

Math 598

Markov chain Monte Carlo

Dr David A. Stephens

Department of Mathematics & Statistics
Room 1225, Burnside Hall

Bayesian Non-Parametrics: Motivation

- ▶ Parametric models are restrictive;
- ▶ We often meet data sets that exhibit non-standard heterogeneity;
- ▶ Likelihood-based inference is more palatable than moment-based estimation (such as GEE, GMM)
- ▶ Because we can report a more complete summary of the inference results, and perform prediction.

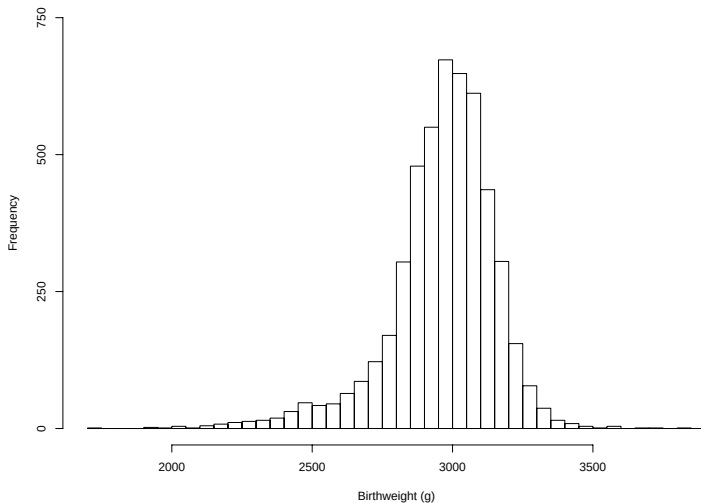
Bayesian Non-Parametrics: Motivation

- Regression Problems
- Survival Analysis
- Clustering
- Measurement Error/Errors-in-Variables Problems
- Density Estimation

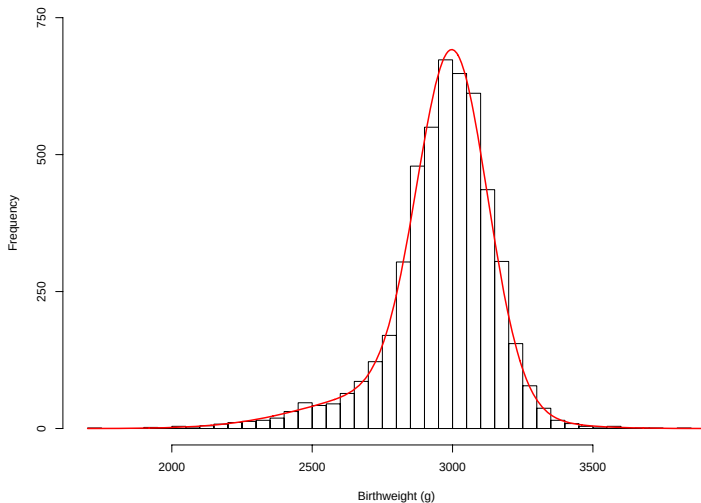
– anywhere where parametric models, or non-parametric maximum likelihood are used.

Some applications in hypothesis testing.

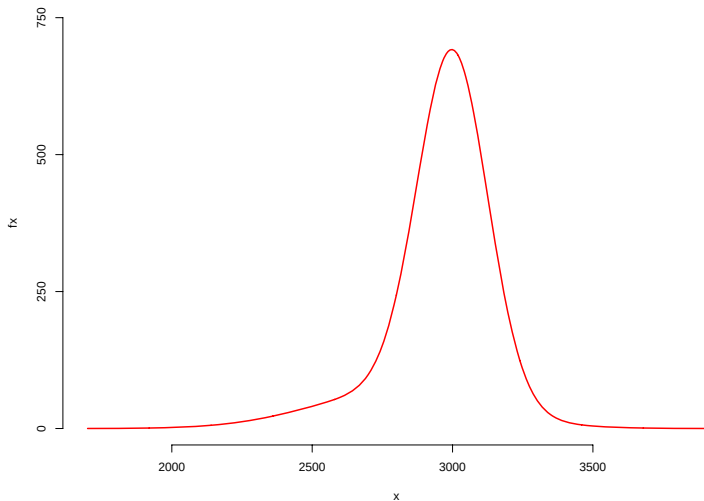
Hypothetical Example: Birthweight data



Hypothetical Example: Birthweight data

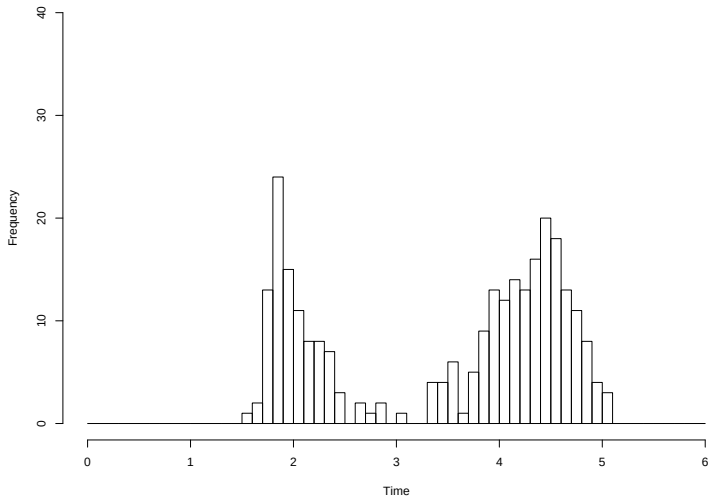


Hypothetical Example: Birthweight data



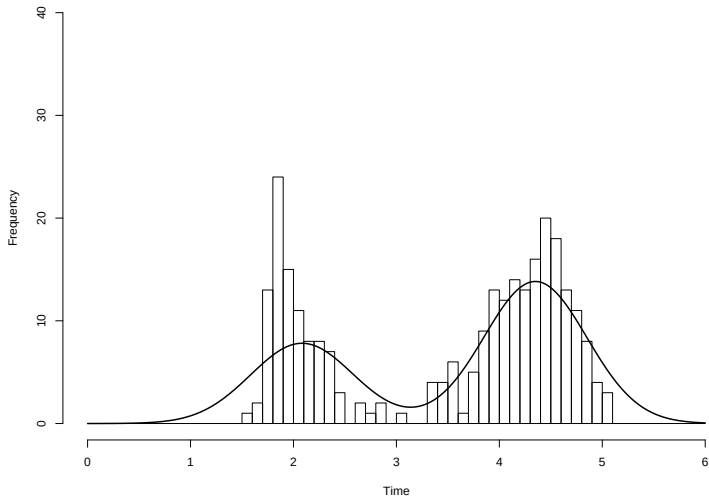
Example: Old Faithful Data

Geyser Data



Example: Old Faithful Data

Geyser Data



Bayesian Inference

In Bayesian statistical inference, we compute and summarize the **posterior distribution** of the unknown parameters in the probability model, in light of observed data.

- In Bayesian **parametric** inference, the parameter is the usual θ , λ , μ say that appears in the presumed (conditional) data generating model.
- In Bayesian **non-parametric** inference, the parameter is the **distribution** from which the data are drawn.

As part of Bayesian inference, we need to specify the **prior** probability distribution for these parameters.

Random Distributions

How do we construct a “random” distribution suitable for non-parametric prior modelling ? In the univariate case:

(I) Choose some locations on the real line

$$x_1, x_2, x_3, \dots$$

(II) Place a mass of probability at each location

$$p_1, p_2, p_3, \dots$$

where the probabilities sum to 1.

Then the function

$$f(x) = \sum_{i=1}^{\infty} p_i \delta_{x_i}(x)$$

is a discrete probability distribution on the set $\{x_1, x_2, x_3, \dots\}$.

Random Distributions

How do we make this random ?

- (I) Choose the locations **randomly** (and independently) from some distribution G_X ; denote them $x_1, x_2, x_3, \dots$
- (II) Choose the probabilities **randomly** in such a way such that they sum to 1; denote them $\pi_1, \pi_2, \pi_3, \dots$

Then

$$\tilde{f}(x) = \sum_{i=1}^{\infty} \pi_i \delta_{x_i}(x)$$

is a random mass function.

Random Distributions

If the number of locations is **finite**, equal to K say, then we can generate the probabilities $\pi_1, \dots, \pi_K$ from a **Dirichlet** distribution

$$\text{Dirichlet}(K; \alpha_1, \dots, \alpha_{K+1})$$

where the α s are fixed constants.

In this case, the x_i s are fixed, or the x_i s simulated from G_X .

Random Distributions

If the number of locations is **infinite**, it is helpful to order the π_i in descending order

$$\pi_1 \geq \pi_2 \geq \pi_3 \geq \cdots$$

so that eventually the terms in the infinite sum become effectively zero, so that the truncation

$$\tilde{f}(x) \simeq \tilde{f}_N(x) = \sum_{i=1}^N \pi_i \delta_{x_i}(x)$$

can be computed.

Random Distributions

For example, consider for $\alpha > 0$

$$V_1, V_2, V_3, \dots \sim \text{Beta}(1, \alpha)$$

independently distributed, and

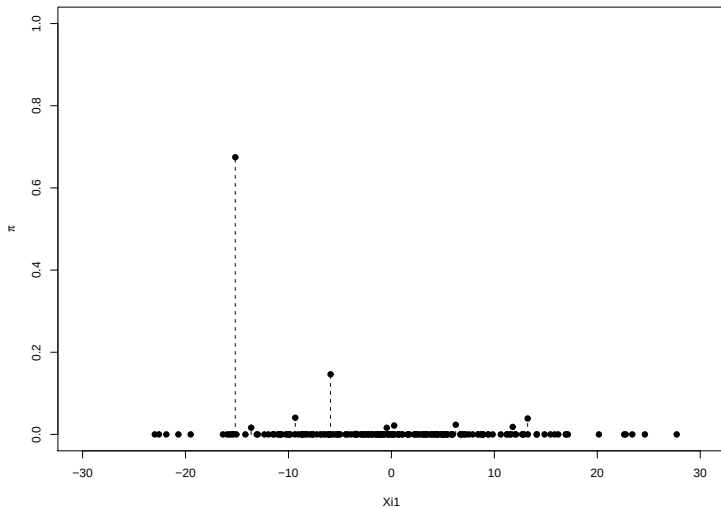
$$\pi_1 = V_1$$

$$\pi_2 = (1 - V_1)V_2$$

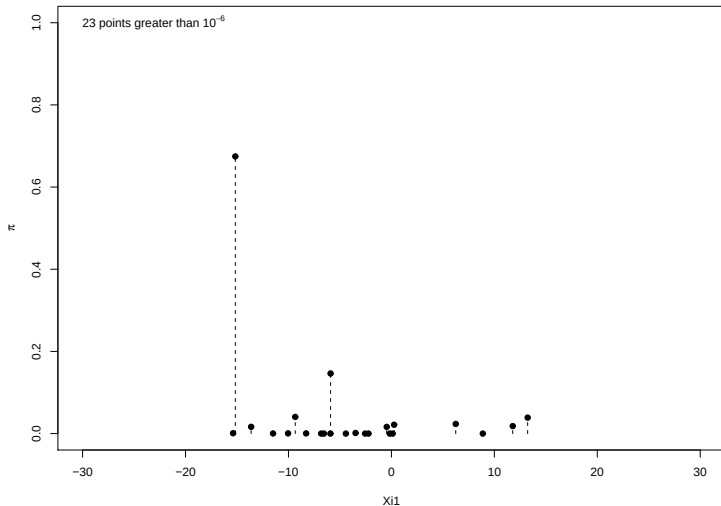
$$\pi_3 = (1 - V_1)(1 - V_2)V_3$$

$$\vdots \quad \vdots \quad \vdots$$

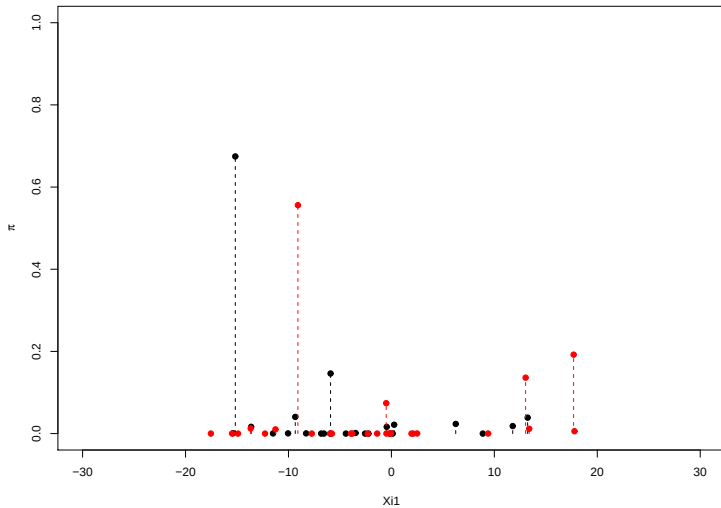
Simulation: $\alpha = 2$



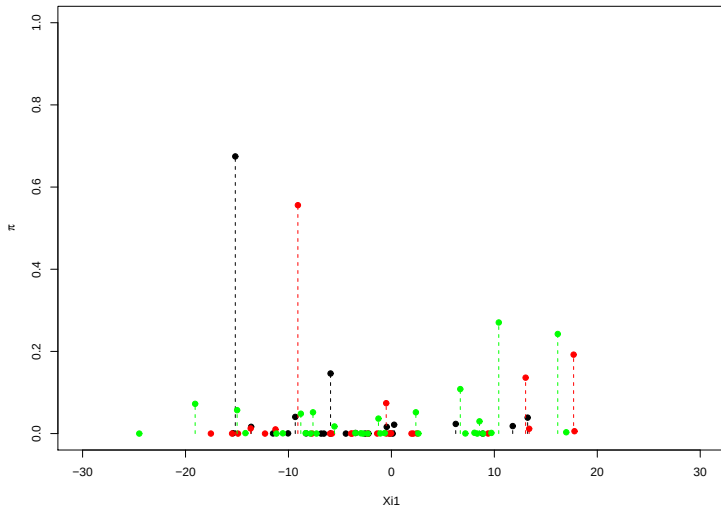
Simulation: $\alpha = 2$, thresholded



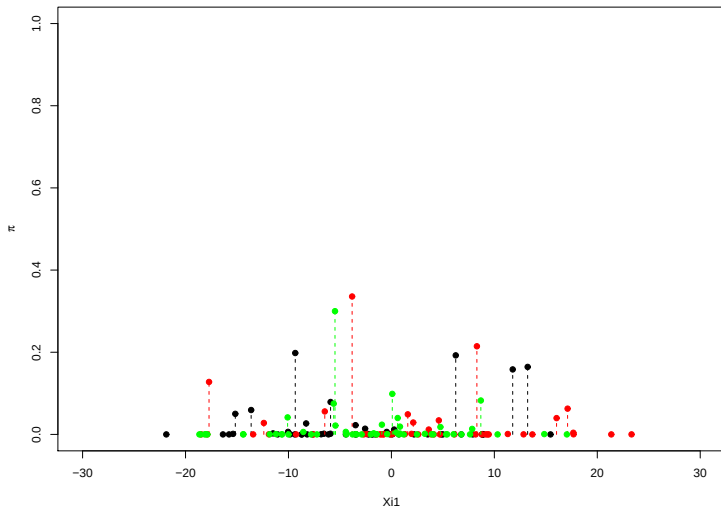
Simulation: $\alpha = 2$, thresholded



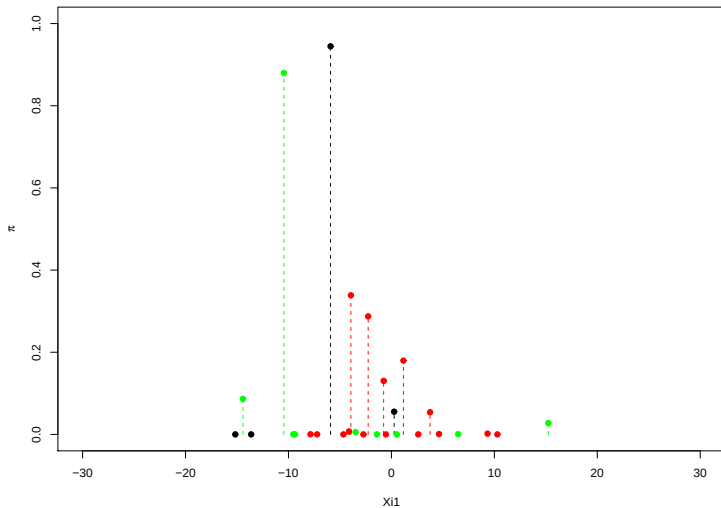
Simulation: $\alpha = 2$, thresholded



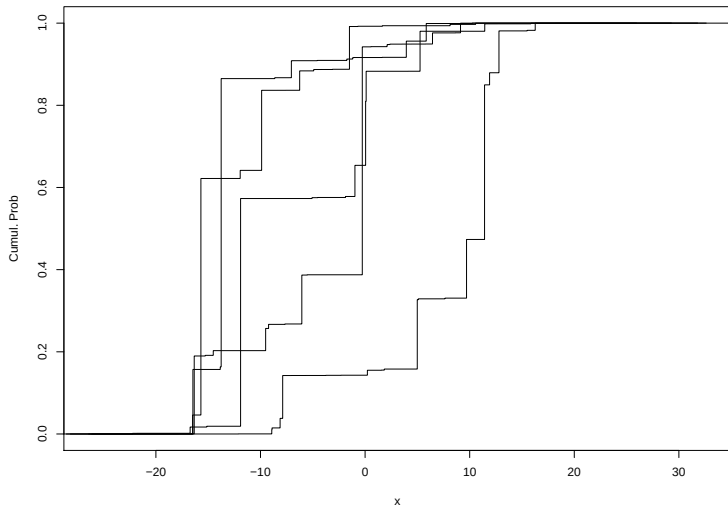
Simulation: $\alpha = 4$



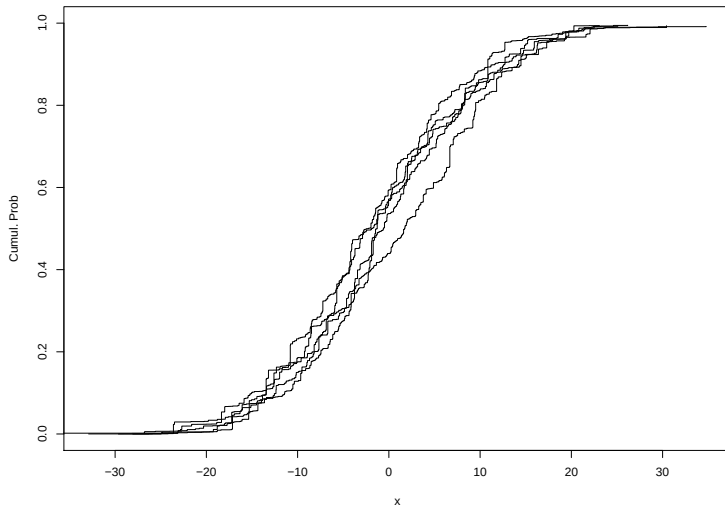
Simulation: $\alpha = 0.5$



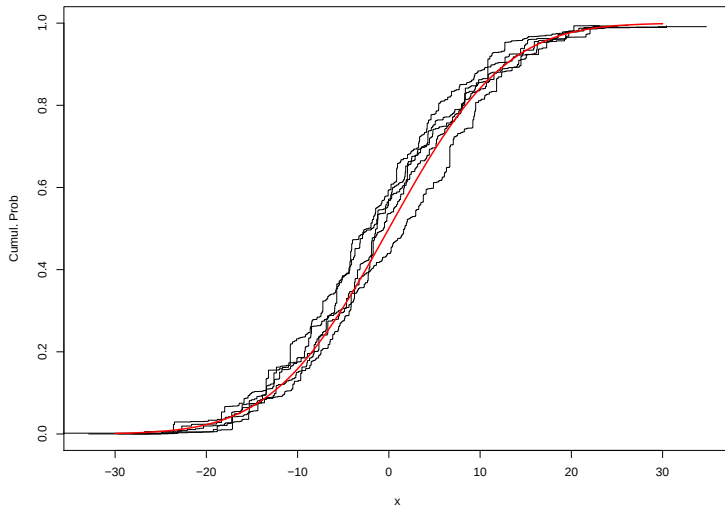
Simulation: $\alpha = 2$ (CDF)



Simulation: $\alpha = 100$ (CDF)



Simulation: $\alpha = 100$ (CDF)



Summary

- Two hyperparameters α, G_X .
- α small gives a few large masses
- α large gives many small masses
- α large reproduces a distribution much like G_X

Thus α is like a **precision** parameter, G_X is like a **location** parameter for the distribution.

Dirichlet Process

The method described above is the **stick-breaking** construction of the **Dirichlet Process** with parameters (α, G_X) .

$$\tilde{f} \sim DP(\alpha, G_X)$$

For any partition of $\mathbb{R}$ into disjoint subsets $B_1, B_2, \dots, B_K, B_{K+1}$ the Dirichlet process assigns random probabilities

$$\mathbf{p} = (p_1, p_2, \dots, p_K, p_{K+1})^\top$$

to the subsets, where

$$\mathbf{p} \sim \text{Dirichlet}(K; \alpha_1, \alpha_2, \dots, \alpha_K, \alpha_{K+1})$$

and $\alpha_k = \alpha G_X(B_k)$ for each k

Posterior Calculation

Suppose *a priori* $\tilde{f} \sim DP(\alpha, G_X)$. What is the posterior distribution if data $y_1, y_2, \dots, y_n$ are observed and presumed independent draws from $\tilde{f}$?

We have a conjugate model: *a posteriori*

$$\tilde{f} \sim DP(\alpha^*, G_X^*)$$

where

$$\begin{aligned}\alpha^* &= \alpha + n \\ G_X^* &= \frac{\alpha G_X + \sum_{j=1}^n \delta_{y_j}}{\alpha + n}\end{aligned}$$

Dirichlet Process

For any partition of $\mathbb{R}$ into disjoint subsets $B_1, B_2, \dots, B_K, B_{K+1}$ with associated probabilities

$$\mathbf{p} = (p_1, p_2, \dots, p_K, p_{K+1})^\top$$

- in the **prior**

$$\mathbf{p} \sim \text{Dirichlet}(K; \alpha_1, \alpha_2, \dots, \alpha_K, \alpha_{K+1})$$

and

$$\alpha_k = \alpha G_X(B_k)$$

- in the **posterior**

$$\mathbf{p} \sim \text{Dirichlet}(K; \alpha_1^*, \alpha_2^*, \dots, \alpha_K^*, \alpha_{K+1}^*)$$

and

$$\alpha_k^* = \alpha^* G_X^*(B_k)$$

Dirichlet Process

It follows that

$$\alpha_k^\star = \alpha^\star G_X^\star(B_k) = G_X(B_k) + n_k$$

where

$$n_k = \text{Number of } y_j \text{ in the set } B_k.$$

Therefore we have the usual kind of Bayesian updating rule.

Also

- α small: posterior looks like empirical cdf
- α large: posterior looks like G_X .

So, overall, things proceed much like parametric inference !

Example: Bayesian Survival Analysis

For possibly censored survival data, the usual approach is to use non-parametric maximum likelihood and the **Kaplan-Meier** curve.

Suppose we have

- event times

$$t_1 \leq \dots \leq t_n$$

with N_j events at time j .

- censoring indicators z_{jk} where

$$z_{jk} = \begin{cases} 1 & \text{Death} \\ 0 & \text{Censored} \end{cases}$$

Can form likelihood in terms of discrete hazards; leads to usual Kaplan-Meier formulation

Example: Bayesian Survival Analysis

In the Bayesian version, we specify a non-parametric model by considering the probability content for the disjoint intervals

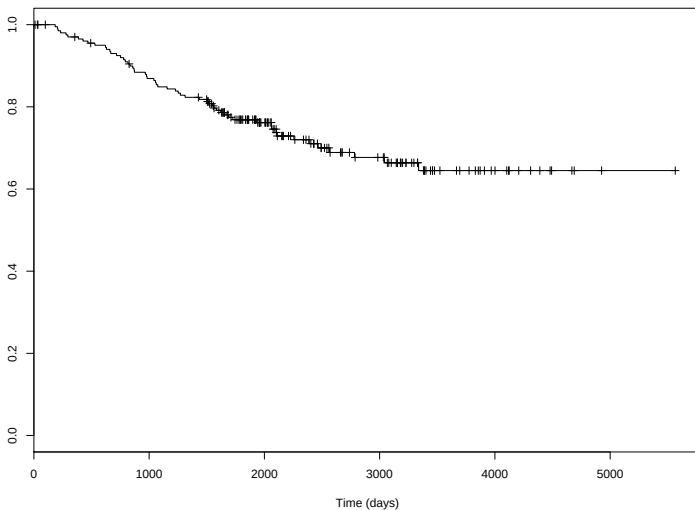
$$(0, t_1], (t_1, t_2], \dots, (t_{n-1}, t_n], (t_n, \infty).$$

We may use a Dirichlet distribution to specify the prior; however, need to specify the α parameters.

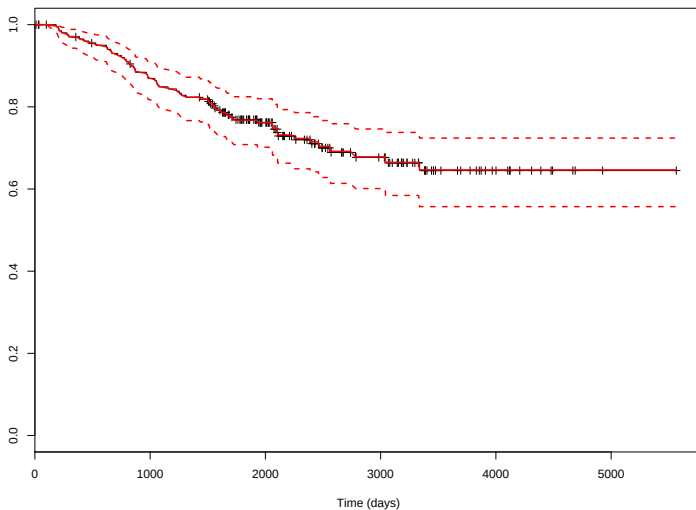
To do this in a **coherent** fashion, we use ideas from the Dirichlet process; we specify

$$\alpha_k = \alpha G_X((t_{k-1}, t_k]) \quad k = 1, 2, \dots$$

Example: Melanoma Data - Kaplan-Meier



Example: Melanoma Data - Bayesian interval



Dirichlet Process: Some issues

1. How do we report the inference ?
 - Which partition, that is, which sets $B_1, B_2, \dots, B_K, B_{K+1}$ should we choose to report the posterior ?
2. In this formulation, the posterior can only support **discrete** distributions
 - that is, any estimate of the true f obtained from the model is almost surely a discrete distribution.

We solve 1. by using **simulation** methods

We solve 2. by **extending the model**.

Simulating from a Dirichlet Process Model

To obtain a sample (of data) of size M from a Dirichlet Process $DP(\alpha, G_X)$:

STEP 1: Sample $Z_1 \sim G_X$.

STEP 2: For $j = 2, \dots, M$, sample

$$Z_j | Z_1, Z_2, \dots, Z_{j-1} \sim \frac{\alpha}{\alpha + j - 1} G_X + \frac{1}{\alpha + j - 1} \sum_{l=1}^{j-1} \delta_{Z_l}(Z_j)$$

that is, given $Z_1, Z_2, \dots, Z_{j-1}$, either sample

Z_j from G_X with probability $\alpha/(\alpha + j - 1)$.

or sample

Z_j from $\{Z_1, \dots, Z_{j-1}\}$ with probability $1/(\alpha + j - 1)$ each.

The Dirichlet Process and Clustering

This algorithm is a **Polya Urn** scheme.

It demonstrates that the Dirichlet process model induces a **clustering** mechanism: in the simulated Z sample, we have many identical values due to the sampling of Z_j uniformly on

$$\{Z_1, \dots, Z_{j-1}\}$$

at each j .

The Dirichlet Process and Clustering

In a sample of n items from a $DP(\alpha, G_X)$ model, the probability of having k clusters is

$$\Pr[K = k] = \frac{\alpha^k B(n, k)}{A_n(\alpha)}$$

where $B(n, k)$ is the **Stirling number of the first kind**.

$$A_n(x) = \sum_{j=1}^n B(n, j) x^j$$

The Dirichlet Process and Clustering

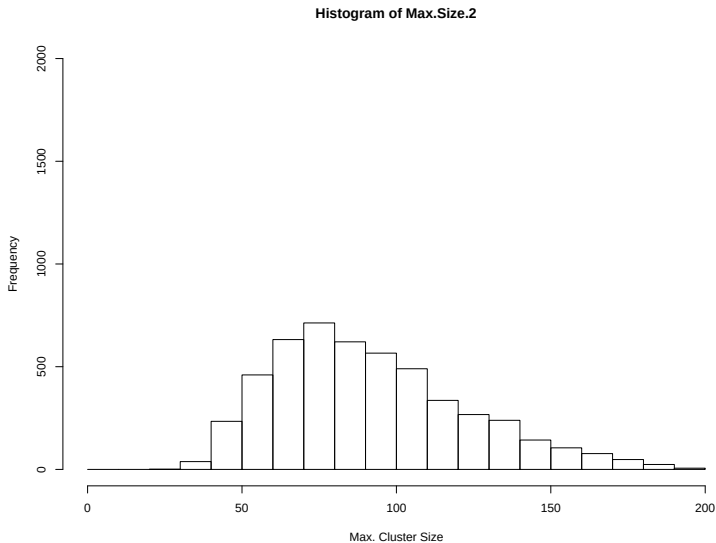
The expected number of clusters *a priori* is

$$\sum_{j=1}^n \frac{\alpha}{\alpha + j - 1} = O(\alpha \log n)$$

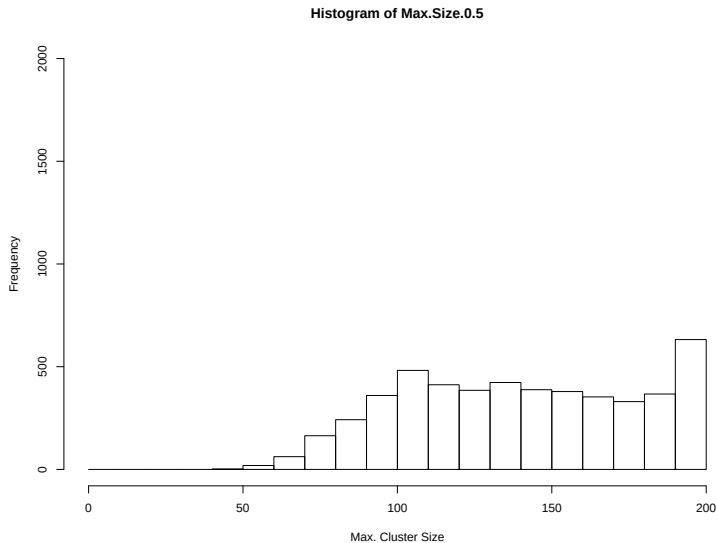
For $n = 200$:

α	$\mathbb{E}[K]$
0.5	3.631
2.0	9.766
4.0	16.238
10.0	30.930

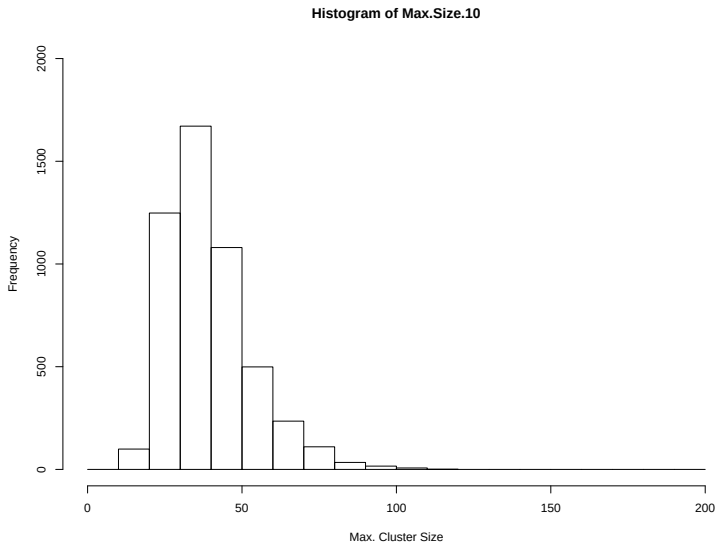
Maximum Cluster size: $\alpha = 2$



Maximum Cluster size: $\alpha = 0.5$

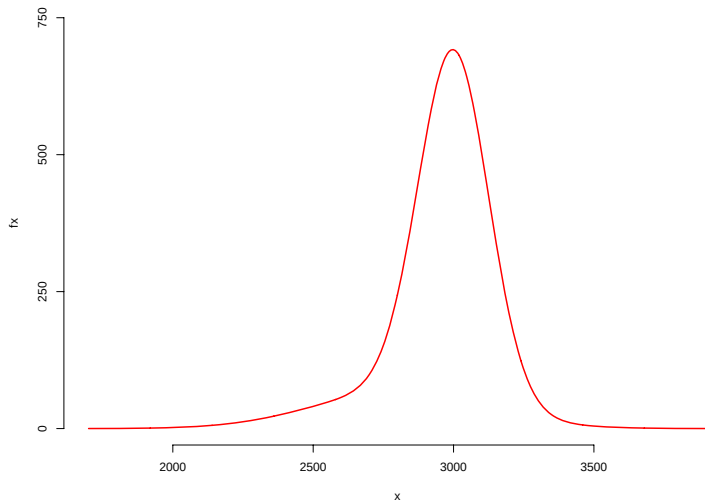


Maximum Cluster size: $\alpha = 10$



Extension to The Continuous Case

We need to model continuous distributions.



Extension to The Continuous Case

We add another stage that brings in a continuous distribution.

For example, could treat each x_i as the location of a normal density, and consider generating a y for each

$$x_1, x_2, \dots \sim G_X$$

$\pi_1, \pi_2, \dots$ generated by stick-breaking.

$$y \sim \phi((y - x_i)/\sigma) \quad i = 1, 2, \dots$$

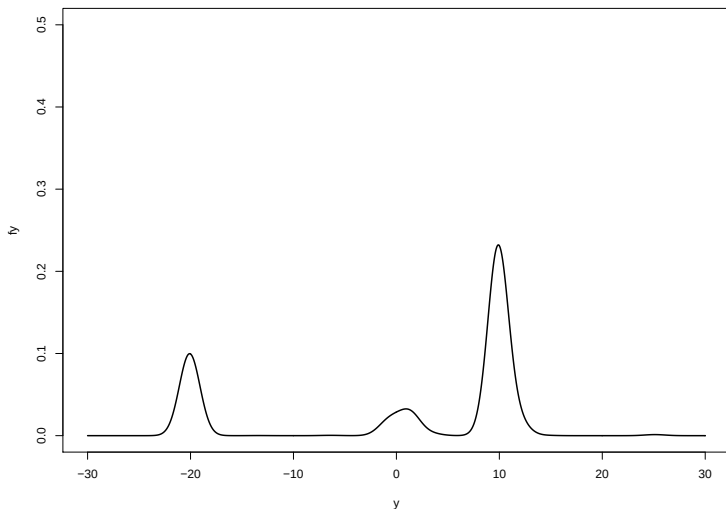
Then,

$$\tilde{f}(y) = \sum_{i=1}^n \pi_i \phi((y - x_i)/\sigma)$$

that is, an **infinite mixture model**.

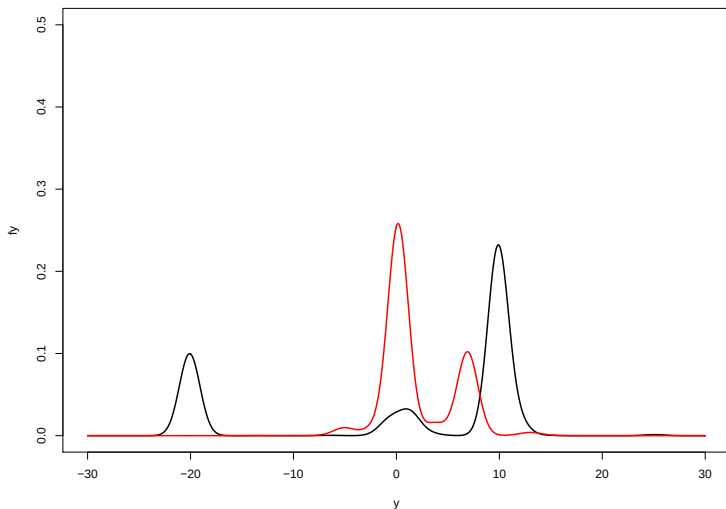
Example: $\alpha = 2, \sigma = 1$

Simulation 1



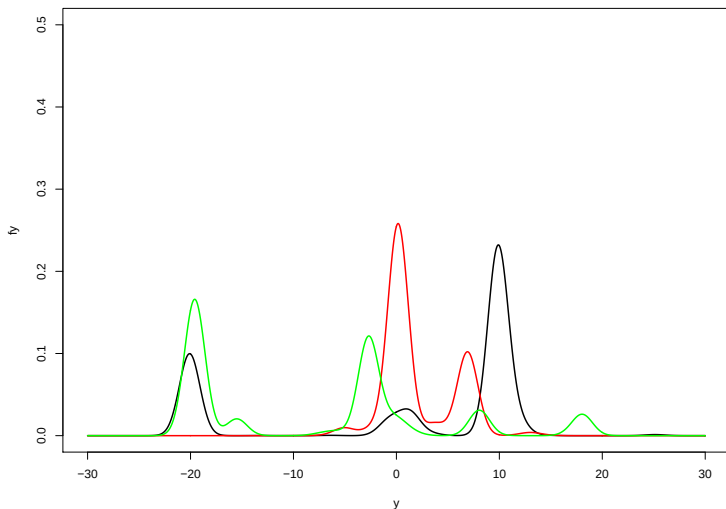
Example: $\alpha = 2, \sigma = 1$

Simulation 2



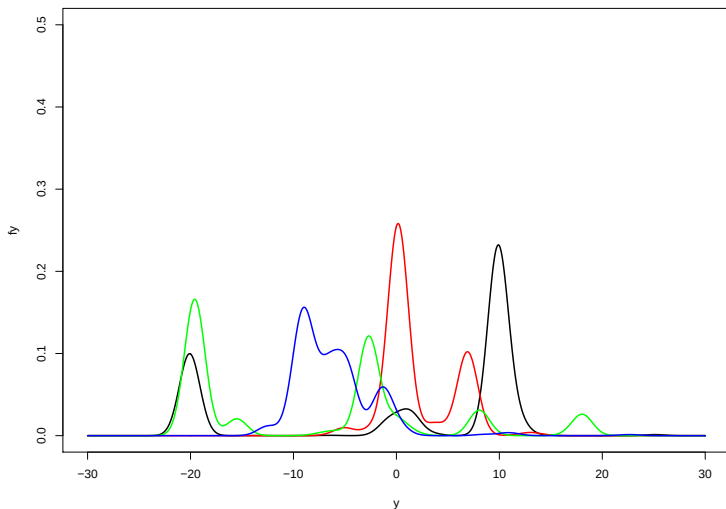
Example: $\alpha = 2, \sigma = 1$

Simulation 3



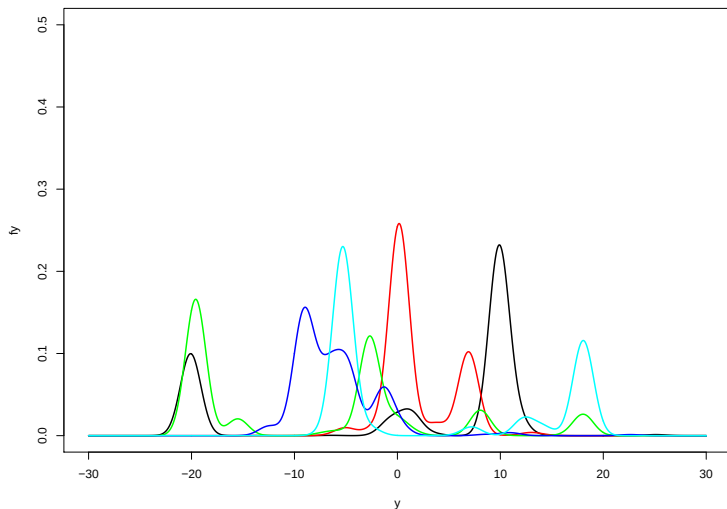
Example: $\alpha = 2, \sigma = 1$

Simulation 4



Example: $\alpha = 2, \sigma = 1$

Simulation 5



Dirichlet Process Mixture

This construction is called the

Dirichlet Process Mixture (DPM)

with a Normal kernel. Any continuous kernel g_Y can be used in place of ϕ .

Under this model, $\tilde{f}$ is almost surely **continuous**.

- α small implies less bumpy
- α large implies more bumpy.

Bayesian Inference

Suppose we have the usual de Finetti construction

- ▶ a prior model for f that is $DPM(\alpha, G_X, g_Y; \theta)$ where θ represents the parameters that appear in G_X and g_Y .
- ▶ conditional on f , data $y_1, y_2, \dots, y_n \sim f$

We wish to compute the posterior for f . We use the hierarchical formulation

$$\begin{aligned}y_j | x_j &\stackrel{\text{ind.}}{\sim} g_Y(y_j | x_j, \theta) & j = 1, \dots, n \\x_1, \dots, x_n &\sim DP(\alpha, G_X; \theta) \\ \theta &\sim p(\theta).\end{aligned}$$

Bayesian Inference: Algorithm 1

The latent variables $x_1, \dots, x_n$ are also treated as parameters. They can be sampled using an MCMC **Gibbs sampler** scheme.

For $j = 1, \dots, n$, we sample

$$x_j \mid \mathbf{x}_{(j)}, \mathbf{y} \sim w_0 p(x_j | y_j) + \sum_{l \neq j} w_l \delta_{x_l}$$

where

- ▶ $\mathbf{y} = (y_1, \dots, y_n)^\top$
- ▶ $\mathbf{x}_{(j)} = (x_1, \dots, x_{j-1}, x_{j+1}, \dots, x_n)^\top$.
- ▶ w_0 is proportional to α times the **prior predictive** of y_j
- ▶ w_l is proportional to the **likelihood** of $y_j | x_l$
- ▶ $p(x_j | y_j)$ is the **posterior** for x_j given y_j .

Bayesian Inference: Algorithm 2

To be more efficient, we can use the **clustering** property: suppose that at a given iteration of the MCMC, there are K clusters labelled 1 to K , where $K \leq n$.

Label the K distinct x values

$$z_1, \dots, z_K$$

and for each j , define the corresponding cluster label c_j where

$$c_j = k \iff x_j = z_k$$

We can update the c_j s instead of the x_j s which will be more computationally efficient; we are clustering x s to the **cluster centres** at the z values.

Bayesian Inference: Algorithm 2

For $i = 1, \dots, n$, let

- ▶ $n_1(i), \dots, n_K(i)$ denote the number of items in clusters $1, \dots, K$
- ▶ $\mathbf{y}_1(i), \dots, \mathbf{y}_K(i)$ denote the vectors of y values currently allocated to the K clusters

if the i th data point is removed.

For $j = 1, \dots, n$, we sample the cluster labels in a Gibbs sampler with conditional probabilities

$$\Pr[c_j = k \mid c_{(j)}] \propto \frac{n_j(i)}{n - 1 + \alpha} p(y_i \mid \mathbf{y}_k(i)) \quad k = 1, \dots, K$$

and

$$\Pr[c_j = K + 1 \mid c_{(j)}] \propto \frac{\alpha}{n - 1 + \alpha} p(y_i)$$

Bayesian Inference: Algorithm 2

In this expression

- $p(y_i | \mathbf{y}_k(i))$ is the **posterior predictive** density for y_i in the DPM model, assuming that y_i comes from cluster k .
- $p(y_i)$ is the **prior predictive** density for y_i in the DPM model, assuming that y_i comes from a **new cluster** not currently represented in the data.

In this formulation, we have **integrated out** the Dirichlet process.

Thus we can simply sample the cluster labels in turn, and then sample the $z_1, \dots, z_k$ values; this will allow us to do density estimation.

Bayesian Inference: Algorithm 2

By the usual calculation

$$p(y_i | \mathbf{y}_k(i)) = \int g_Y(y_i | x) p(x | \mathbf{y}_k(i)) dx$$

where

$$\begin{aligned} p(x | \mathbf{y}_k(i)) &\propto p(\mathbf{y}_k(i) | x) p(x) \\ &= \left\{ \prod_{l \neq i} g_Y(y_l | x) \right\} p(x) \end{aligned}$$

gives the posterior distribution for the k th cluster centre.

Similarly

$$p(y_i) = \int g_Y(y_i | x) p(x) dx$$

Bayesian Inference: Algorithm 2

In the earlier Gaussian model, suppose for simplicity that G_X is the $N(0, \lambda^2)$ density:

$$\begin{aligned}x_i &\sim N(0, \lambda^2) \\ y_i | x_i &\sim N(x_i, \sigma^2)\end{aligned}$$

Then

$$p(y_i | \mathbf{y}_k(i)) \equiv N \left(\frac{n_k(i) \bar{y}_k(i)}{n_k(i)/\sigma^2 + 1/\lambda^2}, \frac{(n_k(i) + 1)/\sigma^2 + 1/\lambda^2}{n_k(i)/\sigma^2 + 1/\lambda^2} \sigma^2 \right)$$

and

$$p(y_i) \equiv N(0, \sigma^2 + \lambda^2)$$

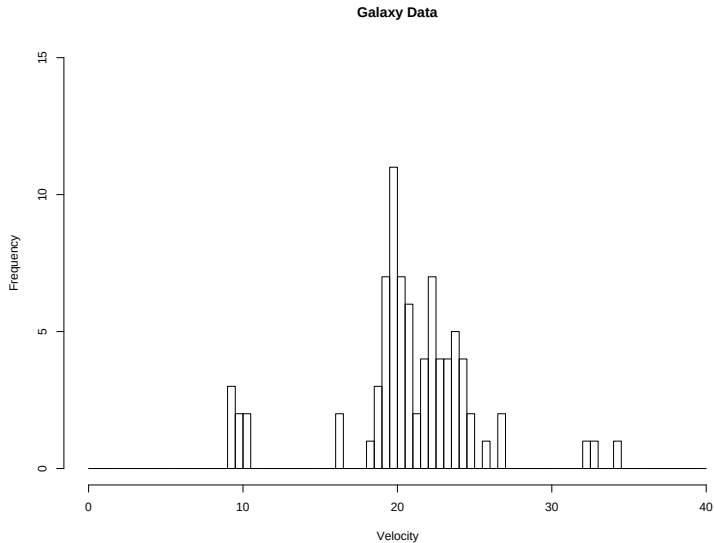
Extensions

Easy to extend to

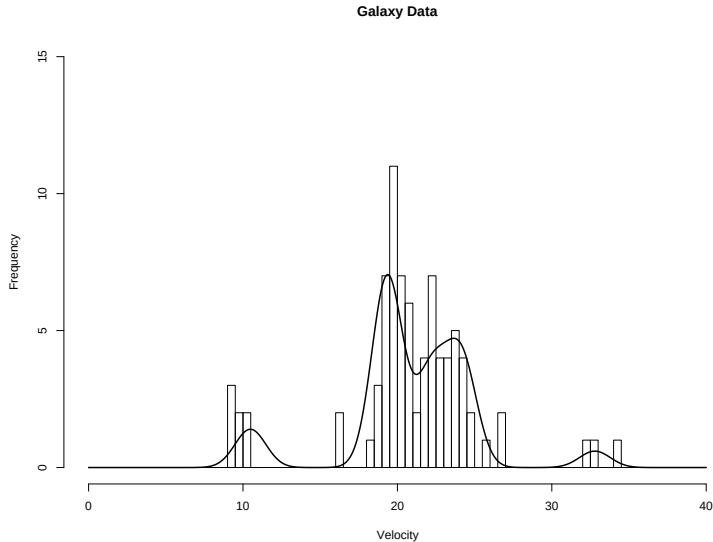
- unknown σ^2
- non-Gaussian conjugate models
- blocked Gibbs sampler
- Metropolis-Hastings MCMC for cluster labels
- multivariate conjugate models

Not so easy to do non-conjugate models.

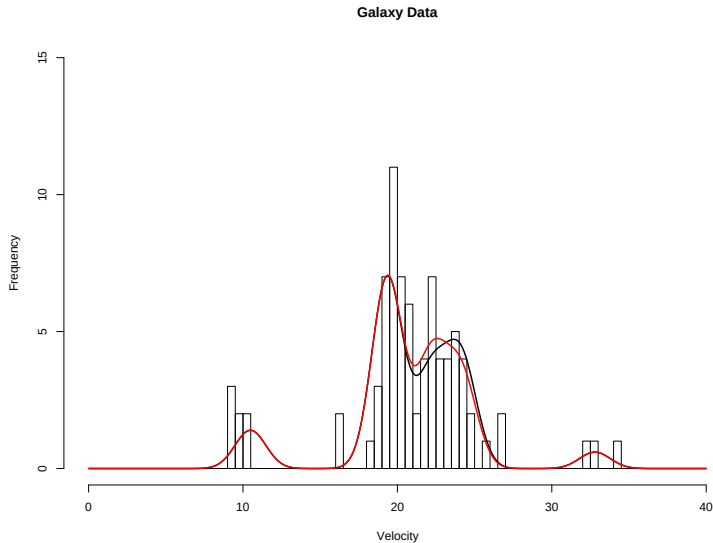
Example: Galaxy Data



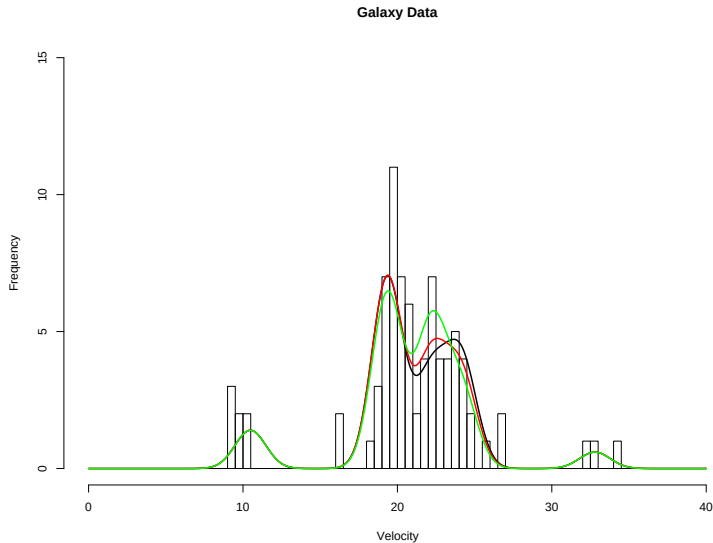
Example: Galaxy Data



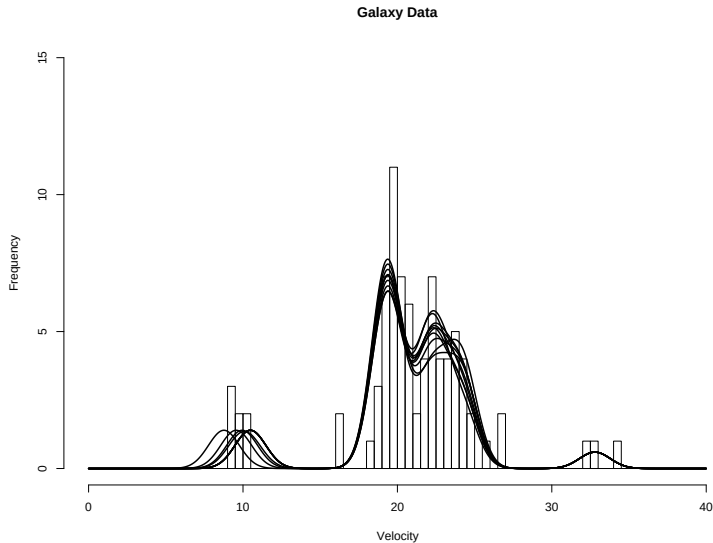
Example: Galaxy Data



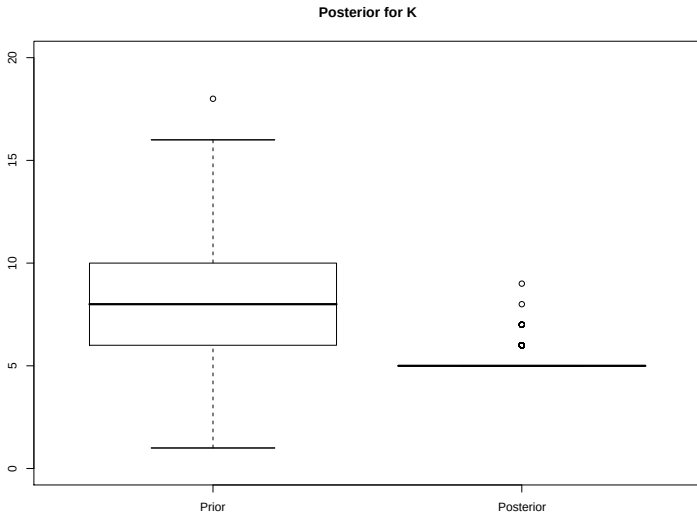
Example: Galaxy Data



Example: Galaxy Data



Example: Galaxy Data



Extensions & Open Problems

- Extensions :
 - Polya Tree Models
 - Hypothesis Testing
 - Spatial Problems
 - Normalized Random Measures
 - Connections with Lévy Processes
- Open Problems:
 - Properties of Estimators (consistency etc.)
 - Model Selection
 - Comparison with **Bayesian Semi-Parametrics**