# Lecture IV – Non-parametric clustering

Marina Meilă
mmp@stat.washington.edu

Department of Statistics
University of Washington

1. Paradigms for clustering
2. Methods based on non-parametric density estimation
3. Model-based: Dirichlet process mixture models

**Reading** AoNPS Ch.: –, HTF Ch.: 14.3 Murphy Ch.: 11.[1], 11.2.1-3, 11.3, Ch 25

# What is clustering? Problem and Notation

- **Informal definition** **Clustering** = Finding groups in data
- **Notation**

| | | |
|---|---|---|
| $\mathcal{D}$ | = | $\{x_1, x_2, \ldots x_n\}$ a **data set** |
| $n$ | = | number of **data points** |
| $K$ | = | number of **clusters** ($K << n$) |
| $\Delta$ | = | $\{C_1, C_2, \ldots, C_K\}$ a partition of $\mathcal{D}$ into disjoint subsets |
| $k(i)$ | = | the **label** of point $i$ |
| $\mathcal{L}(\Delta)$ | = | cost (loss) of $\Delta$ (to be minimized) |

- **Second informal definition** **Clustering** = given $n$ **data points**, separate them into $K$ **clusters**
- Hard vs. soft clusterings
  - **Hard** clustering $\Delta$: an item belongs to only 1 cluster
  - **Soft** clustering $\gamma = \{\gamma_{ki}\}_{k=1:K}^{i=1:n}$
    $\gamma_{ki}$ = the **degree of membership** of point $i$ to cluster $k$

$$\sum_k \gamma_{ki} = 1 \quad \text{for all } i$$

(usually associated with a probabilistic model)

# Clustering Paradigms

Depend on type of data, type of clustering, type of cost (probabilistic or not), and constraints (about $K$, shape of clusters)

- Data = vectors $\{x_i\}$ in $\mathbb{R}^d$

| | |
|---|---|
| Parametric | Cost based [hard] |
| ($K$ known) | Model based [soft] |
| Non-parametric | Dirichlet process mixtures [soft] |
| ($K$ determined | Information bottleneck [soft] |
| by algorithm) | Modes of distribution [hard] |
| | Gaussian blurring mean shift? [hard] |
| | Level sets of distribution [hard] |

- Data = similarities between pairs of points $[S_{ij}]_{i,j=1:n}$, $S_{ij} = S_{ji} \geq 0$

**Similarity based clustering**

| | |
|---|---|
| Graph partitioning | spectral clustering [hard, $K$ fixed, cost based] |
| | typical cuts [hard non-parametric, cost based] |
| Affinity propagation | [hard/soft non-parametric] |

# Classification vs Clustering

| | **Classification** | **Clustering** |
|---|---|---|
| **Cost (or Loss)** $\mathcal{L}$ | Expectd error | many! (probabilistic or not) |
| | Supervised | Unsupervised |
| **Generalization** | Performance on new data is what matters | Performance on current data is what matters |
| $K$ | Known | Unknown |
| **"Goal"** | Prediction | Exploration Lots of data to explore! |
| **Stage of field** | Mature | Still young |

# Methods based on non-parametric density estimation

**Idea** The clusters are the isolated peaks in the (empirical) data density

- group points by the peak they are under
- some outliers possible
- $K = 1$ possible(no clusters)
- shape and number of clusters $K$ determined by algorithm
- **structural parameters**
  - **smoothness** of the **density estimate**
  - what is a peak

**Algorithms**

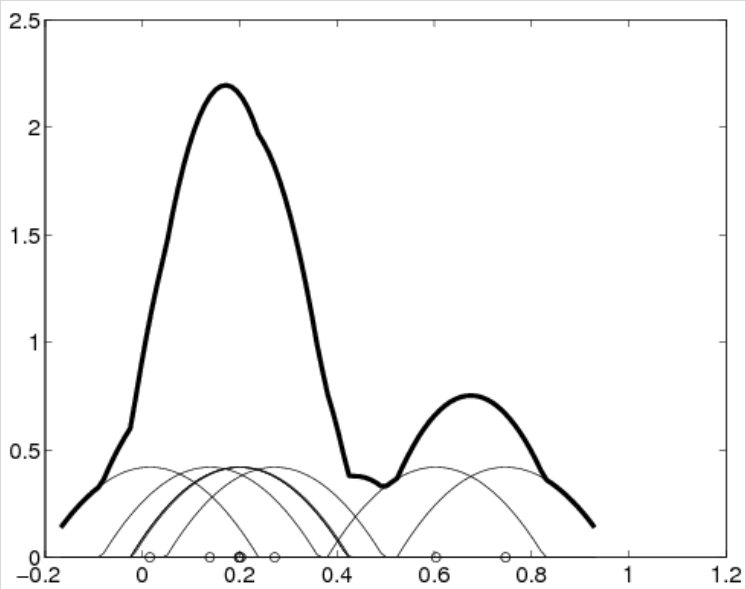
- peak finding algorithms Mean-shift algorithms
- level sets based algorithms
  - Nugent-Stuetzle, Support Vector clustering
- Information Bottleneck **?**

# Kernel density estimation

**Input**
- data $\mathcal{D} \subseteq \mathbb{R}^d$
- **Kernel** function $K(z)$
- parameter **kernel width** $h$ (is a **smoothness parameter**)

**Output** $f(x)$ a **probability density** over $\mathbb{R}^d$

$$f(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$




- $f$ is sum of Gaussians centered on each $x_i$
- $f$ is smoother (less variation) if $h$ larger
- **caveat:** dimension $d$ can't be too large

## The kernel function

- Example $K(z) = \frac{1}{(2\pi)^{d/2}} e^{-||z||^2/2}$, $z \in \mathbb{R}^d$ is the Gaussian kernel
- In general
  - $K()$ should represent a density on $\mathbb{R}^d$, i.e $K(z) \geq 0$ for all $z$ and $\int K(z)dz = 1$
  - $K()$ symmetric around 0, decreasing with $||z||$
- In our case, $K$ must be differentiable

# Mean shift algorithms

**Idea** find points with $\nabla f(x) = 0$

Assume $K(z) = e^{-||z||^2/2}/\sqrt{2\pi}$ Gaussian kernel

$$\nabla f(x) = -\frac{1}{nh^d} \sum_{i=1}^{n} K(\frac{x - x_i}{h})(x - x_i)/h$$

Local max of $f$ is solution of implicit equation

$$x = \underbrace{\frac{\sum_{i=1}^{n} x_i K(\frac{x-x_i}{h})}{\sum_{i=1}^{n} K(\frac{x-x_i}{h})}}_{\text{the mean shift } m(x)}$$

**Algorithm Simple Mean Shift**

**Input** Data $\mathcal{D} = \{x_i\}_{i=1:n}$, kernel $K(z)$, $h$

1. for $i = 1 : n$
   1. $x \leftarrow x_i$
   2. iterate $x \leftarrow m(x)$ until convergence to $m_i$
2. group points with same $m_i$ in a cluster

## Remarks

- mean shift iteration guaranteed to converge to a max of $f$
- computationally expensive
- a faster variant...

**Algorithm Mean Shift (Comaniciu-Meer)**

**Input** Data $\mathcal{D} = \{x_i\}_{i=1:n}$, kernel $K(z)$, $h$

1. select $q$ points $\{x_j\}_{j=1:q} = \mathcal{D}_q \subseteq \mathcal{D}$ that cover the data well
2. for $j \in \mathcal{D}_q$
   1. $x \leftarrow x_j$
   2. iterate $x \leftarrow m(x)$ until convergence to $m_j$
3. group points in $\mathcal{D}_q$ with same $m_j$ in a cluster
4. assign points in $\mathcal{D} \setminus \mathcal{D}_q$ to the clusters by the **nearest-neighbor** method

$$k(i) = k(\operatorname*{argmin}_{j \in \mathcal{D}_q} ||x_i - x_j||)$$

# [Supplement: Gaussian blurring mean shift]

**Idea**

- like **Simple Mean Shift** but points are shifted to new locations
- the density estimate $f$ changes
- becomes concentrated around peaks very fast

**Algorithm Gaussian Blurrring Mean Shift (GBMS)**

**Input** Data $\mathcal{D} = \{x_i\}_{i=1:n}$, Gaussian kernel $K(z)$, $h$

1. Iterate until STOP
   1. for $i = 1 : n$ compute $m(x_i)$
   2. for $i = 1 : n$, $x_i \leftarrow m(x_i)$

**Remarks**

- all $x_i$ converge to a single point
  $\Rightarrow$ need to stop before convergence

Empirical stopping criterion ?

- define $e_i^t = ||x_i^t - x_i^{t-1}||$ the change in $x_i$ at $t$
- define $H(e^t)$ the **entropy** of the **histogram** of $\{e_i^t\}$
- STOP when $\sum_{i=1}^n e_i^t/n <$`tol` OR $|H(e^t) - H(e^{t-1})| <$`tol'`

**Convergence rate** If true $f$ Gaussian, convergence is **cubic**

$$||x_i^t - x^*|| \leq C||x_i^{t-1} - x^*||^3$$

very fast!!

# The Nugent-Stuetzle algorithm

**Algorithm Nugent-Stuetzle**

**Input** Data $\mathcal{D} = \{x_i\}_{i=1:n}$, kernel $K(z)$

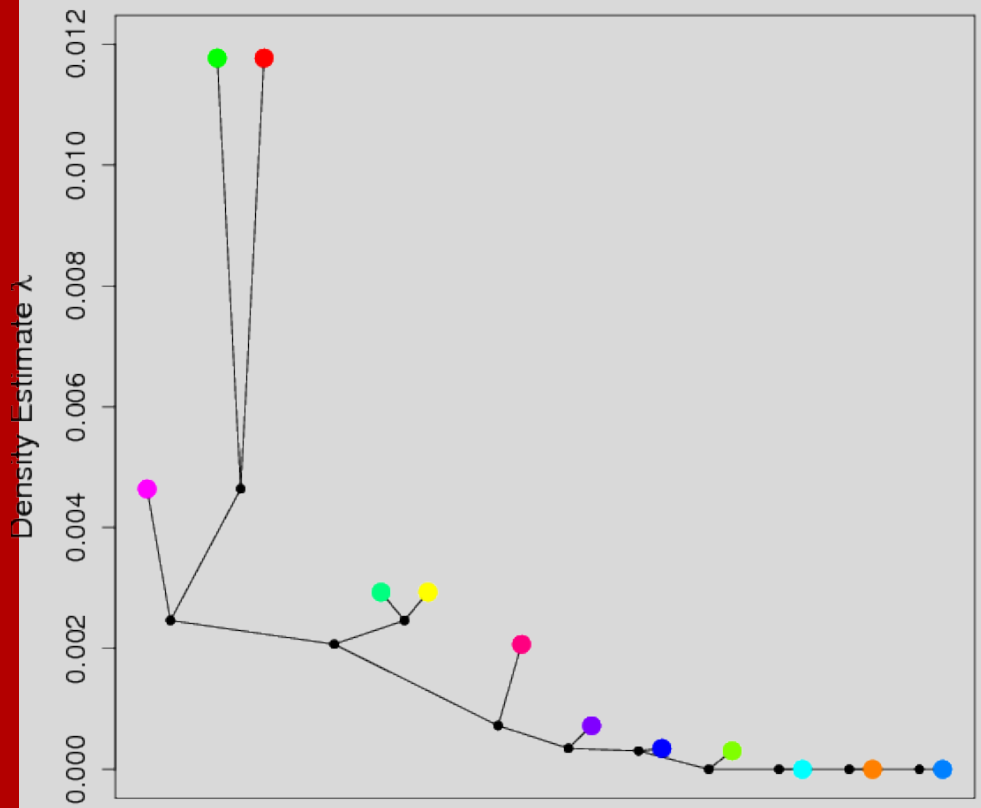
1. Compute KDE $f(x)$ for chosen $h$
2. for levels $0 < l_1 < l_2 < \ldots < l_r < \ldots < l_R \geq \sup_x f(x)$
   1. find **level set** $L_r = \{x \,|\, f(x) \geq l_r\}$ of $f$
   2. if $L_r$ **disconnected** then each connected component is a cluster $\rightarrow (C_{r,1}, C_{r,2}, \ldots C_{r,K_r})$

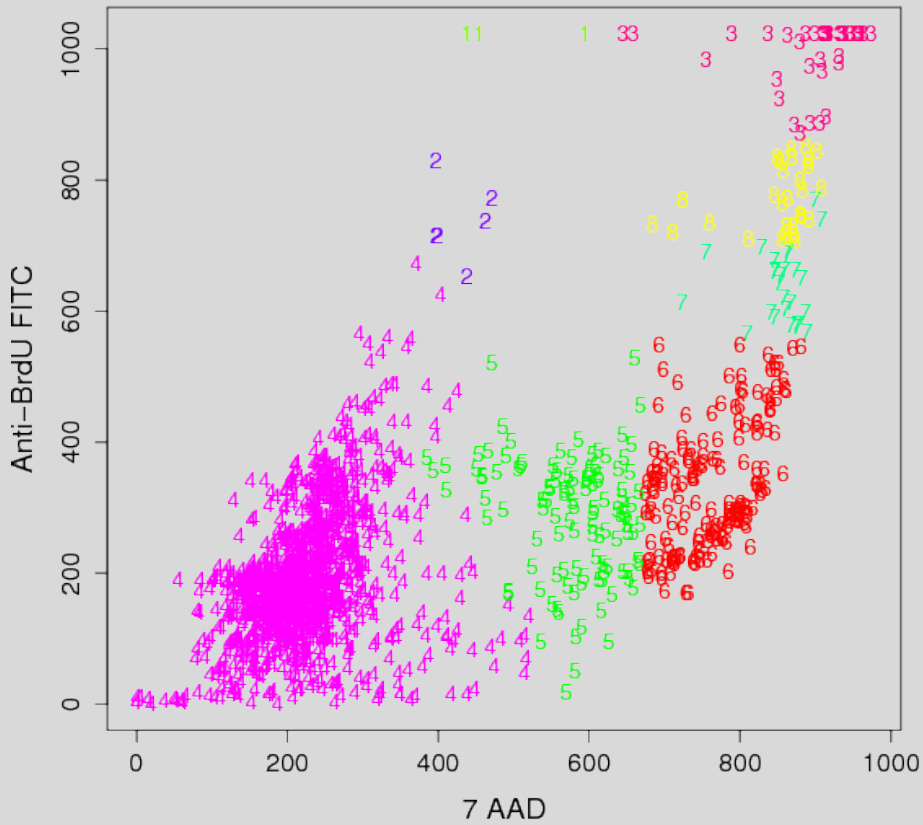
**Output** clusters $\{(C_{r,1}, C_{r,2}, \ldots C_{r,K_r})\}_{r=1:R}$

# Remarks

- every cluster $C_{r,k} \subseteq$ some cluster $C_{r-1,k'}$
- therefore output is hierarchical clustering
- some levels can be pruned (if no change, i.e. $K_r = K_{r-1}$)
- algorithm can be made recursive, i.e. efficient
- finding level sets of $f$ tractable only for $d = 1, 2$
- for larger $d$, $L_r = \{x_i \in \mathcal{D} \,|\, f(x_i) \geq l_r\}$
- to find connected components
  - for $i \neq j \in L_r$
    if $f(tx_i + (1-t)x_j) \geq l_r$ for $t \in [0, 1]$
    then $k(i) = k(j)$
- confidence intervals possile by resampling

Cluster tree with 13 leaves (8 clusters, 5 artifacts)




(from ?)

# Chaudhuri-Dasgupta Algorithm

- Uses $k$-**nearest neighbor** graphs (filtration)
- Parameters $k$ (nearest neigbhors) and $\alpha \in [1, 2]$
- for $r \geq 0$, $G_r = (V_r, E_r)$ with
  - $x_i \in V_r$ iff **distance to $k$-nn of $x_i$** $\leq r$
  - $(x_i, x_j) \in E_r$ iff $||x_i - x_j|| \leq \alpha r$

**Consistency Theorem** For any $\epsilon$ (separation parameter) and $\delta$ (confidence), $\alpha \in [\sqrt{2}, 2]$ (graph density), if $k = C \log^2(1/\delta) \frac{d \log n}{\epsilon^2}$
for any two clusters $C, C'$ in cluster tree, there exists a level $r$ so that $C \cap \mathcal{D}, C' \cap \mathcal{D}$ are clusters at level $r$

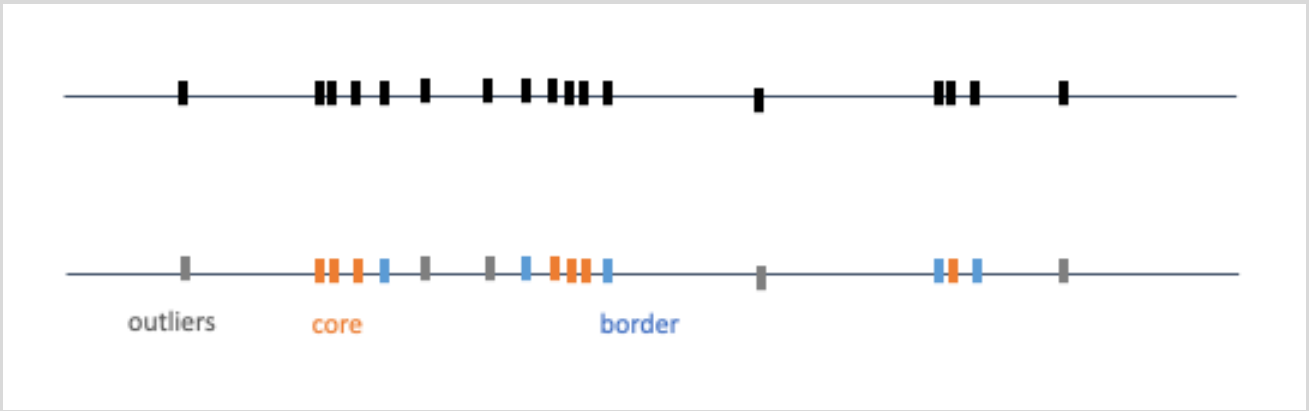
# The K-nn density estimator

**The K-nn** density estimator

- Let $B_r(x)$ be the (closed) ball of radius $r$ centered at $x$
- If $|B_r(x^i) \cap \mathcal{D}| = k$ then $\hat{p}(x^i) = \frac{1}{r^n \omega_n} \frac{k}{n}$ is an estimate of the density at $x^i$
  - $\omega_n = \pi^{n/2}/\Gamma(n/2+1)$ is the volume of the unit ball in $\mathbb{R}^n$
  - intuitively, the ball of radius $r$ contains $k/n$ probability mass
  - Note that the density $\hat{p}$ is not required to integrate to 1

# DBScan

- Introduced with no proof, but widely used. Implicitly based on the K-nn estimator
- Parameters $r$ radius, $m$ minimum number points
- Definitions **core** $Q = \{x^i \in \mathcal{D}, \text{ with } B_r(x^i) \cap \mathcal{D}| \geq m\}$
- **border** $B = \{x^i \in \mathcal{D} \setminus Q, \text{ so that } x^i \in B_r(x^j),\ x^j \in Q\}$
- **outliers (noise)** $O = \mathcal{D} \setminus (Q \cup B)$




- Algorithm idea
- Construct directed graph $\mathcal{G}$ with edges $(i, j)$ where $x^i \in Q, j \in B_r(x^i)$
- The graph edges between core points are undirected/symmetric, the other are from core to border
- Clusters are determined by the connected components of the graph restricted to $Q$.
- The border points are assigned to a cluster containing $x^j$ so that $x^i \in B_r(x^j)$, $x^j \in Q$ Note that this assignment is not unique!

- Heuristic algorithm estimates $r, m$

# [Supplement: Chaudhuri-Dasgupta Algorithm]

**Consistency Theorem** For any $\epsilon$ (separation parameter) and $\delta$ (confidence), $\alpha \in [\sqrt{2}, 2]$ (graph density), if $k = C \log^2(1/\delta) \frac{d \log n}{\epsilon^2}$
for any two clusters $C, C'$ in cluster tree, there exists a level $r$ so that $C \cap \mathcal{D}, C' \cap \mathcal{D}$ are clusters at level $r$

- $r$ depends on $\lambda =$"bridge" between $C, C'$ (and $\sigma > 0$ "tube" width)

$$r^d \omega_d \lambda = \frac{k}{n} + \ldots \textit{confidence term}$$

- it follows that the needed sample size $n$ at level $\lambda$

$$n = \mathcal{O}\left( \frac{d}{\lambda \epsilon^2 (\sigma/2)^d \omega_d} \log \frac{d}{\lambda \epsilon^2 (\sigma/2)^d \omega_d} \right)$$

- this **sample complexity** $n$ is almost tight
- for $\alpha < \sqrt{2}$ sample complexity is exponential in $d$

- New results **[Kent, B. P., Rinaldo, A. and Verstynen, T. 2013]**

- Remark: algorithm(s) can be applied in any metric space

# [Supplement: Support Vector (SV) clustering]

**Idea** same as for Nugent-Stuetzle, but use **kernelized density estimator** instead of KDE

**Algorithm SV**

**Input** data $\mathcal{D}$, parameters $q$ kernel width, $p \in (0,1)$ proportion of outliers

1. construct a 1-class SVM with parameters $q$, $C = 1/np$
   this is equivalent to enclosing the data in a sphere in feature space
   for any $x$ its distance from center of sphere is

$$R^2(x) = K(x,x) - 2\sum_j \alpha_j K(x,x_j) + \sum_{i,j} K(x_i,x_j)$$

   for $x_i$ support vector, $R(x_i) = R$ (same for all)
2. for all pairs $i,j = 1:n$
   - $i,j$ in same cluster if segment $[i,j]$ is within sphere with radius $R$ in feature space
   - practically, test if $R(tx_i + (1-t)x_j) < R$ for $t$ on a grid over [0,1]

# Remarks

- the **kernel** used by SV is $K(x, x') = e^{-q||x-x'||^2}$
- $q$ controls boundary smoothness
- SV's lie on cluster boundaries, "margin error" points lie outside clusters (are outliers)
- SV theory $\frac{\text{margin errors}}{n} \rightarrow \frac{1}{nC} = p$ for large $n$
- hence $p$ controls the proportion of outliers
- $p, q$ together control $K$
  $p$ larger, $q$ smaller $\Rightarrow$ $K$ smaller

# The Dirichlet distribution

- $Z \in \{1 : r\}$ a discrete random variable, let $\theta_j = P_z(j)$, $j = 1, \ldots r$.
- **Multinomial distribution** Probability of i.i.d. sample of size $N$ from $P_z$

$$P(z^{1,\ldots n}) = \prod_{j=1}^{r} \theta_j^{n_j}$$

where $n_j = \#$the value $j$ is observed, $j = 1, \ldots r$

- $n_{1:r}$ are the **sufficient statistics** of the data.
- The **Dirichlet distribution** is defined over domain of $\theta_{1,\ldots r}$, with real parameters $N'_{1,\ldots r} > 0$ by

$$D(\theta_{1,\ldots r}; n'_{1,\ldots r}) = \frac{\Gamma(\sum_j n'_j)}{\prod_j \Gamma(n'_j)} \prod_j \theta_j^{n'_j - 1}$$

where $\Gamma(p) = \int_0^\infty t^{p-1} e^{-t} dt$.

# Dirichlet process mixtures

- Model-based
- generalization of mixture models to
  - infinite $K$
  - Bayesian framework
- denote $\theta_k$ = parameters for component $f_k$
- assume $f_k(x) \equiv f(x, \theta_k) \in \{f(x, \theta)\}$
- assume prior distributions for parameters $g_0(\theta)$
- prior with hyperparameter $\alpha > 0$ on the number of clusters
- very flexible model

# A sampling model for the data

- **Example: Gaussian mixtures**, $d = 1$, $\sigma_k = \sigma$ fixed
- $\theta = \mu$
- prior for $\mu$ is $Normal(0, \sigma_0^2 I_d)$
- Sampling process
  - for $i = 1 : n$ sample $x_i, k(i)$ as follows

    denote $\{1 : K\}$ the clusters after step $i - 1$
    define $n_k$ the size of cluster $k$ after step $i - 1$

    1. $$k(i) = \begin{cases} k & \text{w.p } \frac{n_k}{i-1+\alpha}, \ k = 1 : K \\ K+1 & \text{w.p } \frac{\alpha}{i-1+\alpha} \end{cases} \quad (1)$$
    2. if $k(i) = K + 1$ sample $\mu_i \equiv \mu_{K+1}$ from $Normal(0, \sigma_0^2)$
    3. sample $x_i$ from $Normal(\mu_{k(i)}, \sigma^2)$
- can be shown that the distribution of $x_{1:n}$ is **interchangeable** (does not depend on data permutation)

# The hyperparameters

- $\sigma_0$ controls spread of centers
  - should be large
- $\alpha$ controls number of cluster centers
  - $\alpha$ large $\Rightarrow$ many clusters
- cluster sizes non-uniform (larger clusters attract more new points)
- many single point clusters possible

**General Dirichlet mixture model**

- cluster densities $\{f(x, \theta)\}$
- parameters $\theta$ sampled from prior $g_0(\theta, \beta)$
- cluster membership $k(i)$ sampled as in (1)
- $x_i$ sampled from $f(x, \theta_{k(i)})$
- **Model Hyperparameters** $\alpha, \beta$

# Clustering with Dirichlet mixtures

**The clustering problem**

- $\alpha, g_0, \beta, \{f\}$ given
- $\mathcal{D}$ given
- wanted $\theta_{1:n}$ (not all distinct!)
- note:
  - $\theta_{1:n}$ determines a hard clustering $\Delta$
  - the posterior of $\theta_{1:n}$ given the data determines a soft clustering via $P(x_i \mid k) \propto \int f(x_i|\theta_k) g_k(\theta_k) d\theta_k$

Estimating $\theta_{1:n}$ cannot be solved in closed form
Usually solved by MCMC (Markov Chain Monte Carlo) sampling

# Clustering with Dirichlet mixtures via MCMC

**MCMC estimation for Dirichlet mixture**

**Input** $\alpha, g_0, \beta, \{f\}, \mathcal{D}$

**State** cluster assignments $k(i)$, $i = 1:n$,
parameters $\theta_k$ for all distinct $k$

**terate**

1. for $i = 1:n$ (reassign data to clusters)
   1. remove $i$ from its cluster (hence $\sum_k n_k = n-1$)
   2. resample $k(i)$ by

$$k(i) = \begin{cases} \text{existing } k & \text{w.p } \propto \frac{n_k}{n-1+\alpha} f(x_i, \theta_k) \\ \text{new cluster} & \text{w.p } \frac{\alpha}{n-1+\alpha} \int f(x_i, \theta) g_0(\theta) d\theta \end{cases} \quad (2)$$

   3. if $k(i)$ is new label, sample a new $\theta_{k(i)} \propto g_0 f(x_i, \theta)$
2. for $k \in \{k(1:n)\}$ (resample cluster parameters)
   1. sample $\theta_k$ from posterior $g_k(\theta) \propto g_0(\theta, \beta) \prod_{i \in C_k} f(x_i, \theta)$

$g_k$ can be computed in closed form if $g_0$ is conjugate prior

**)utput** a state with high posterior

# Summary: Parametric vs. non-parametric

**Parametric clustering**

- Optimizes a cost $\mathcal{L}$
- Most costs are NP-hard to optimize
- Assumes more detailed knowledge of cluster shapes
- Assumes $K$ known (But there are wrapper methods to select $K$)
- Gets harder with larger $K$
- Older, more used and better studied

**Non-parametric clustering**

- Variety of paradigms
  - density-based methods have no cost function
  - (Max Likelihood: non-parametric mixture models)
  - Bayesian: Dirichlet Process Mixtures (samples from posterior of $k(1:n), \{\theta_k\}$ given $\mathcal{D}$)
- Do not depend critically on initialization
- $K$ and outliers selected automatically, naturally
- Require hyperparameters (= smoothness parameters)

# When to use

- Parametric
  - shape of clusters known
  - $K$ not too large or known
  - clusters of comparable sizes
- Non-parametric (density based)
  - shape of clusters arbitrary
  - $K$ large or many outliers
  - clusters sizes in large range (a few large clusters and many small ones)
  - dimension $d$ small (except for **SV**)
  - lots of data
- Dirichlet Process mixtures
  - shape of clusters known
  - clusters sizes in large range

# Notation

$||x - y||$ Euclidean distance for $x, y \in \mathbb{R}^d$, $||x - y|| = \sqrt{\sum_{j=1}^{d}(x_j - y_j)^2}$

# Links

- Yee Whye Teh's tutorial on DP Mixtures http://mlg.eng.cam.ac.uk/tutorials/07/ywt.pdf
- Lecture on exponential family models http: