Bayesian Distance Clustering

David Dunson (with Leo Duan)

Departments of Statistical Science & Mathematics
Duke University
dunson@duke.edu

Motivation

Survey on Clustering Methods

Clustering

- One of the main tools for unsupervised statistical analysis

Survey on Clustering Methods

Clustering

- One of the main tools for unsupervised statistical analysis
- Often 1st step to simplify complex data — dividing them into small & homogeneous groups.

Survey on Clustering Methods

Clustering

- One of the main tools for unsupervised statistical analysis
- Often 1st step to simplify complex data — dividing them into small & homogeneous groups.

Survey on Clustering Methods

Clustering

- One of the main tools for unsupervised statistical analysis
- Often 1st step to simplify complex data — dividing them into small & homogeneous groups.

Great uncertainty exists in clustering

- Clusters tend to overlap.

Survey on Clustering Methods

Clustering

- One of the main tools for unsupervised statistical analysis
- Often 1st step to simplify complex data — dividing them into small & homogeneous groups.

Great uncertainty exists in clustering

- Clusters tend to overlap.
- Generative models via mixture likelihood are extremely useful & popular. Consider $y_i \in \mathcal{Y} \subseteq \mathbb{R}^p$

$$y_i \overset{iid}{\sim} f, \quad f(y) = \sum_{h=1}^{k} \pi_h \mathcal{K}(y; \theta_h).$$

Equivalently, there is a latent clustering label $c_i \in \{1, \ldots, k\}$

$$y_i \sim \mathcal{K}(\theta_{c_i}), \quad \text{pr}(c_i = h) = \pi_h.$$

Survey on Clustering Methods

How to choose $\mathcal{K}$???

- Dilemma: balancing complexity vs flexibility.

Survey on Clustering Methods

How to choose $\mathcal{K}$???

- Dilemma: balancing complexity vs flexibility.
- Simple $\mathcal{K}$ – fewer parameters, easy computation.

Survey on Clustering Methods

How to choose $\mathcal{K}$???

- Dilemma: balancing complexity vs flexibility.
- Simple $\mathcal{K}$ – fewer parameters, easy computation.
- e.g. Gaussian mixture (GMM) and its approximation (K-means).

Survey on Clustering Methods

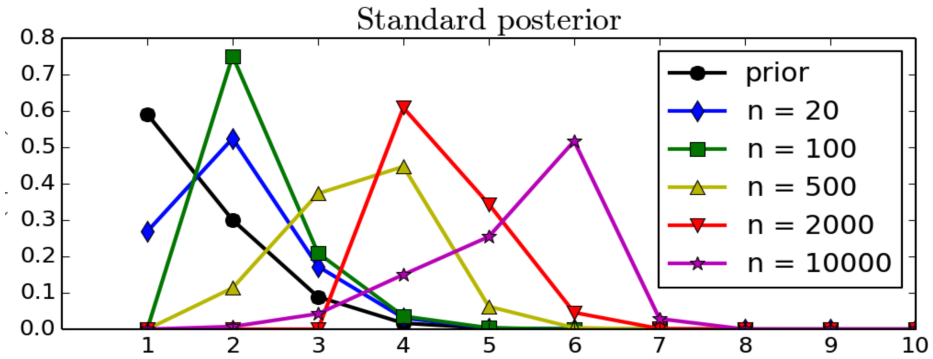
How to choose $\mathcal{K}$???

- Dilemma: balancing complexity vs flexibility.
- Simple $\mathcal{K}$ – fewer parameters, easy computation.
- e.g. Gaussian mixture (GMM) and its approximation (K-means).
- But strong assumptions lack robustness: sub-Gaussian tails, symmetry, elliptical contour.

Survey on Clustering Methods

How to choose $\mathcal{K}$???

- Dilemma: balancing complexity vs flexibility.
- Simple $\mathcal{K}$ – fewer parameters, easy computation.
- e.g. Gaussian mixture (GMM) and its approximation (K-means).
- But strong assumptions lack robustness: sub-Gaussian tails, symmetry, elliptical contour.
- Many practical problems. e.g. as n increases, slight violation to assumption yields unbounded number of clusters (Miller and Dunson 2018).



# of clusters in fitting GMM to 2 *mildly skewed* Gaussians

Survey on Clustering Methods

- More flexible $\mathcal{K}$:
  1. Skewed-Gaussian (Lin, Lee and Yen 2007)
  2. t-kernel (Peel and McLachlan 2000)
  3. Another layer of uniform mixture (Rodriguez and Walker 2014)
  4. Copula (Kosmidis and Karlis 2015)

Survey on Clustering Methods

- More flexible $\mathcal{K}$:
  1. Skewed-Gaussian (Lin, Lee and Yen 2007)
  2. t-kernel (Peel and McLachlan 2000)
  3. Another layer of uniform mixture (Rodriguez and Walker 2014)
  4. Copula (Kosmidis and Karlis 2015)
- # of parameters becomes unwieldy — easy to overfit in finite n & create intractable computation for large p.

Survey on Clustering Methods

- More flexible $\mathcal{K}$:
  1. Skewed-Gaussian (Lin, Lee and Yen 2007)
  2. t-kernel (Peel and McLachlan 2000)
  3. Another layer of uniform mixture (Rodriguez and Walker 2014)
  4. Copula (Kosmidis and Karlis 2015)
- # of parameters becomes unwieldy — easy to overfit in finite n & create intractable computation for large p.
- Imagine: How many parameters do you need to handle skewness in $y_i \in \mathbb{R}^{100}$? How do you run MCMC on those?

Survey on Clustering Methods

- Some new solutions by 'assumption weakening'
  1. Gaussian mixture with one extra component of improper uniform (Coretto and Hennig 2017)
  2. Creating a KL-divergence ball around exact Gaussian mixture likelihood (Miller and Dunson 2018)

Survey on Clustering Methods

- Some new solutions by 'assumption weakening'
  1. Gaussian mixture with one extra component of improper uniform (Coretto and Hennig 2017)
  2. Creating a KL-divergence ball around exact Gaussian mixture likelihood (Miller and Dunson 2018)
- Not clear on how to extend those to more complicated objects — e.g. clustering multiple time series in brain EEG data.

Survey on Clustering Methods

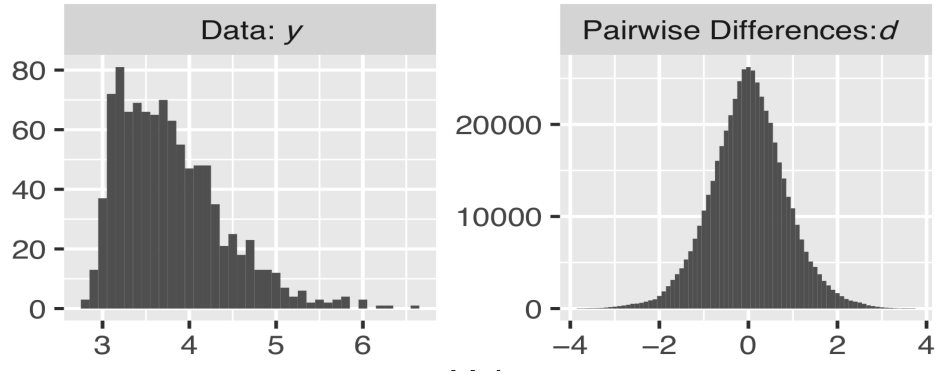
Apparently, we want some new tool that is:

- Probabilistic — critical for UQ.
- Simple — small # of parameters, even for complicated data.
- Theoretically well justified and (almost) tuning free.

Bayesian Distance Clustering

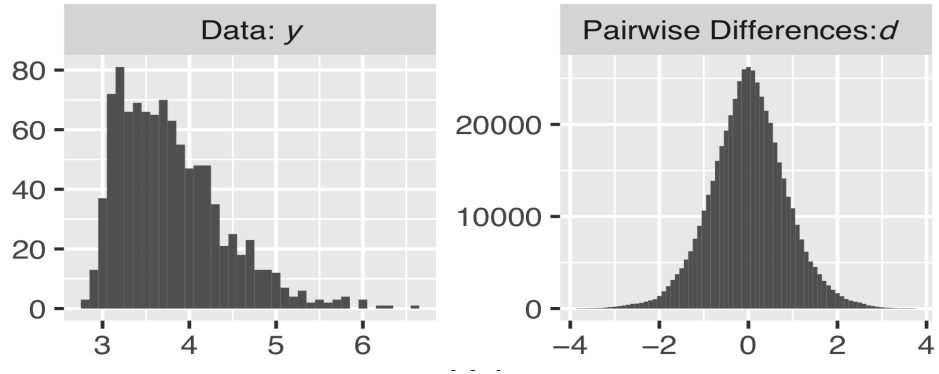
Bayesian Distance Clustering

- Why using distances? First, consider pairwise differences: $d_{i,i'} = y_i - y_i'$



Bayesian Distance Clustering

- Why using distances? First, consider pairwise differences: $d_{i,i'} = y_i - y_i'$

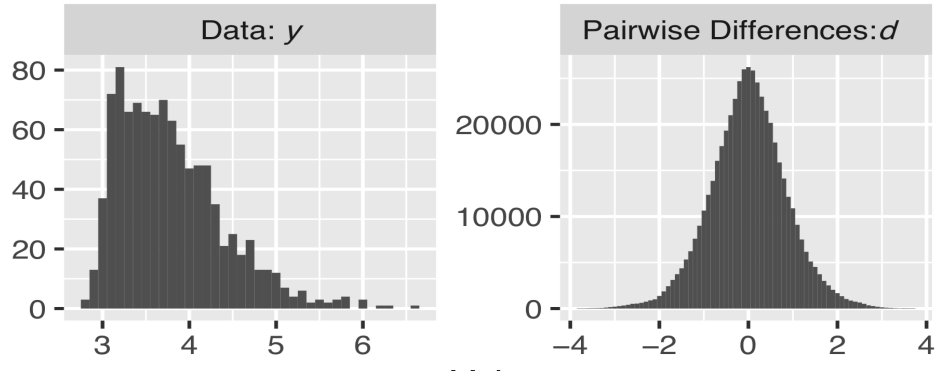


1. If y_i and y_i' are in the same cluster, due to iid:

$$\mathbb{E} d_{i,i'}^r = \vec{0} \qquad \text{for } r = 1, 3, 5, \ldots$$

Bayesian Distance Clustering

- Why using distances? First, consider pairwise differences: $d_{i,i'} = y_i - y_{i'}$



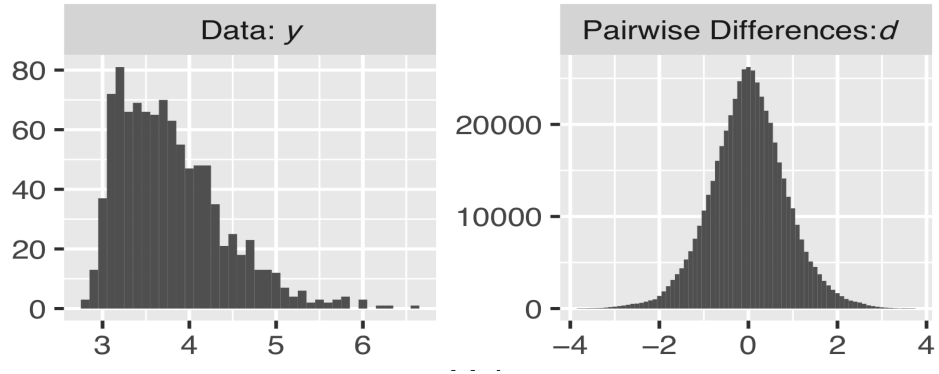
1. If y_i and $y_{i'}$ are in the same cluster, due to iid:

$$\mathbb{E} d_{i,i'}^r = \vec{0} \qquad \text{for } r = 1, 3, 5, \ldots$$

2. Unimodal at $\vec{0}$ as long as y_i is unimodal (Hodges and Lehmann 1954).

Bayesian Distance Clustering

- Why using distances? First, consider pairwise differences: $d_{i,i'} = y_i - y_i'$



1. If y_i and y_i' are in the same cluster, due to iid:

$$\mathbb{E} d_{i,i'}^r = \vec{0} \qquad \text{for } r = 1, 3, 5, \ldots$$

2. Unimodal at $\vec{0}$ as long as y_i is unimodal (Hodges and Lehmann 1954).
3. Intuitively, the within-cluster distances $\|d_{i,i'}\|$ will now concentrate at 0 (for most norms $\|.\|$).

Bayesian Distance Clustering

- $d_{i,i'}$ are much easier to model than y_i!

Bayesian Distance Clustering

- $d_{i,i'}$ are much easier to model than y_i!
- How do we obtain a 'coherent' likelihood for $d_{i,i'}$?

Bayesian Distance Clustering

- $d_{i,i'}$ are much easier to model than y_i!
- How do we obtain a 'coherent' likelihood for $d_{i,i'}$?
- Imagine an *unknown* oracle likelihood, conditioned on latent labels $c_{(n)}$

$$L^*(y_{(n)}; c_{(n)}) = \prod_{h=1}^{k} \prod_{i:c_i=h} \mathcal{K}_h(y_i) = \prod_{h=1}^{k} L_h(y^{[h]})$$

Bayesian Distance Clustering

- $d_{i,i'}$ are much easier to model than y_i!
- How do we obtain a 'coherent' likelihood for $d_{i,i'}$?
- Imagine an *unknown* oracle likelihood, conditioned on latent labels $c_{(n)}$

$$L^*(y_{(n)}; c_{(n)}) = \prod_{h=1}^{k} \prod_{i:c_i=h} \mathcal{K}_h(y_i) = \prod_{h=1}^{k} L_h(y^{[h]})$$

- Transform $L_h(y^{[h]})$ using $y_1^{[h]}$ and $(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]})$

$$L_h^*(y^{[h]}) = \mathcal{K}_h(y_1^{[h]}) \prod_{i=2}^{n_h} G_h(y_i^{[h]} - y_1^{[h]} \mid y_1^{[h]})$$
$$= \mathcal{K}_h\left(y_1^{[h]} \mid d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right) G_h\left(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right)$$

Bayesian Distance Clustering

- $d_{i,i'}$ are much easier to model than y_i!
- How do we obtain a 'coherent' likelihood for $d_{i,i'}$?
- Imagine an *unknown* oracle likelihood, conditioned on latent labels $c_{(n)}$

$$L^*(y_{(n)}; c_{(n)}) = \prod_{h=1}^{k} \prod_{i:c_i=h} \mathcal{K}_h(y_i) = \prod_{h=1}^{k} L_h(y^{[h]})$$

- Transform $L_h(y^{[h]})$ using $y_1^{[h]}$ and $(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]})$

$$\begin{aligned} L_h^*(y^{[h]}) &= \mathcal{K}_h(y_1^{[h]}) \prod_{i=2}^{n_h} G_h(y_i^{[h]} - y_1^{[h]} \mid y_1^{[h]}) \\ &= \mathcal{K}_h\left(y_1^{[h]} \mid d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right) G_h\left(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right) \end{aligned}$$

- Discard $y_1^{[h]}$ (requiring heavy assumptions) by integrating it out.

Bayesian Distance Clustering

- We get a *Distance Likelihood*

$$L(y_{(n)}; c_{(n)}) = \prod_{h=1}^{k} G_h\left(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right).$$

Bayesian Distance Clustering

- We get a *Distance Likelihood*

$$L(y_{(n)}; c_{(n)}) = \prod_{h=1}^{k} G_h\left(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right).$$

- Since we don't know the oracle, we need to specify G_h.

Bayesian Distance Clustering

- We get a *Distance Likelihood*

$$L(y_{(n)}; c_{(n)}) = \prod_{h=1}^{k} G_h\left(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right).$$

- Since we don't know the oracle, we need to specify G_h.
- This is much easier to do, since we only need to worry about covariance and tails!

Bayesian Distance Clustering

- We get a *Distance Likelihood*

$$L(y_{(n)}; c_{(n)}) = \prod_{h=1}^{k} G_h\left(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right).$$

- Since we don't know the oracle, we need to specify G_h.
- This is much easier to do, since we only need to worry about covariance and tails!
- Note $d_{i,i'}^{[h]} = d_{i,1}^{[h]} - d_{i',1}^{[h]}$, we propose an over-complete form as the simple product:

$$\begin{aligned} G_h\left(d_{2,1}^{[h]}, \ldots, d_{n_h,1}^{[h]}\right) &= \prod_{i=2}^{n_h} g_h^{\alpha_h}(d_{i,1}^{[h]}) \Big\{ \prod_{i=3}^{n_h} \prod_{1<i'<i} g_h^{\alpha_h}(d_{i,i'}^{[h]}) \Big\}, \\ &= \prod_{i=2}^{n_h} \prod_{i'<i} g_h^{\alpha_h}(d_{i,i'}^{[h]}), \end{aligned}$$

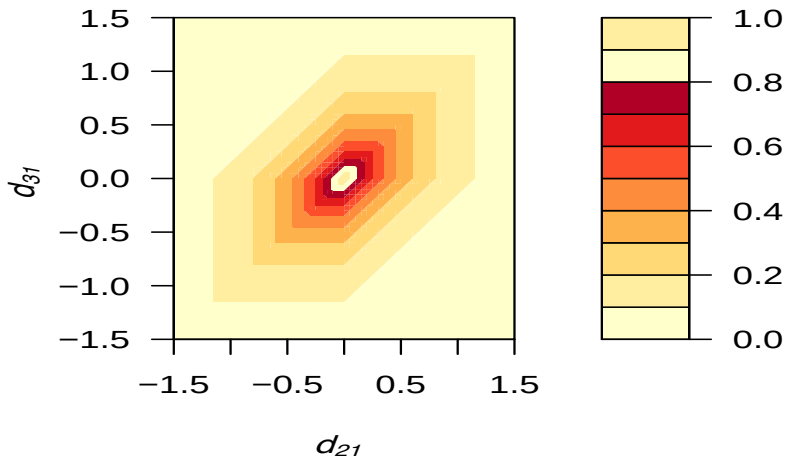
g_h: marginal density; α_h: calibration parameter (to be set later).

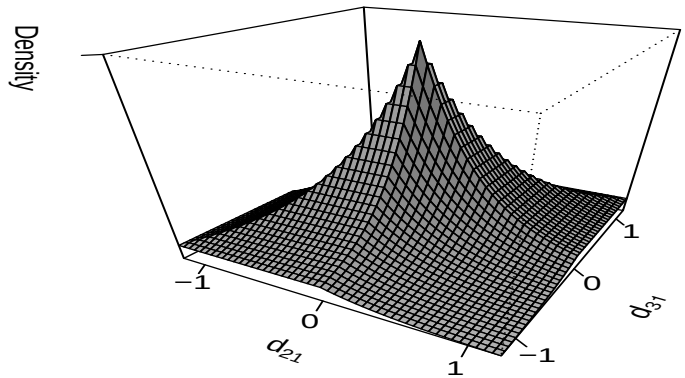
Bayesian Distance Clustering

Due to linear constraint $d_{i,i'}^{[h]} = d_{i,1}^{[h]} - d_{i',1}^{[h]}$, covariances among $d_{i,i'}^{[h]}$'s are automatically induced:

Lemma

$$\text{var}(d_{i,i'}^{[h]}) = 2\Sigma, \ \text{cov}(d_{i,i'}^{[h]}, d_{i,i''}^{[h]}) = \Sigma \ \ \textit{for } i' \neq i'', i \neq i', i \neq i''.$$





Likelihood for 3 distances $G_h(D^{[h]}) = \exp(-|d_{21}|)\exp(-|d_{31}|)\exp(-|d_{32}|)$.

Bayesian Distance Clustering

Product form makes it easy to satisfy **coherence** conditions:

Lemma

(Exchangeability)

$$L(y_{(n)}; c_{(n)}) = L(y_{(n^*)}; c_{(n^*)})$$

with $(n^*) = \{1_*, \ldots, n_*\}$ *denoting a set of permuted indices.*

Lemma

(Marginalization)

$$L(y_{(n)}) = \int_{\mathcal{Y}} L(y_{(n+1)}) \mathrm{d}y_{n+1}.$$

Bayesian Distance Clustering

- Choosing marginal $g_h(d_{i,i'}^{[h]})$ is straightforward, mostly based on tail assumption and how to handle the covariance of y_i.

Bayesian Distance Clustering

- Choosing marginal $g_h(d_{i,i'}^{[h]})$ is straightforward, mostly based on tail assumption and how to handle the covariance of y_i.
- How $d_{i,i'}^{[h]}$ enters g_h also determines what type of distance is used.

Bayesian Distance Clustering

- Choosing marginal $g_h(d_{i,i'}^{[h]})$ is straightforward, mostly based on tail assumption and how to handle the covariance of y_i.
- How $d_{i,i'}^{[h]}$ enters g_h also determines what type of distance is used.
- For example, a multivariate Gaussian density

$$g_h(d_{i,i'}^{[h]}) \propto \exp\left(-\frac{1}{2}\sigma_h^{-1} {d_{i,i'}^{[h]}}^{\mathrm{T}} S^{-1} d_{i,i'}^{[h]} \right)$$

corresponds to Mahalanobis distance
$\Delta(y_i^{[h]}, y_{i'}^{[h]}) = (y_i^{[h]} - y_{i'}^{[h]})^{\mathrm{T}} S^{-1} (y_i^{[h]} - y_{i'}^{[h]})$.

Bayesian Distance Clustering

- Choosing marginal $g_h(d_{i,i'}^{[h]})$ is straightforward, mostly based on tail assumption and how to handle the covariance of y_i.
- How $d_{i,i'}^{[h]}$ enters g_h also determines what type of distance is used.
- For example, a multivariate Gaussian density

$$g_h(d_{i,i'}^{[h]}) \propto \exp\left(-\frac{1}{2}\sigma_h^{-1} d_{i,i'}^{[h]\mathrm{T}} S^{-1} d_{i,i'}^{[h]} \right)$$

corresponds to Mahalanobis distance
$\Delta(y_i^{[h]}, y_{i'}^{[h]}) = (y_i^{[h]} - y_{i'}^{[h]})^{\mathrm{T}} S^{-1} (y_i^{[h]} - y_{i'}^{[h]})$.

- For simplicity, we will assume diagonal covariance for y_i and focus on guarantee on tail robustness.

Bayesian Distance Clustering

- Choosing marginal $g_h(d_{i,i'}^{[h]})$ is straightforward, mostly based on tail assumption and how to handle the covariance of y_i.
- How $d_{i,i'}^{[h]}$ enters g_h also determines what type of distance is used.
- For example, a multivariate Gaussian density

$$g_h(d_{i,i'}^{[h]}) \propto \exp\left(-\frac{1}{2}\sigma_h^{-1} d_{i,i'}^{[h]\mathrm{T}} S^{-1} d_{i,i'}^{[h]} \right)$$

  corresponds to Mahalanobis distance $\Delta(y_i^{[h]}, y_{i'}^{[h]}) = (y_i^{[h]} - y_{i'}^{[h]})^{\mathrm{T}} S^{-1} (y_i^{[h]} - y_{i'}^{[h]})$.
- For simplicity, we will assume diagonal covariance for y_i and focus on guarantee on tail robustness.
- Computation is simple via Gibbs sampling.

Theory: A Deeper Dive

Tail bound on pairwise distances

Chernoff bound on distances

Lemma

(Concentration inequality) If all $y_i^{[h]}$*'s are sub-exponential with bound parameters* (ν_h, b_h)*, then* $\|d_{i,i'}^{[h]}\|_\infty = \max_{j=1}^{p} |d_{i,i',j}^{[h]}|$ *has*

$$\text{pr}(\|d_{i,i'}^{[h]}\|_\infty > t) \leq 2p \exp\{-t/(2b_h)\} \quad \text{for } t > 2\nu_h^2/b_h.$$

- For p not too large, $d_{i,i'}^{[h]}$ can be well approximated by a Laplace.

Tail bound on pairwise distances

Chernoff bound on distances

Lemma

(Concentration inequality) If all $y_i^{[h]}$'s are sub-exponential with bound parameters (ν_h, b_h), then $\|d_{i,i'}^{[h]}\|_\infty = \max_{j=1}^p |d_{i,i',j}^{[h]}|$ has

$$\text{pr}(\|d_{i,i'}^{[h]}\|_\infty > t) \leq 2p \exp\{-t/(2b_h)\} \quad \text{for } t > 2\nu_h^2/b_h.$$

- For p not too large, $d_{i,i'}^{[h]}$ can be well approximated by a Laplace.
- p-dimensional Laplace with diagonal covariance is equivalent to using weighted-ℓ_1 distances

$$g_h(d_{i,i'}^{[h]}) = (2^{-p} \prod_{j=1}^p \sigma_{h,j}^{-1}) \exp\Big(-\sum_{j=1}^p |d_{i,i',j}^{[h]}|/\sigma_{h,j}\Big).$$

Information Theory on Clustering

One natural question: what we lose when discarding information in $\mathcal{K}(y_1^{[h]} \mid .)$?

- A relevant information concept: Bregman divergence (Bregman 1967) between two random variables x and y

$$B_\phi(x, y) = \phi(x) - \phi(y) - (x - y)' \nabla \phi(y)$$

$\phi : \mathbf{dom}\ \phi \to \mathbb{R}$ a strictly convex and differentiable function; $\nabla \phi(y)$ the gradient of ϕ at y.

Information Theory on Clustering

One natural question: what we lose when discarding information in $\mathcal{K}(y_1^{[h]} \mid .)$?

- A relevant information concept: Bregman divergence (Bregman 1967) between two random variables x and y

$$B_\phi(x, y) = \phi(x) - \phi(y) - (x - y)' \nabla \phi(y)$$

ϕ : **dom** $\phi \to \mathbb{R}$ a strictly convex and differentiable function; $\nabla \phi(y)$ the gradient of ϕ at y.

- Generality: For any regular exponential $\mathcal{K}_h$, there always exists a Bregman re-parameterization [Banerjee et al (2005)]

$$\mathcal{K}_h(y_i; \theta_h) = \exp\{T(y_i)'\theta_h - \psi(\theta_h)\}\, \kappa(y_i)$$
$$\Leftrightarrow \exp\left[-B_\phi\{T(y_i), \mu_h\}\right] b_\phi\{T(y_i)\},$$

T: minimal sufficient statistics for θ_h; $\mu_h = \mathbb{E}_{y_i \sim \mathcal{K}_h} T(y_i)$.

Information Theory on Clustering

- For model-based clustering, maximizing mixture likelihood $\Leftrightarrow$ minimizing total Bregman divergence wrt the mean of $T(y_i)$:

$$H_y = \sum_{h=1}^{k} H_y^{[h]}, \quad H_y^{[h]} = \sum_{i=1}^{n_h} B_\phi \left\{ T(y_i), \mu_h \right\}.$$

Information Theory on Clustering

- For model-based clustering, maximizing mixture likelihood $\Leftrightarrow$ minimizing total Bregman divergence wrt the mean of $T(y_i)$:

$$H_y = \sum_{h=1}^{k} H_y^{[h]}, \quad H_y^{[h]} = \sum_{i=1}^{n_h} B_\phi \left\{ T(y_i), \mu_h \right\}.$$

- For distance clustering, now view each distance as some form of pairwise Bregman divergence

$$H_d = \sum_{h=1}^{k} H_d^{[h]}, \quad H_d^{[h]} = \alpha_h \lambda_h^{-1} \sum_{i=1}^{n_h} \sum_{i'=1}^{n_h} B_\phi \left\{ T(y_i^{[h]}), T(y_{i'}^{[h]}) \right\}.$$

Information Theory on Clustering

- For model-based clustering, maximizing mixture likelihood $\Leftrightarrow$ minimizing total Bregman divergence wrt the mean of $T(y_i)$:

$$H_y = \sum_{h=1}^{k} H_y^{[h]}, \quad H_y^{[h]} = \sum_{i=1}^{n_h} B_\phi \left\{ T(y_i), \mu_h \right\}.$$

- For distance clustering, now view each distance as some form of pairwise Bregman divergence

$$H_d = \sum_{h=1}^{k} H_d^{[h]}, \quad H_d^{[h]} = \alpha_h \lambda_h^{-1} \sum_{i=1}^{n_h} \sum_{i'=1}^{n_h} B_\phi \left\{ T(y_i^{[h]}), T(y_{i'}^{[h]}) \right\}.$$

- How do they compare?

Information Theory on Clustering

Lemma

(Expected Bregman Divergences) The model-based and distance-based Bregman divergences have this relationship:

$$\mathbb{E}_{y^{[h]}} H_d^{[h]} = (2n_h\alpha_h\lambda_h^{-1})\,\mathbb{E}_{y^{[h]}}\left[\sum_{i=1}^{n_h}\frac{H_y^{[h]} + B_\phi\{\mu_h, T(y_i^{[h]})\}}{2}\right],$$

where the expectation over $y^{[h]}$ is taken with respect to $\mathcal{K}_h$.

RHS after $\mathbb{E}$: symmetrized Bregman divergence b/w $T(y_i^{[h]})$ and μ_h.

- We can make two expected divergences equal, by setting $\alpha_h = 1/n_h$ and learn λ_h adaptively from its posterior.

Information Theory on Clustering

Lemma

(Expected Bregman Divergences) The model-based and distance-based Bregman divergences have this relationship:

$$\mathbb{E}_{y^{[h]}} H_d^{[h]} = (2n_h\alpha_h\lambda_h^{-1})\,\mathbb{E}_{y^{[h]}}\left[\sum_{i=1}^{n_h}\frac{H_y^{[h]} + B_\phi\{\mu_h, T(y_i^{[h]})\}}{2}\right],$$

where the expectation over $y^{[h]}$ is taken with respect to $\mathcal{K}_h$.

RHS after $\mathbb{E}$: symmetrized Bregman divergence b/w $T(y_i^{[h]})$ and μ_h.

- We can make two expected divergences equal, by setting $\alpha_h = 1/n_h$ and learn λ_h adaptively from its posterior.
- NO Bregman information is lost for clustering!

Information Theory on Clustering

Lemma

(Expected Bregman Divergences) The model-based and distance-based Bregman divergences have this relationship:

$$\mathbb{E}_{y^{[h]}} H_d^{[h]} = (2n_h\alpha_h\lambda_h^{-1})\, \mathbb{E}_{y^{[h]}} \left[\sum_{i=1}^{n_h} \frac{H_y^{[h]} + B_\phi\{\mu_h, T(y_i^{[h]})\}}{2} \right],$$

where the expectation over $y^{[h]}$ is taken with respect to $\mathcal{K}_h$.

RHS after $\mathbb{E}$: symmetrized Bregman divergence b/w $T(y_i^{[h]})$ and μ_h.

- We can make two expected divergences equal, by setting $\alpha_h = 1/n_h$ and learn λ_h adaptively from its posterior.
- NO Bregman information is lost for clustering!
- Although we implicitly assume T and ϕ are chosen in the same way in both $H_d^{[h]}$ and $H_y^{[h]}$, this is easy to achieve / approximate via sensible choice of distance density.

Information Theory on Clustering

A toy example, for $h = 1, \ldots, k$,

$$y_i^{[h]} \sim \text{No}(\mu_h, \sigma_h^2)$$

with $y_i^{[h]} \in \mathbb{R}$.

- Model-based $H_y^{[h]} = \sum_i (y_i^{[h]} - \mu_h)^2 / \sigma_h^2$.

Information Theory on Clustering

A toy example, for $h = 1, \ldots, k$,

$$y_i^{[h]} \sim \text{No}(\mu_h, \sigma_h^2)$$

with $y_i^{[h]} \in \mathbb{R}$.

- Model-based $H_y^{[h]} = \sum_i (y_i^{[h]} - \mu_h)^2 / \sigma_h^2$.
- Distance-based $H_d^{[h]} = (2 n_h \alpha_h \lambda_h^{-1}) \sum_{i=1}^{n_h} \sum_{i'=1}^{n_h} (d_{i,i'}^{[h]})^2$.

Information Theory on Clustering

A toy example, for $h = 1, \ldots, k$,

$$y_i^{[h]} \sim \text{No}(\mu_h, \sigma_h^2)$$

with $y_i^{[h]} \in \mathbb{R}$.

- Model-based $H_y^{[h]} = \sum_i (y_i^{[h]} - \mu_h)^2 / \sigma_h^2$.
- Distance-based $H_d^{[h]} = (2 n_h \alpha_h \lambda_h^{-1}) \sum_{i=1}^{n_h} \sum_{i'=1}^{n_h} (d_{i,i'}^{[h]})^2$.
- When $\alpha_h = 1/n_h$ and $\lambda_h = 2\sigma_h^2$,

$$\mathbb{E}_y H_y^{[h]} = \mathbb{E}_y H_d^{[h]} = n - 1$$

Information Theory on Clustering

A toy example, for $h = 1, \ldots, k$,

$$y_i^{[h]} \sim \text{No}(\mu_h, \sigma_h^2)$$

with $y_i^{[h]} \in \mathbb{R}$.

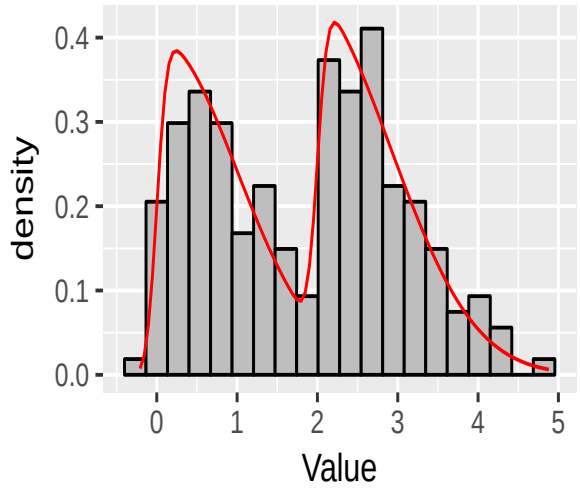
- Model-based $H_y^{[h]} = \sum_i (y_i^{[h]} - \mu_h)^2 / \sigma_h^2$.
- Distance-based $H_d^{[h]} = (2 n_h \alpha_h \lambda_h^{-1}) \sum_{i=1}^{n_h} \sum_{i'=1}^{n_h} (d_{i,i'}^{[h]})^2$.
- When $\alpha_h = 1/n_h$ and $\lambda_h = 2\sigma_h^2$,

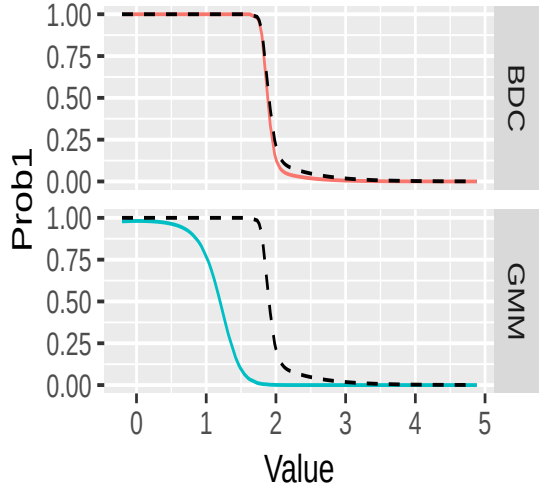
$$\mathbb{E}_y H_y^{[h]} = \mathbb{E}_y H_d^{[h]} = n - 1$$

- $\lambda_h = 2\sigma_h^2$ can be learned from the posterior because $\text{var}(d_{i,i'}^{[h]}) = 2\text{var}(y_i^{[h]})$.

Data Application

Sim1: Robust to skewness





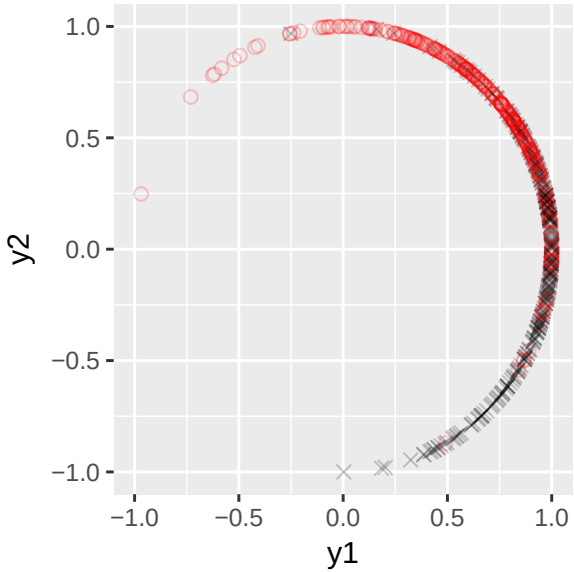
True density (red) of a mixture of two right skewed Gaussians.

Posterior assignment $\text{pr}(c_i = 1)$. Dashed: oracle probability. Red: Bayes dist. clustering. Green: Gaussian mixture (GMM)

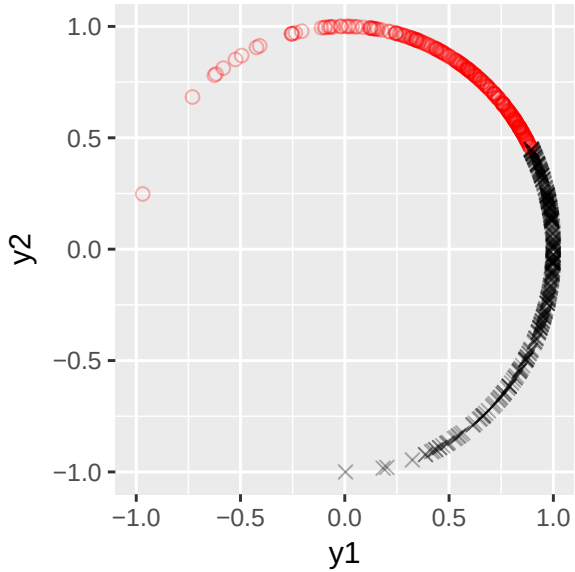
Interestingly, at small $n = 200$ and increasing p, Bayesian Distance Clustering performs even better (in adjusted Rand index) than mixture of skewed Gaussians (the 'true' kernel), likely due to fewer parameters.

p	Bayes Dist. Clustering	Mix. of Gaussians	Mix. of Skewed Gaussians
5	0.76 (0.71, 0.81)	0.55 (0.40, 0.61)	0.76 (0.72, 0.80)
10	0.72 (0.68, 0.76)	0.33(0.25, 0.46)	0.62 (0.53, 0.71)
30	0.71 (0.67, 0.76)	0.25 (0.20, 0.30)	0.43 (0.37, 0.50)

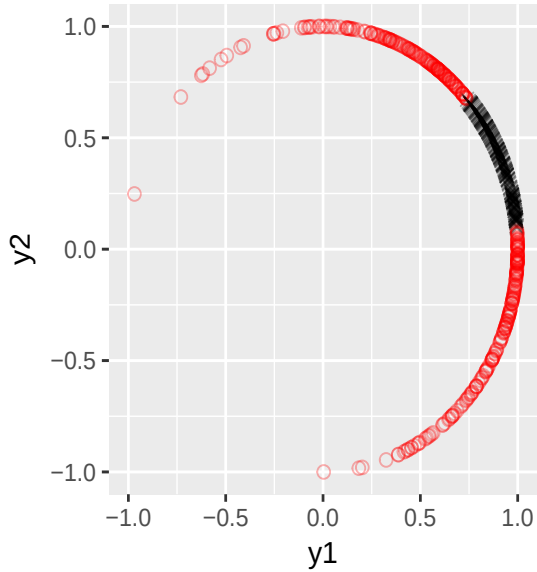
Sim2: Easy to handle constrained data.



(a) Data on unit circle colored by true cluster labels.



(b) Point clustering estimates from BDC.



(c) Point clustering estimates from a mixture of Gaussian model.

Figure: Clustering data from two-component mixture of von-Mises Fisher with $\mu_1 = (1, 0)$ and $\mu_2 = (1/\sqrt{2}, 1/\sqrt{2})$.

Neuroscience application: Clustering brain voxels using gene expression.

- Goal: Estimate functional partitions by segmenting the mouse brain according to the gene expression levels, and compare with structural partitions as defined in brain anatomy.

Neuroscience application: Clustering brain voxels using gene expression.

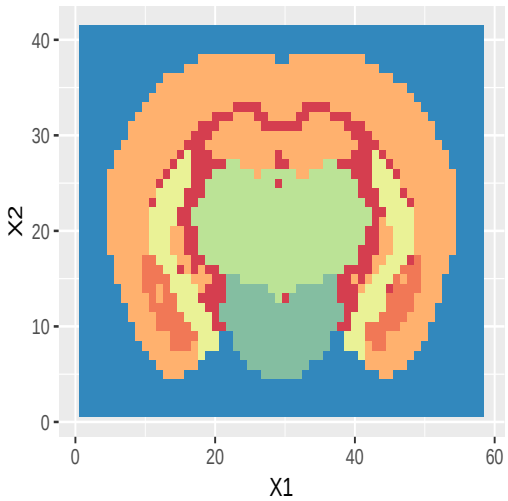
- Goal: Estimate functional partitions by segmenting the mouse brain according to the gene expression levels, and compare with structural partitions as defined in brain anatomy.
- Data: Gene transcriptome over mid-coronal section of 41×58 voxels, excluding the empty ones, sample size $n = 1781$. Each voxel has expression levels for 3241 genes. Data are obtained from Allen Mouse Brain Atlas (Lein et al 2007).

Neuroscience application: Clustering brain voxels using gene expression.

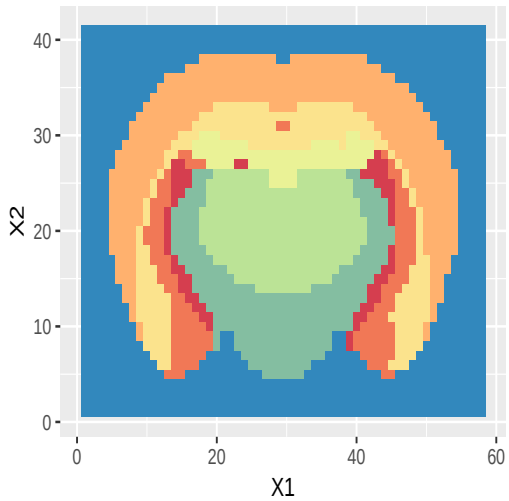
- Goal: Estimate functional partitions by segmenting the mouse brain according to the gene expression levels, and compare with structural partitions as defined in brain anatomy.
- Data: Gene transcriptome over mid-coronal section of 41×58 voxels, excluding the empty ones, sample size $n = 1781$. Each voxel has expression levels for 3241 genes. Data are obtained from Allen Mouse Brain Atlas (Lein et al 2007).
- To avoid curse-of-dimensionality on distances: we follow previous visualization application on the same dataset (Mahfouz et al 2014), and extract the first $p = 30$ principal components as the source data y_i.

Neuroscience application: Clustering brain voxels using gene expression.

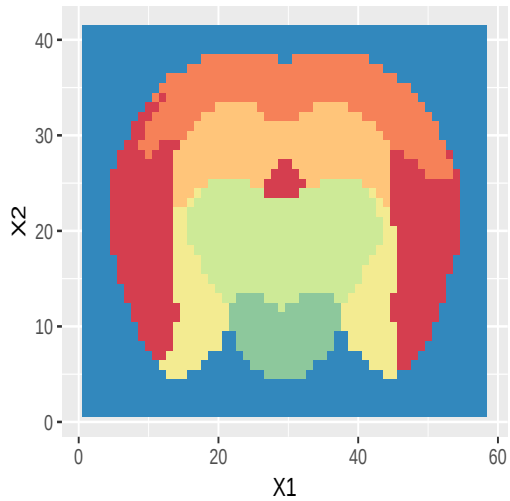
We fit an overfitted model with $k = 20$ and Dirichlet-$(1/20)$ on the component weights. Comparing point estimates:



(a) Structural Partitions (known anatomical labels).

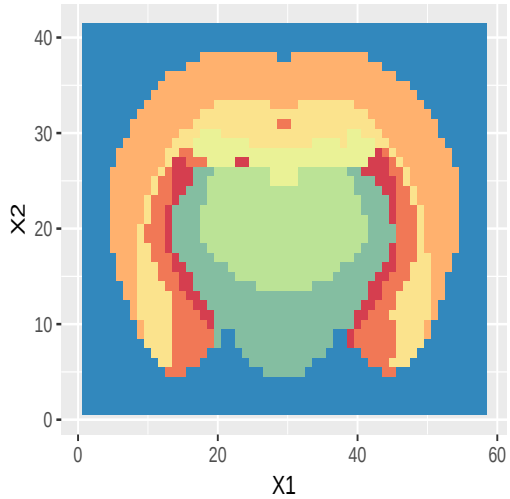


(b) Functional Partitions (clustered by BDC).

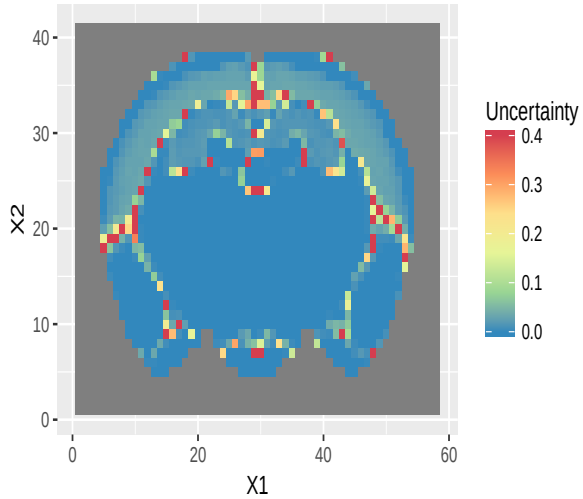


(c) Functional Partitions (clustered by Gaussian Mixture Model).

Neuroscience application: Clustering brain voxels using gene expression.



(a) Functional Partitions: point estimates $\{\hat{c}_i\}$ by BDC



(b) Uncertainty: $\text{pr}(c_i \neq \hat{c}_i)$

Most uncertainty is in the inner layers of the cortical plate (upper parts of the brain).

Comparing point estimates against others:

Table: Comparison of label point estimates using Bayesian distance clustering (BDC), Gaussian mixture model (GMM), spectral clustering (SC), DBSCAN and Mixture of Factor Analyzers (MFA). The similarity measure is computed with respect to the anatomical structure labels .

	BDC	GMM	SC	DBSCAN	MFA
Adjusted Rand Index	0.49	0.31	0.45	0.43	0.43
Normalized Mutual Information	0.51	0.42	0.46	0.44	0.47
Adjusted Mutual Information	0.51	0.42	0.47	0.45	0.47

Discussion

Discussion

- More general distances / divergences can be included: e.g. geodesic, transport distances, etc.
- So far we have focused on moderate n, as the number of distances increases in $\mathcal{O}(n^2)$. Interesting to develop new scalable solution.
- For high dimension data, "Dimension reduction + Clustering" can cause loss of information. Useful to develop new distances that preserve discriminability.

Primary References

- **Duan, L. L., & Dunson, D. B. (2018) Bayesian Distance Clustering. *arXiv* preprint arXiv:1810.08537.**
- Miller, J. W. & Dunson, D. B. (2018). Robust Bayesian inference via coarsening. *Journal of the American Statistical Association*, Online.
- Lein, E. S., Hawrylycz, M. J., AO, N., Ayres, M., Bensinger, A., Bernard, A., BOE, A. F., Boguski, M. S., Brockway, K. S. & Byrnes, E. J. (2007). Genome-wide atlas of gene expression in the adult mouse brain. *Nature* 445, 168.
- Hodges, J. & Lehmann, E. (1954). Matching in paired comparisons. *The Annals of Mathematical Statistics* 25, 787-791.
- Coretto, P. & Hennig, C. (2016). Robust improper maximum likelihood: tuning, computation, and a comparison with other methods for robust Gaussian clustering. *Journal of the American Statistical Association* 111, 1648- 1659.
- Banerjee, A., Merugu, S., Dhillon, I. S. & Ghosh, J. (2005). Clustering with Bregman divergences. *Journal of Machine Learning Research* 6, 1705-1749.
- Bregman, L. M. (1967). The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics* 7, 200-217.