

# Discovering signals in fMRI data; a Bayesian nonparametric approach

Ahmed Bou-Rabee, Wanrong Zhu, Zheng Xu, Mo Zhou

STAT 30850  
University of Chicago

March 14, 2017

## Project Goal

- ▶ Formulate a method which can adaptively identify clusters of signals in functional magnetic resonance imaging (fMRI) data.
- ▶ Evaluate the proposed method by drawing comparison between it and the existing p-filter algorithm.

## What is fMRI data?

- ▶ fMRI data measures the change in brain blood flow associated with mental activity [HSM04].
- ▶ fMRI data is in the form (voxel, time, intensity of reading).
- ▶ Example: To identify regions of the brain associated with hunger, fMRI readings can be taken while hungry subjects are shown pictures of food.
- ▶ Multiple comparison problem due to hundreds of thousands of voxels
- ▶ Identify significant clusters (not just individual voxels)

## What's our method

- ▶ Inspired by Stephens (2000), we describe a bayesian nonparametric method by creating a Markov birth-death process with stationary distribution to detect clusters of signals.
- ▶ View each cluster as a point in parameter space.
- ▶ Posterior distribution of the parameters being stationary distribution.
- ▶ Theoretically, this method works for multiple-dimensional data which incorporates spatial and temporal information.

## Details of the Method: Priors

- ▶ number of signal clusters:  $k \sim \text{Truncated Poisson}(\lambda, 1, k_{max})$ .
- ▶ signal centers:  $c_j \sim U(\mathcal{D})$  for  $j = 1, \dots, k$ .
- ▶ signal radius:  
 $r_j \sim \text{Truncated Normal}(\mu, \sigma, r_{min}, r_{max})$  for  $j = 1, \dots, k$ .
- ▶ signal strength:  $\beta_j \sim U(\beta_{min}, \beta_{max})$  for  $j = 1, \dots, k$ .
- ▶ p-values in signal clusters:  $p_i \sim \text{Beta}(\frac{1}{\beta_j}, \beta_j)$ , when  $x_i$  is in cluster  $j$ .
- ▶ p-values not in signal clusters:  $p_i \sim U(0, 1)$ .

## Details of the Method (continued): inventing the chain

- ▶ Birth: generating a new cluster.
- ▶ Death: "killing" an existing cluster.
- ▶ Birth rate: constant  $\lambda$  is pre-defined and independent of clusters.
- ▶ Death rate:  $\mu_i$  depends on "current" clusters and is updated each step.
- ▶ Flip a weighted coin to decide birth (w/ prob  $\frac{\lambda}{\lambda + \mu_i}$ ) or death (w/ prob  $\frac{\mu_i}{\lambda + \mu_i}$ ).

## Details of the Method (continued): death rate calculation using likelihoods

- ▶  $K$  clusters with prior  $Beta(\frac{1}{\beta_j}, \beta_j)$  for  $j = 1, 2, \dots, K$ .  $K$  itself is random with prior  $F_K$ .
- ▶ Label specify which cluster each data point belongs.
- ▶ Current cluster likelihood:  $l = \log L(\text{data} | Beta(\frac{1}{\beta_j}, \beta_j)'s, \text{labels});$   
 $c = \log L(K | F_K)$
- ▶ Cluster likelihood after "killing" cluster  $j$ :  
 $l_{-j} = \log L(\text{data} | Beta(\frac{1}{\beta_j}, \beta_j)'s, \text{labels}_{-j});$   $c_{-j} = \log L(K - 1 | F_K)$
- ▶  $u_j = \log(\lambda) + (l_{-j} - l) + (c_{-j} - \log(K) - c)$  for  $j = 1, 2, \dots, K$ .
- ▶  $u = \sum_{j=1}^K e^{u_j}.$

## Details of the Method (continued)

- ▶ At the end of each step, run metropolis-hasting algorithm to sample from the posterior of the beta distribution
- ▶ Purpose: TODO



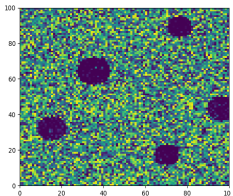
## Details of the Method (continued)

- ▶ Run the chain long enough before starting collect sample labels.
- ▶ Sample labels from evenly space grid along the chain to avoid autocorrelation.
- ▶ Average over sample labels to determine if it is signal or null.

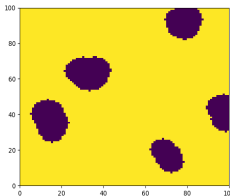
## Toy Data: Preliminaries

- ▶ 100-by-100 grid with  $k=5$  clusters of signals and the rest is null.
- ▶ The centers  $C_k \sim$  uniform from the grid while being distinct for  $k = 1, 2, \dots, 5$
- ▶ The radius  $R_k \sim TN(7, 2, 5, 10)$  for  $k = 1, 2, \dots, 5$
- ▶ Signals in clusters  $p_{ki} \sim Beta(1, \beta_k)$  where  $\beta_k \sim TN(8, 5, 2, 200)$  for  $k = 1, 2, \dots, 5$

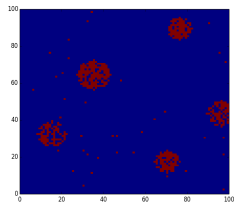
## Toy Data (continued): Performance



(a) Actual grid



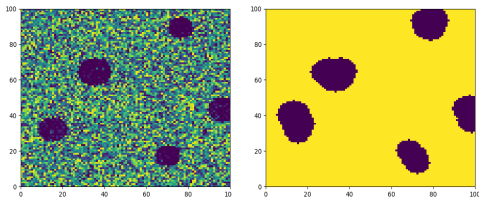
(b) Birth death chain



(c) p-filter

Figure: Simulated data

## Toy Data (continued): More on Priors



(a) Actual grid

(b)  $k \sim$

$$Trun(Poi(50), 1, 100);$$

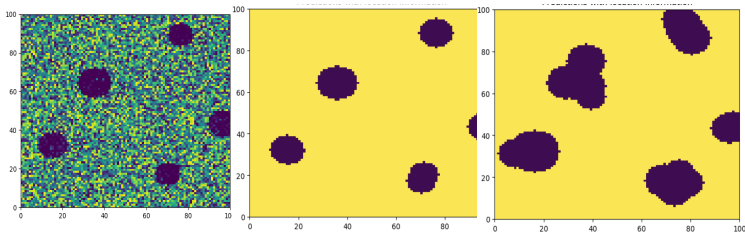
$$r \sim TN(7, 2, 5, 10);$$

$$\beta \sim TN(8, 5, 2, 200)$$

Figure: Simulated data

## Toy Data (continued): More on Priors

First, let's compare different priors on beta



(a) Actual grid

(b)  $k \sim$

$Trun(Poi(50), 1, 100);$   
 $r \sim TN(7, 2, 5, 10);$   
 $\beta \sim TN(30, 3, 2, 200)$

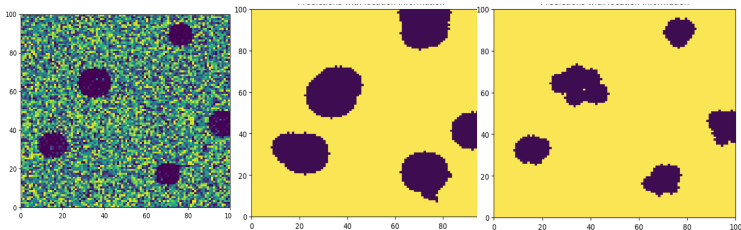
(c)  $k \sim$

$Trun(Poi(50), 1, 100);$   
 $r \sim TN(7, 2, 5, 10);$   
 $\beta \sim TN(2, 1, 2, 200)$

Figure: Simulated data

## Toy Data (continued): More on Priors

Now, let's compare different priors on radius.



(a) Actual grid

(b)  $k \sim$

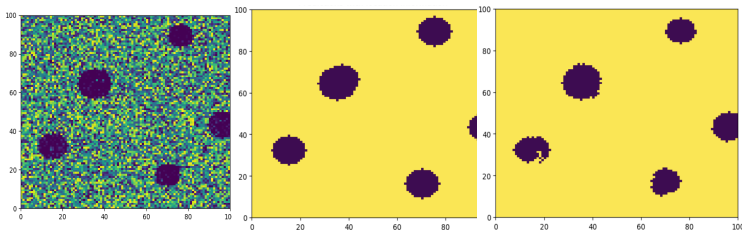
$Trun(Poi(50), 1, 100)$ ;  $Trun(Poi(50), 1, 100)$ ;  
 $r \sim TN(10, 2, 7, 13)$ ;  $r \sim TN(2, 2, 1, 10)$ ;  
 $\beta \sim TN(8, 5, 2, 200)$   $\beta \sim TN(8, 5, 2, 200)$

(c)  $k \sim$

Figure: Simulated data

## Toy Data (continued): More on Priors

What if we make signal stronger?



(a) Actual grid

(b)  $k \sim$

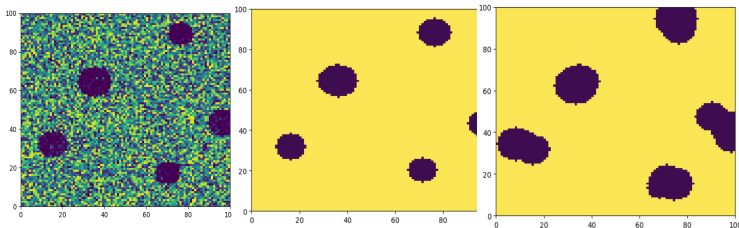
(c)  $k \sim$

$Trun(Poi(50), 1, 100); Trun(Poi(50), 1, 100);$   
 $r \sim TN(10, 2, 7, 13); \quad r \sim TN(2, 2, 1, 10);$   
 $\beta \sim TN(30, 3, 2, 200) \quad \beta \sim TN(30, 3, 2, 200)$

Figure: Simulated data

## Toy Data (continued): More on Priors

Next, we change the priors on number of clusters.



(a) Actual grid

(b)  $k \sim$

$Trun(Poi(3), 1, 10);$

$r \sim TN(7, 2, 5, 10);$

$\beta \sim TN(8, 5, 2, 200)$

(c)  $k \sim$

$Trun(Poi(300), 1, 1000);$

$r \sim TN(7, 2, 5, 10);$

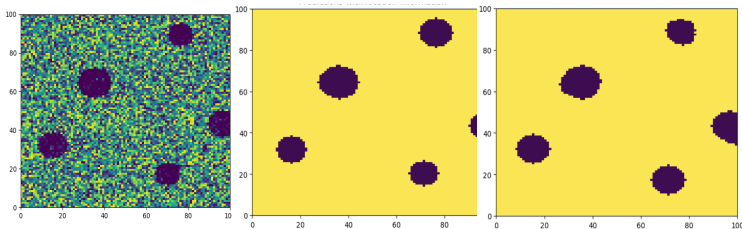
$\beta \sim TN(8, 5, 2, 200)$

Figure: Simulated data



## Toy Data (continued): More on Priors

Again, let's make signal stronger



(a) Actual grid

(b)  $k \sim$

$\text{Trun}(\text{Poi}(3), 1, 10);$

$r \sim \text{TN}(7, 2, 5, 10);$

$\beta \sim \text{TN}(30, 3, 2, 200)$

(c)  $k \sim$

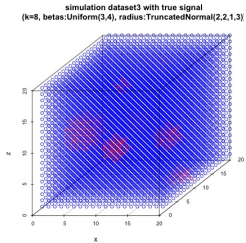
$\text{Trun}(\text{Poi}(300), 1, 1000);$

$r \sim \text{TN}(7, 2, 5, 10);$

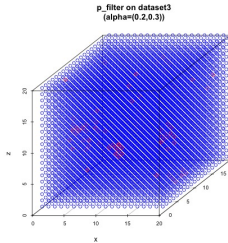
$\beta \sim \text{TN}(30, 3, 2, 200)$

Figure: Simulated data

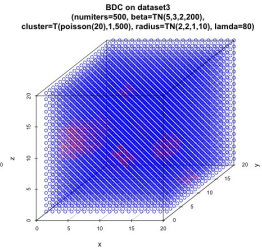
## Toy Data 2 (3-D)



(a)



(b)



(c)

Figure: Simulated data

## Toy Data 2 (3-D)

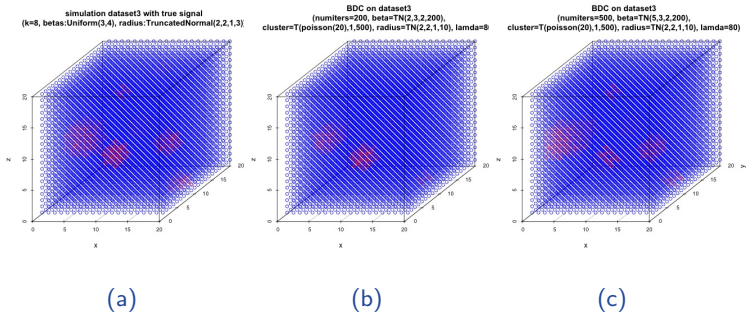


Figure: Simulated data

## Toy Data 2 (3-D)

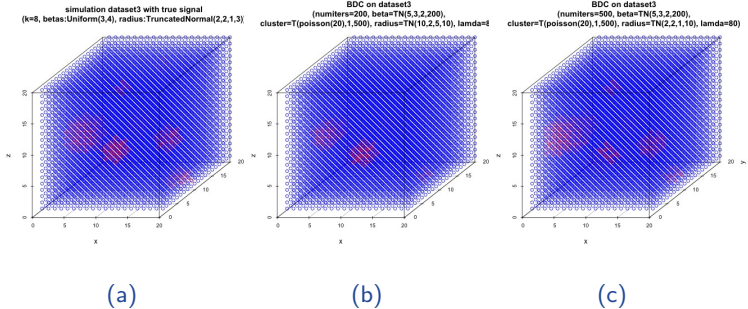


Figure: Simulated data

## Toy Data 2 (3-D)

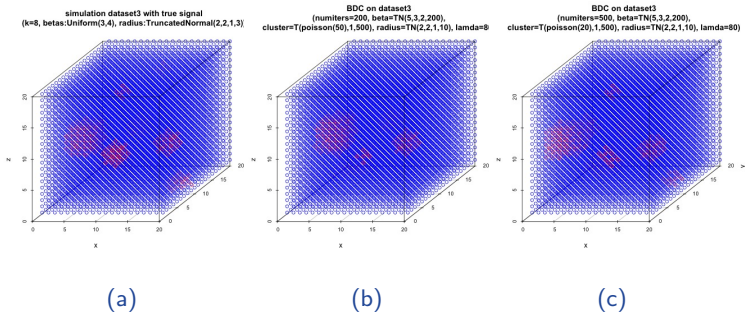
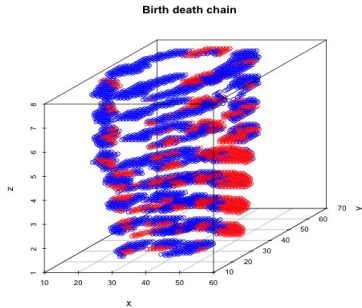
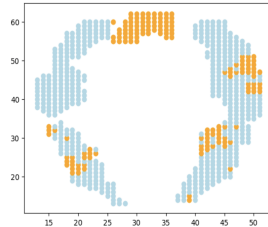


Figure: Simulated data

## Performance on real fMRI data



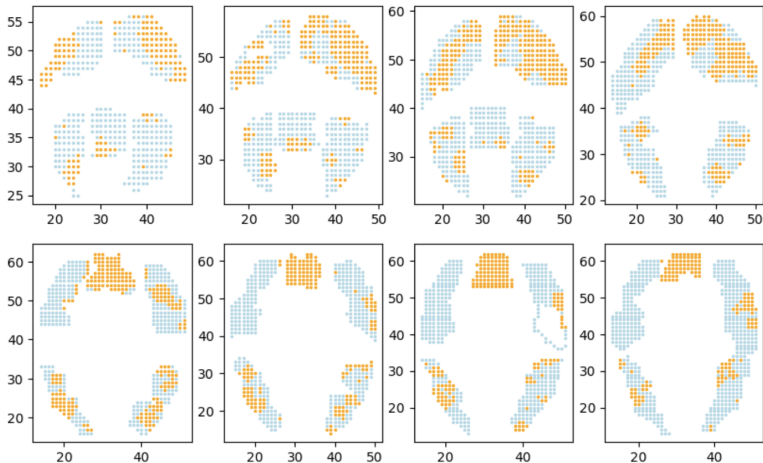
(a) 3-D view



(b) 2-D view: one slice

Figure: fMRI data

## Performance on real fMRI data (continued)



## Performance on real fMRI data (continued): Comparison to p-filter

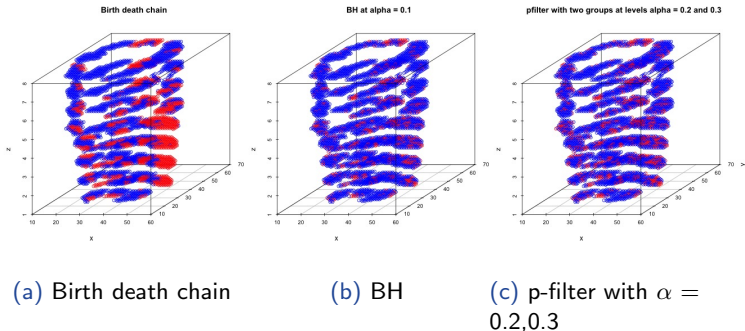


Figure: fMRI data



## Conclusion and Future Work

- ▶ Formulated and tested a nonparametric bayesian method to adaptively identify clusters of signals.
- ▶ Showed promising results on both simulation and real fMRI data.
- ▶ Extend from p-values to intensities directly by specifying appropriate priors for null distributions and for signal distributions.
- ▶ Put priors on the hyper-parameters and maximize this priors using EM. That is uniform prior over hyper-parameters.

**Thanks!**