Machine Learning Department
School of Computer Science
Carnegie Mellon University

Bayesian Nonparametrics:
DPMM
+
Indian Buffet Process

Matt Gormley

Lecture 23
Apr. 23, 2021

Reminders

- Cloud Credits (AWS or GCP)
  - first request deadline: Thu at 11:59pm
- Quiz 3
  - Mon, May 3 during lecture slot
  - Topics: Lectures 16 - 23

Exchangability

Question:

Select All: Which of the following properties of an infinite sequence of random variables $X_1, X_2, X_3, \ldots$ ensure that they are infinitely exchangeable?

- For any pair of orderings $(i_1, i_2, \ldots, i_n)$ and $(j_1, j_2, \ldots, j_n)$ of the indices $(1,\ldots,n)$ the joint probability of the two orderings is the same
- The joint distribution is invariant to permutation
- The joint distribution of the first n random variables can be represented as a mixture
- The random variables are independent and identically distributed

Answer:

Chinese Restaurant Process & Stick-breaking Constructions

DIRICHLET PROCESS MIXTURE MODEL

CRP Mixture Model

- Draw n cluster indices from a CRP:

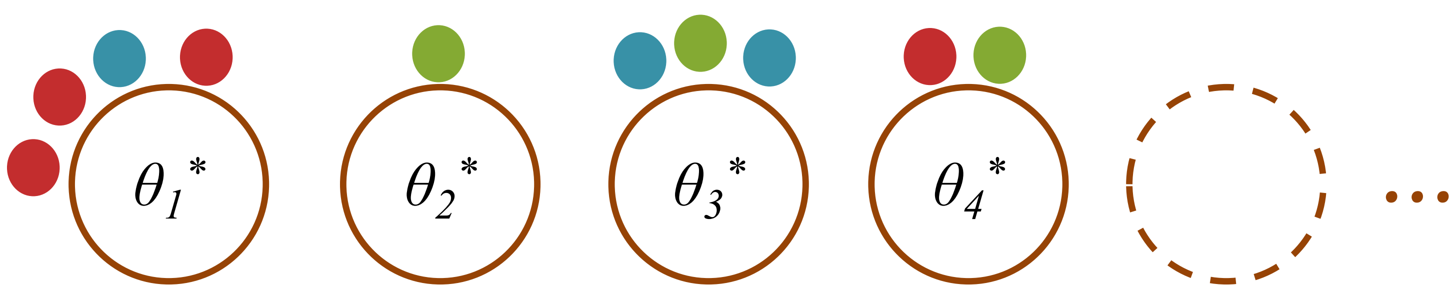
  $z_1, z_2, \ldots, z_n \sim CRP(\alpha)$
- For each of the resulting K clusters:

  $\theta_k^* \sim H$

  where H is a base distribution
- Draw n observations:

$$x_i \sim p(x_i \mid \theta^*_{z_i})$$

Customer i orders a dish x_i (observation) from a table-specific distribution over dishes θ_k^* (cluster parameters)



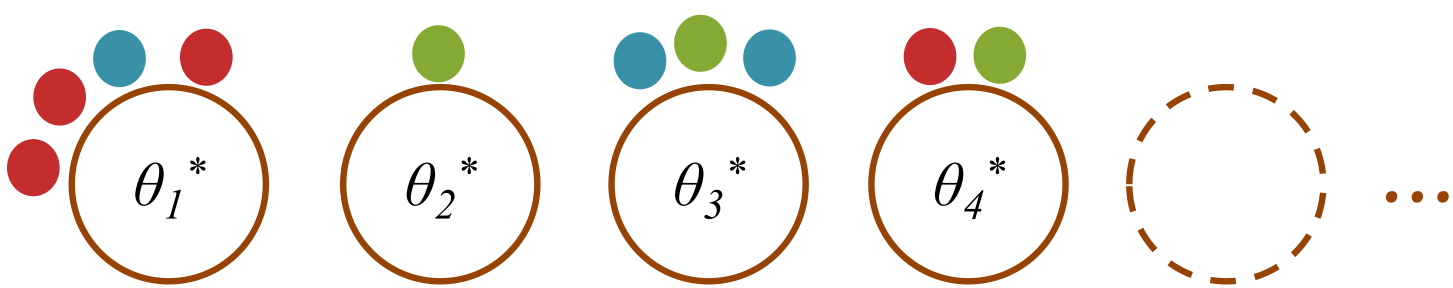
(color denotes different values of x_i)

CRP Mixture Model

- Draw n cluster indices from a CRP:
  $z_1, z_2, \ldots, z_n \sim CRP(\alpha)$
- For each of the resulting K clusters:
  $\theta_k^* \sim H$
  where H is a base distribution
- Draw n observations:
  $$x_i \sim p(x_i \mid \theta^*_{z_i})$$

- The Gibbs sampler is easy thanks to **exchangeability**
- For each observation, we remove the customer / dish from the restaurant and resample as if they were the **last to enter**
- If we **collapse out the parameters,** the Gibbs sampler draws from the conditionals:

  $z_i \sim p(z_i \mid \boldsymbol{z}_{-i}, \boldsymbol{x})$



(color denotes different values of x_i)

CRP Mixture Model

Overview of 3 Gibbs Samplers for Conjugate Priors

- Alg. 1: (uncollapsed)
  - Markov chain state: per-customer parameters $\theta_1, \ldots, \theta_n$
  - For $i = 1, \ldots, n$: Draw $\theta_i \sim p(\theta_i \mid \boldsymbol{\theta}_{-i}, \boldsymbol{x})$

All the thetas except θ_i

- Alg. 2: (uncollapsed)
  - Markov chain state: per-customer cluster indices $z_1, \ldots, z_n$ and per-cluster parameters $\theta_1^*, \ldots, \theta_k^*$
  - For $i = 1, \ldots, n$: Draw $z_i \sim p(z_i \mid \boldsymbol{z}_{-i}, \boldsymbol{x}, \boldsymbol{\theta}^*)$
  - Set K = number of clusters in $\boldsymbol{z}$
  - For $k = 1, \ldots, K$: Draw $\theta_k^* \sim p(\theta_k^* \mid \{x_i : z_i = k\})$
- Alg. 3: (collapsed)
  - Markov chain state: per-customer cluster indices $z_1, \ldots, z_n$
  - For $i = 1, \ldots, n$: Draw $z_i \sim p(z_i \mid \boldsymbol{z}_{-i}, \boldsymbol{x})$

CRP Mixture Model

- Q: How can the Alg. 2 Gibbs samplers permit an infinite set of clusters in finite space?
- A: Easy!
  - We are only representing a finite number of clusters at a time – those to which the data have been assigned
  - We can always bring back the parameters for the "next unoccupied table" if we need them

Dirichlet Process Mixture Model

Whiteboard

- Dirichlet Process Mixture Model (stick-breaking version)

CRP-MM vs. DP-MM

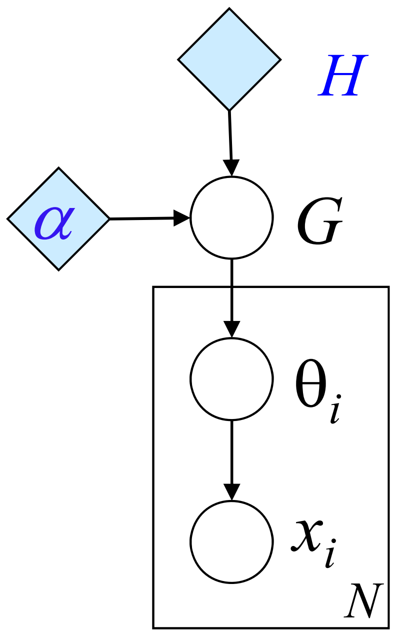
Dirichlet Process: For both the CRP and stick-breaking constructions, if we marginalize out G, we have the following predictive distribution:

$$\theta_{n+1} | \theta_1, \ldots, \theta_n \sim \frac{1}{\alpha + n} \left(\alpha H + \sum_{i=1}^{n} \delta_{\theta_i} \right)$$

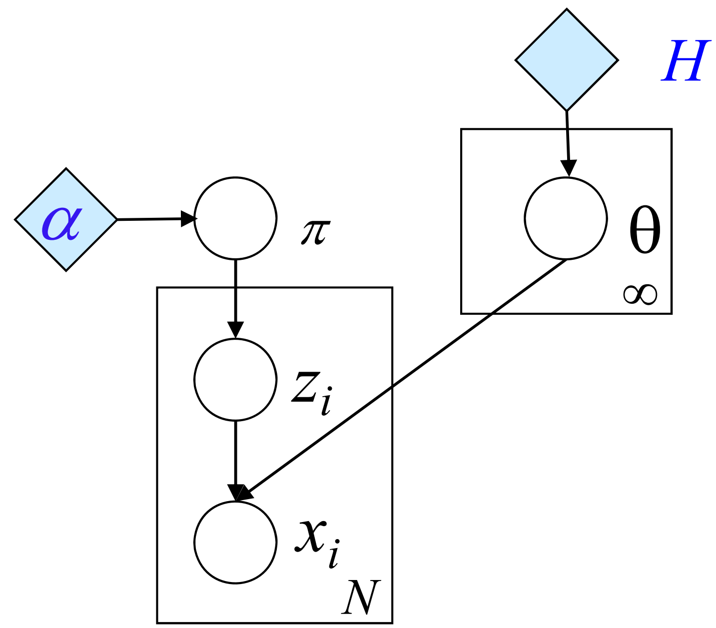
(Blackwell-MacQueen Urn Scheme)

The Chinese Restaurant Process Mixture Model is just a different construction of the Dirichlet Process Mixture Model where we have marginalized out G

Graphical Models for DPMMs



The Pólya urn construction



The Stick-breaking construction

Figure from Blei & Jordan (2006)

Example: DP Gaussian Mixture Model

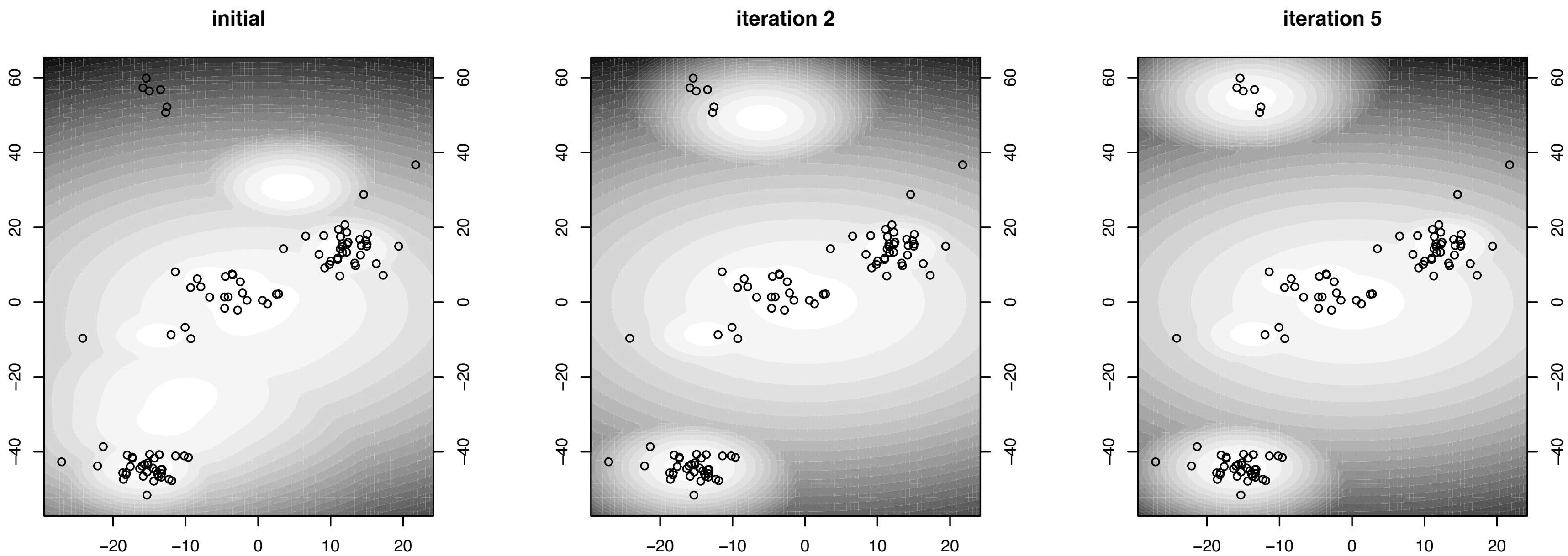


Figure 2: The approximate predictive distribution given by variational inference at different stages of the algorithm. The data are 100 points generated by a Gaussian DP mixture model with fixed diagonal covariance.

Example: DP Gaussian Mixture Model

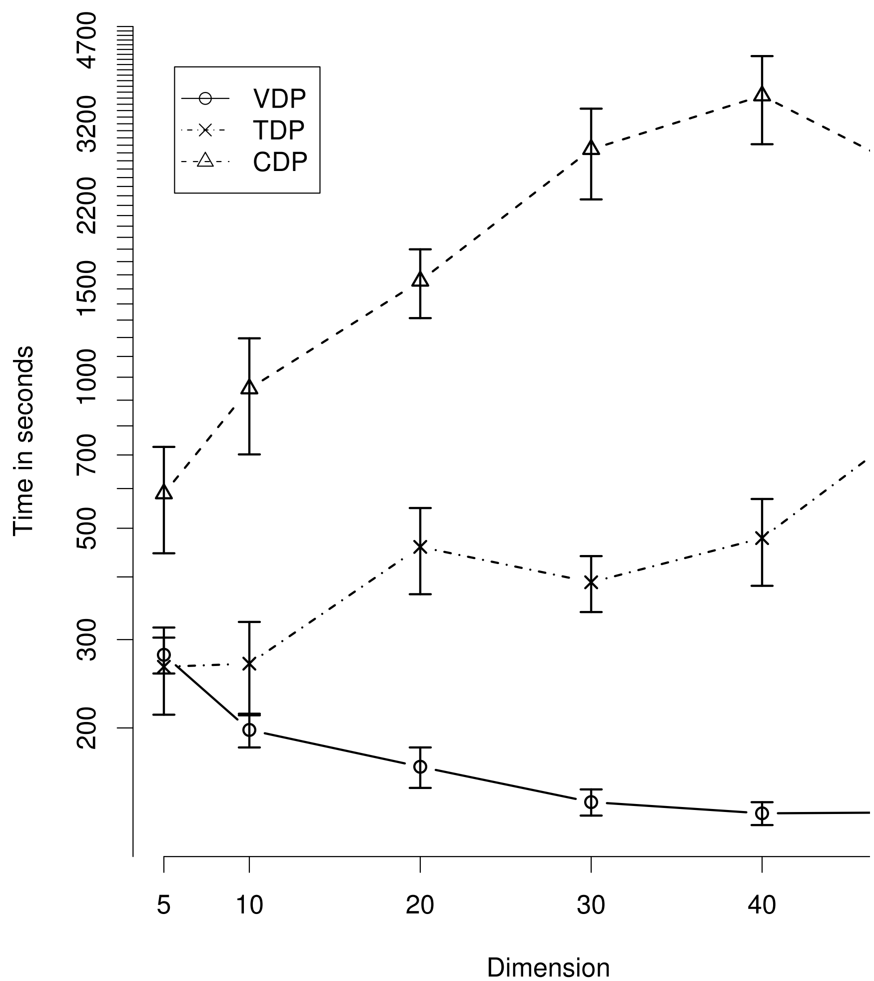


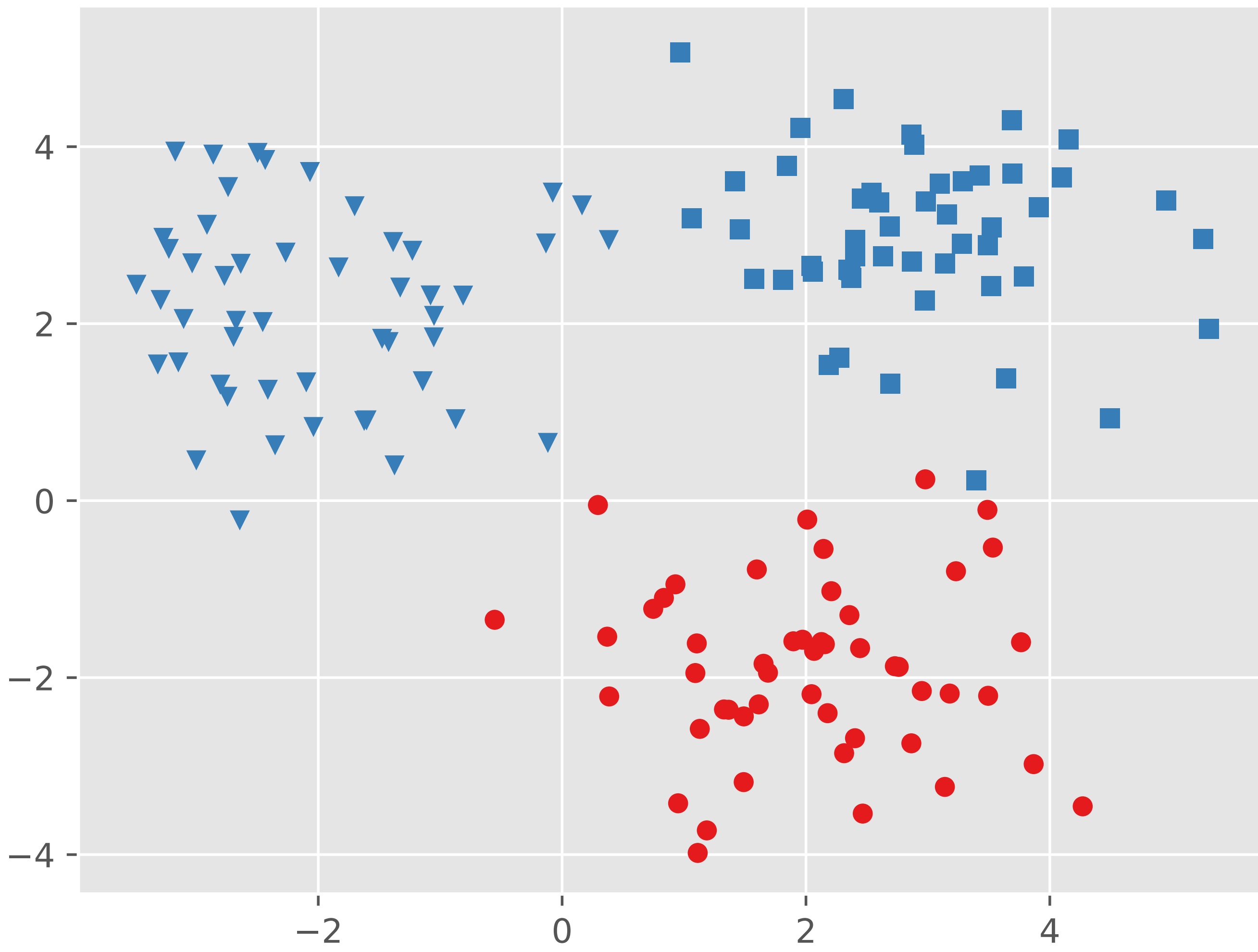
Figure 3: Mean convergence time and standard error across ten data sets per dimension for variational inference, TDP Gibbs sampling, and the collapsed Gibbs sampler.

Summary of DP and DP-MM

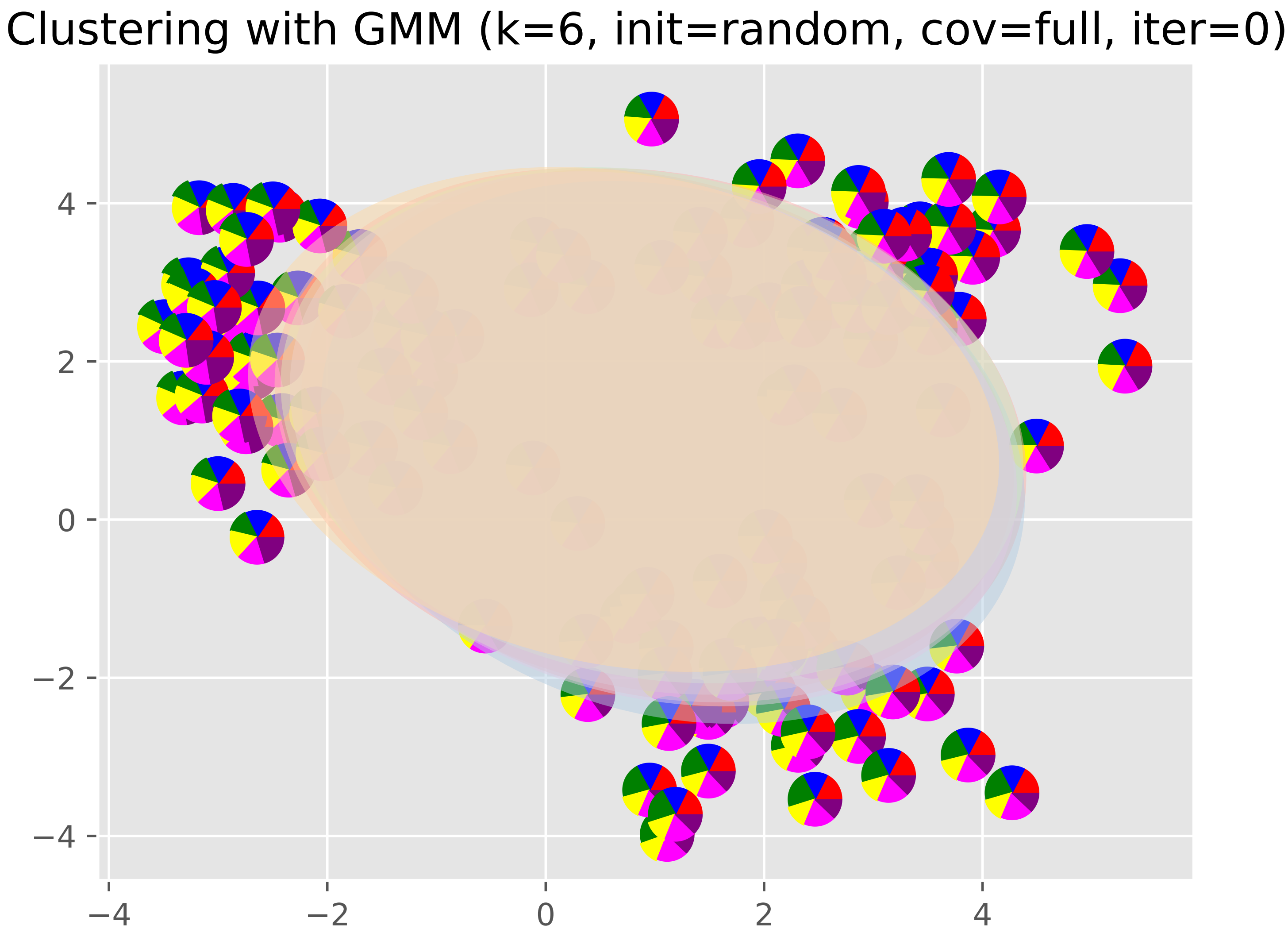
- **DP** has many **different representations:**
  - Chinese Restaurant Process
  - Stick-breaking construction
  - Blackwell-MacQueen Urn Scheme
  - Limit of finite mixtures
  - etc.
- These representations give rise to a variety of **inference techniques** for the **DP-MM** and related models
  - Gibbs sampler (CRP)
  - Gibbs sampler (stick-breaking)
  - Variational inference (stick-breaking)
  - etc.

GMM VS. DPMM EXAMPLE

Example: Dataset

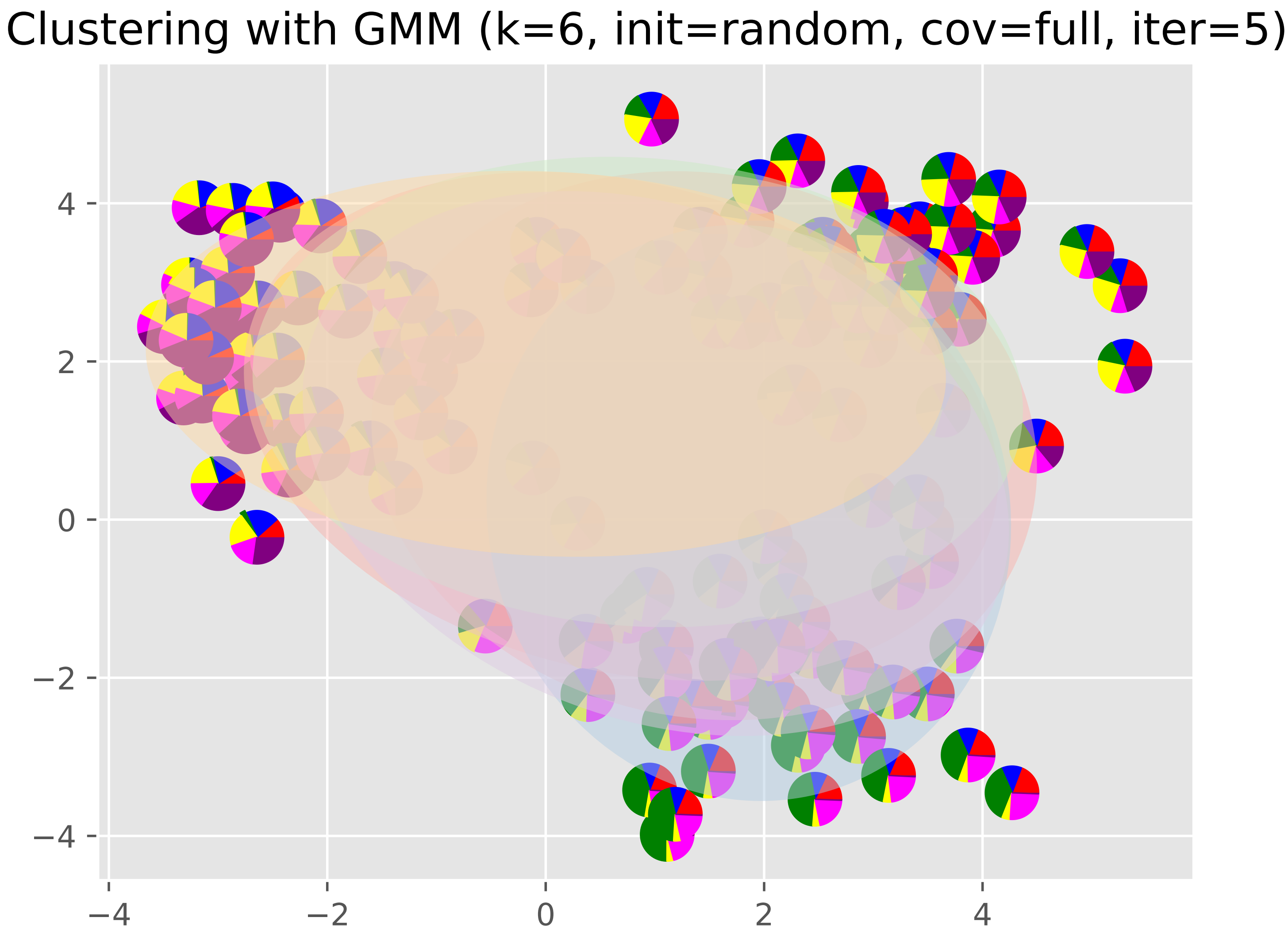


Example: GMM

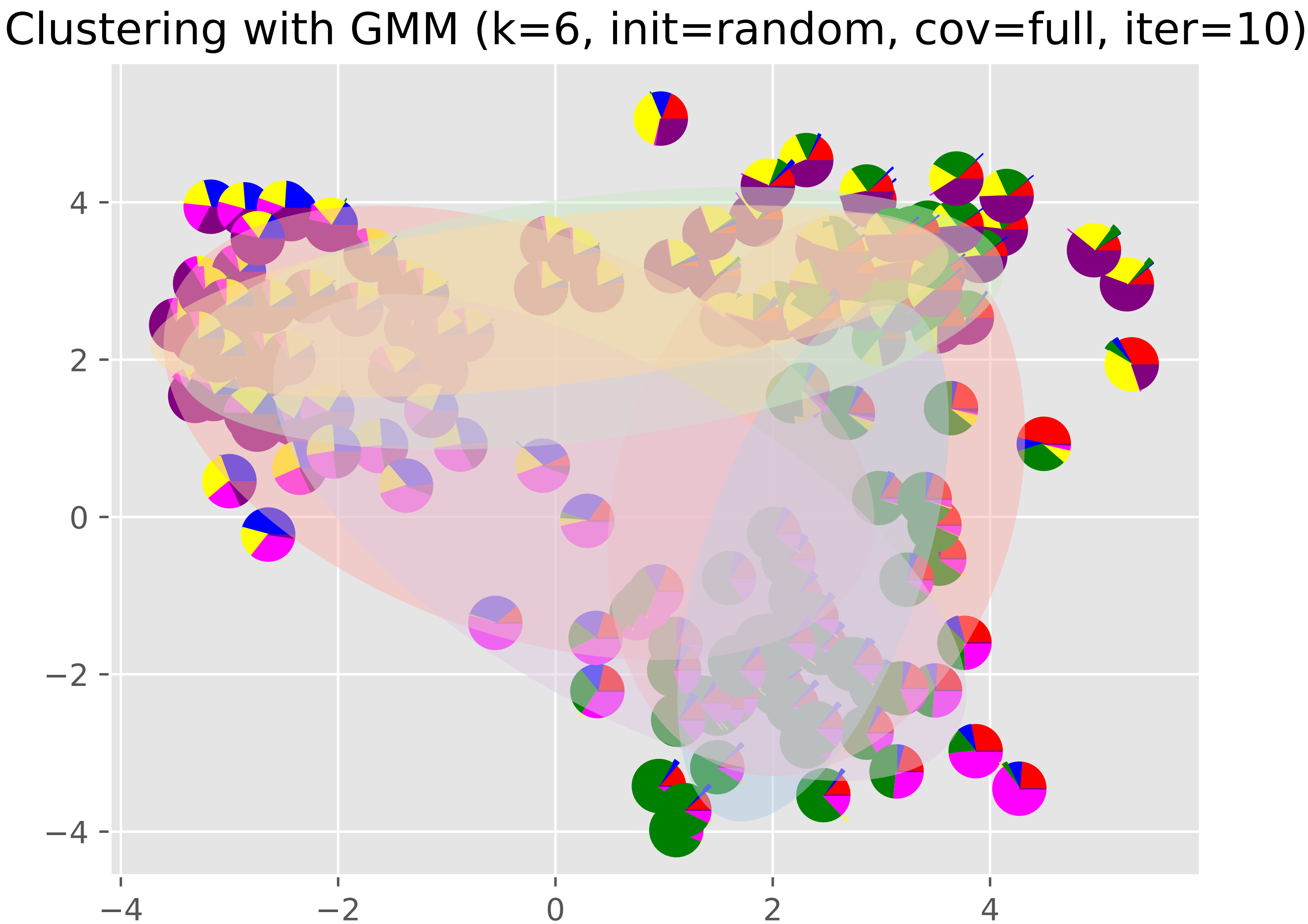


Clustering with GMM (k=6, init=random, cov=full, iter=0)
4
2
0
−2
−4
−4
−2
0
2
4

Example: GMM

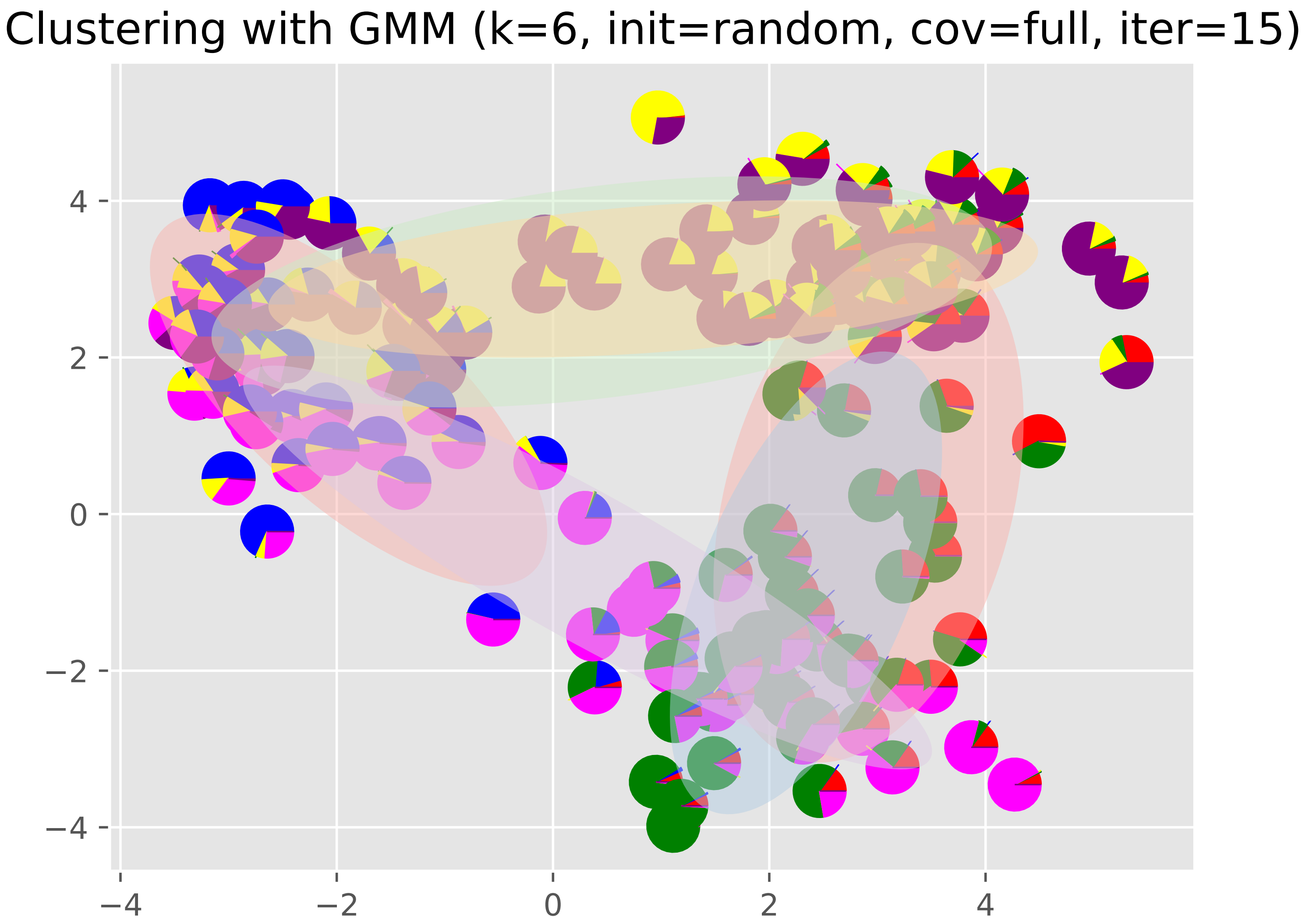

Clustering with GMM (k=6, init=random, cov=full, iter=5)
4
2
0
−2
−4
−4
−2
0
2
4

Example: GMM



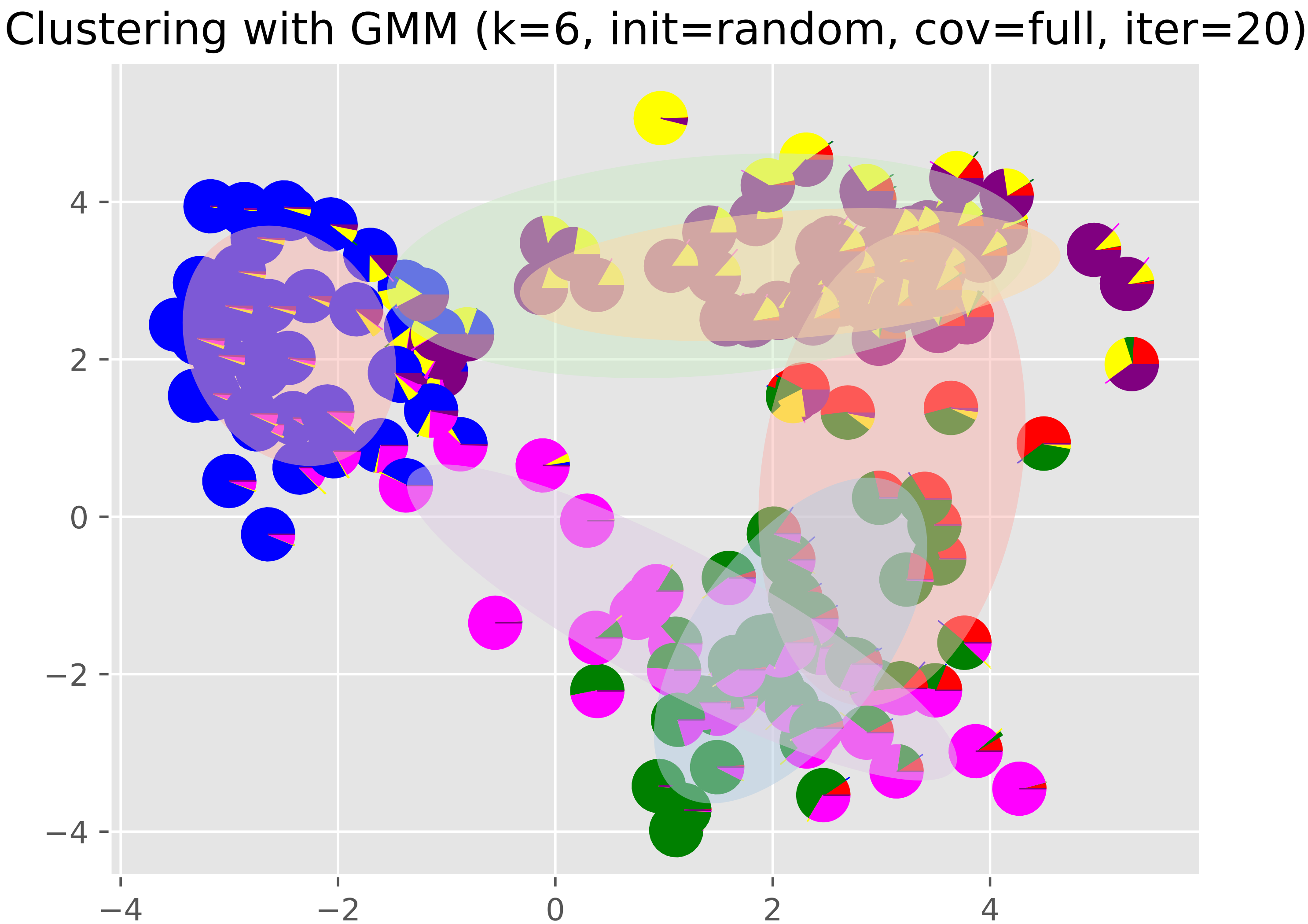
Clustering with GMM (k=6, init=random, cov=full, iter=10)
4
2
0
−2
−4
−4
−2
0
2
4

Example: GMM

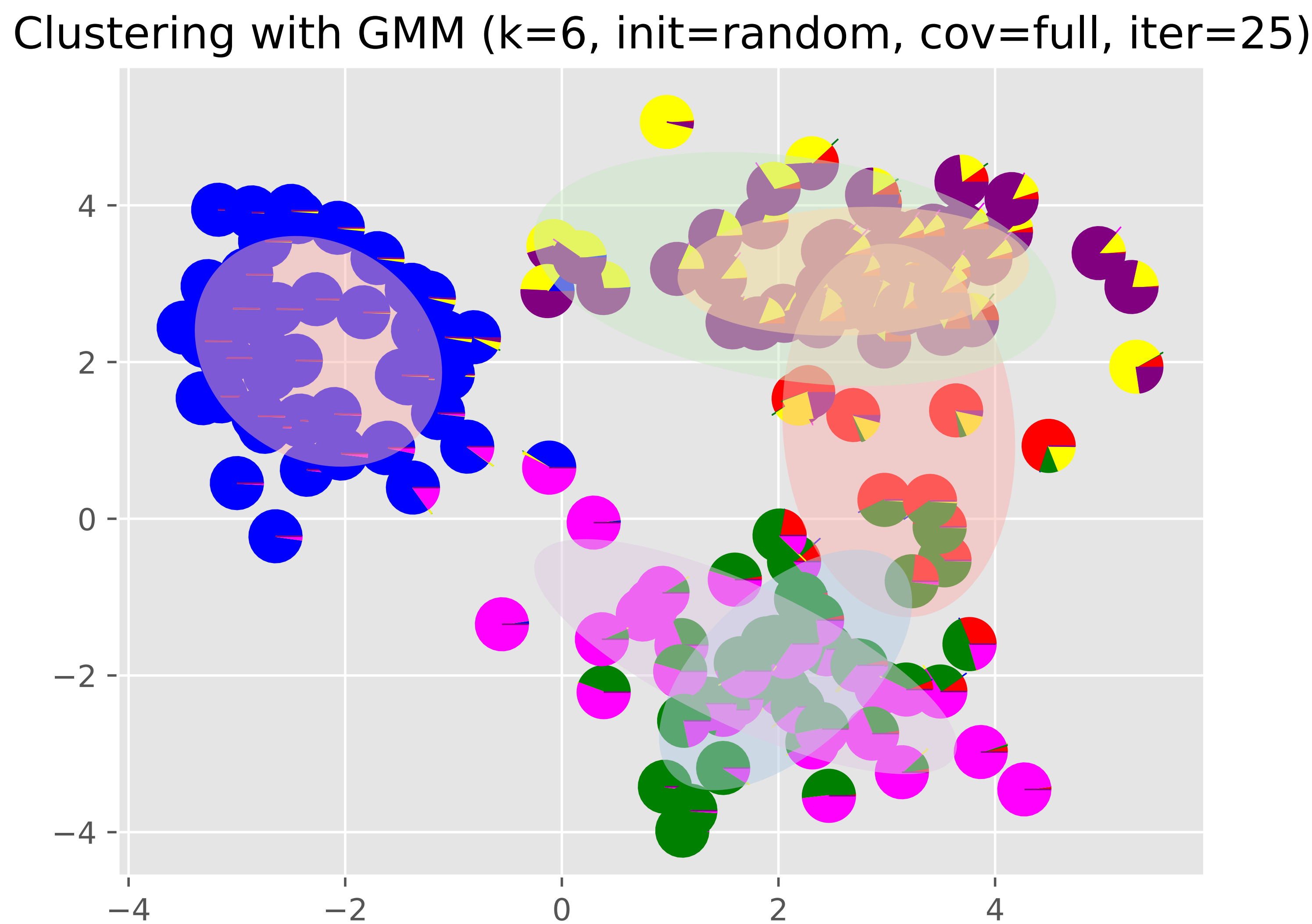


Clustering with GMM (k=6, init=random, cov=full, iter=15)
4
2
0
−2
−4
−4
−2
0
2
4

Example: GMM

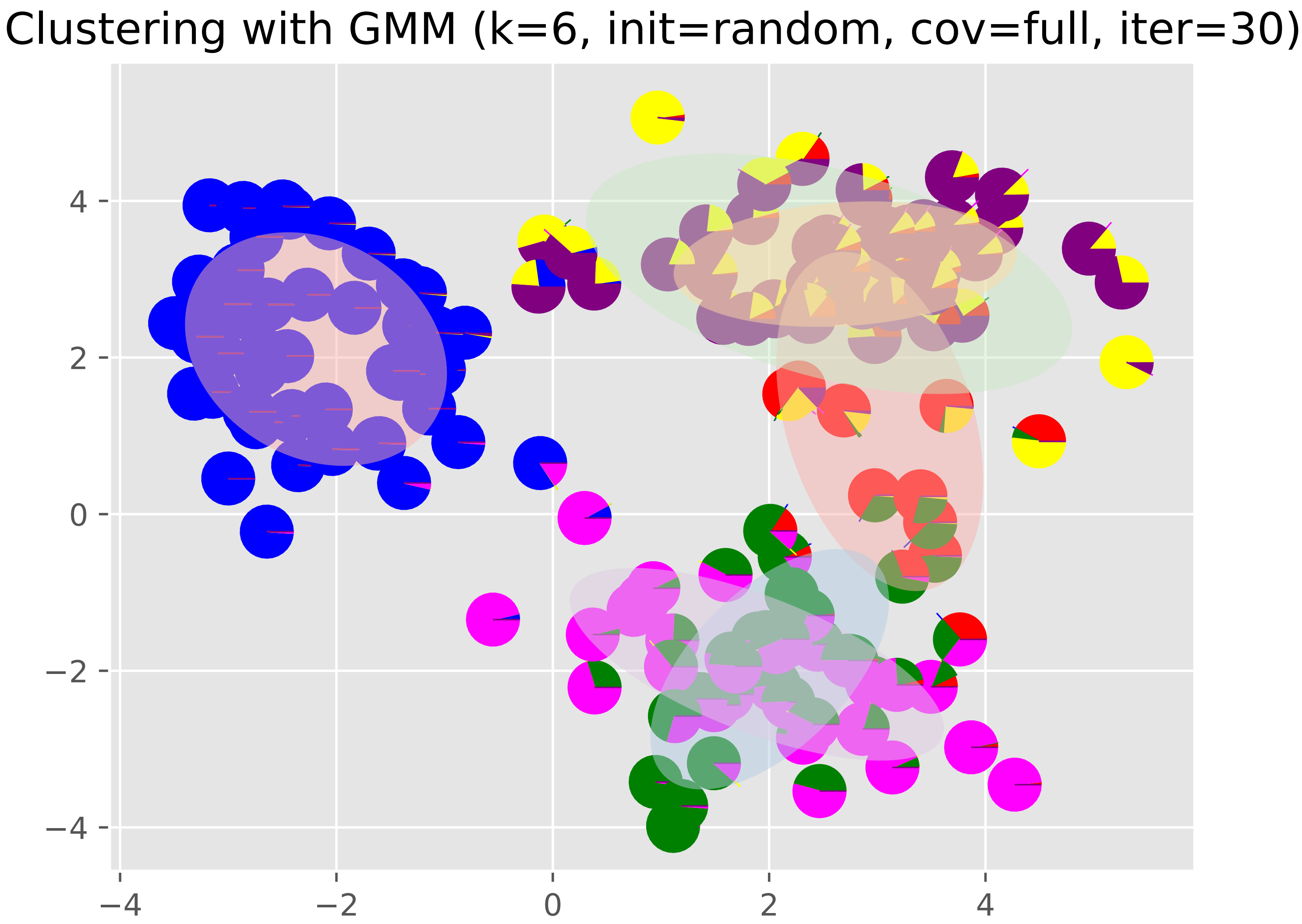

Clustering with GMM (k=6, init=random, cov=full, iter=20)
4
2
0
−2
−4
−4
−2
0
2
4

Example: GMM



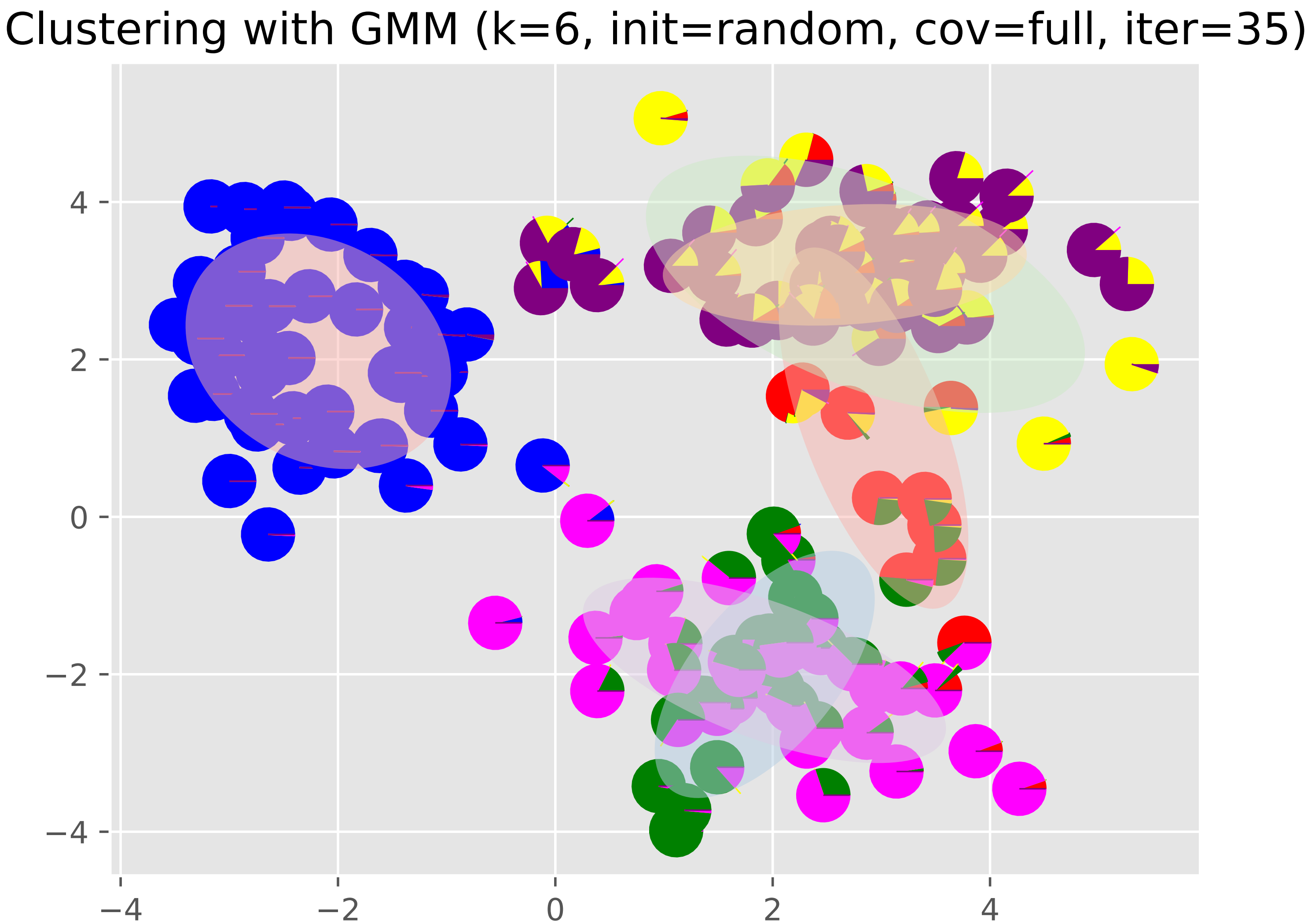
Clustering with GMM (k=6, init=random, cov=full, iter=25)
4
2
0
−2
−4
−4
−2
0
2
4

Example: GMM

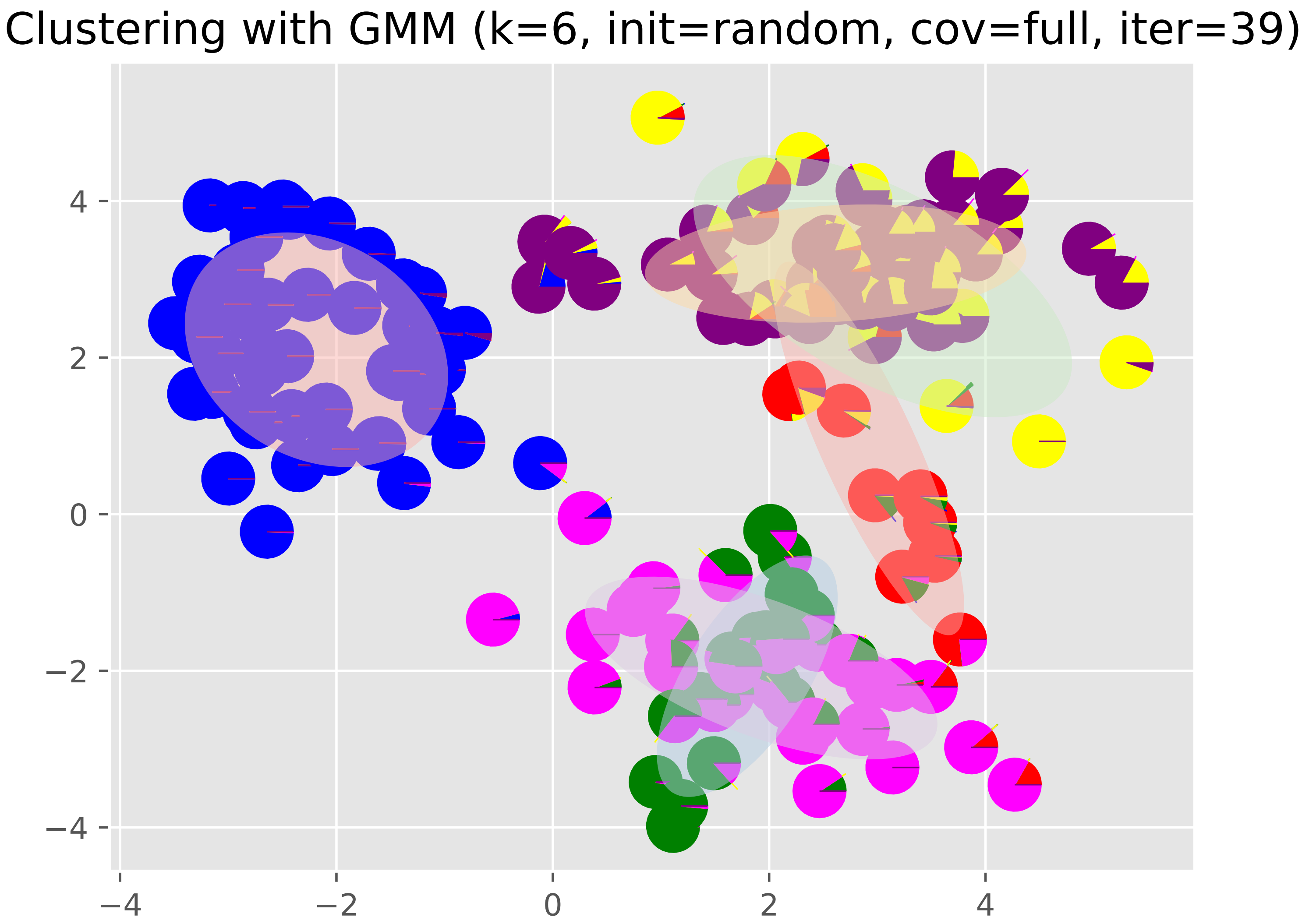


Clustering with GMM (k=6, init=random, cov=full, iter=30)
4
2
0
−2
−4
−4
−2
0
2
4

Example: GMM

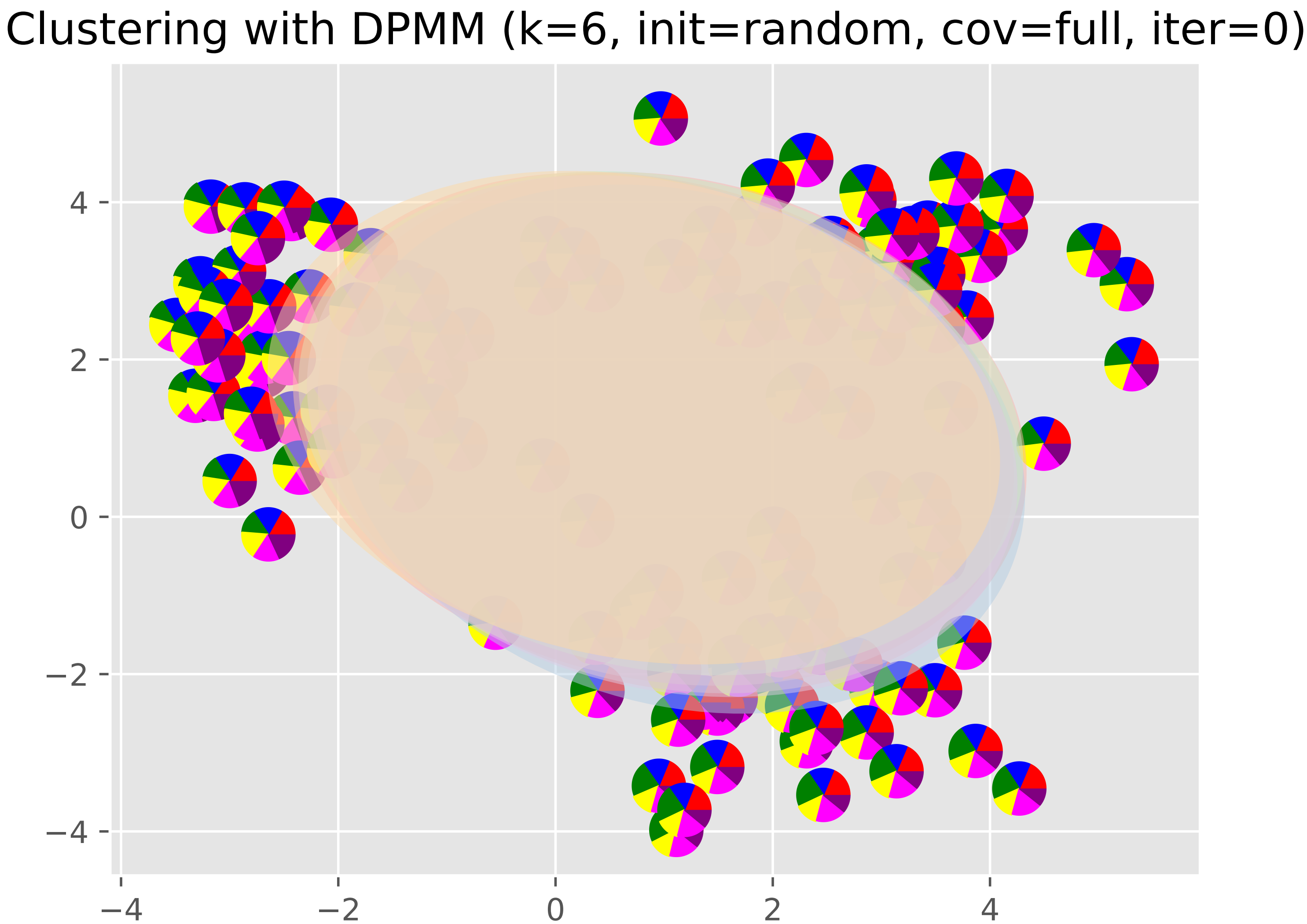

Clustering with GMM (k=6, init=random, cov=full, iter=35)
4
2
0
−2
−4
−4
−2
0
2
4

Example: GMM

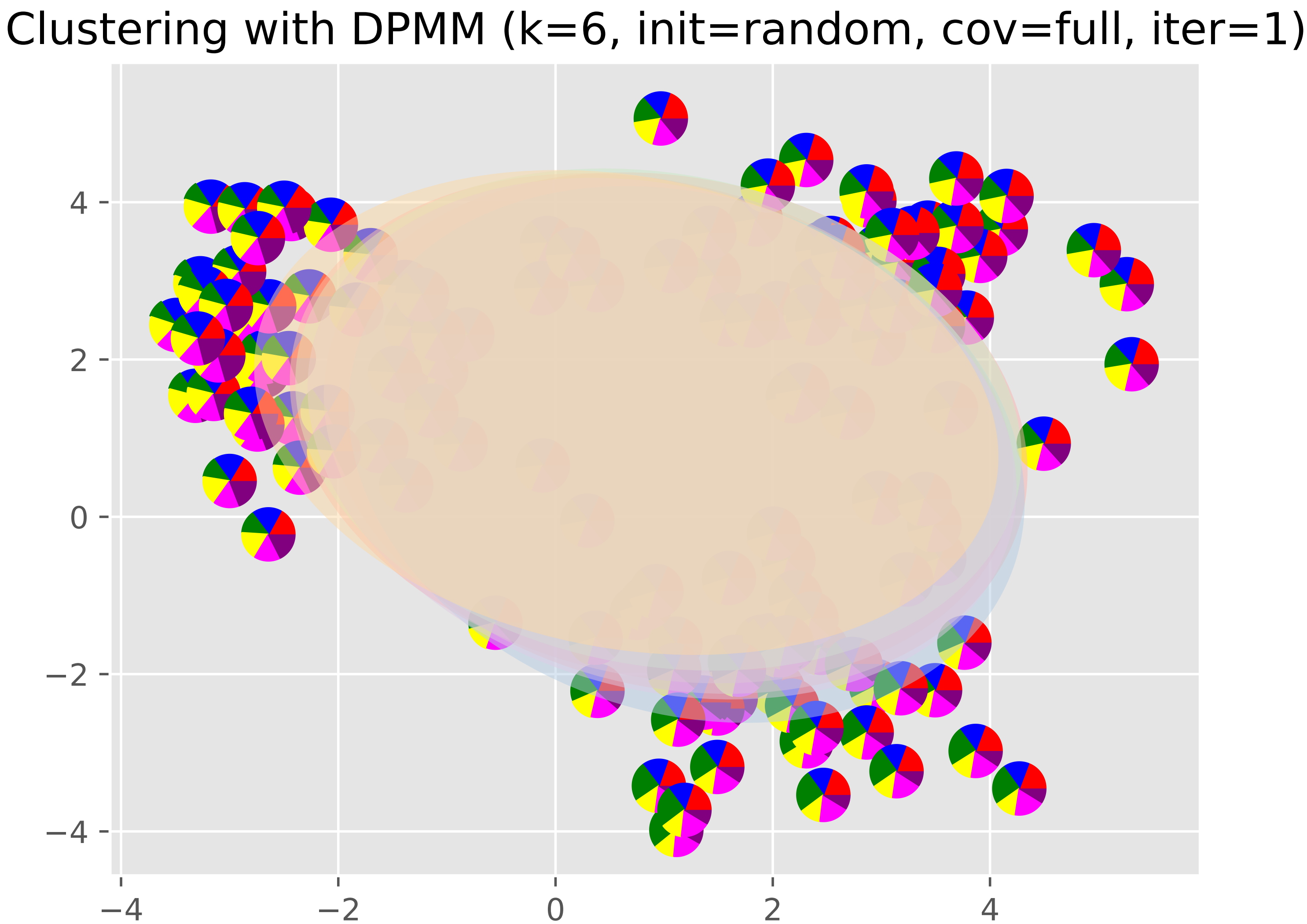


Clustering with GMM (k=6, init=random, cov=full, iter=39)
4
2
0
−2
−4
−4
−2
0
2
4

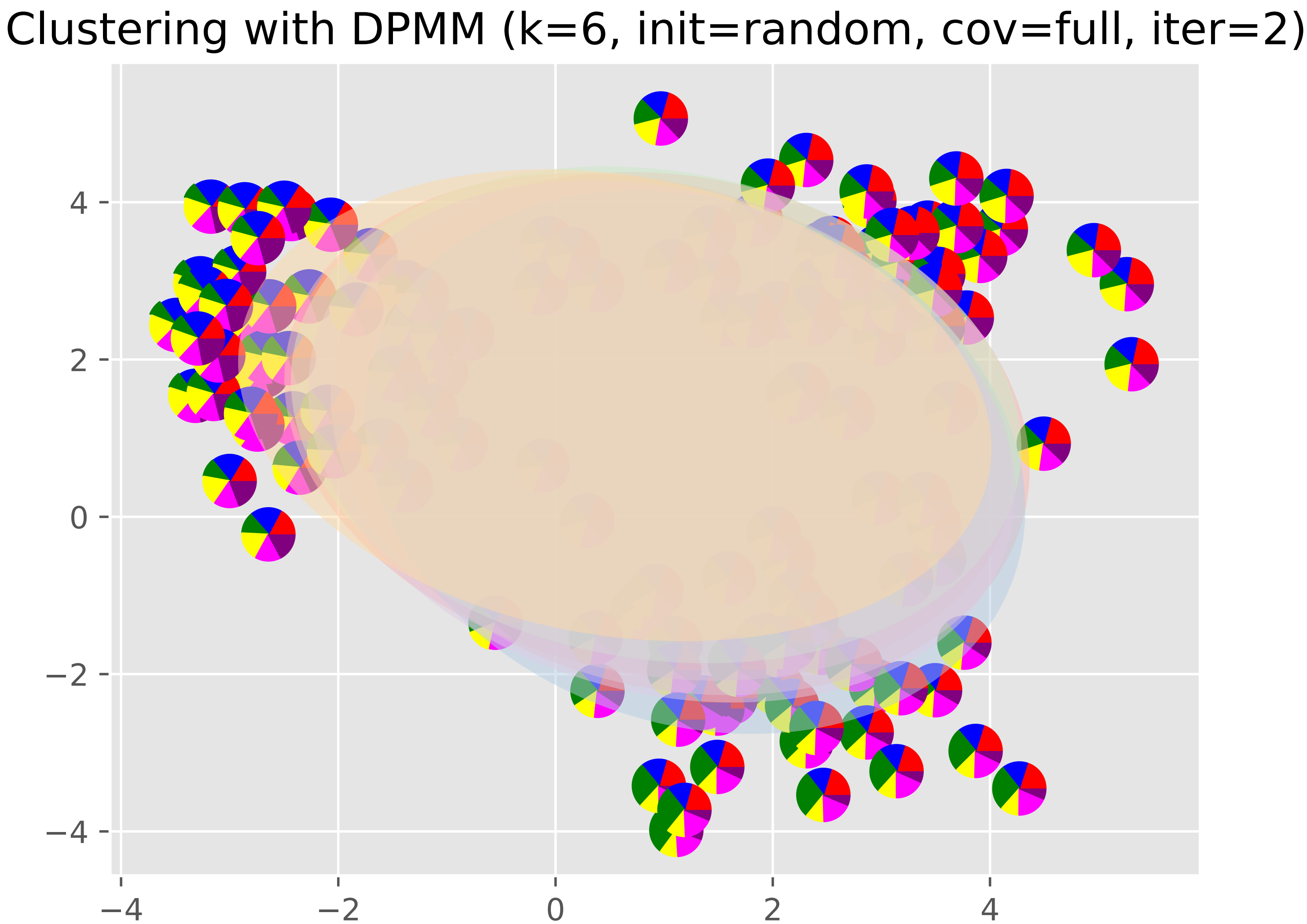
Example: DPMM



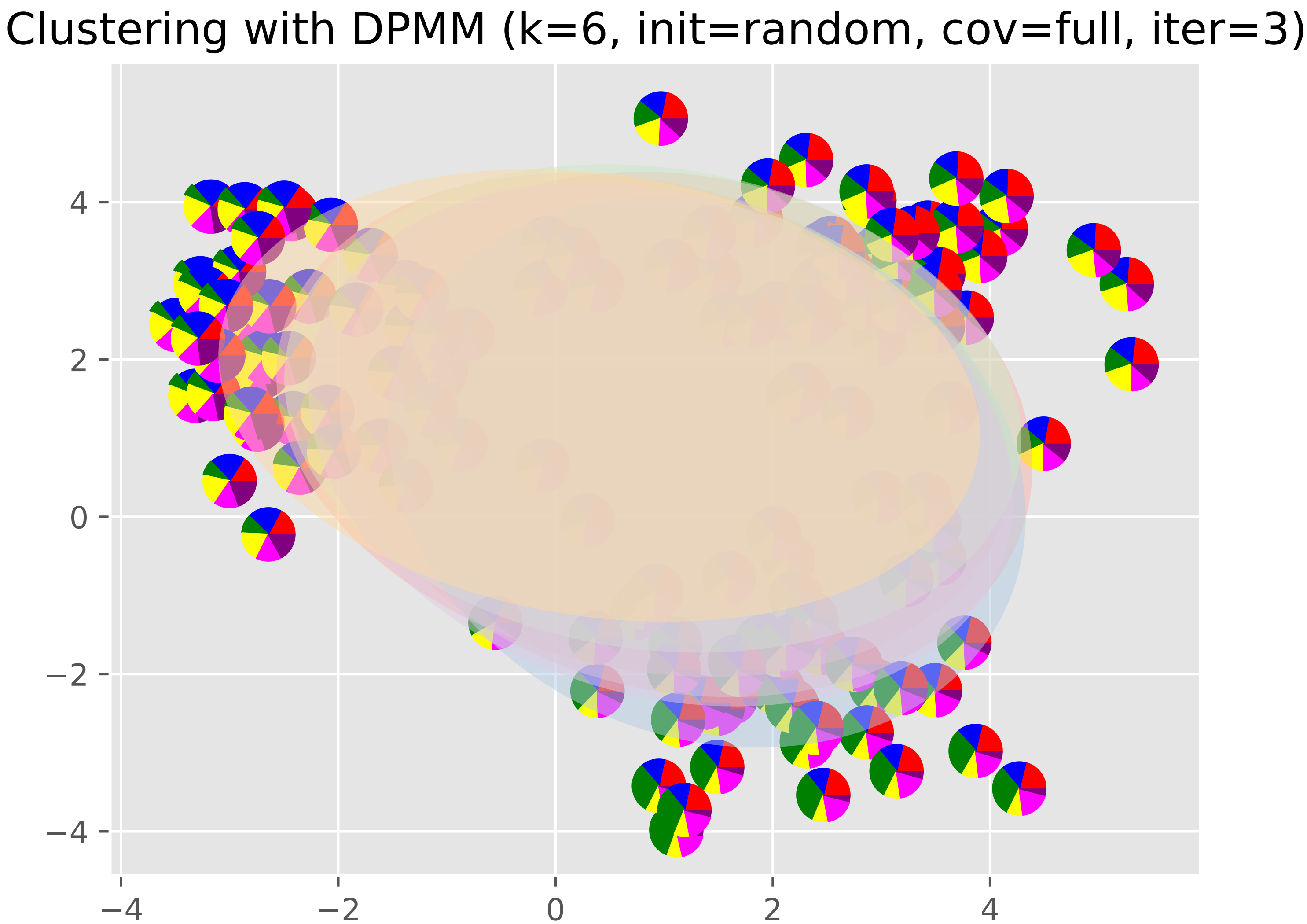
Example: DPMM



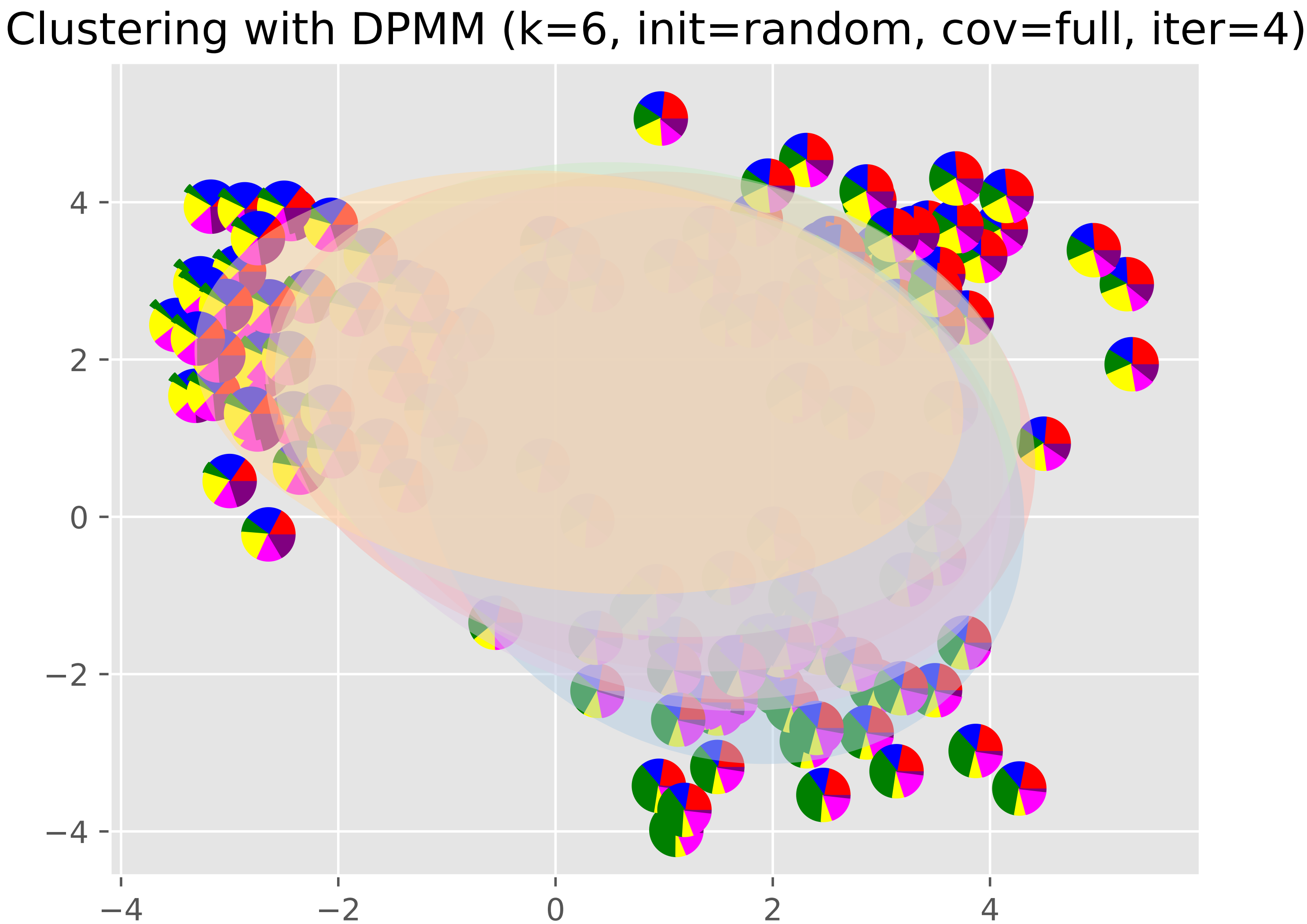
Example: DPMM



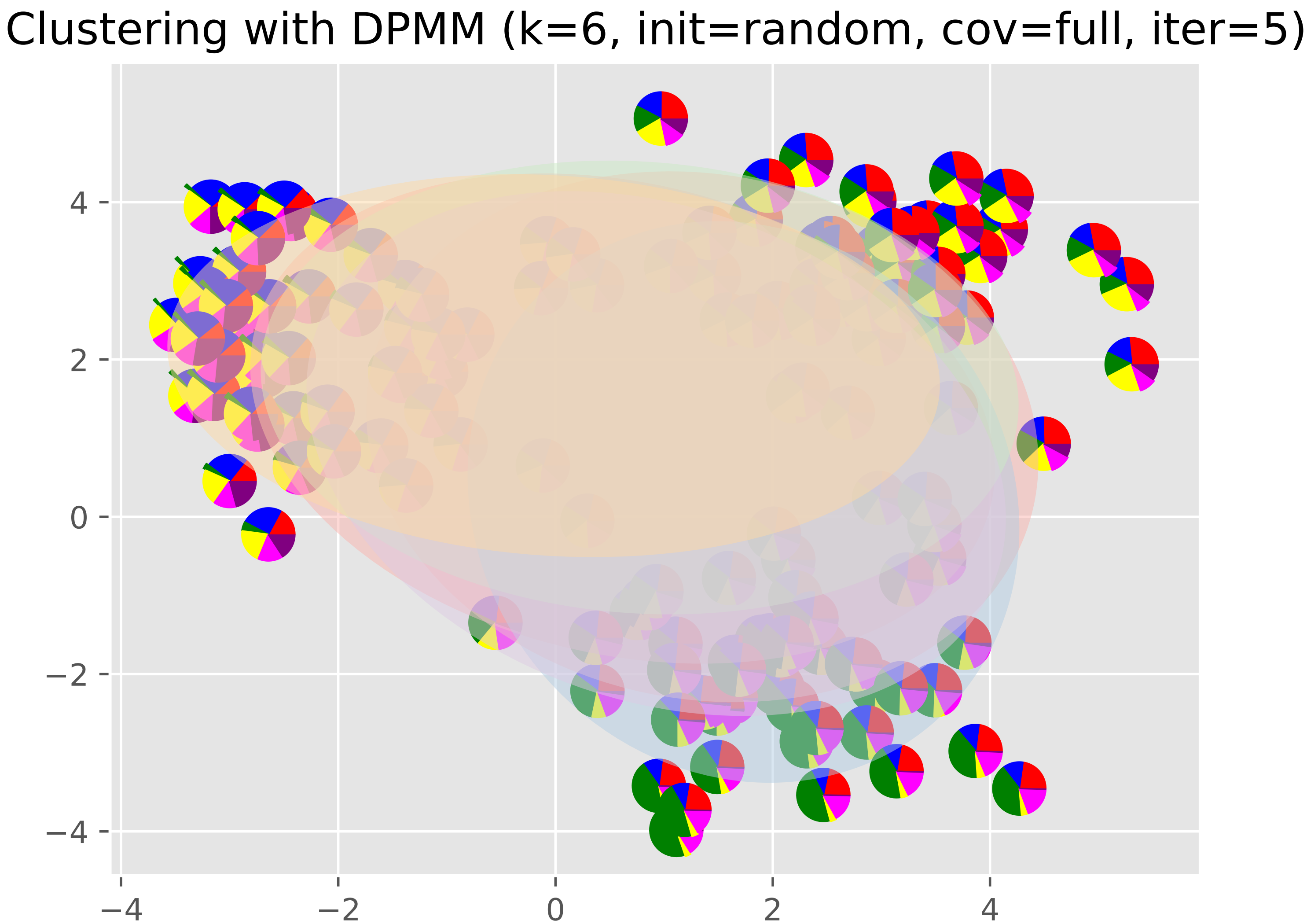
Example: DPMM



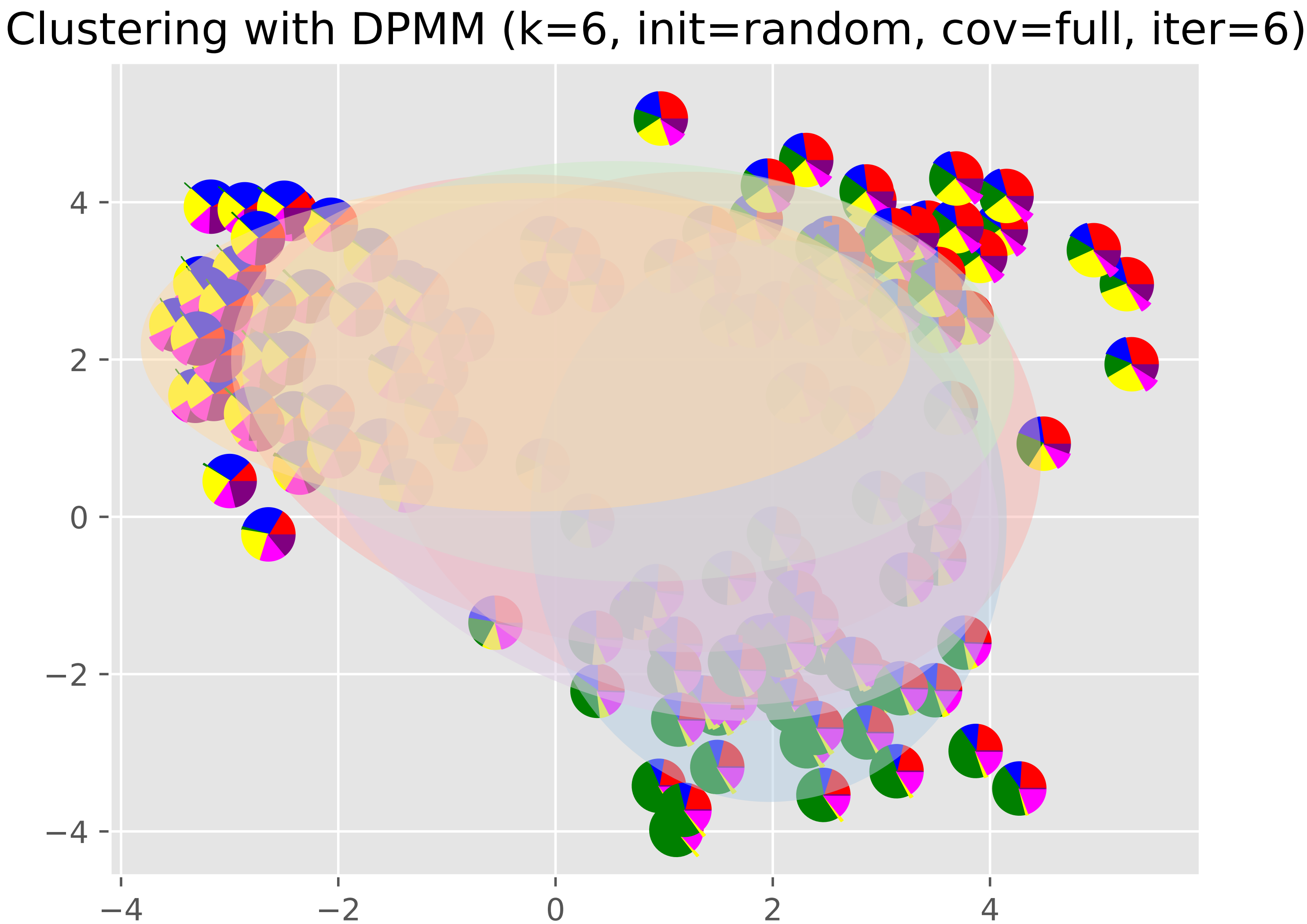
Example: DPMM



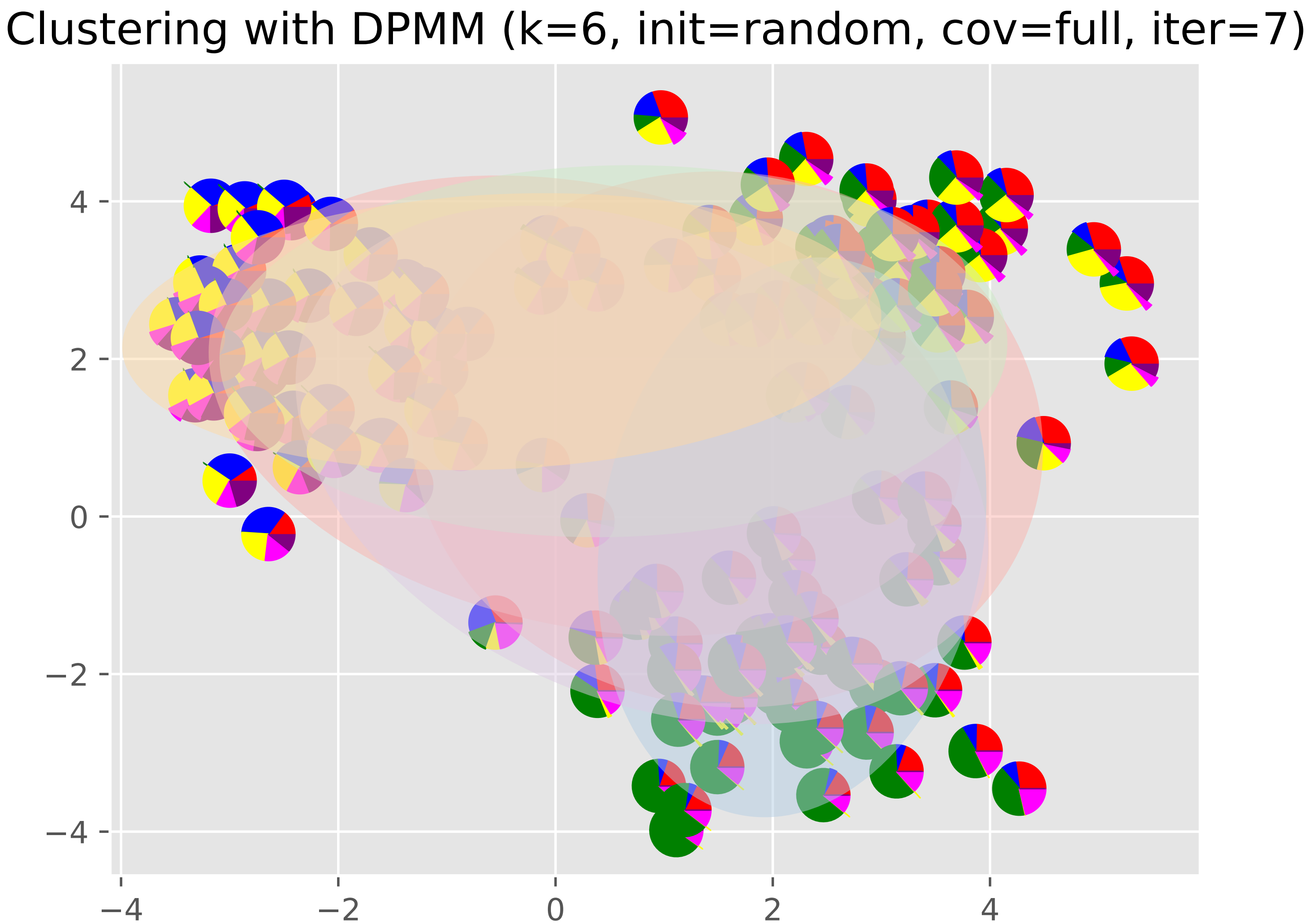
Example: DPMM



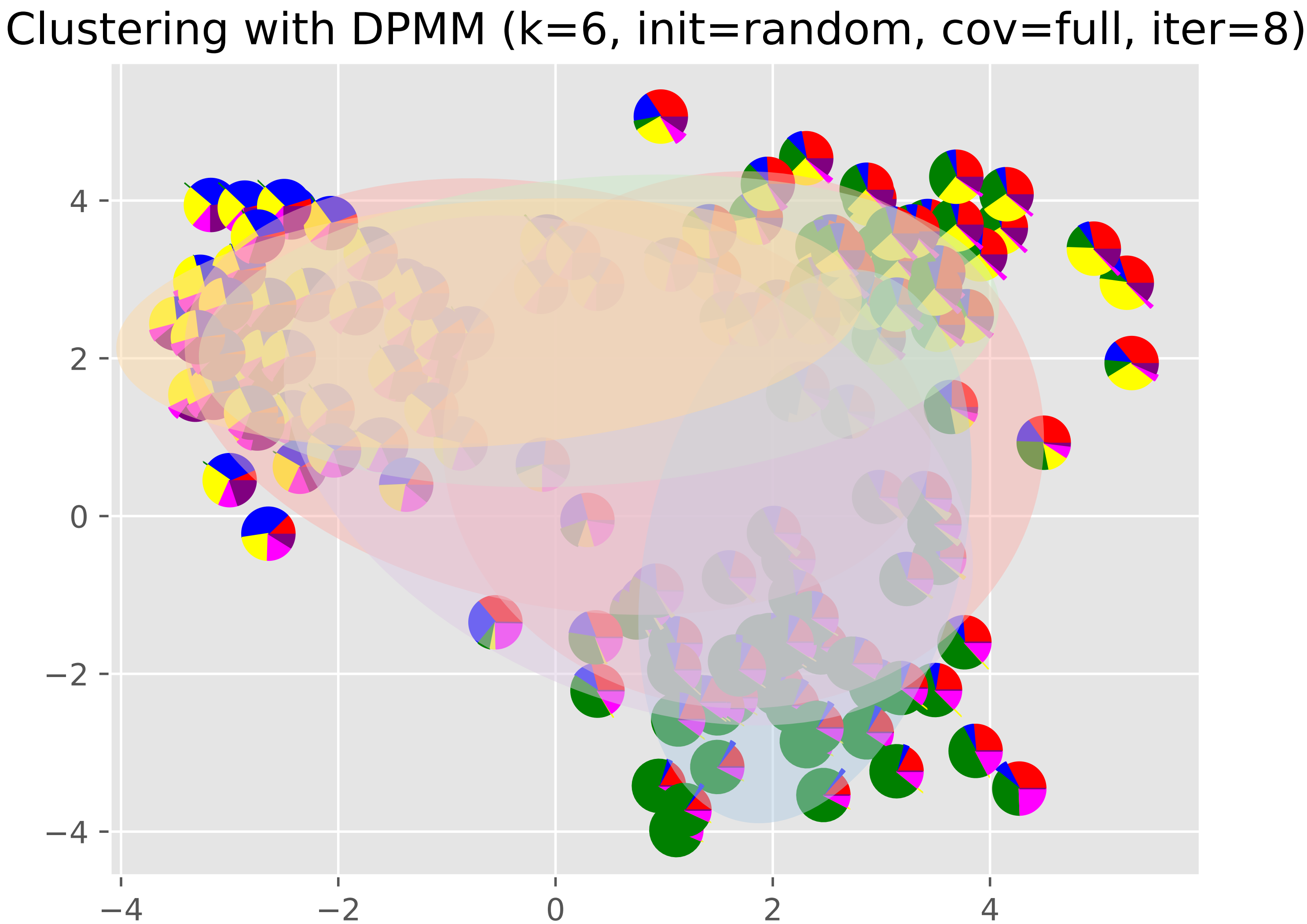
Example: DPMM



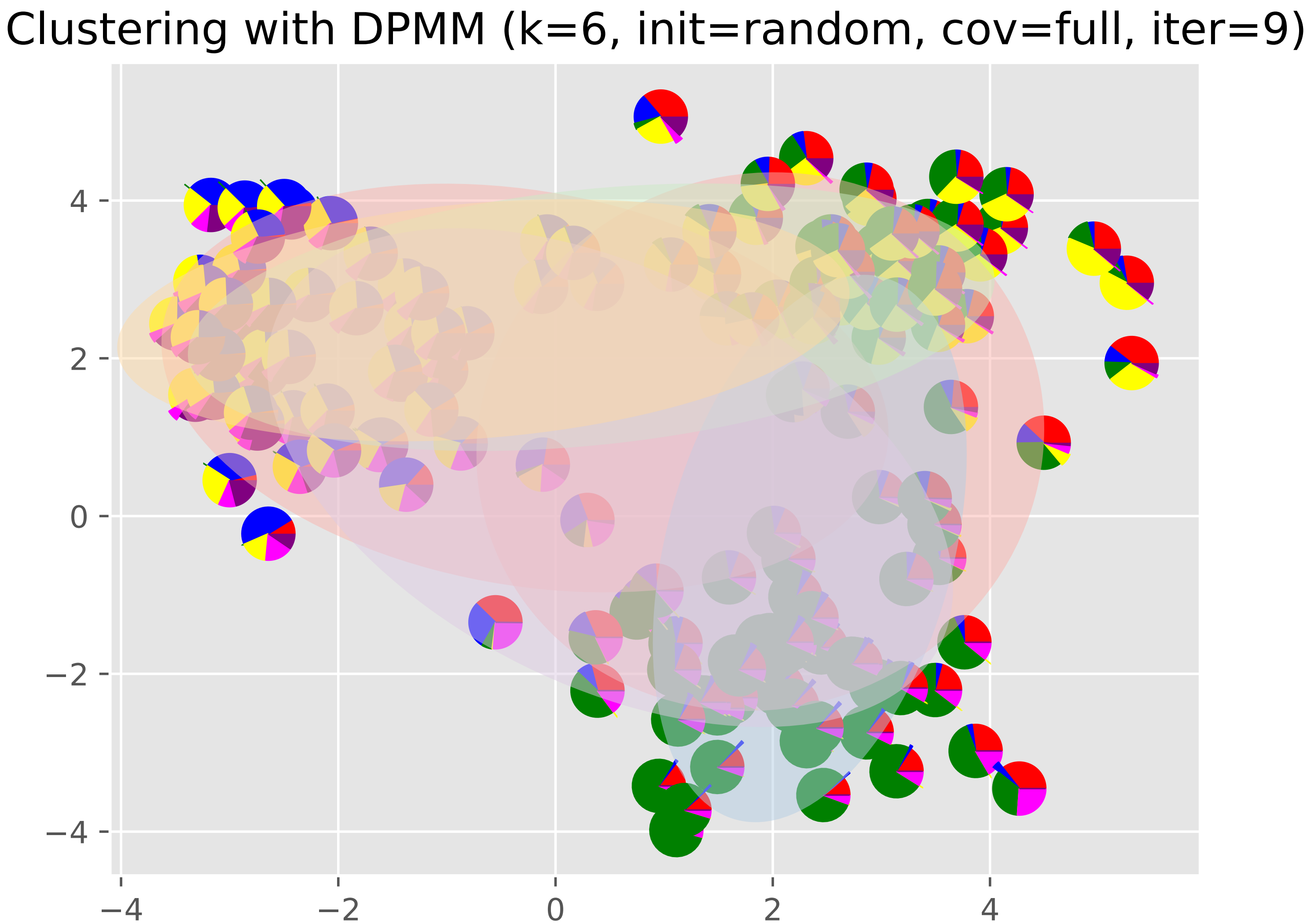
Example: DPMM



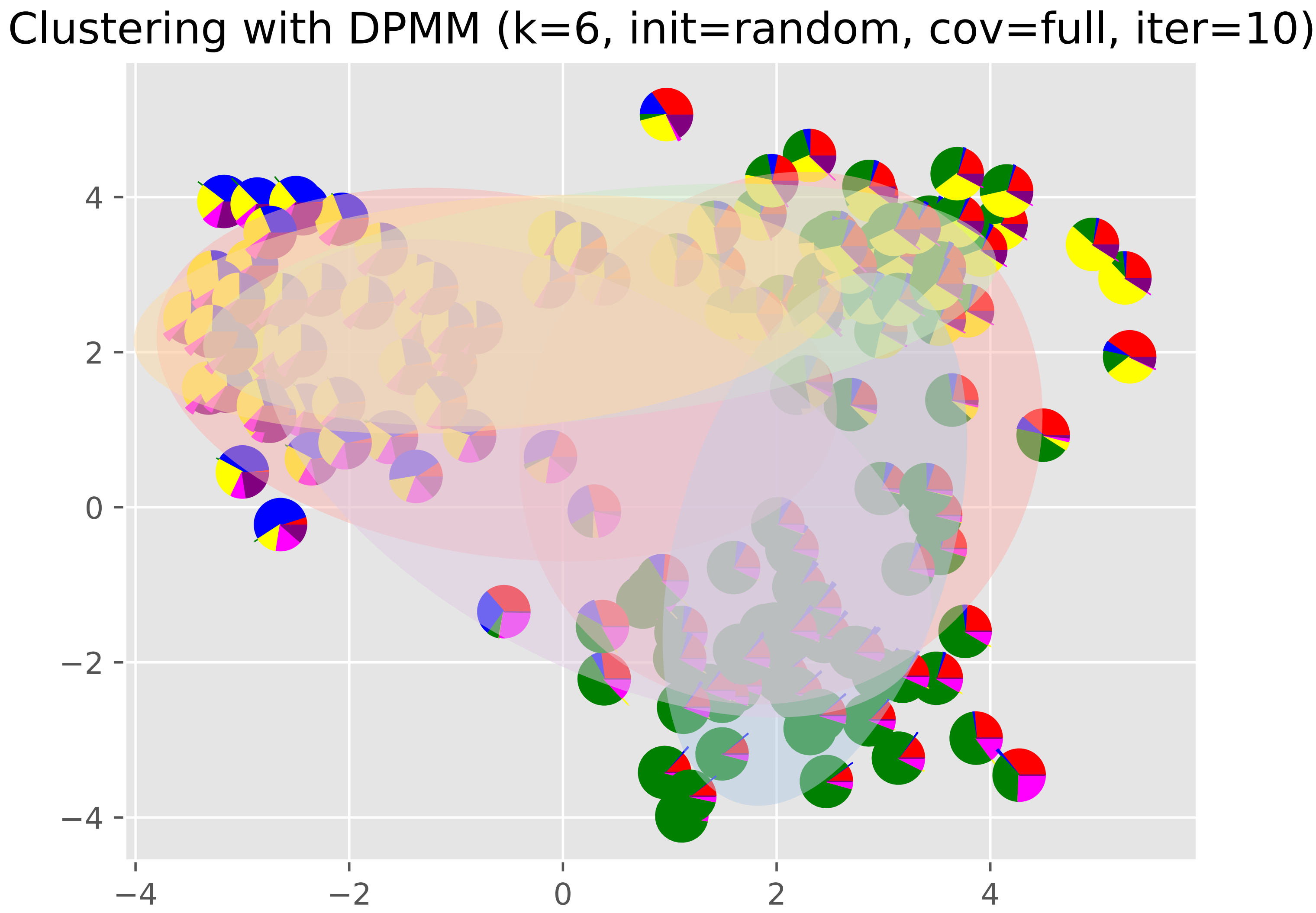
Example: DPMM



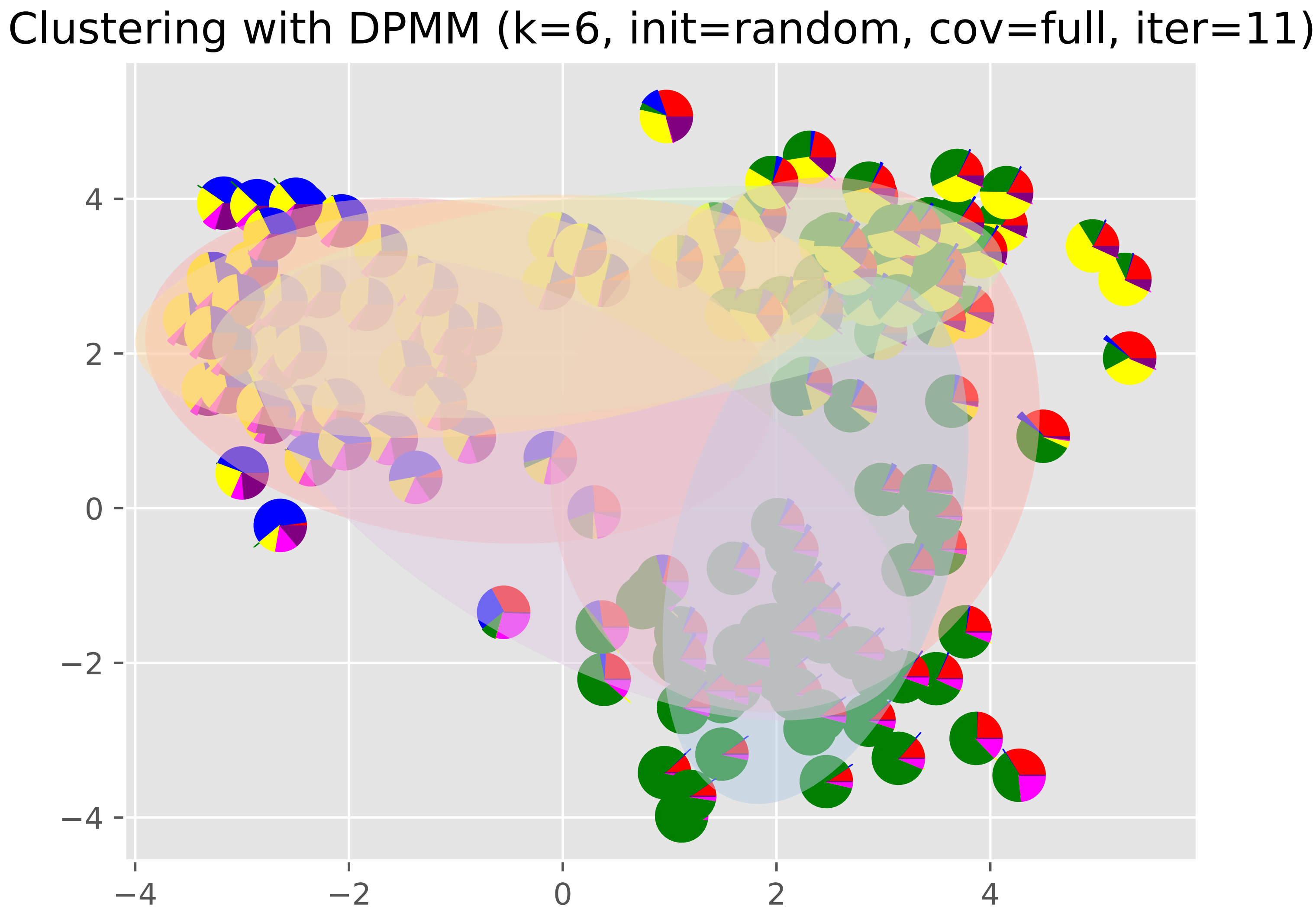
Example: DPMM



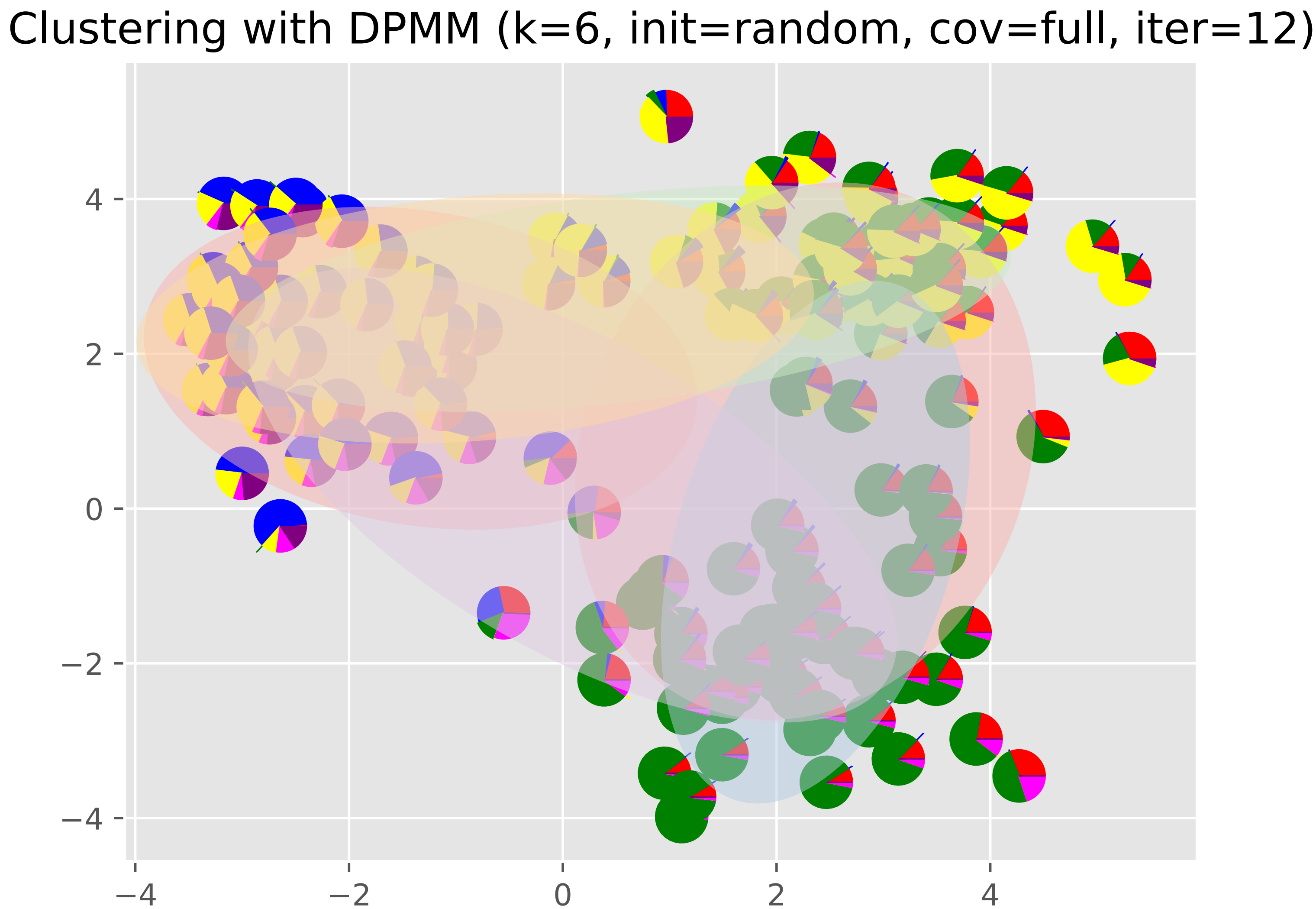
Example: DPMM



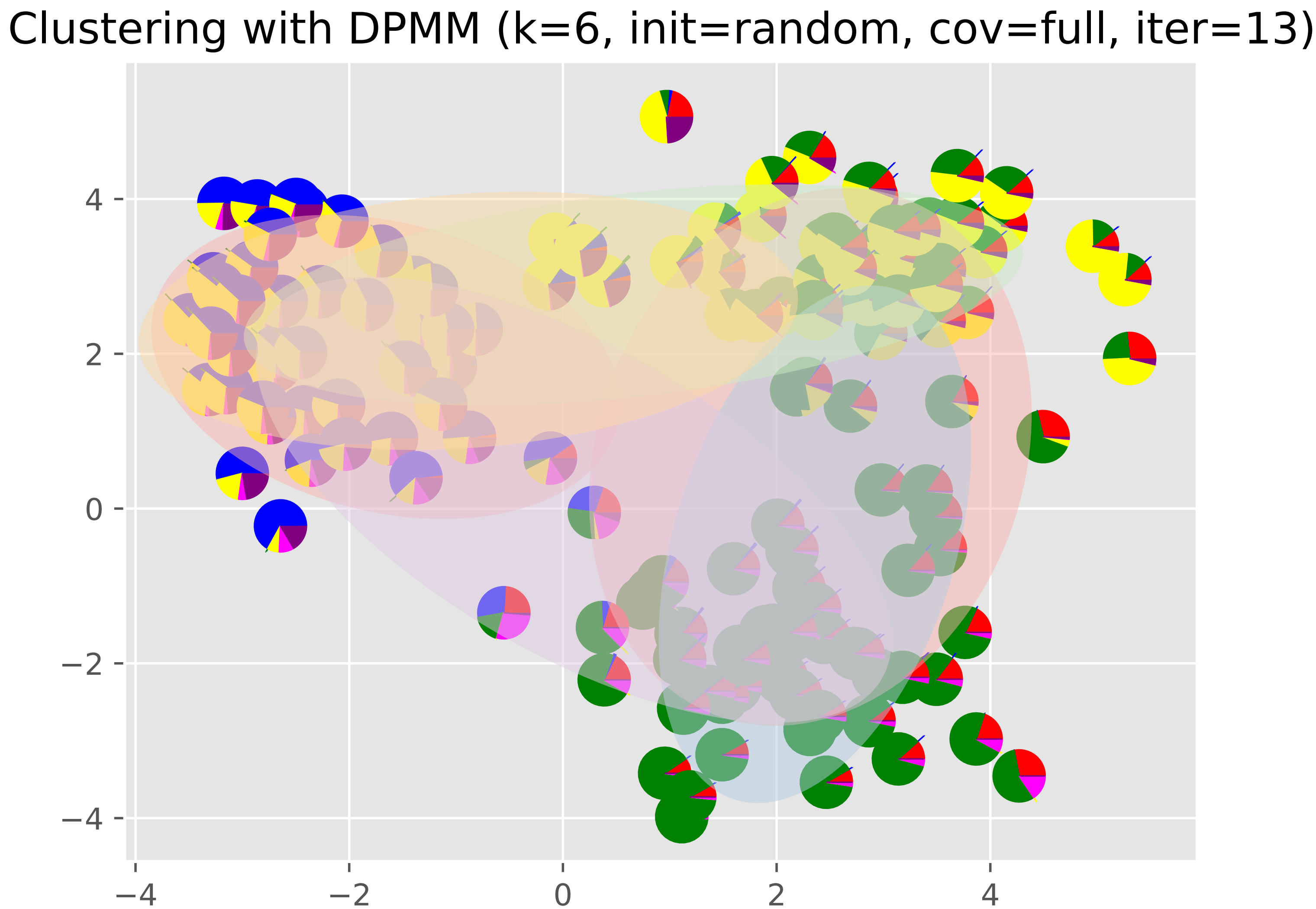
Example: DPMM



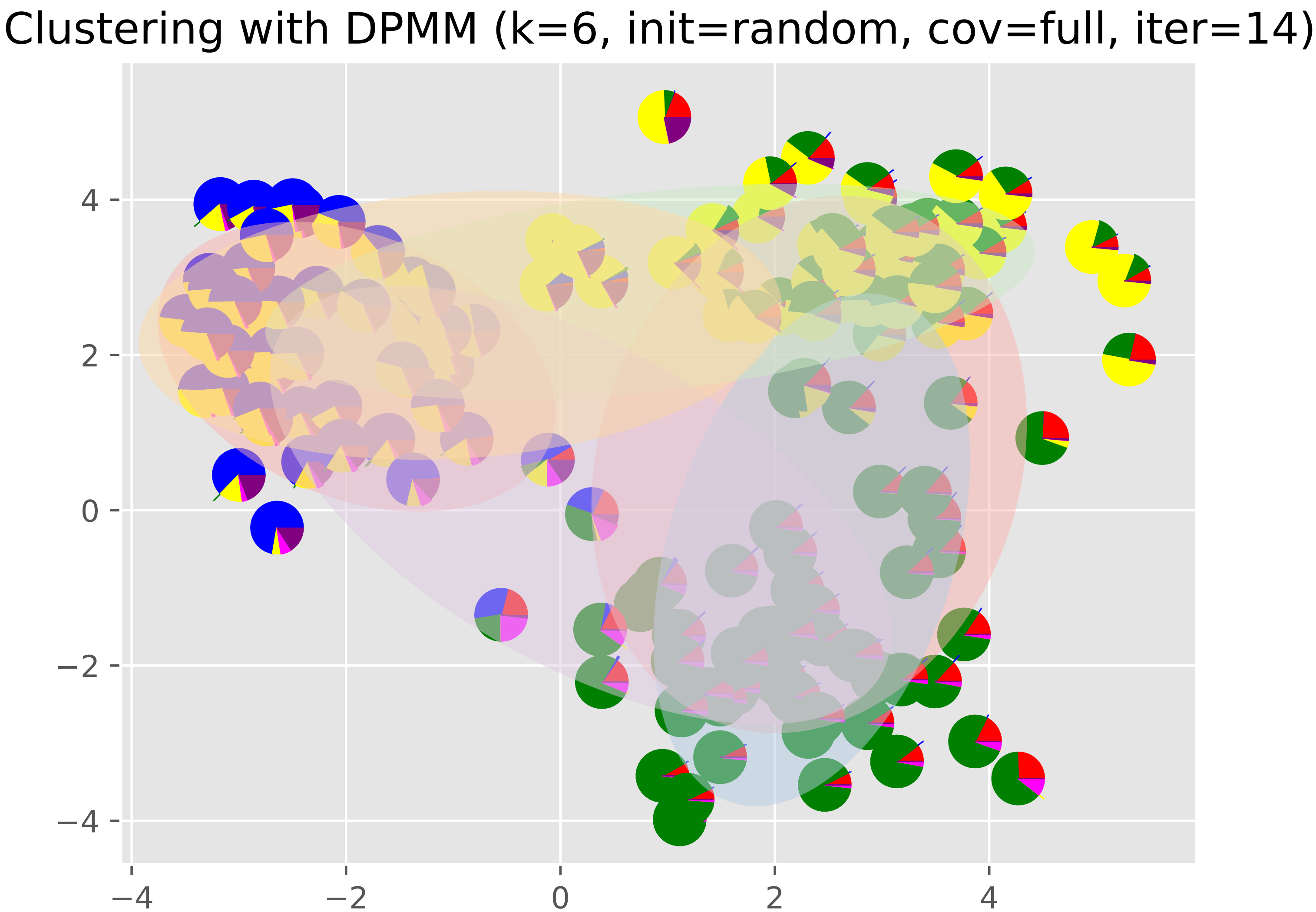
Example: DPMM



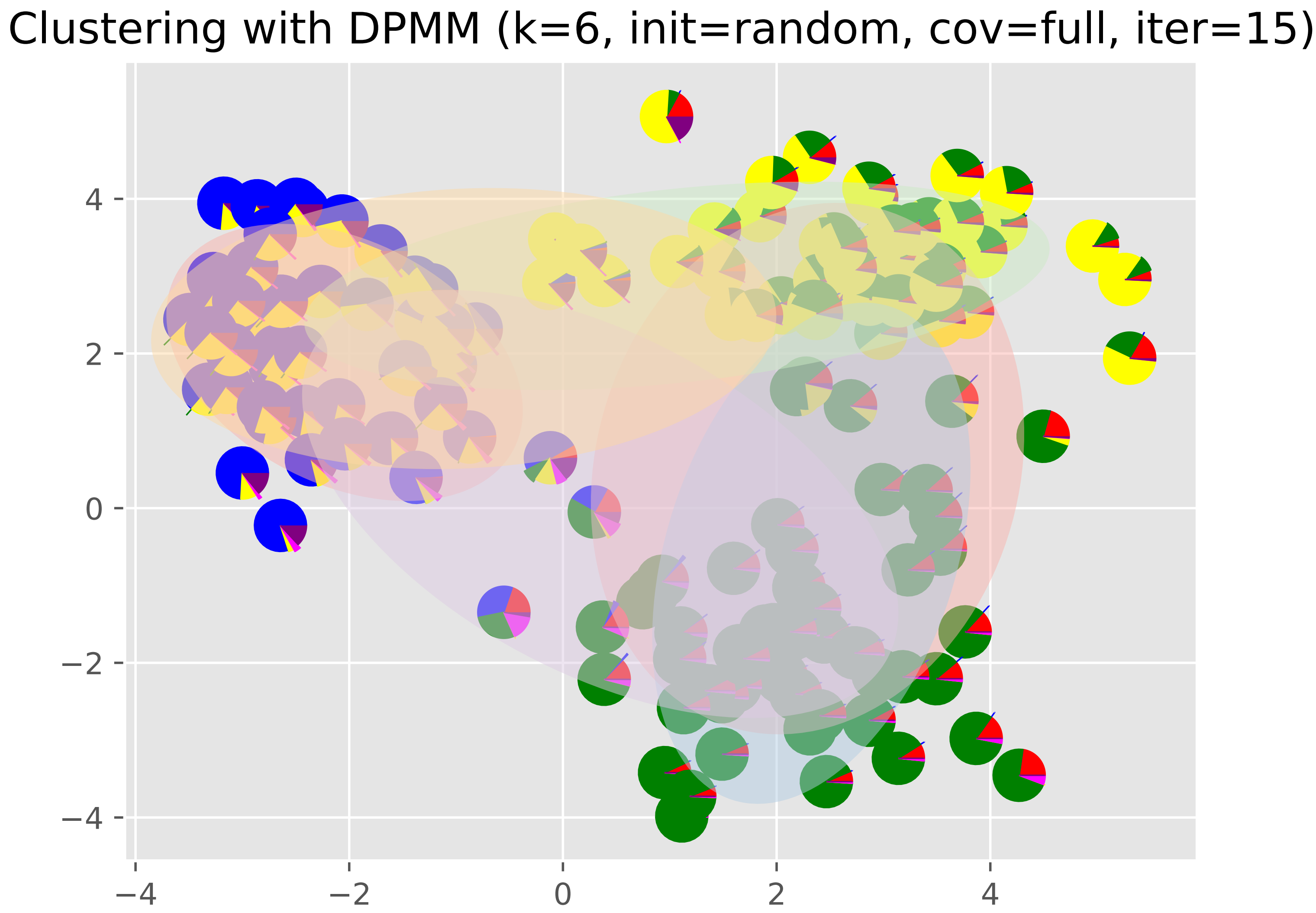
Example: DPMM



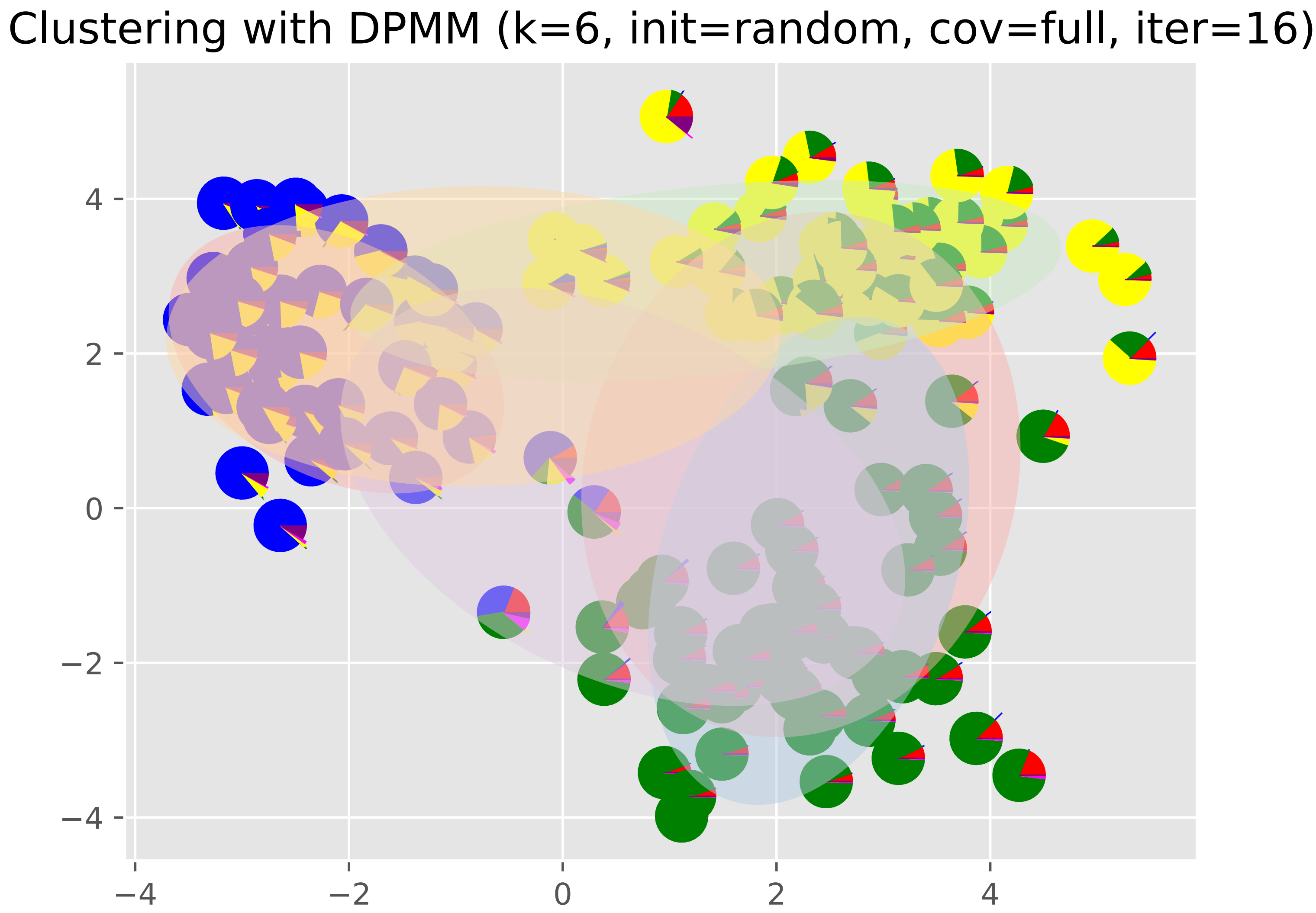
Example: DPMM



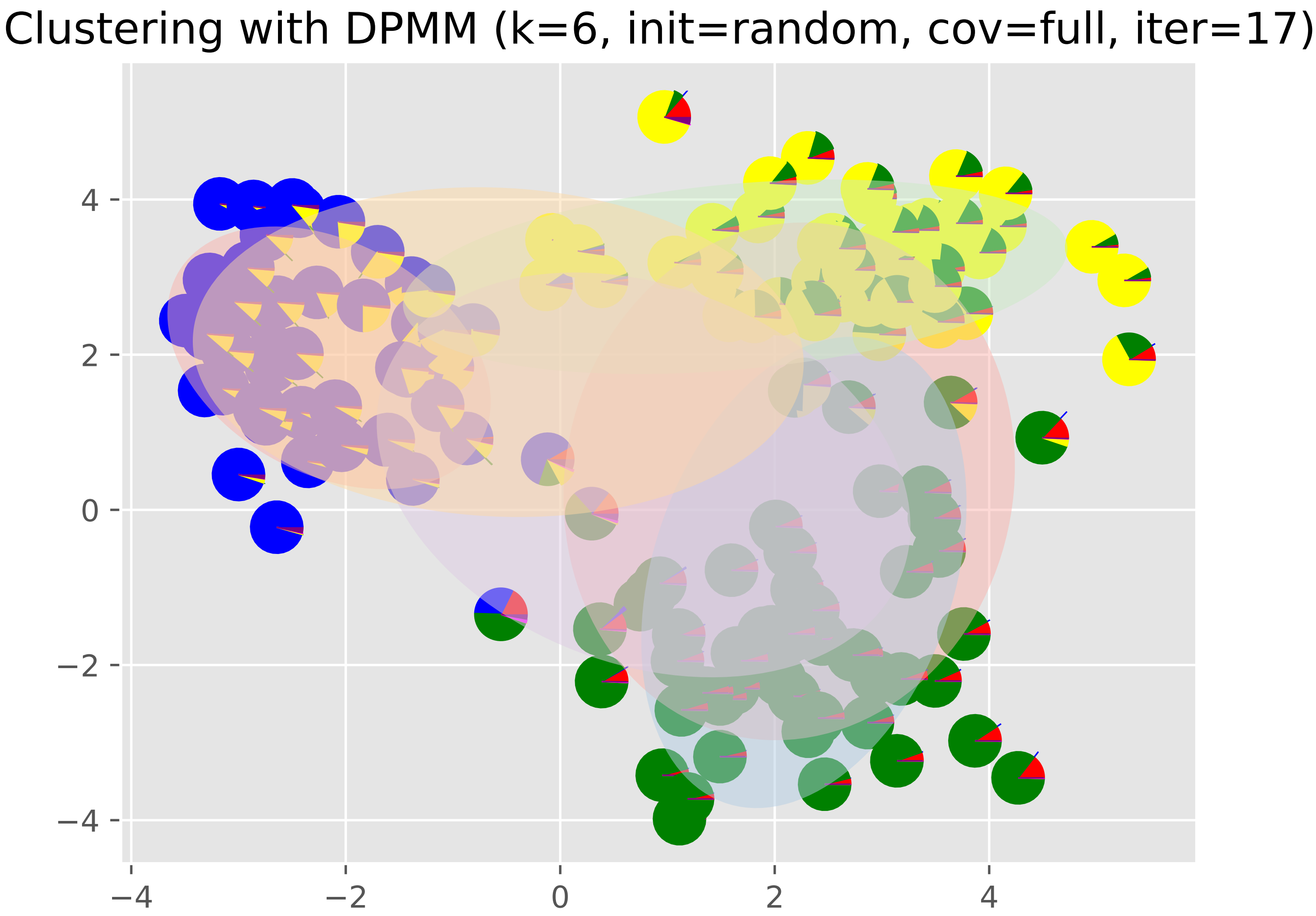
Example: DPMM



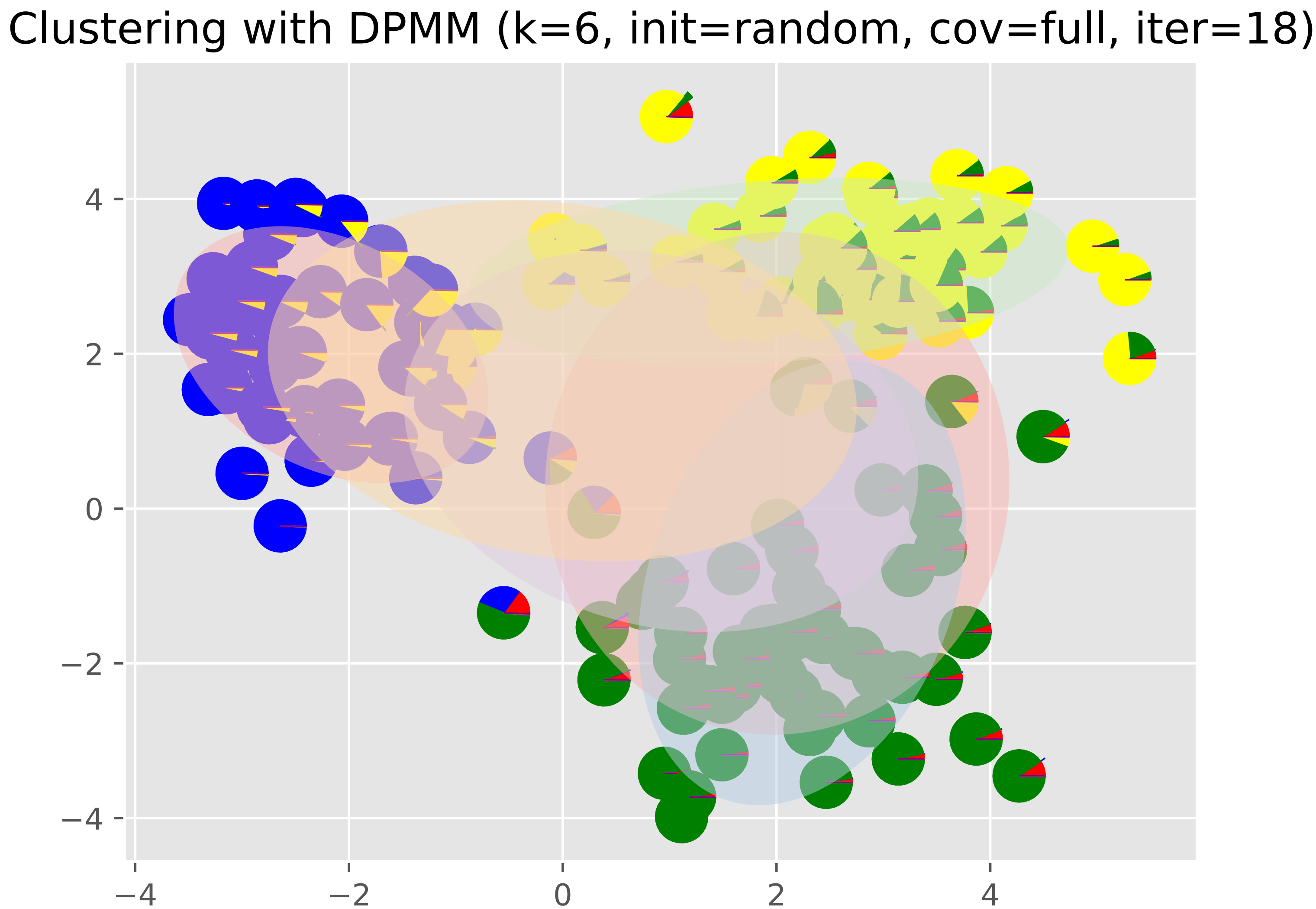
Example: DPMM



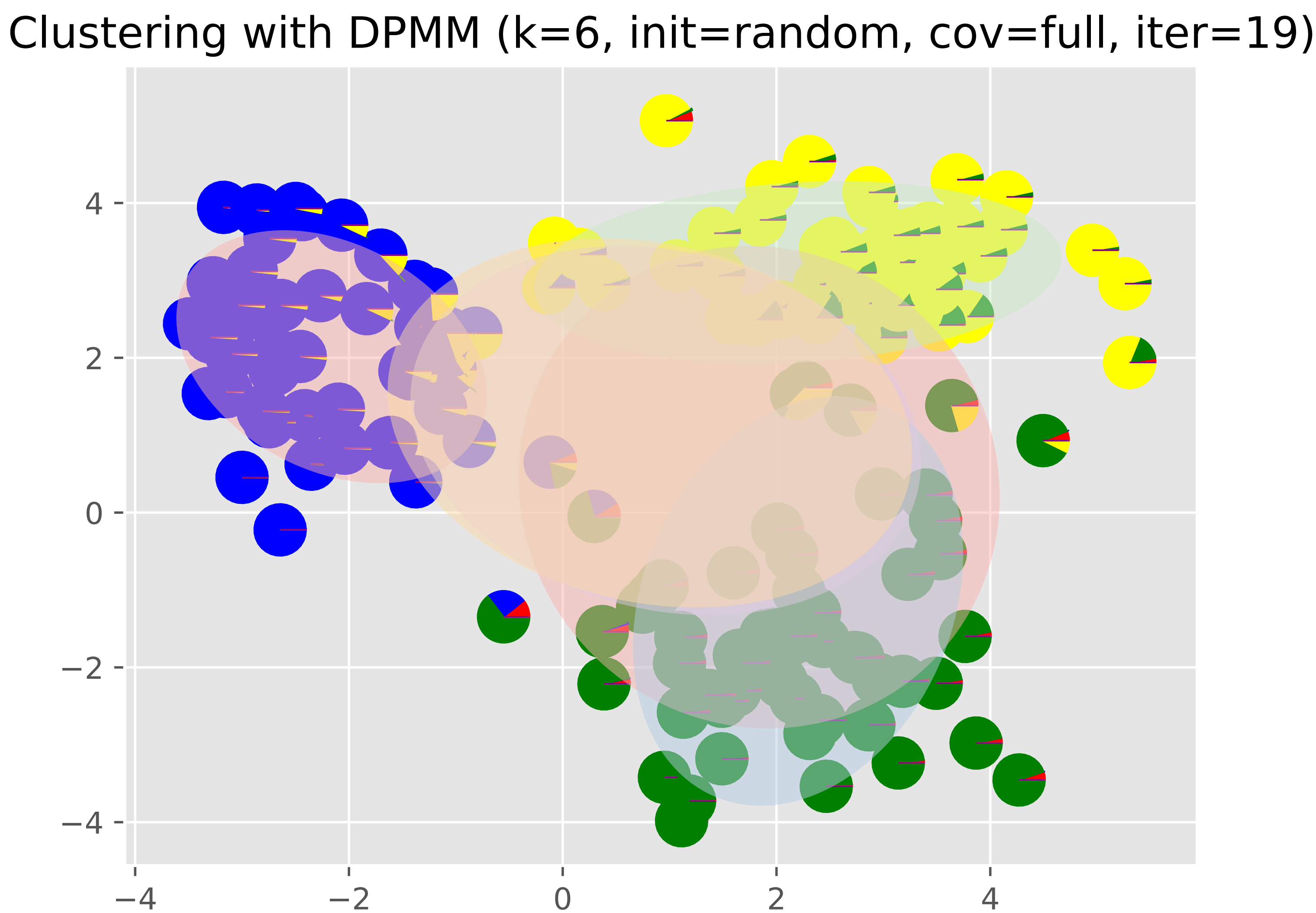
Example: DPMM



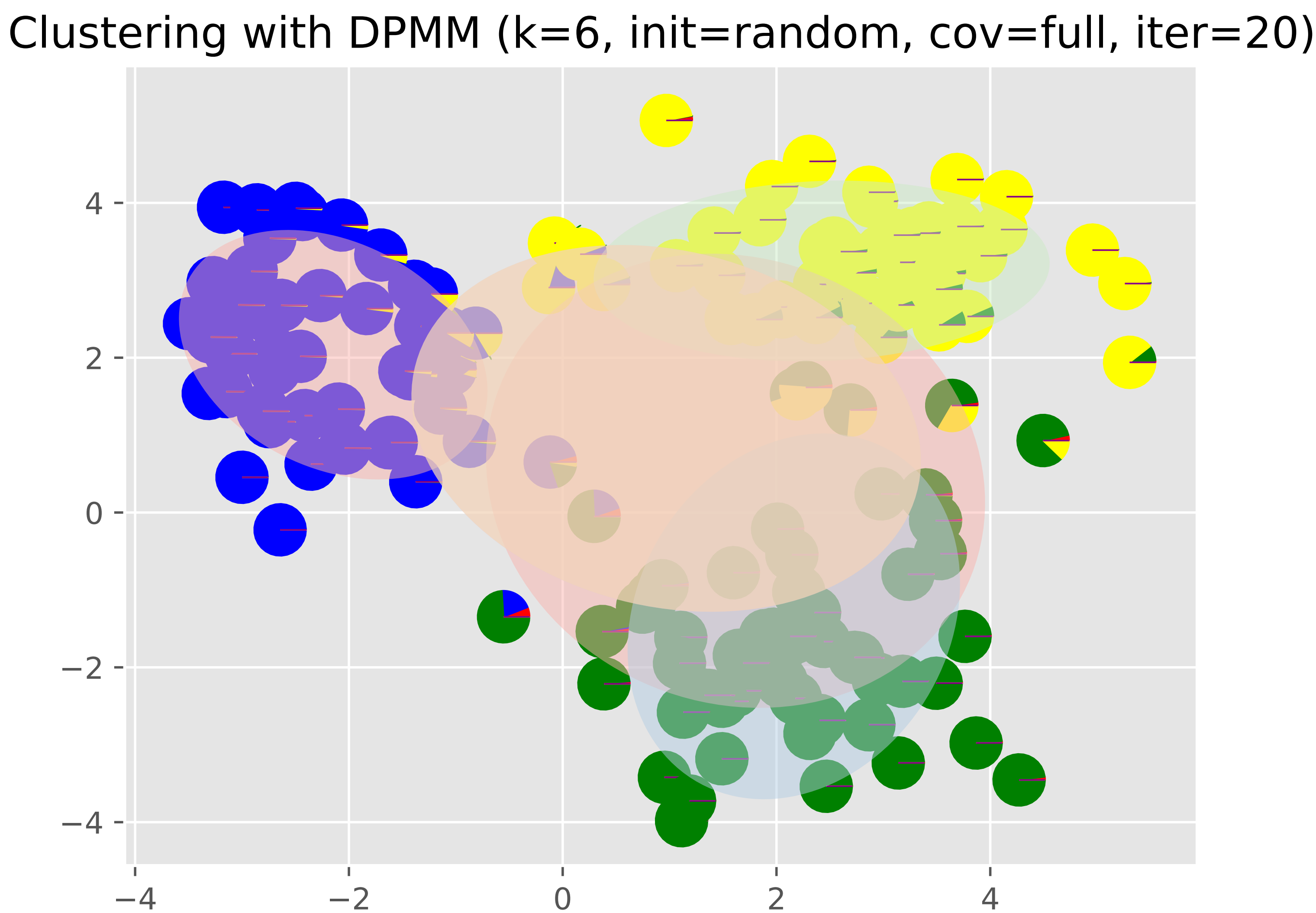
Example: DPMM



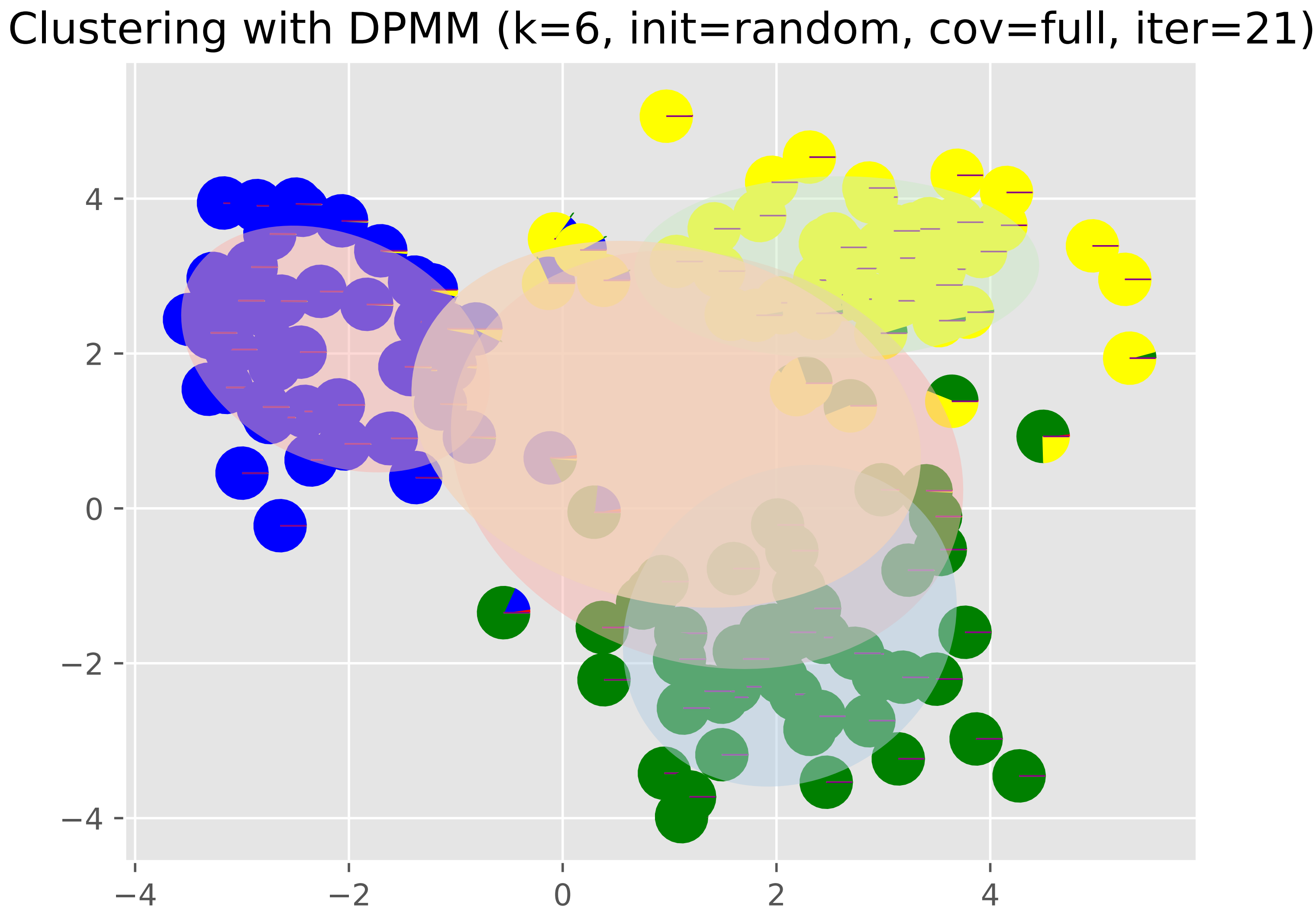
Example: DPMM



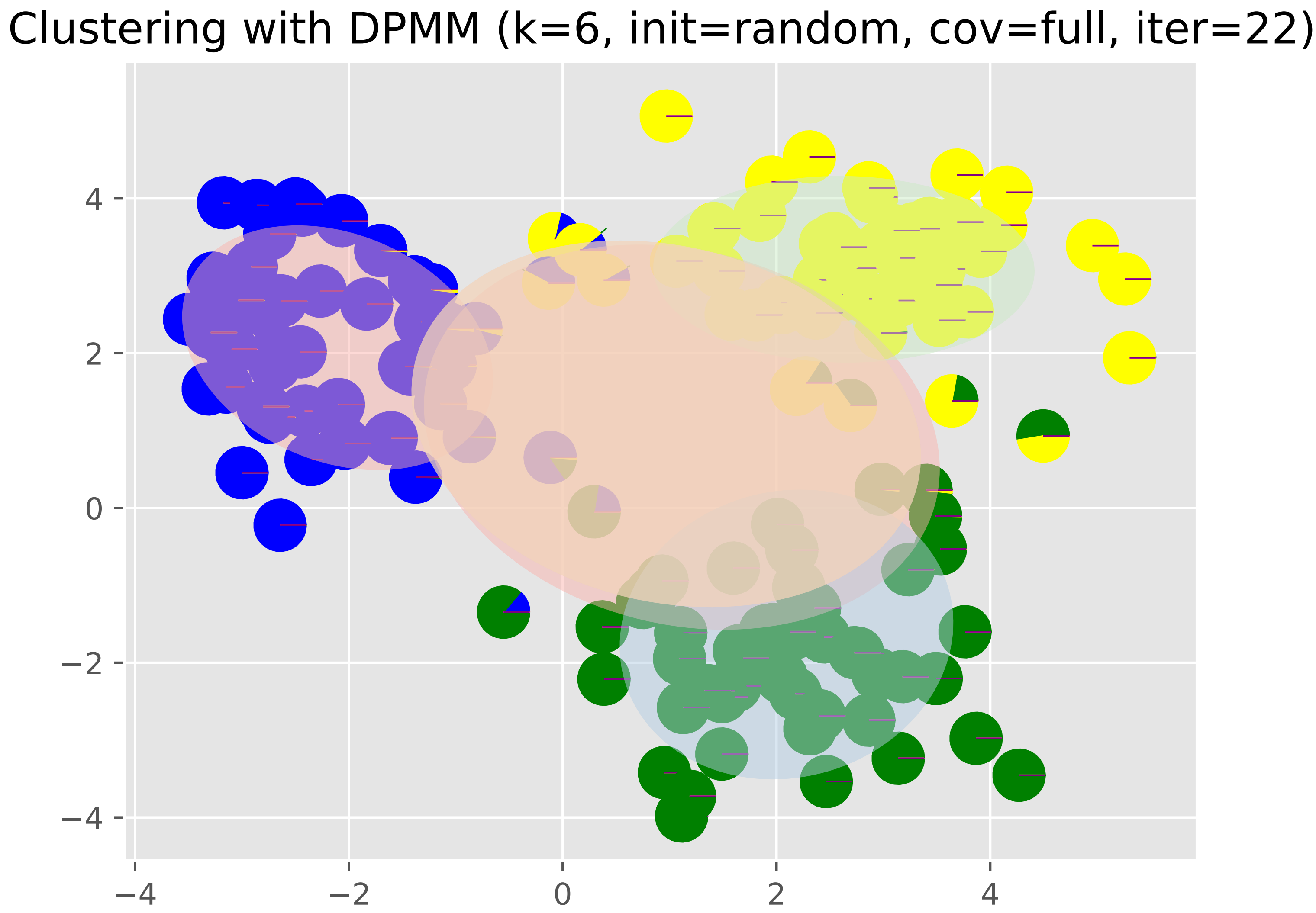
Example: DPMM



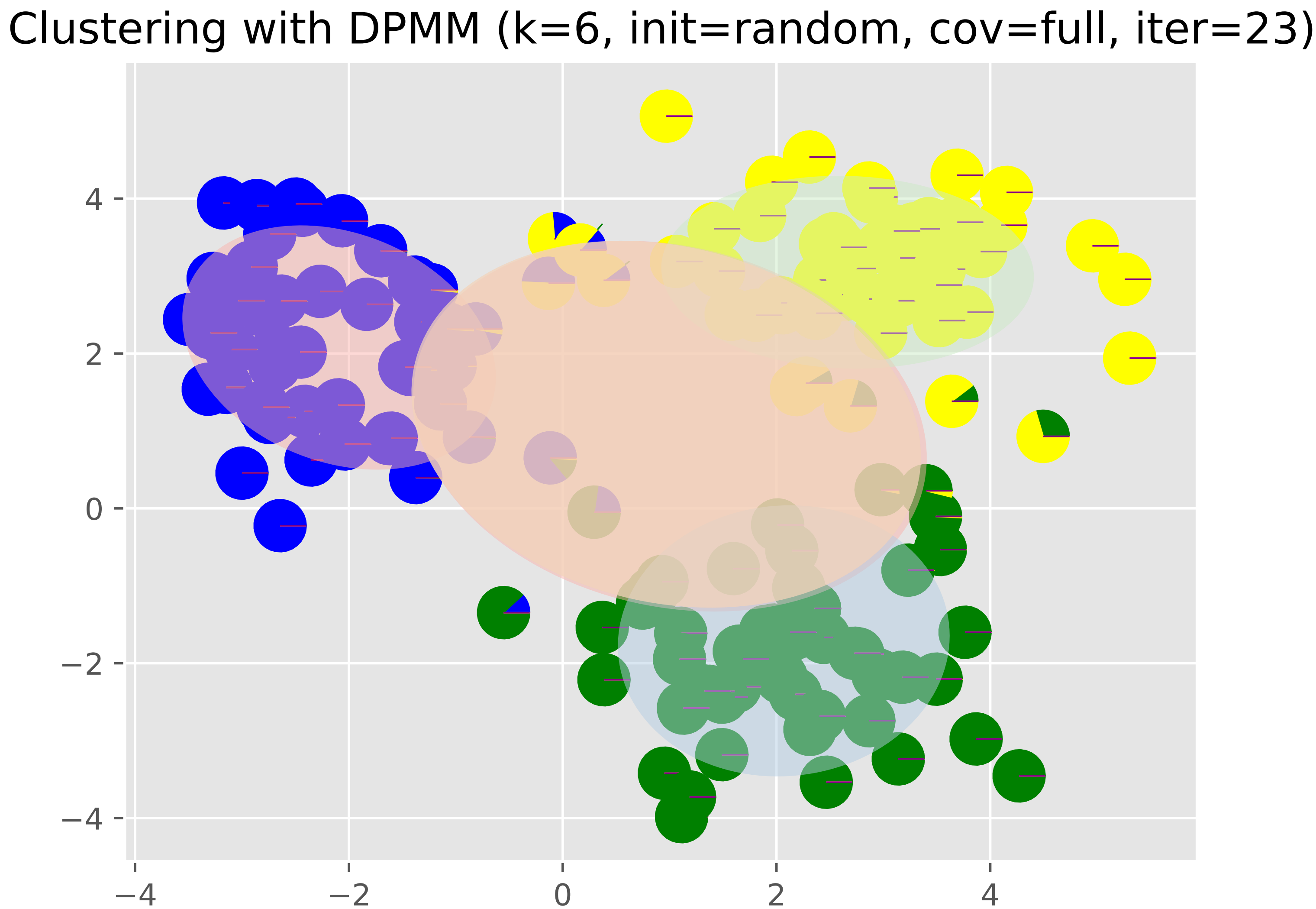
Example: DPMM



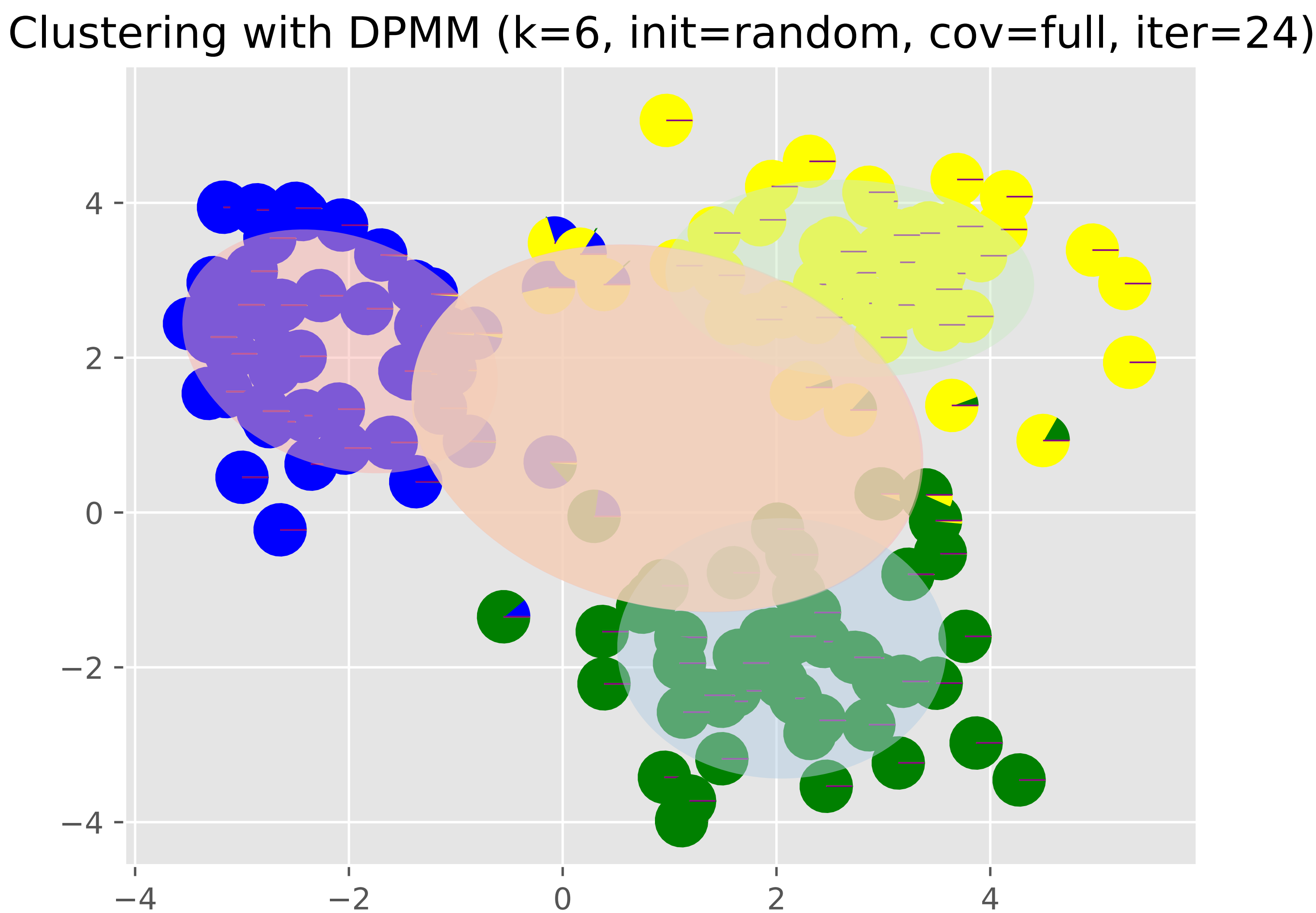
Example: DPMM



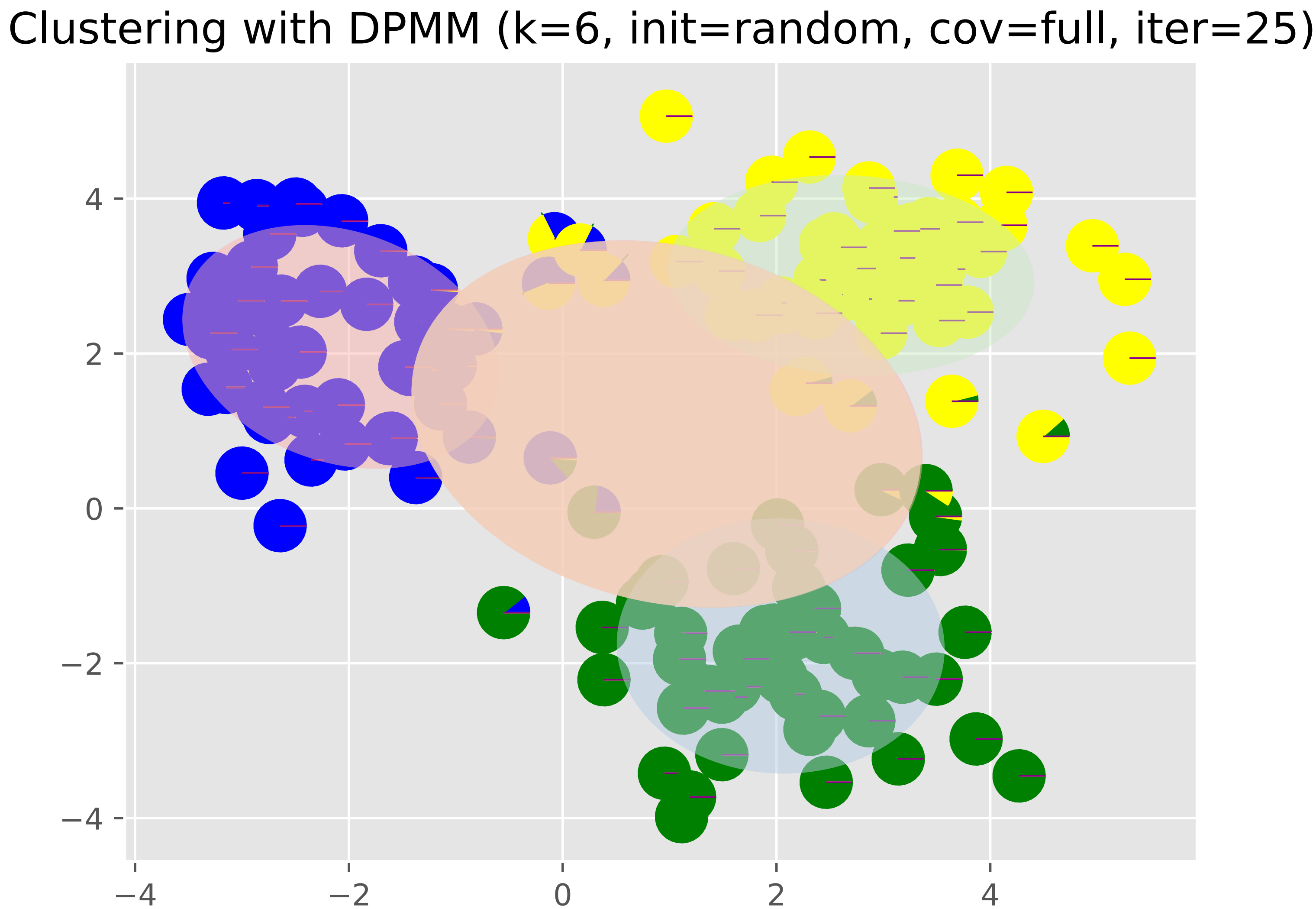
Example: DPMM



Example: DPMM



Example: DPMM



HIERARCHICAL DIRICHLET PROCESS (HDP)

Related Models

- Hierarchical Dirichlet Process Mixture Model (HDP-MM)
- Infinite HMM
- Infinite PCFG

HDP-MM

- In LDA, we have *M* independent samples from a Dirichlet distribution.
- The weights are different, but the topics are fixed to be the same.
- If we replace the Dirichlet distributions with Dirichlet processes, each atom of each Dirichlet process will pick a topic *independently* of the other topics.

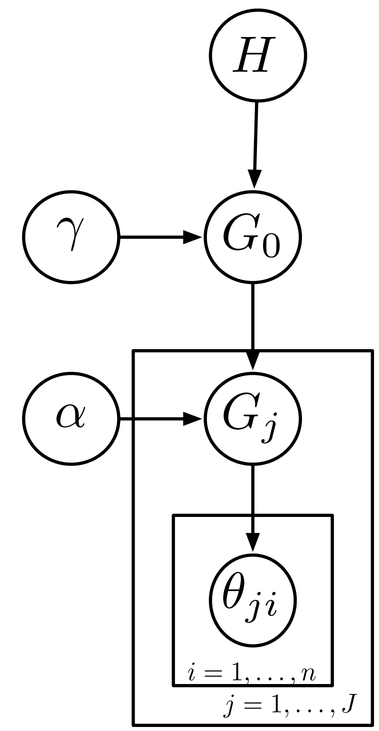
- Because the base measure is *continuous*, we have zero probability of picking the same topic twice.
- If we want to pick the same topic twice, we need to use a *discrete* base measure.
- For example, if we chose the base measure to be $H = \sum_{k=1}^{K} \alpha_k \delta_{\beta_k}$ then we would have LDA again.

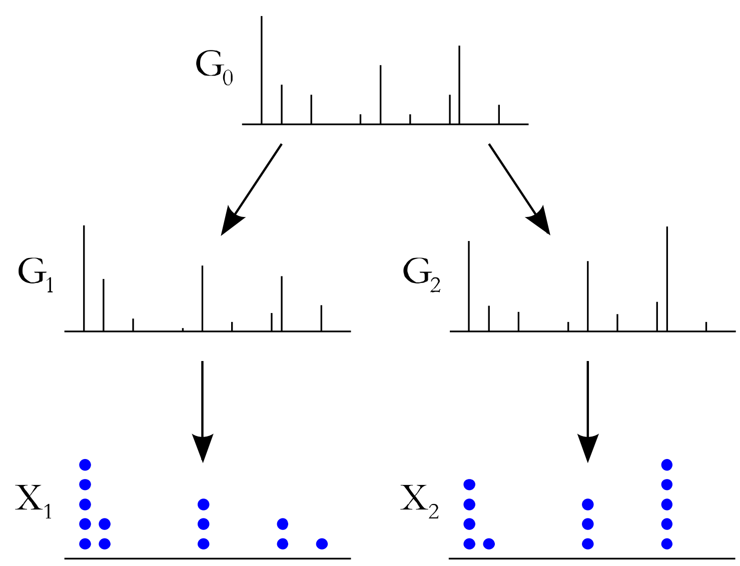
- We want there to be an infinite number of topics, so we want an *infinite, discrete* base measure.
- We want the location of the topics to be random, so we want an *infinite, discrete, random* base measure.

HDP-MM

Hierarchical Dirichlet process:

$$G_0|\gamma, H \sim \mathrm{DP}(\gamma, H)$$
$$G_j|\alpha, G_0 \sim \mathrm{DP}(\alpha, G_0)$$
$$\theta_{ji}|G_j \sim G_j$$





HDP-MM

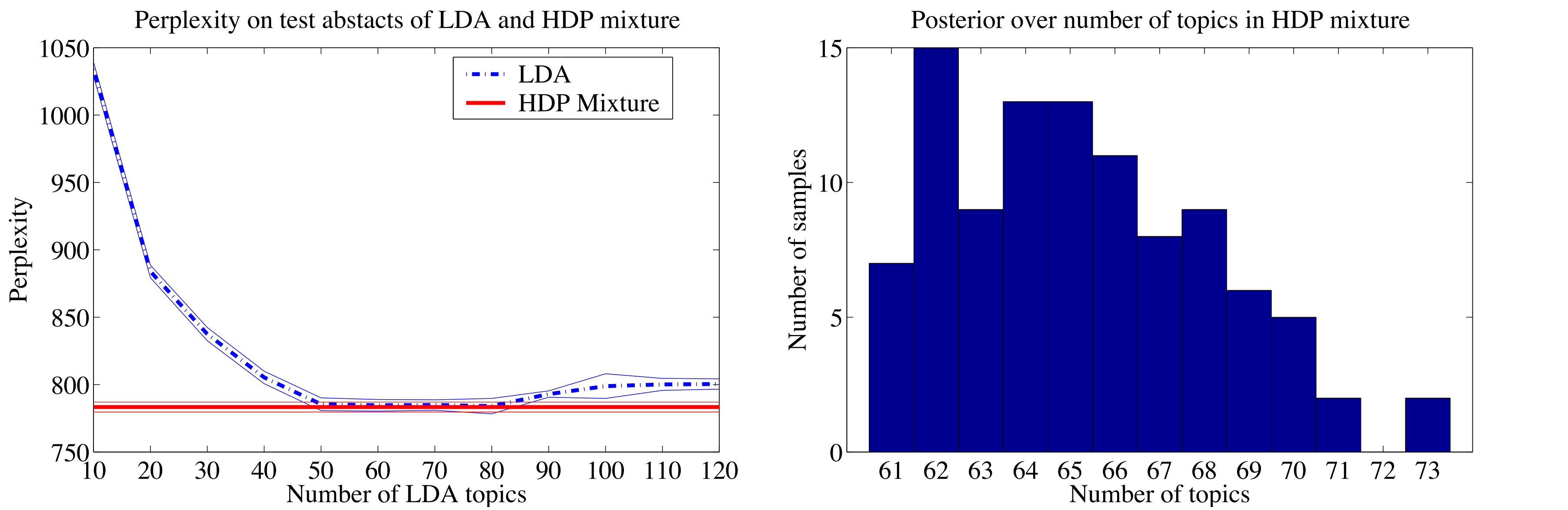
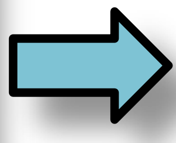


Figure 6: (Left) Comparison of latent Dirichlet allocation and the hierarchical Dirichlet process mixture. Results are averaged over 10 runs; the error bars are one standard error. (Right) Histogram of the number of topics for the hierarchical Dirichlet process mixture over 100 posterior samples.

HDP-HMM (Infinite HMM)

Number of hidden states in Infinite HMM is **countably infinite**



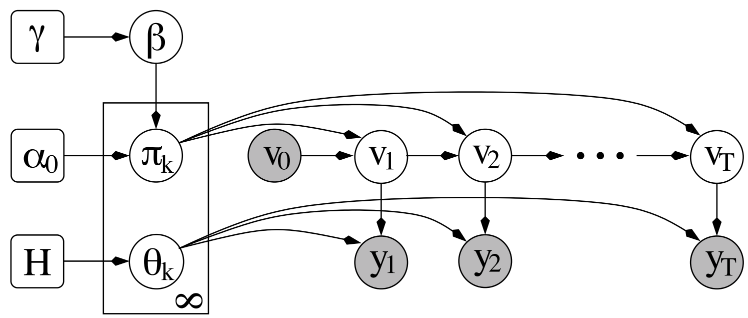


Figure 9: A hierarchical Bayesian model for the infinite hidden Markov model.

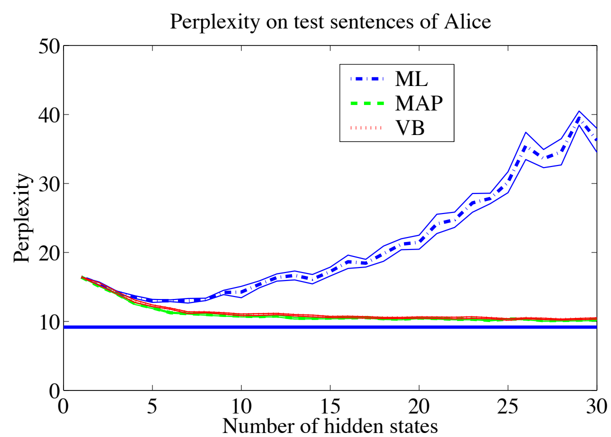


Figure 10: Comparing the infinite hidden Markov model (solid horizontal line) with ML, MAP and VB trained hidden Markov models. The error bars represent one standard error (those for the HDP-HMM are too small to see).

Figures from Liang et al. (2007)

HDP-PCFG (Infinite PCFG)

HDP-PCFG

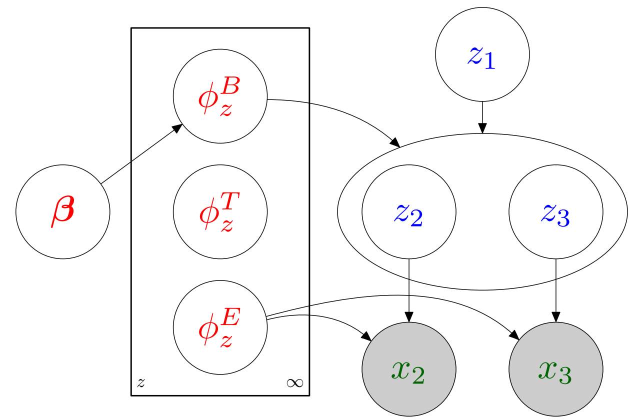
$\boldsymbol{\beta} \sim \mathrm{GEM}(\alpha)$ [draw top-level symbol weights]

For each grammar symbol $z \in \{1, 2, \dots\}$:

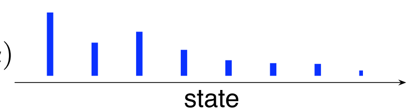
- $\phi_z^T \sim \mathrm{Dirichlet}(\alpha^T)$ [draw rule type parameters]
- $\phi_z^E \sim \mathrm{Dirichlet}(\alpha^E)$ [draw emission parameters]
- $\phi_z^B \sim \mathrm{DP}(\alpha^B, \boldsymbol{\beta}\boldsymbol{\beta}^T)$ [draw binary production parameters]

For each node i in the parse tree:

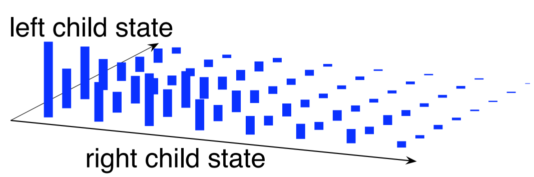
- $t_i \sim \mathrm{Multinomial}(\phi_{z_i}^T)$ [choose rule type]
- If $t_i = \textsc{Emission}$:
  - $x_i \sim \mathrm{Multinomial}(\phi_{z_i}^E)$ [emit terminal symbol]
- If $t_i = \textsc{Binary-Production}$:
  - $(z_{L(i)}, z_{R(i)}) \sim \mathrm{Multinomial}(\phi_{z_i}^B)$ [generate children symbols]



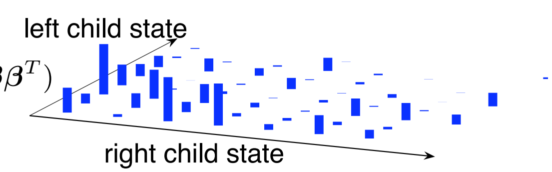
$\boldsymbol{\beta} \sim \mathrm{GEM}(\alpha)$



$\boldsymbol{\beta}\boldsymbol{\beta}^T$



$\phi_z^B \sim \mathrm{DP}(\boldsymbol{\beta}\boldsymbol{\beta}^T)$



Parametric vs. Nonparametric

Type of Model	Parametric Example	Nonparametric Example	
		Construction #1	Construction #2
distribution over counts	Dirichlet-Multinomial Model	Dirichlet Process (DP)	
		Chinese Restaurant Process (CRP)	Stick-breaking construction
mixture	Gaussian Mixture Model (GMM)	Dirichlet Process Mixture Model (DPMM)	
		CRP Mixture Model	Stick-breaking construction
admixture	Latent Dirichlet Allocation (LDA)	Hierarchical Dirichlet Process Mixture Model (HDPMM)	
		Chinese Restaurant Franchise	Stick-breaking construction

Summary of DP and DP-MM

- **DP** has many **different representations:**
  - Chinese Restaurant Process
  - Stick-breaking construction
  - Blackwell-MacQueen Urn Scheme
  - Limit of finite mixtures
  - etc.
- These representations give rise to a variety of **inference techniques** for the **DP-MM** and related models
  - Gibbs sampler (CRP)
  - Gibbs sampler (stick-breaking)
  - Variational inference (stick-breaking)
  - etc.

INDIAN BUFFET PROCESS (IBP)

Outline

- **Motivation:** *Infinite* Latent Feature Models
- **Finite Feature Model**
  - Beta-Bernoulli Model
  - Marginalized Beta-Bernoulli Model
  - Expected # of non-zeros
  - Taking the **Infinite** Limit
  - Left-ordered form (equivalence classes)
- **The Indian Buffet Process (IBP)**
  - Nonexchangeable IBP
  - Exchangeable IBP
  - Gibbs Sampling with Exchangeable IBP
- **IBP properties**
- **Applications**
- **Summary**

Motivation

- **Latent Feature Models**
  - Examples:
    - factor analysis
    - probabilistic PCA
    - cooperative vector quantization
    - sparse PCA
- **Applications**
  - object detection in images
  - choice behavior (i.e. option A over option B)
  - proteomics: modeling the functional interactions of proteins – which can belong to multiple complexes at the same time
  - collaborative filtering: modeling features of movie preferences (a la. Netflix challenge)
  - structure learning for graphical models (i.e. bipartite graphs)

Latent Feature Models

Let $\mathbf{x}_i$ be the ith data instance

$\mathbf{f}_i$ be its features

Define $\mathbf{X} = [\mathbf{x}_1^T, x_2^T, \ldots, x_N^T]$

$\mathbf{F} = [\mathbf{f}_1^T, f_2^T, \ldots, f_N^T]$

Model: $p(\mathbf{X}, \mathbf{F}) = p(\mathbf{X}|\mathbf{F})p(\mathbf{F})$

Latent Feature Models

Decompose the feature matrix, F, into a sparse binary matrix, Z, and a value matrix, V.

$$\mathbf{F} = \mathbf{Z} \otimes \mathbf{V}$$

where $\otimes$ is the elementwise product

$$z_{ij} \in \{0, 1\}$$

$$v_{ij} \in \mathcal{R}$$

K features

N objects

$$\begin{bmatrix} 1.2 & 10 & 0 \\ 0 & 9 & 0 \\ 0.5 & 0 & -.1 \\ 0 & 10 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 1.2 & 10 & 0.3 \\ -.3 & 9 & 0.5 \\ 0.5 & 9 & -.1 \\ 0.7 & 10 & -.1 \end{bmatrix}$$

Latent Feature Models

Decompose the feature matrix, F, into a sparse binary matrix, Z, and a value matrix, V.

$$\mathbf{F} = \mathbf{Z} \otimes \mathbf{V}$$

where $\otimes$ is the elementwise product

$$z_{ij} \in \{0, 1\}$$

$$v_{ij} \in \mathcal{R}$$

Model: $p(\mathbf{X}, \mathbf{F}) = p(\mathbf{X}|\mathbf{F})p(\mathbf{F})$

$$= p(\mathbf{X}|\mathbf{F})p(\mathbf{Z})p(\mathbf{V})$$

The IBP will provide p(Z) for the case of infinite columns!

Figure from Griffiths & Ghahramani (2011)

Finite Feature Model

Beta-Bernoulli Model

Generative Story:

- for each feature $k \in \{1, \dots, K\}$:
  - $\pi_k \sim$ Beta$(\frac{\alpha}{K}, 1)$ where $\alpha > 0$
  - for each object $i \in \{1, \dots, N\}$:
    - $z_{ik} \sim$ Bernoulli(π_k)

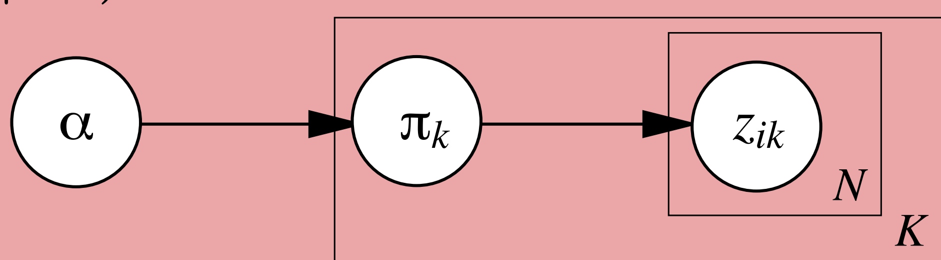
[row]

[prob. of feat. k]

[column]

[is feat. ON/OFF]

$p(\mathbf{Z}, \boldsymbol{\pi} \mid \alpha)$



Finite Feature Model

Marginalized Beta-Bernoulli Model

Because of the **conjugacy** of the **Beta** and **Bernoulli**, we can analytically **marginalize out** the feature prevalence parameters, π_k.

$$
\begin{aligned}
P(\mathbf{Z}) &= \prod_{k=1}^{K} \int \left(\prod_{i=1}^{N} P(z_{ik} | \pi_k) \right) p(\pi_k)\, d\pi_k \\
&= \prod_{k=1}^{K} \frac{\frac{\alpha}{K} \Gamma(m_k + \frac{\alpha}{K}) \Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})}.
\end{aligned}
$$

where $m_k = \sum_{i=1}^{N} z_{ik}$ is # features ON in column k,

Γ is the Gamma function

Finite Feature Model

Expected # of non-zeroes

Generative Story:

- for each feature $k \in \{1, \ldots, K\}$: [row]
  - $\pi_k \sim \text{Beta}(\frac{\alpha}{K}, 1)$ where $\alpha > 0$ [prob. of feat. k]
  - for each object $i \in \{1, \ldots, N\}$: [column]
    - $z_{ik} \sim \text{Bernoulli}(\pi_k)$ [is feat. ON/OFF]

Recall: if $X \sim \text{Beta}(r, s)$, then $\mathbb{E}[X] = \frac{r}{r+s}$

if $Y \sim \text{Bernoulli}(p)$, then $\mathbb{E}[Y] = p$

$$\mathbb{E}[z_{ik}] = \frac{\frac{\alpha}{K}}{1 + \frac{\alpha}{K}}$$

$$\Rightarrow \mathbb{E}[\mathbf{1}^T \mathbf{Z} \mathbf{1}] = \mathbb{E}\left[\sum_{i=1}^{N} \sum_{k=1}^{K} z_{ik}\right] = \frac{N\alpha}{1 + \frac{\alpha}{K}}$$

So the expected number of non-zero entries in Z is ≤ $N\alpha$

What happens as $K \to \infty$?

Finite Feature Model

*Taking the **Infinite** Limit*

$$\lim_{K\to\infty} p(\mathbf{Z}) = \lim_{K\to\infty} \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})}.$$

$$= 0$$

Problem: Every **matrix** has **zero** probability!

Finite Feature Model

Left-Ordered Form (lof)

Topic Modeling:

- Consider many samples of the kth topic from the Markov chain: $\phi_k^{(1)}, \phi_k^{(2)}, \ldots, \phi_k^{(T)}$

  This topic will "drift" over time (e.g. from {politics} at time (t) to {geology} at time (t+m))
- Instead of averaging, it's common to use a MAP estimate of the topics
- The **order** of the topics is **not important** to the model (i.e. the topics are not identifiable)

Finite Feature Model

Left-Ordered Form (lof)

Back to our model:

- Q: In a **latent** feature model, what’s the difference between feature k=13 and k=27?
- A: Nothing!

The use of left-ordered form **capitalizes** on the fact that **features are not identifiable** (i.e. order of features doesn’t matter to the model)

Finite Feature Model

Left-Ordered Form (lof)

Define the history of feature k to be the magnitude of the binary value given by the column:

$$h_k = \sum_{i=1}^{N} 2^{(N-i)} z_{ik}$$

K_h = # of features with history h

K_0 = # of features with $m_k = 0$ (i.e. $h = 0$)

$K_+ = \sum_{h=1}^{2^N - 1} K_h$, # of features with non-zero history

$$\Rightarrow K = K_0 + K_+$$

Same history

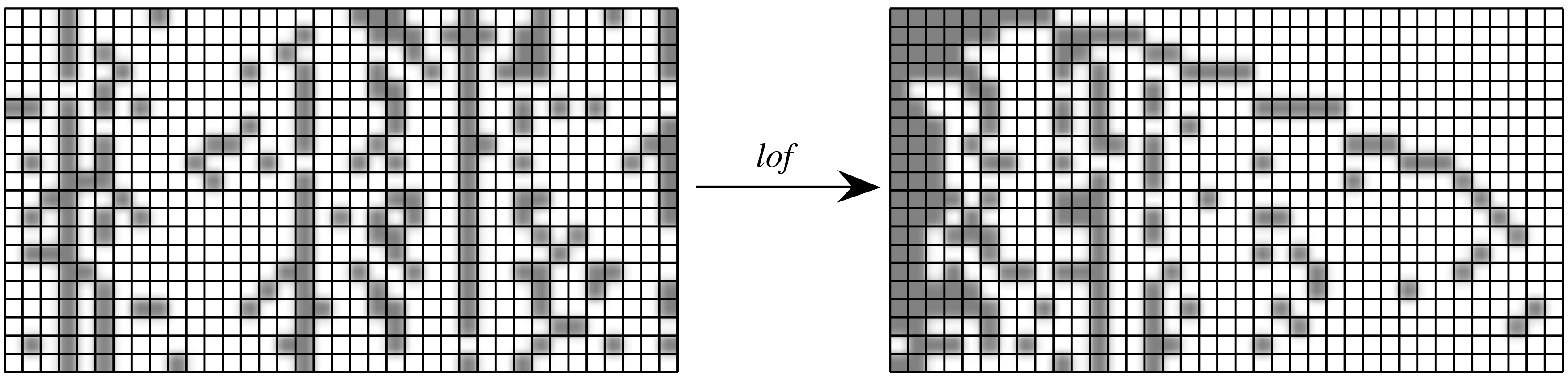
$$\begin{array}{cccc} 10 & 13 & 2 & 13 \\ \hline 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{array}$$

Define lof(Z) to be sorted left-to-right by the history of each feature.

Finite Feature Model

Left-Ordered Form (lof)

Define *lof(Z)* to be sorted left-to-right
by the history of each feature.



Define equivalence class $[Z] = \{Z' : lof(Z') = lof(Z)\}$

Cardinality of $[Z] = \dfrac{K!}{\prod_{h=0}^{2^N-1} K_h!}$

Finite Feature Model

*Taking the **Infinite** Limit*

$$\lim_{K\to\infty} p(\mathbf{Z}) = \lim_{K\to\infty} \prod_{k=1}^{K} \frac{\frac{\alpha}{K}\Gamma(m_k + \frac{\alpha}{K})\Gamma(N - m_k + 1)}{\Gamma(N + 1 + \frac{\alpha}{K})}.$$

$$= 0$$

Problem: Every **matrix** has **zero** probability!

$$\lim_{K\to\infty} p([\mathbf{Z}]) = \lim_{K\to\infty} \frac{K!}{\prod_{h=0}^{2^N-1} K_h!} p(\mathbf{Z})$$

$$= \frac{\alpha^{K_+}}{\prod_{h=1}^{2^N-1} K_h!} \cdot \exp\{-\alpha H_N\} \cdot \prod_{k=1}^{K_+} \frac{(N - m_k)!(m_k - 1)!}{N!},$$

where $H_N = \sum_{j=1}^{N} \frac{1}{j}$ is the Nth harmonic number

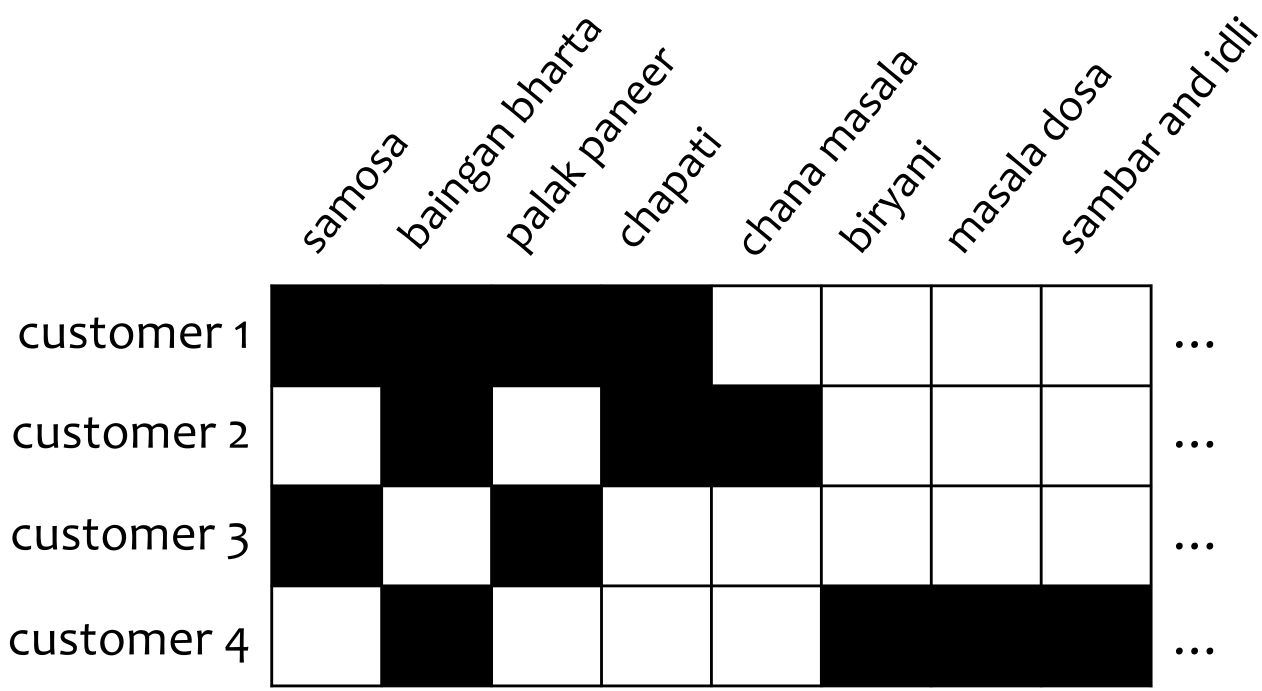
Solution: Every **equivalence class** has **non-zero** probability!

Figure from Griffiths & Ghahramani (2005)

The Indian Buffet Process

^Non-exchangeable

- Imagine an Indian restaurant with a buffet containing an **infinite** # of dishes.
- N customers make a plate by selecting dishes from the buffet:
  - **1st customer:**
    Starts at the left and selects a Poisson(α) number of dishes
  - **ith customer:**
    1. Samples *previously sampled* dishes according to their popularity:
       (i.e. with prob. m_k/i where m_k is the # of previous customers who tried dish k)
    2. Then selects a Poisson(α/i) number of new dishes



Problem: the process is **not exchangeable** – dishes sampled as "new" **depend on the customer order.**

Figure from Griffiths & Ghahramani (2005)

The Indian Buffet Process
^Exchangeable

- Imagine an Indian restaurant with a buffet containing an **infinite** # of dishes.
- N customers make a plate by selecting dishes from the buffet:
  - **1st customer:**
    Starts at the left and selects a Poisson(α) number of dishes
  - **ith customer:**
    1. Makes a single decision for dishes with same history, h:
       (i.e. If there are K_h dishes w/history h sampled by m_h customers, then she samples a Binomial(m_h/I, K_h) number starting at the left)
    2. Then selects a Poisson(α/i) number of new dishes

This yields a *lof* matrix, Z.

Does so with probability p([Z])!

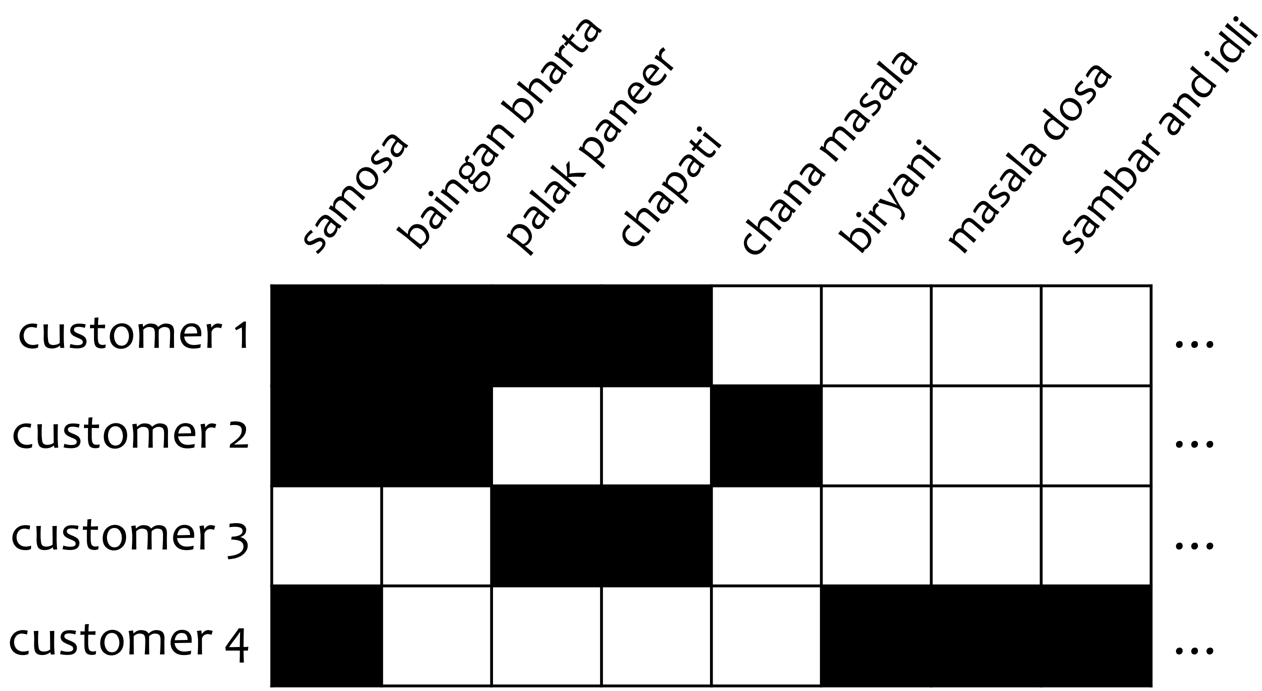
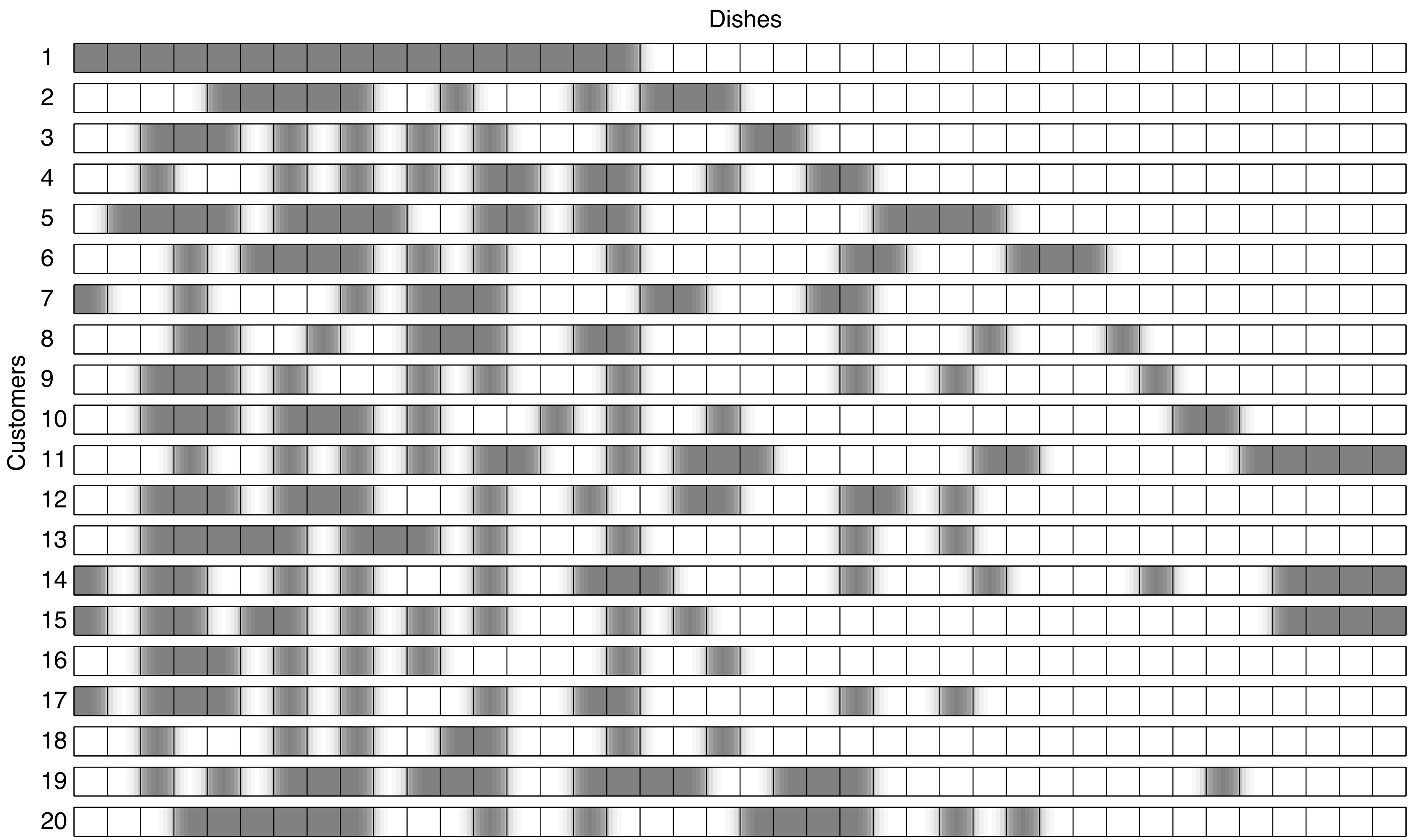


Figure from Griffiths & Ghahramani (2005)

The Indian Buffet Process

Example:



Gibbs Sampler for IBP

Consider a "prior only" sampler of p(Z | α)

- For finite K:

$$\begin{aligned} P(z_{ik}=1|\mathbf{z}_{-i,k}) &= \int_0^1 P(z_{ik}|\pi_k)p(\pi_k|\mathbf{z}_{-i,k})\,d\pi_k \\ &= \frac{m_{-i,k}+\frac{\alpha}{K}}{N+\frac{\alpha}{K}}, \end{aligned}$$

where $z_{-i,k}$ is the kth column except row i,

$m_{-i,k}$ is the # of rows w/feat. k except i

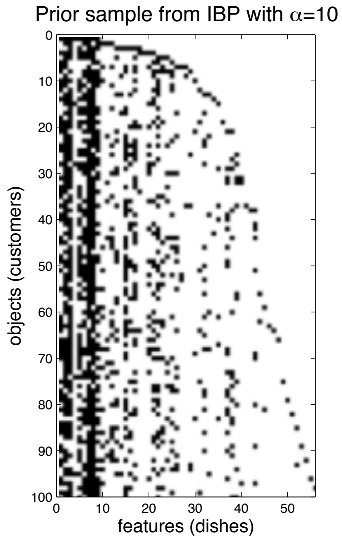
- For infinite K:
  - The "Exchangeable IBP" is *exchangeable*!
  - Choose an order s.t. the i^{th} customer was the last to enter (just like CRP sampler)
  - For any k s.t. $m_{-i,k} > 0$, resample:

$$P(z_{ik}=1|\mathbf{z}_{-i,k}) = \frac{m_{-i,k}}{N},$$

  - Then draw a Poisson(α/i) # of new dishes.

Properties of the Indian buffet process

$$P([\mathbf{Z}]|\alpha) = \exp\big\{-\alpha H_N\big\} \frac{\alpha^{K_+}}{\prod_{h>0} K_h!} \prod_{k \le K_+} \frac{(N-m_k)!(m_k-1)!}{N!}$$



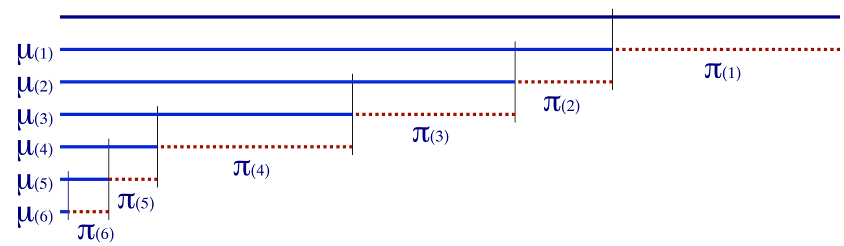


Figure 1: Stick-breaking construction for the DP and IBP. The black stick at top has length 1. At each iteration the vertical black line represents the break point. The brown dotted stick on the right is the weight obtained for the DP, while the blue stick on the left is the weight obtained for the IBP.

Shown in (Griffiths and Ghahramani, 2005):

- It is infinitely exchangeable.
- The number of ones in each row is Poisson(α)
- The expected total number of ones is αN.
- The number of nonzero columns grows as $O(\alpha \log N)$.

Additional properties:

- Has a stick-breaking representation (Teh, Görür, Ghahramani, 2007)
- Can be interpreted using a Beta-Bernoulli process (Thibaux and Jordan, 2007)

Posterior Inference in IBPs

$$P(\mathbf{Z}, \alpha|\mathbf{X}) \propto P(\mathbf{X}|\mathbf{Z})P(\mathbf{Z}|\alpha)P(\alpha)$$

Gibbs sampling: $P(z_{nk} = 1|\mathbf{Z}_{-(nk)}, \mathbf{X}, \alpha) \propto P(z_{nk} = 1|\mathbf{Z}_{-(nk)}, \alpha)P(\mathbf{X}|\mathbf{Z})$

- If $m_{-n,k} > 0$, $P(z_{nk} = 1|\mathbf{z}_{-n,k}) = \frac{m_{-n,k}}{N}$
- For infinitely many k such that $m_{-n,k} = 0$: Metropolis steps with truncation* to sample from the number of new features for each object.
- If α has a Gamma prior then the posterior is also Gamma $\rightarrow$ Gibbs sample.

Conjugate sampler: assumes that $P(\mathbf{X}|\mathbf{Z})$ can be computed.

Non-conjugate sampler: $P(\mathbf{X}|\mathbf{Z}) = \int P(\mathbf{X}|\mathbf{Z}, \theta)P(\theta)d\theta$ cannot be computed, requires sampling latent θ as well (c.f. (Neal 2000) non-conjugate DPM samplers).

***Slice sampler:** non-conjugate case, is not approximate, and has an adaptive truncation level using a stick-breaking construction of the IBP (Teh, et al, 2007).

Particle Filter: (Wood & Griffiths, 2007).

Accelerated Gibbs Sampling: maintaining a probability distribution over some of the variables (Doshi-Velez & Ghahramani, 2009).

Variational inference: (Doshi-Velez, Miller, van Gael, & Teh, 2009).

Modelling Data

Latent variable model: let $\mathbf{X}$ be the $N \times D$ matrix of observed data, and $\mathbf{Z}$ be the $N \times K$ matrix of binary latent features

$$P(\mathbf{X}, \mathbf{Z}|\alpha) = P(\mathbf{X}|\mathbf{Z})P(\mathbf{Z}|\alpha)$$

By combining the IBP with different likelihood functions we can get different kinds of models:

- Models for graph structures (w/ Wood, Griffiths, 2006)
- Models for protein complexes (w/ Chu, Wild, 2006)
- Models for overlapping clusters (w/ Heller, 2007)
- Models for choice behaviour (Görür, Jäkel & Rasmussen, 2006)
- Models for users in collaborative filtering (w/ Meeds, Roweis, Neal, 2006)
- Sparse latent factor models (w/ Knowles, 2007)

Summary

- Beta-Bernoulli model is a **simple** *finite* feature model
- Can treat features as **latent**
- **Infinite limit** of Beta-Bernoulli yields the Indian Buffet Process (IBP)
- Many properties of the IBP are similar to the CRP