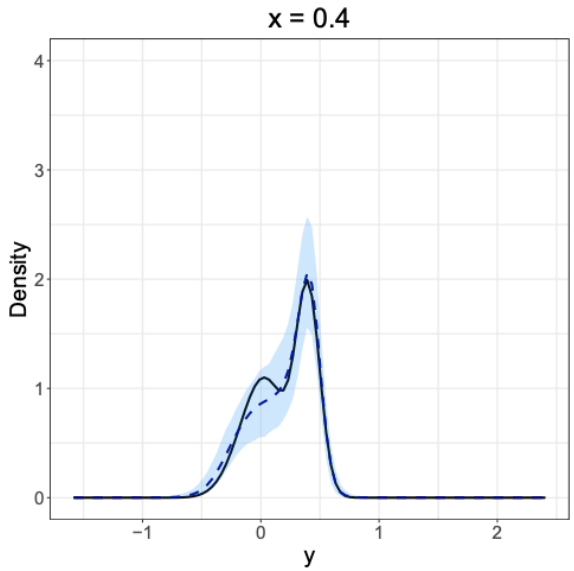
- ↪ When our interest in only in the conditional density, this approach unnecessarily requires the modelling of the marginal of $\mathbf{x}$.

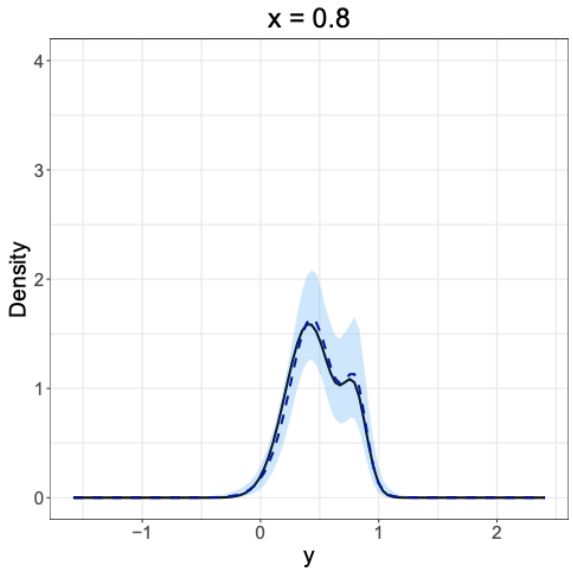
Dependent Dirichlet process mixtures

Density regression

$$\hookrightarrow \quad y_i \mid x_i \overset{\text{ind.}}{\sim} \exp(-2x_i)\mathrm{N}(x_i, 0.01) + \{1 - \exp(-2x_i)\}\mathrm{N}(x_i^4, 0.04),$$
$$x_i \overset{\text{iid}}{\sim} \mathrm{U}(0,1), \quad i = 1, \ldots, 1000.$$







Dependent Dirichlet process mixtures

Density regression

↪ Another possibility is to define a flexible model for the collection of conditional densities $\{f(y \mid \mathbf{x}) : \mathbf{x} \in \mathcal{X} \subset R^p\}$ by allowing the mixing measure to depend on $\mathbf{x}$

$$f(y \mid \mathbf{x}) = \int k(y \mid \theta) \mathrm{d}G_{\mathbf{x}}(\theta).$$

↪ The question is then which prior to assign to the collection of mixing measures $\{G_x : x \in \mathcal{X}\}$.

↪ Two possible choices are:

1. All $G_{\mathbf{x}}$ are assumed to be identical, e.g., $G_{\mathbf{x}} \equiv G \sim \mathrm{DP}(\alpha, G_0)$ for all $\mathbf{x} \in \mathcal{X}$.
2. All $G_{\mathbf{x}}$ are assumed to be distinct and independent, e.g., $G_{\mathbf{x}} \sim \mathrm{DP}(\alpha, P_0)$, independently for each $\mathbf{x}$.

↪ We seek a compromise between these two extreme choices as (1) is too restrictive and corresponds to maximum borrowing of strength across covariate values, and (2) is wasteful and corresponds to no borrowing of strength.

Dependent Dirichlet process mixtures

Density regression

↪ Indeed, Chung and Dunson (2011) lists some desirable properties of a prior for the collection of dependent mixture measures, which include:

1. Increasing dependence between $G_{\mathbf{x}}$ and $G_{\mathbf{x}^*}$ as the distance between $\mathbf{x}$ and $\mathbf{x}^*$ decreases.

2. Simple and interpretable expressions for the expectation and variance of each $G_{\mathbf{x}}$ as well as the correlation between $G_{\mathbf{x}}$ and $G_{\mathbf{x}^*}$.

3. Efficient posterior simulation in a broad variety of applications.

Dependent Dirichlet process mixtures

Density regression

- One possibility for a prior on $\{G_{\mathbf{x}} : \mathbf{x} \in \mathcal{X}\}$ is the dependent Dirichlet process of MacEachearn (1999, 2001), which builds upon the stick-breaking representation. In full generality, it is specified as

$$G_{\mathbf{x}}(\cdot) = \sum_{l=1}^{\infty} \omega_l(\mathbf{x}) \delta_{\theta_l(\mathbf{x})}(\cdot).$$

- Here
  - $\{\theta_l(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}$, for $l \geq 1$, are independent realisations from a centring stochastic process $G_{0\mathcal{X}}$ defined on $\mathcal{X}$.

    myitem Stick-breaking weights $\omega_1(\mathbf{x}) = v_1(\mathbf{x})$, $\omega_l(\mathbf{x}) = v_l(\mathbf{x}) \prod_{m<l}(1 - v_m(\mathbf{x}))$, for $l \geq 2$, where $\{v_l(\mathbf{x}) : \mathbf{x} \in \mathcal{X}\}$ are independent realisations from a stochastic process on $\mathcal{X}$ with marginals $v_l(\mathbf{x}) \sim \text{Beta}(1, \alpha(\mathbf{x}))$ (or with common $\alpha(\mathbf{x}) \equiv \alpha$).

Dependent Dirichlet process mixtures

Density regression

- ↪ Applications of the fully general dependent Dirichlet process mixture model are hard to find.
- ↪ Formulations with flexible weights and with constant atoms $\theta_l(\mathbf{x}) \equiv \theta_l$) or atoms relying on a linear formulation (e.g., $\theta_l(\mathbf{x}) = \mathbf{x}^T\boldsymbol{\beta}_l$) are very flexible but computations can be burdensome.
- ↪ The approach of considering covariate independent weights ($\omega_l(\mathbf{x}) \equiv \omega_l$), known as the single-weights dependent Dirichlet process, is very popular due to its computational convenience (as all samplers developed for DPMs can be used), but can have limited flexibility in practice.
- ↪ However, quoting MacEachern (2000) in his seminal paper

  "*They (single-weights DDPs) thus provide a general framework that covers a vast territory*"

- ↪ Here we will focus on single-weights dependent Dirichlet processes. A review of methods and applications of approaches allowing the weights to depend on covariates can be found in Wade, Inacio, and Petrone (2023)

  `https://arxiv.org/abs/2307.16298`

Dependent Dirichlet process mixtures

Density regression

↪ In the most popular version of the single weights dependent Dirichlet process mixtures of normals model (De Iorio et al., 2009), the conditional density takes the form

$$
\begin{aligned}
f(y \mid \mathbf{x}) &= \int \phi(y; \mu, \sigma^2) \mathrm{d}G_{\mathbf{x}}(\mu, \sigma^2) \\
&= \sum_{l=1}^{\infty} \omega_l \phi(y; \mu_l(\mathbf{x}), \sigma_l^2), \quad \mu_l(\mathbf{x}) = \mathbf{x}\boldsymbol{\beta}_l, \\
G_{\mathbf{x}} &= \sum_{l=1}^{\infty} \omega_l \delta_{(\mathbf{x}\beta_l, \sigma_l^2)}, \quad (\boldsymbol{\beta}_l, \sigma_l^2) \overset{\text{iid}}{\sim} G_0.
\end{aligned}
$$

where $\phi(y; \mu, \sigma^2)$ denotes the density of the normal distribution with mean μ and variance σ^2.

↪ One can imagine a non-homogeneous population, where a subject's response behaviour may be described by one of the models in the infinite collection of linear regression models, and allocation to a specific component is independent of x.

Dependent Dirichlet process mixtures

Density regression

- Note that this model simply corresponds a DP mixture of normal linear regression models, that is,

$$f(y \mid \mathbf{x}) = \int \phi(y; \mathbf{x}\boldsymbol{\beta}, \sigma^2) \mathrm{d}G(\beta, \sigma^2), \quad G \sim \mathrm{DP}(\alpha, G_0).$$

- The weights match those from a 'standard' stick breaking representation.

- Although the variance of each component does not depend on the covariates, the overall variance of the mixture model

$$\mathrm{var}(y \mid \mathbf{x}) = \sum_{l=1}^{\infty} \omega_l \sigma_l^2 + \sum_{l=1}^{\infty} \omega_l \left\{ \mu_l^2(\mathbf{x}) - \left(\sum_{l=1}^{\infty} \omega_l \mu_l(\mathbf{x}) \right)^2 \right\},$$

  does depend on $\mathbf{x}$

Dependent Dirichlet process mixtures

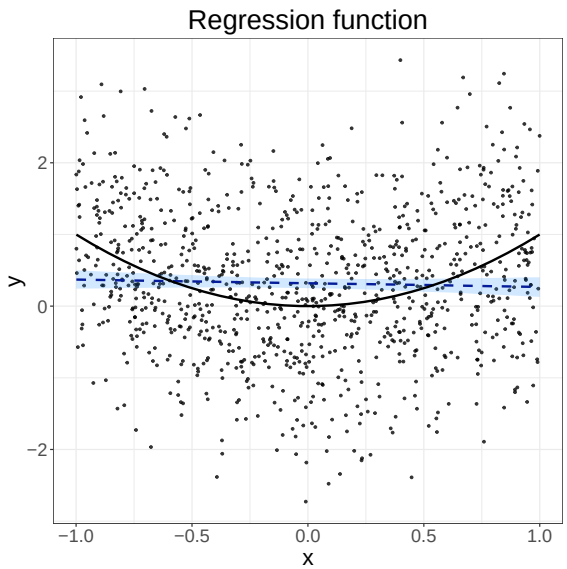
Density regression

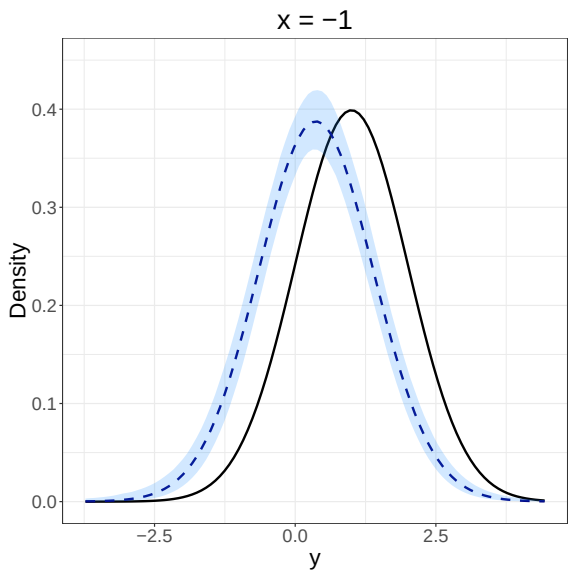
- ↪ This model incorporates an infinite number of normal linear regressions and this might seem very flexible at a first glance.
- ↪ However, and for instance, the mean regression structure is linear; we have a weighted combination of parametric regression functions, but without the local adjustment afforded by covariate-dependent weights.
- ↪ That is, the single-weights DDP mixture (of normals) model is flexible in terms of non-Gaussian response, but not in terms of regression relationships.
- ↪ For increased model flexibility, in terms of the implied mean regression structure, higher-dimensional transformations of the continuous covariates are needed. One possibility is the use of (cubic) B-splines basis.

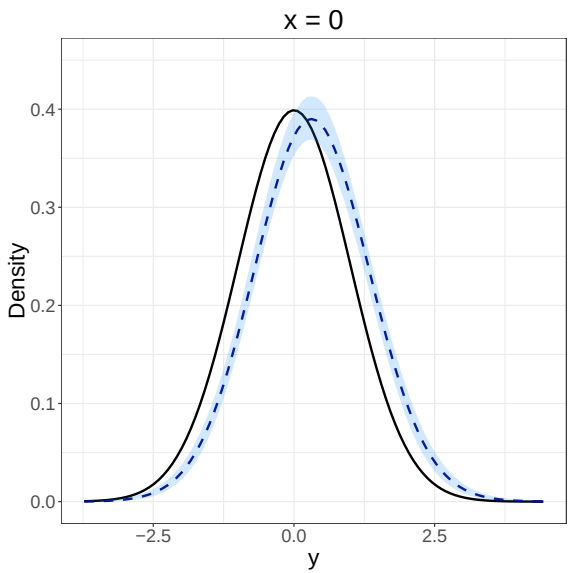
Dependent Dirichlet process mixtures

Density regression

$$\hookrightarrow \quad y_i \mid x_i \overset{\text{ind.}}{\sim} \mathrm{N}\left(x_i^2, 1^2\right), \quad x_i \overset{\text{iid}}{\sim} \mathrm{U}(-1, 1), \quad i = 1, \ldots, 1000.$$







Dependent Dirichlet process mixtures

Density regression

↪ To overcome the lack of flexibility, but retaining the computational tractability, an easy fix is to model the mean of each component in a flexible manner

$$\mu_l(\mathbf{x}) = \beta_{l0} + f_{l1}(x_1) + \cdots + f_{lp}(x_p),$$

where $f_{lh}(\cdot)$ are smooth and unknown functions, $l \geq 1$ and $h = 1, \ldots, p$.

↪ Each smooth function $f_{lh}(\cdot)$ can be approximated by a linear combination of cubic B-splines basis functions defined over a sequence of knots $\xi_{h0} < \xi_{h1} < \cdots < \xi_{hK_h} < \xi_{h,K_h+1}$

$$\mu_l(\mathbf{x}) = \beta_{l0} + \underbrace{\mathbf{B}'_{\xi_1}(x_1)\boldsymbol{\beta}_{l1}}_{f_{l1}(x_1)} + \cdots + \underbrace{\mathbf{B}'_{\xi_p}(x_p)\boldsymbol{\beta}_{lp}}_{f_{lp}(x_p)} = \mathbf{z}'\boldsymbol{\beta}_l.$$

↪ To assist in the selection of the number of internal knots the LPML and/or WAIC can be used.

Dependent Dirichlet process mixtures

Density regression

- ↪ A conditionally conjugate centring distribution $G_0(\boldsymbol{\beta}, \sigma^2)$, which greatly simplifies computations, is specified

$$(\boldsymbol{\beta}_l, \sigma_l^2) \overset{\text{iid}}{\sim} N_Q(\boldsymbol{\beta}_l \mid \mathbf{m}, \mathbf{S})\Gamma(\sigma_l^{-2} \mid a, b), \quad l \geq 1,$$

  with conjugate hyperpriors $\mathbf{m} \sim \mathrm{N}_Q(\mathbf{m}_0, \mathbf{S}_0)$ and $\mathbf{S}^{-1} \sim \mathrm{W}_Q(\nu, (\nu\Psi)^{-1})$, where Q denotes the dimension of the vector $\mathbf{z}$.

- ↪ As in the density estimation setup, posterior inference can be conducted using a blocked Gibbs sampler and, as such, the final model for the conditional density is

$$f(y \mid \mathbf{x}) = \sum_{l=1}^{L} \omega_l \phi(y \mid \mathbf{z}'\boldsymbol{\beta}_l, \sigma_l^2).$$

- ↪ This model characterises the conditional density of the responses using a mixture of normal distributions with the component means varying differentially and nonlinearly with covariates.

Dependent Dirichlet process mixtures

Density regression

- Upon the introduction of latent variables, $z_1, \ldots, z_n$, that identify the mixture component where each observation belongs to, the full conditional distributions of all model parameters are available in closed form.

- The latent variables are updated through

$$\Pr(z_i = l \mid \text{else}) = \frac{\omega_l \phi(y_{ii} \mid \mathbf{z}_i' \boldsymbol{\beta}_l, \sigma_l^2)}{\sum_{k=1}^{L} \omega_k \phi(y_i \mid \mathbf{z}_i' \boldsymbol{\beta}_k, \sigma_k^2)}, \qquad \omega_l = v_l \prod_{r<l} (1 - v_r), \qquad l = 1, \ldots, L.$$

- Update the stick-breaking weights from their conjugate beta posterior distribution

$$v_l \mid \text{else} \sim \text{Beta}\left(n_l + 1, \alpha + \sum_{r=l+1}^{L} n_r \right), \qquad l = 1, \ldots, L-1,$$

with $n_l = \sum_{i=1}^{n} I(z_i = l)$ being the number of observations from component l.

Dependent Dirichlet process mixtures

Density regression

↪ Update the component specific parameters, $\boldsymbol{\beta}_l$ and σ_l^2 ($l = 1, \ldots, L$), as in a standard Bayesian normal linear regression model, using observations from component l, i.e.,

$$\boldsymbol{\beta}_l \mid \text{else} \sim \mathrm{N}_Q \left(\mathbf{V}_l \left(\mathbf{S}^{-1}\mathbf{m} + \sigma_l^{-2} \sum_{\{i:z_i=l\}} \mathbf{z}_i y_i \right), \mathbf{V}_l \right),$$

$$\mathbf{V}_l = \left(\mathbf{S}^{-1} + \sigma_l^{-2} \sum_{\{i:z_i=l\}} \mathbf{z}_i \mathbf{z}_i' \right)^{-1},$$

$$\sigma_l^2 \mid \text{else} \sim \mathrm{IG} \left(a + \frac{n_l}{2}, b + \frac{1}{2} \sum_{\{i:z_i=l\}} \left(y_i - \mathbf{z}_i' \boldsymbol{\beta}_l \right)^2 \right).$$

Dependent Dirichlet process mixtures

Density regression

↪ Update the hyperparameters, $\mathbf{m}$ and $\mathbf{S}$, by sampling from their full conditional distributions, namely

$$\mathbf{m} \mid \text{else} \sim \mathrm{N}_Q\left(\mathbf{V}\left(\mathbf{S}_0^{-1}\mathbf{m}_0 + \mathbf{S}^{-1}\sum_{l=1}^{L}\boldsymbol{\beta}_l\right), \mathbf{V}\right), \quad \mathbf{V} = (\mathbf{S}_0^{-1} + L\mathbf{S}^{-1})^{-1},$$

$$\mathbf{S}^{-1} \mid \text{else} \sim \mathrm{W}_Q\left(\nu + L, \left(\nu\boldsymbol{\Psi} + \sum_{l=1}^{L}(\boldsymbol{\beta}_l - \mathbf{m})(\boldsymbol{\beta}_l - \mathbf{m})'\right)^{-1}\right).$$

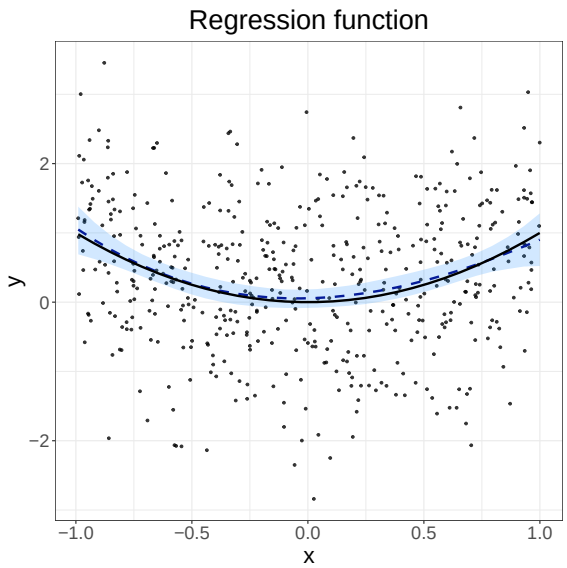
↪ If given a $\Gamma(a_\alpha, b_\alpha)$ prior, posterior samples for α can be drawn from

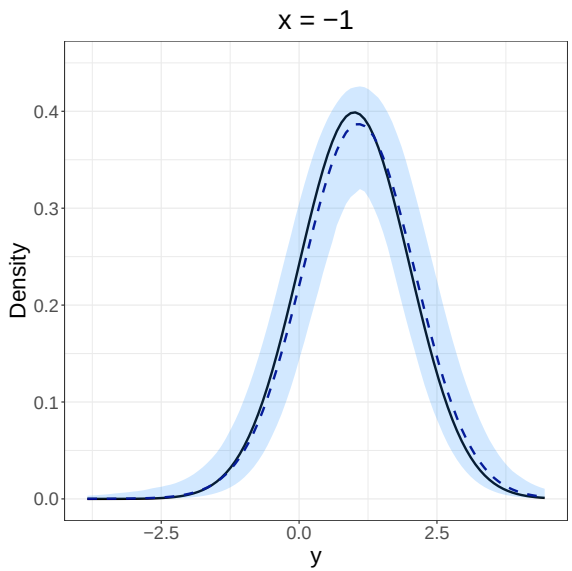
$$\alpha \mid \text{else} \sim \Gamma\left(a_\alpha + L - 1, b_\alpha - \sum_{l=1}^{L-1}(1 - v_l)\right).$$

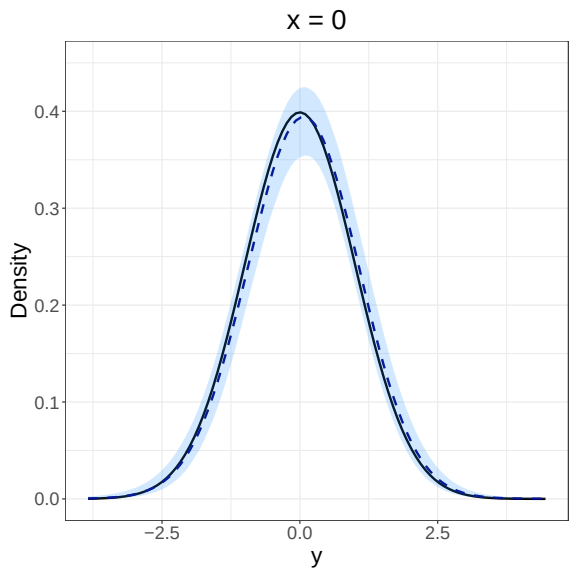
Dependent Dirichlet process mixtures

Density regression

$\hookrightarrow \quad y_i \mid x_i \overset{\text{ind.}}{\sim} \mathrm{N}\left(x_i^2, 1^2\right), \quad x_i \overset{\text{iid}}{\sim} \mathrm{U}(-1, 1), \quad i = 1, \ldots, 500.$







Dependent Dirichlet process mixtures

Density regression: application to ROC regression

- We revisit the ROC curves application, now focusing on ROC regression.
- The conditional or covariate-specific ROC curve, given a covariate value $\mathbf{x}$, is defined as

$$\text{ROC}(p \mid \mathbf{x}) = 1 - F_D\{F_{\bar{D}}^{-1}(1-p \mid \mathbf{x}) \mid \mathbf{x}\},$$

  where $F_{\bar{D}}(y \mid \mathbf{x}) = \Pr(Y_{\bar{D}} \leq y \mid \mathbf{X}_{\bar{D}} = \mathbf{x})$ and $F_D(y \mid \mathbf{x}) = \Pr(Y_D \leq y \mid \mathbf{X}_D = \mathbf{x})$ are the conditional CDFs of the test in the nondiseased and diseased groups, respectively.
- In this case, a number of possibly different ROC curves (and therefore discriminatory accuracies) may be obtained for different values of $\mathbf{x}$.
- Thus, the covariate-specific ROC curve is an important tool that helps to understand and determine the optimal and suboptimal populations where to apply the tests on.
- That is, the covariate-specific ROC curve allows determining the populations, defined by or homogeneous with respect to $\mathbf{x}$, where the diagnostic test has a 'good' or 'poor' discriminatory capacity.

Dependent Dirichlet process mixtures

Density regression: application to ROC regression

- ↪ As it was the case for the ROC curve, flexibly estimating the covariate-specific ROC curve is a matter of flexibly estimating the conditional CDFs in the diseased and nondiseased group.
- ↪ Here we will use a single-weights dependent Dirichlet process with the effect of the covariates in each normal component modelled through cubic B-splines basis functions.
- ↪ We will also show how to evaluate, through the covariate-specific ROC curve, the possible modifying effect of age and gender on the discriminatory capacity of the BMI.
- ↪ Results and code to reproduce them are available in the Supplementary file.

Mixtures of finite Polya trees

PT prior

- We now focus on another popular nonparametric prior for density/distribution estimation. The discussion closely follows Branscum, Johnson, and Baron (2013).
- Polya tree priors have been discussed as early as Freedman (1963), Fabius (1964), and Ferguson (1974).
- However, the natural starting point for understanding their potential use in modeling data is Lavine (1992, 1994), while Hanson (2006) considered the fundamental computational details and popularise them.
- Polya trees are way less popular than DPs but they are a powerful tool as well.

Mixtures of finite Polya trees

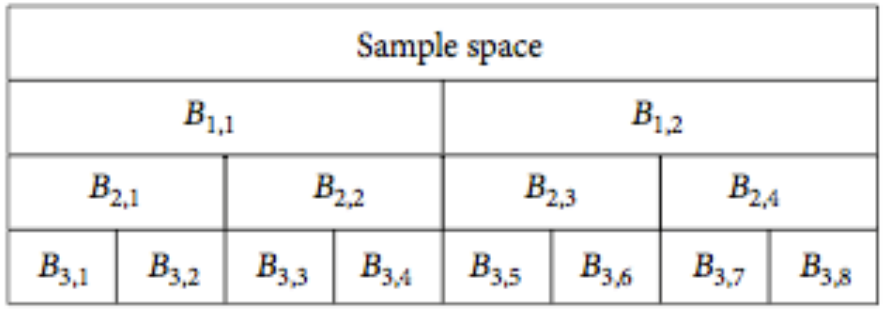
PT prior

- Suppose that a random sample $y_1, \ldots, y_n$ is obtained from an unknown/random distribution F.
- A Polya tree for a distribution F is constructed by dividing the sample space into finer-and-finer disjoint sets using successive binary partitioning.
- For instance, the first partition splits the sample space into two non overlapping intervals.
- In the second partition, those two intervals are each split, yielding a finer partition that contain four intervals.
- Then, these four intervals are each split to give an eight interval third level partition of the sample space.
- At level j of the tree, the sample space is partitioned into 2^j intervals, $j = 1, \ldots$

Mixtures of finite Polya trees

PT prior

↪ Let $B_{j,k}$ denotes the kth interval at level j of the tree, for $j = 1, \ldots,$ and $k = 1, \ldots, 2^j$.



↪ Observe that the partitions are nested within one another, starting at the top of the tree and working up.

↪ For example, by definition, $B_{1,1} = B_{2,1} \cup B_{2,2}$ and $B_{j-1,1} = B_{j,1} \cup B_{j,2}$, etc.

↪ These intervals can be thought as the bins in the histogram.

Mixtures of finite Polya trees

Finite PT prior

- ↪ For a full tree the splitting continues ad infinitum.
- ↪ However, in practice, we truncate to a fixed J, hence the term, finite Polya tree.
- ↪ Generally, setting J equal to 4, 5, or 6 often works well in practice.
- ↪ Another option is to select J so that roughly $J = \log_2 n$ (Hanson, 2006).

Mixtures of finite Polya trees

Finite PT prior

- ↪ Informally speaking, the unknown distribution F assigns the data points y_is to the intervals at level J of the tree and the task is to use the observed distribution of the data into the intervals to estimate F.

- ↪ Although all levels of the tree are important for the purpose of estimating F, of primary importance is level J.

- ↪ The goal here is to produce a data-driven estimate of F that assigns high probability to intervals that contain lots of data, assigns low probability to empty intervals, and assigns midrange probability to intervals that contain some (but not a lot) of the data.

Mixtures of finite Polya trees

Finite PT prior

- Let us consider first the simplest case $J = 1$.
- Then data are assigned to either $B_{1,1}$ or $B_{1,2}$.
- The probability of a data point y_i being assigned to $B_{1,1}$ is $F(B_{1,1}) = \Pr(y_i \in B_{1,1})$ which since F is a cdf and $B_{1,1}$ is an interval of the form (L, U) is to be interpreted as $F(B_{1,1}) = F(U) - F(L)$.
- Denote this unknown probability by $\pi_{1,1}$.
- Then, by the complement rule, $\pi_{1,2} = 1 - \pi_{1,1}$ is the probability assigned to set $B_{1,2}$.
- In notation, $\Pr(y_i \in B_{1,1}) = F(B_{1,1}) = \pi_{1,1}$ and $\Pr(y_i \in B_{1,2}) = F(B_{1,2}) = \pi_{1,2}$.
- Since F is unknown, $\pi_{1,1}$ and $\pi_{1,2}$ are also unknown.

Mixtures of finite Polya trees

Finite PT prior

- ↪ To help interpret these parameters, suppose the data arise from a right-skewed distribution (lots of more data in $B_{1,1}$ than in $B_{1,2}$), then $\pi_{1,1}$ would be large and hence $\pi_{1,2}$ would be small, and vice versa for left-skewed data.

- ↪ Obviously, in practice, $J = 1$ is never used because it would lead to a crude estimate of the density function f, much like estimating a density using a relative frequency histogram that contains only two bins.

Mixtures of finite Polya trees

Finite PT prior

- Let us now consider, again for simplicity, the case of $J = 2$. We have now four sets.
- Data assignment is based on whether the data point was in $B_{1,1}$ or $B_{1,2}$ at the previous level $j = 1$.
- If $y_i \in B_{1,1}$ then y_i is assigned to interval $B_{2,1}$ with unknown probability $\pi_{2,1}$ or to interval $B_{2,2}$ with probability $\pi_{2,2} = 1 - \pi_{2,1}$.
- Similarly, define $\pi_{2,3}$ and $\pi_{2,4}(= 1 - \pi_{2,3})$ to be the probability of y_i being assigned to set $B_{2,3}$ or $B_{2,4}$, respectively, given that y_i was on $B_{1,2}$.
- The $\pi_{j,k}$s are conditional parameters, since $\pi_{2,1} = \Pr(y_i \in B_{2,1} \mid y_i \in B_{1,1})$ and $\pi_{2,3} = \Pr(y_i \in B_{2,3} \mid y_i \in B_{1,2})$.
- To relate the $\pi_{j,k}$s to F we must determine the marginal probability of assignment to the various intervals at level $J(= 2)$.

Mixtures of finite Polya trees

Finite PT prior

↪ Observe that interval $B_{2,1}$ is nested on interval $B_{1,1}$, so the marginal probability of interval $B_{2,1}$ is

$$\begin{aligned} F(B_{2,1}) &= \Pr(y_i \in B_{2,1}) \\ &= \Pr(y_i \in B_{2,1} \cap B_{1,1}) \\ &= \Pr(y_i \in B_{2,1} \mid y_i \in B_{1,1}) \Pr(y_i \in B_{1,1}) \\ &= \pi_{2,1}\pi_{1,1}. \end{aligned}$$

↪ Similar steps lead to $F(B_{2,2}) = (1 - \pi_{2,1})\pi_{1,1}$, $F(B_{2,3}) = \pi_{2,3}\pi_{1,2}$, and $F(B_{2,4}) = (1 - \pi_{2,3})\pi_{1,2}$.

↪ Suppose again that F is right skewed . Then the data will estimate $\pi_{1,1}$ to be large, and it will estimate $\pi_{2,1}$ to be large since most of the n data points will be assigned to set $B_{2,1}$.

↪ Therefore, the estimate of $F(B_{2,1})$ will be (relatively) large.

Mixtures of finite Polya trees

Finite PT prior

- ↪ Notice that level 1 has only one unique parameter, $\pi_{1,1}$, associated with it because $\pi_{1,2}$ is completely determined by $\pi_{1,1}$.
- ↪ Similarly, level 2 has two unique parameters, $\pi_{2,1}$ and $\pi_{2,3}$ associated with it.
- ↪ We can continue the partitioning to any level J.
- ↪ For $J = 3$, we add eight conditional probabilities parameters, $\pi_{3,1}, \pi_{3,2}, \ldots, \pi_{3,8}$, but only four of these are unique.
- ↪ For instance, $\pi_{3,1}$ is the probability that y_i is in interval $B_{3,1}$ given that it is in interval $B_{2,1}$, and

$$\begin{aligned} F(B_{3,1}) &= \Pr(y_i \in B_{3,1}) \\ &= \Pr(y_i \in B_{3,1} \cap B_{2,1}) \\ &= \Pr(y_i \in B_{3,1} \mid y_i \in B_{2,1}) \Pr(y_i \in B_{2,1}) \\ &= \pi_{3,1}\pi_{2,1}\pi_{1,1}. \end{aligned}$$

- ↪ In general, we have

$$F(B_{j,k}) = \prod_{l=1}^{j} \pi_{l,\text{Int}\{(k-1)2^{l-j}+1\}}, \quad j = 1, \ldots, J, \quad k = 1, \ldots, 2^j.$$

Mixtures of finite Polya trees

Finite PT prior

- The key point is that if we can estimate all of the $\pi_{j,k}$s, then we can estimate the probability that it is allocated by F to each interval at level J.
- So far, we have modeled the probability of assignment to each set at level J, but we have not modeled how probability mass is distributed within each interval at level J.
- For instance, all the y_is can be clumped together in the center of the interval.
- Alternatively, the data could be uniformly distributed, clumped to the right or left side of the interval or have any other dispersion pattern within each interval.
- To address this issue, we model the data according to how a user-specified parametric distribution F_0 allocated probability mass within the intervals at level J.
- So, as it was in the DP (or DPM), here with finite Polya trees, the user also needs to specify a probability distribution (and we will see in a few slides that F_0 is also a centring distribution).

Mixtures of finite Polya trees

Finite PT prior

- The distribution F_0 is also used to determine the lower and upper endpoints of all intervals in the tree.
- The median of F_0 is used to split the sample space into two intervals at level 1 of the tree.
- The quartiles of F_0 define cut points for intervals at level 2.
- Writing the 25th percentile as $F_0^{-1}(1/4)$, the median as $F_0^{-1}(2/4)$, and the 75th percentile as $F_0^{-1}(3/4)$, we have for a sample space that covers the real line

$$B_{2,1} = \left(-\infty, F_0^{-1}(1/4)\right), \quad B_{2,2} = \left(F_0^{-1}(1/4), F_0^{-1}(2/4)\right)$$
$$B_{2,3} = \left(F_0^{-1}(2/4), F_0^{-1}(3/4)\right), \quad B_{2,4} = \left(F_0^{-1}(3/4), \infty\right)$$

- In general, the (j,k)th interval is

$$B_{j,k} = \left(F_0^{-1}\left(\frac{k-1}{2^j}\right), F_0^{-1}\left(\frac{k}{2^j}\right)\right), \quad j = 1, \ldots, J, \quad k = 1, \ldots, 2^j.$$

Mixtures of finite Polya trees

Finite PT prior

- The collection $\Pi = \{\pi_{j,k} : j = 1, \ldots, J, k = 1, \ldots, 2^j\}$ constitutes the unknown parameters corresponding to F.
- The probabilities in Π are assumed mutually independent. That is, for instance, (π_{21}, π_{22}) and (π_{23}, π_{24}) are independent.
- We thus need to specify a prior distribution over this collection.
- Due to the fact that when k is an even number between 2 and 2^j, $\pi_{j,k} = 1 - \pi_{j,k-1}$, priors are needed only on $\pi_{j,k}$ when k is odd.
- Since the $\pi_{j,k}$s are probabilities, it is standard to use independent beta priors, specifically

$$\pi_{j,2k-1} \sim \text{Beta}(c\rho(j), c\rho(j)), \quad j = 1, \ldots, J, \quad k = 1, \ldots, 2^{j-1}.$$

- In most of the applications $\rho(j) = j^2$ as this guarantees an absolutely continuous F (in an infinite tree) (Ferguson, 1974).

Mixtures of finite Polya trees

Finite PT prior

- ↪ Before proceeding on further considerations about the role of c, let us note that under this parametrisation
$$E[F(B(j,k))] = \frac{1}{2^j} = F_0(B_{j,k}).$$
- ↪ Thus, F_0 is the prior expectation of the unknown distribution function F.
- ↪ F_0 is usually selected based on our best prior assessment of the data-generating distribution F.
- ↪ The parameter $c > 0$, also referred to as the weight parameter, acts much like as the precision parameter α in the Dirichlet process.
- ↪ As in the Dirichlet process, large values of c leads to realizations of F that are close to F_0.
- ↪ A very low value of c (e.g., $c = 0.1$) will often lead to an estimate of F that is similar to the empirical cdf. Usually $c = 1$ works well in practice.
- ↪ Just like α, c can be regarded as random and a prior placed on it.

Mixtures of finite Polya trees

Finite PT prior

- ↪ Once we have selected J, F_0, and c we have all the elements needed to specify a finite Polya tree for F.
- ↪ The formula for the density function f (our interest here) is given by

$$f(y) = 2^J p(B_{J,k(y)}) f_0(y). \tag{1}$$

- ↪ Here $k(y) \in \{1, \ldots, 2^J\}$ identifies the interval at level J containing y and $p(B_{J,k(y)})$ is the probability of that interval (the product of J of the $\pi_{j,k}$s) (Hanson, 2006).
- ↪ The interval that contains y at level J can be determined using the formula $k(y) = \text{Int}(2^J F_0(y) + 1)$.
- ↪ Note that the density at stage J is just the product of a weighting factor $2^J p(B_{J,k(y)})$ and the original used-specified parametric density f_0.
- ↪ Prior or posterior distributions that focus high probability on regions around $\pi_{jk} = 0.5$ for all j and k will behave very much like the f_0 density.

Mixtures of finite Polya trees

Finite PT prior

- Given Π, F is known.
- So, in order to compute posterior estimates of F (or f) we only need to know how to update the $\pi_{j,k}$s.
- Fortunately, just like the DP, Polya trees enjoy also a simple conjugacy result.
- Specifically, if

$$y_1, \ldots, y_n \mid F \overset{\text{iid}}{\sim} F, \quad F \sim \text{FPT}_J(c, F_0),$$

then $F \mid \mathbf{y}$ is updated through

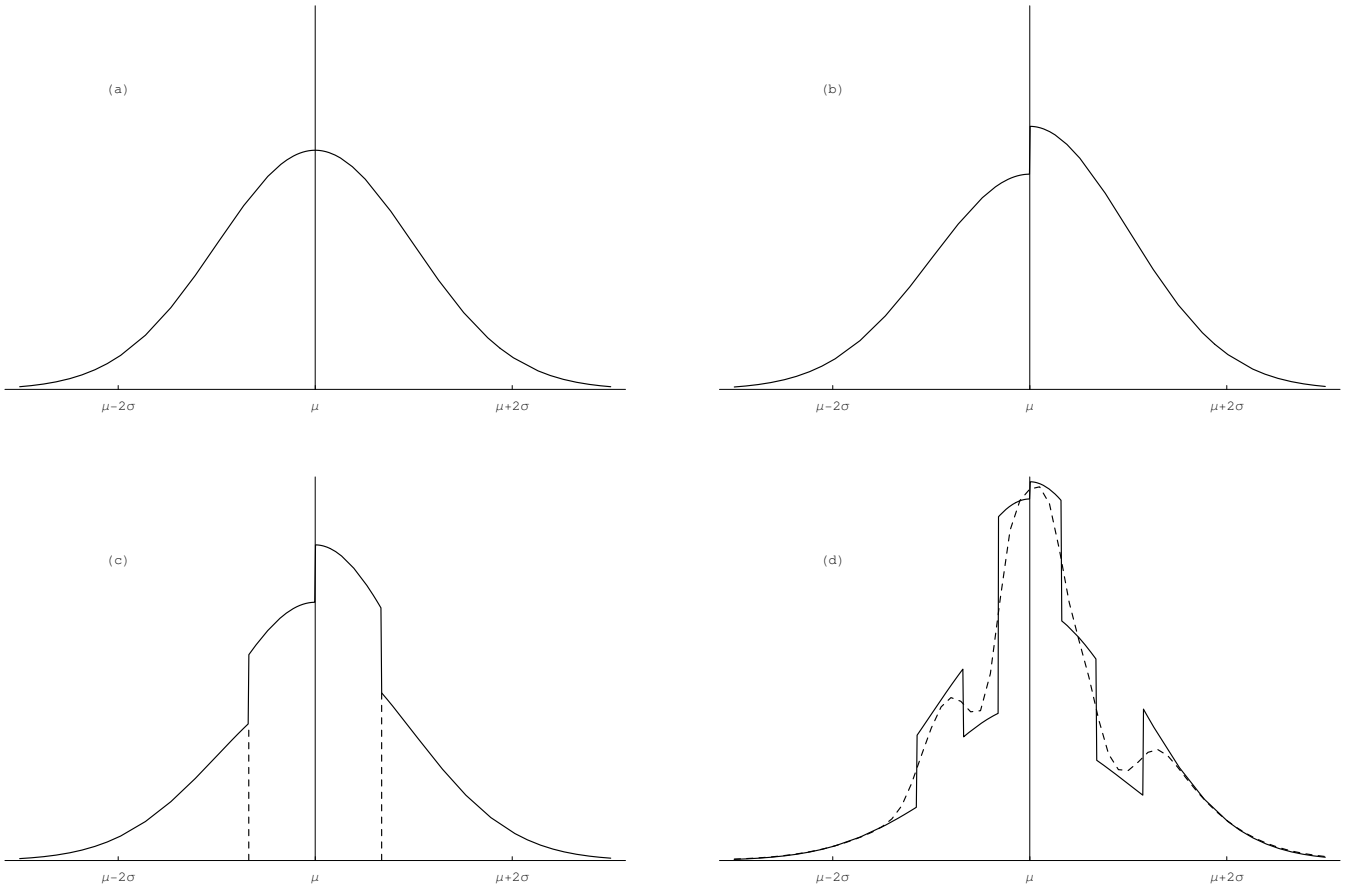
$$\pi_{j,2k-1} \mid \mathbf{y} \overset{\text{ind.}}{\sim} \text{Beta}\left(cj^2 + \sum_{i=1}^{n} I(y_i \in B_{j,k}), cj^2 + \sum_{i=1}^{n} I(y_i \in B_{j,k+1})\right), \tag{2}$$

for $j = 1, \ldots, J$ and $k = 1, \ldots, 2^{j-1}$.

- In words, we update the Beta parameters by counting the number of observations that fall in each set of each level of the tree.
- That is, $F \mid \mathbf{y}$ is a PT with Beta parameters updated through (2).

Mixtures of finite Polya trees

Finite PT prior: example



FPT density estimates considering $F_0 = \mathrm{N}(\mu, \sigma^2)$ and $J = 3$. (a) $\mathrm{N}(\mu, \sigma^2)$ density. (b) $j = 1$; $\pi_{1,1} = 0.45$. (c) $j = 2$; $\pi_{2,1} = 0.4$, $\pi_{2,3} = 0.6$. (d) J=3; $\pi_{3,1} = 0.3$, $\pi_{3,3} = 0.3$, $\pi_{3,5} = 0.6$, $\pi_{3,7} = 0.3$.

Mixtures of finite Polya trees

- ↪ All densities in the previous figure are too jagged, which turns out to be the result of using a fixed F_0.
- ↪ In fact, one of the major criticisms of Polya trees is that, unlike the DP, inferences are somewhat sensitive to the choice of a fixed partition.
- ↪ A remedy is to place a prior distribution on the parameters of F_0, say $\boldsymbol{\theta}$, we denote the resulting centering distribution as $F_{0,\boldsymbol{\theta}}$ to emphasize the dependence on $\boldsymbol{\theta}$.
- ↪ A prior on $\boldsymbol{\theta}$ implies that the starting and endpoints of the sets of the tree are uncertain/random.
- ↪ This has the effect of smoothing out the abrupt jumps at these points that are noticeable in the previous figure.
- ↪ In fact, in panel (d) of the previous figure it is also shown the estimate obtained by considering $\boldsymbol{\theta} = (\mu, \sigma^2)$ as random (dashed line).

Mixtures of finite Polya trees

↪ So, the final model is

$$\begin{aligned} y_1, \ldots, y_n \mid F &\overset{\text{iid}}{\sim} F, \\ F \mid c, \boldsymbol{\theta} &\sim \text{FPT}_J(F_{0,\boldsymbol{\theta}}, c), \\ \boldsymbol{\theta} &\sim p(\boldsymbol{\theta}), \end{aligned}$$

and it is known as a mixture of finite Polya trees.

↪ It can be alternatively written as

$$F \sim \int \text{FPT}_J(F_{0,\boldsymbol{\theta}}, c) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}.$$

↪ The formula for the density function f is identical to that given in (1).

↪ To conduct posterior inference we will now need to know how to sample $\boldsymbol{\theta} \mid \mathbf{y}, \Pi$ (we already know how to sample $\Pi \mid \mathbf{y}, \boldsymbol{\theta}$).

↪ We will make this concrete considering the particular case of $F_{0,\boldsymbol{\theta}} = \text{N}(\mu, \sigma^2)$.

Mixtures of finite Polya trees

- We center random F at $F_{0,\boldsymbol{\theta}} = \mathrm{N}(\mu, \sigma^2)$, where $\boldsymbol{\theta} = (\mu, \sigma^2)$.
- Using (1) the likelihood $L(\Pi, \boldsymbol{\theta}; \mathbf{y})$ is

$$\prod_{i=1}^{n} 2^J \phi(y \mid \mu, \sigma^2) p(k_{\boldsymbol{\theta}}(J, y_i)).$$

- We write $p(k_{\boldsymbol{\theta}}(J, y_i))$ instead of $p(B_{J,k(y_i)})$ to alleviate notation and to make clear the dependence on $\boldsymbol{\theta}$.
- As in Branscum et al. (2008), we assume $\mu \sim \mathrm{N}(a_\mu, b_\mu^2)$ and $\sigma \sim \Gamma(a_\sigma, b_\sigma)$.
- Assuming further that $\boldsymbol{\theta}$ and Π are a priori independent, the joint posterior density is proportional to

$$p(\boldsymbol{\theta}, \Pi \mid \mathbf{y}) \propto L(\Pi, \boldsymbol{\theta}; \mathbf{y}) p(\boldsymbol{\theta}) p(\Pi).$$

- The full conditionals for μ and σ are not recognizable as belonging to a parametric family thus these parameters are updated through Metropolis–Hastings steps.

Mixtures of finite Polya trees

MCMC

1. μ is updated by sampling $\mu^* \sim \mathrm{N}(\mu, s_1)$ and accepted with probability

$$\min\left\{1, \frac{\exp\{-0.5b_\mu^{-2}(\mu^* - a_\mu)^2\}}{\exp\{-0.5b_\mu^{-2}(\mu - a_\mu)^2\}} \frac{\prod_{i=1}^n p(k_{\mu^*,\sigma}(J, y_i))}{\prod_{i=1}^n p(k_{\mu,\sigma}(J, y_i))} \times \frac{\exp\{-0.5\sigma^{-2}\sum_{i=1}^n (y_i - \mu^*)^2\}}{\exp\{-0.5\sigma^{-2}\sum_{i=1}^n (y_i - \mu)^2\}}\right\}.$$

Here s_1 is a tuning parameter that needs to be calibrated to achieve a desirable acceptance rate.

2. σ is updated by sampling $\sigma^* \sim \Gamma(\sigma s_2, s_2)$ and accepted with probability

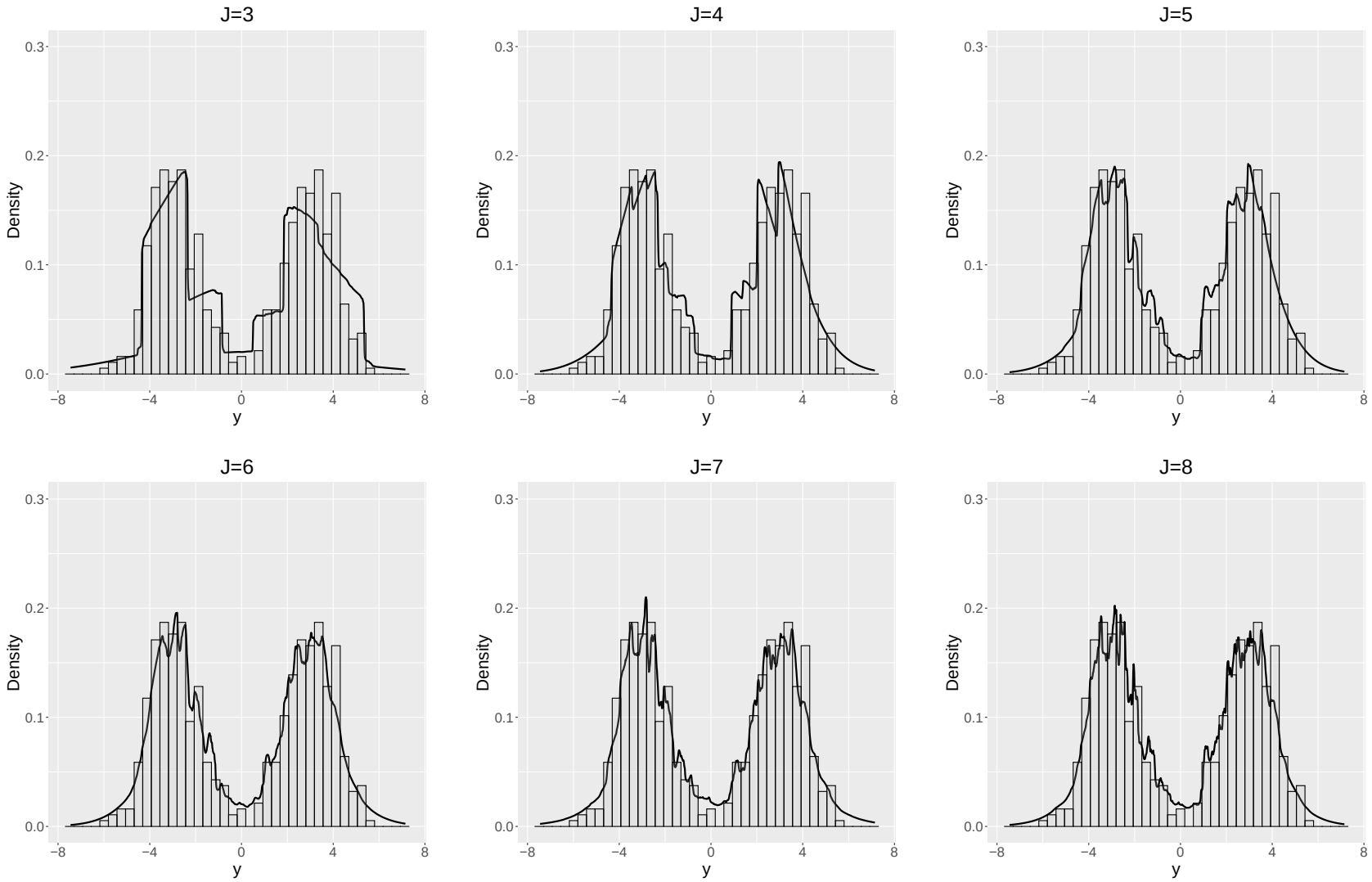
$$\min\left\{1, \frac{f_\Gamma(\sigma^*; a_\sigma, b_\sigma)}{f_\Gamma(\sigma; a_\sigma, b_\sigma)} \frac{\prod_{i=1}^n p(k_{\mu,\sigma^*}(J, y_i))}{\prod_{i=1}^n p(k_{\mu,\sigma}(J, y_i))} \times \frac{\sigma^n \exp\{-0.5(\sigma^*)^{-2}\sum_{i=1}^n (y_i - \mu)^2\}}{\sigma^{*n} \exp\{-0.5\sigma^{-2}\sum_{i=1}^n (y_i - \mu)^2\}} \frac{f_\Gamma(\sigma; \sigma^* s_2, s_2)}{f_\Gamma(\sigma^*; \sigma s_2, s_2)}\right\},$$

where s_2 has the same meaning as s_1.

3. Use (2) to update $\pi_{j,k}$, for $k = 1, \ldots, 2^{j-1}$ and $j = 1, \ldots, J$.

Mixtures of finite Polya trees

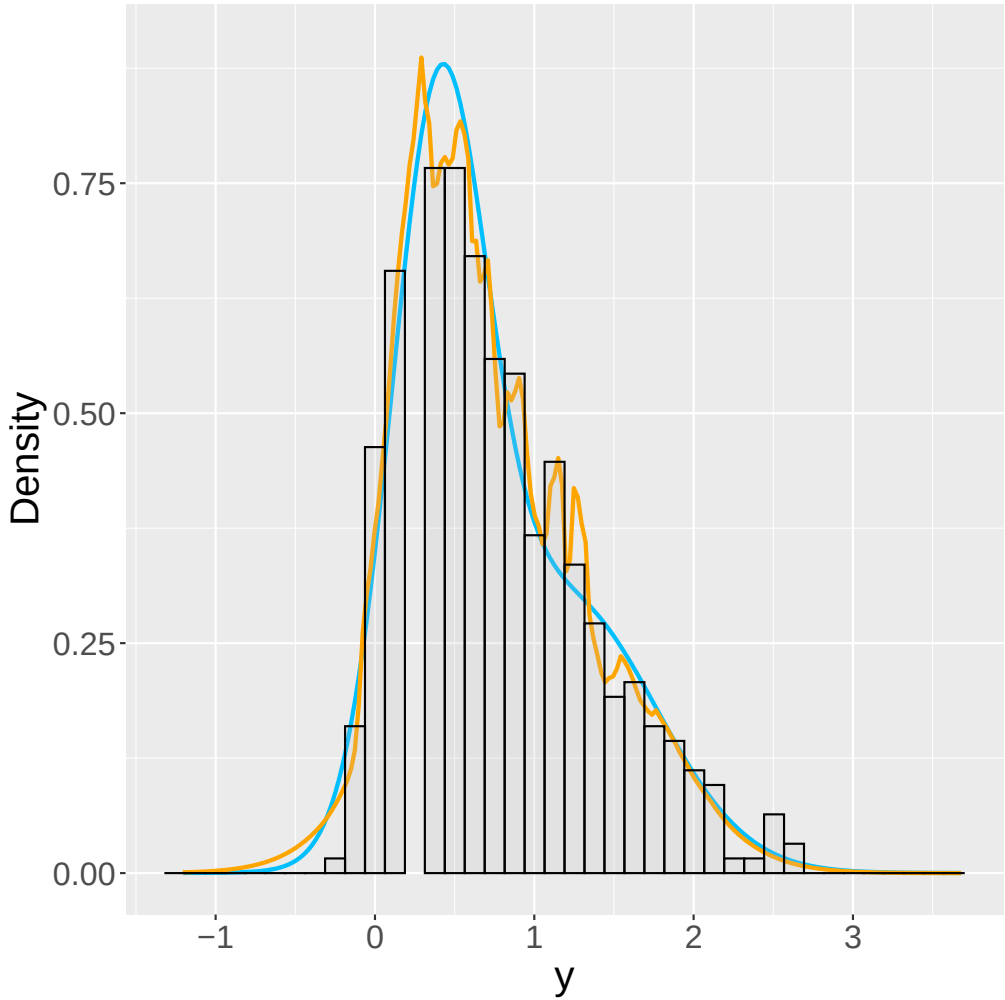
Example: $y_i \overset{\text{iid}}{\sim} 0.5\phi(y \mid -3, 1) + 0.5\phi(y \mid 3, 1)$, $n = 500$.



Mixtures of finite Polya trees

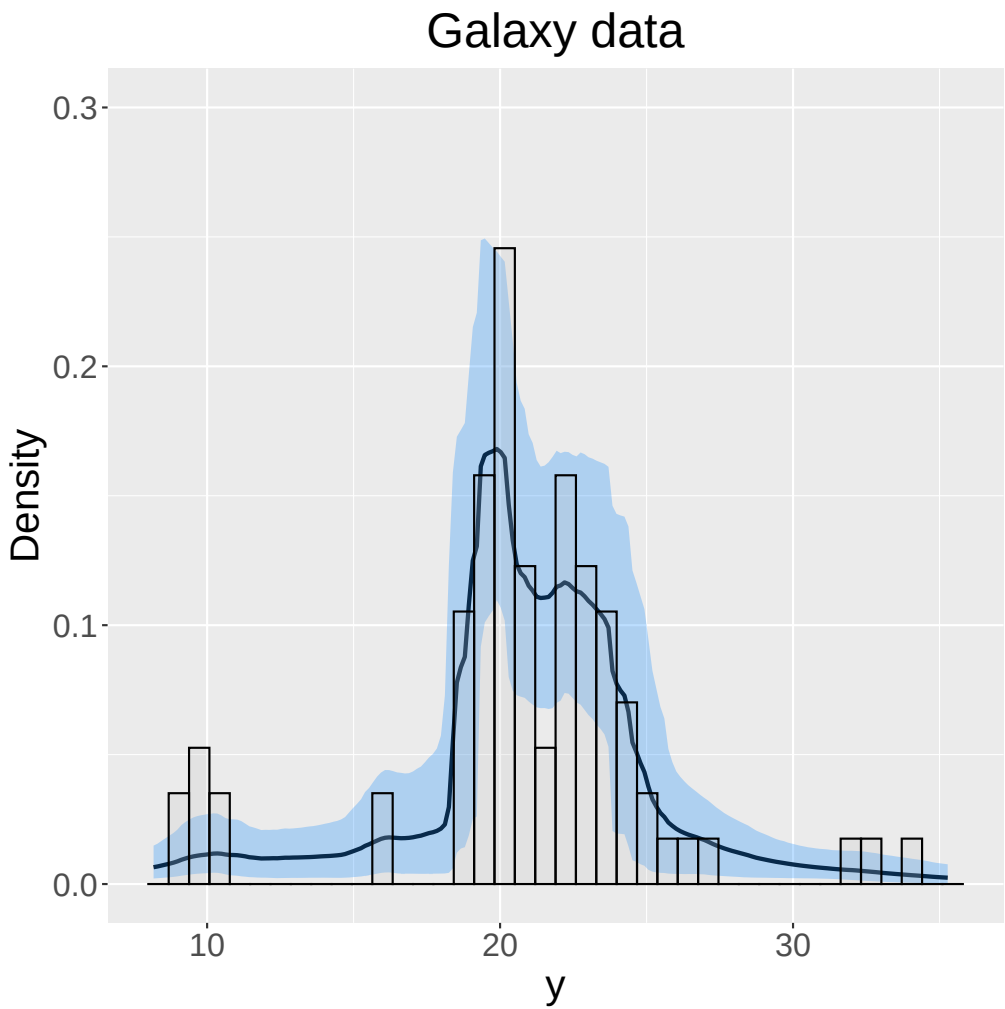
Examples

↪ True data generating mechanism: $\phi_{\text{SN}}(y \mid \mu = 0, \sigma^2 = 1, \lambda = 8), \quad n = 500.$



Mixtures of finite Polya trees

Examples



Linear dependent tailfree process

- A Polya tree defines the conditional probabilities $\pi_{j+1,2k-1}$, $\pi_{j+1,2k}$ as beta distributions.
- To accommodate covariates, and in a spirit of density regression, Jara and Hanson (2011) proposed to model these probabilities through logistic regression.
- Specifically, given covariates $\mathbf{x}$, the probabilities $(\pi_{j+1,2k-1}, \pi_{j+1,2k})$ are defined as

$$\log\left(\frac{\pi_{j+1,2k-1}}{\pi_{j+1,2k}}\right) = \mathbf{x}^T \boldsymbol{\tau}_{j,k}.$$

- The resulting model is known as linear dependent tail free process. For further details see Jara and Hanson, 2011, *Biometrika*.
- The function `LDTFPdensity` in `DPpackage` implements this model.

Thank you...

...for your attention!!!