

Lecture VIII: Introduction to Clustering – Parametric clustering

Marina Meilă
mmp@stat.washington.edu

Department of Statistics
University of Washington

December, 2021

Paradigms for clustering

Parametric clustering algorithms (K given)

Cost based / hard clustering

Basic algorithms

K-means clustering and the quadratic distortion

Model based / soft clustering

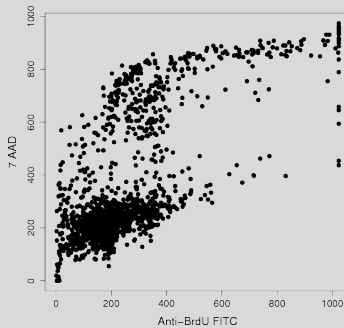
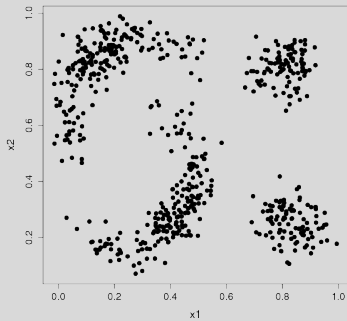
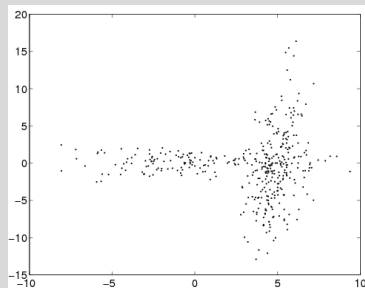
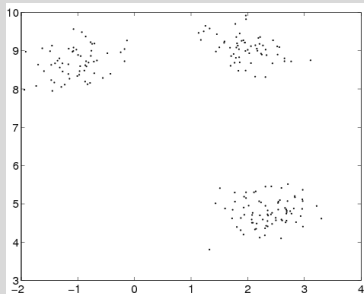
Reading HTF Ch.: 14.3, Murphy Ch.: Ch 11.[1], 11.2.1-3, 11.3, Ch 25, Bach Ch.:

What is clustering? Problem and Notation

- ▶ **Informal definition Clustering** = Finding groups in data
- ▶ **Notation**
 - \mathcal{D} = $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ a **data set**
 - n = number of **data points**
 - K = number of **clusters** ($K \ll n$)
 - Δ = $\{C_1, C_2, \dots, C_K\}$ a partition of \mathcal{D} into disjoint subsets
 - $k(i)$ = the **label** of point i
 - $\mathcal{L}(\Delta)$ = cost (loss) of Δ (to be minimized)
- ▶ **Second informal definition Clustering** = given n **data points**, separate them into K **clusters**
- ▶ Hard vs. soft clusterings
 - ▶ **Hard** clustering Δ : an item belongs to only 1 cluster
 - ▶ **Soft** clustering $\gamma = \{\gamma_{ki}\}_{k=1:K}^{i=1:n}$
 γ_{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)



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 (K known)	Cost based [hard] Model based [soft]
-----------------------------------	---

Non-parametric (K determined by algorithm)	Dirichlet process mixtures [soft] Information bottleneck [soft] Modes of distribution [hard] Gaussian blurring mean shift[?] [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}	Expected 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 <i>Lots of data to explore!</i>
Stage of field	Mature	Still young

Parametric clustering algorithms

- ▶ Cost based
 - ▶ Single linkage (min spanning tree)
 - ▶ Min diameter
 - ▶ Fastest first traversal (HS initialization)
 - ▶ K-medians
 - ▶ K-means
- ▶ Model based (cost is derived from likelihood)
 - ▶ EM algorithm
 - ▶ “Computer science” / “Probably correct” algorithms

Minimum diameter clustering

▶ **Cost** $\mathcal{L}(\Delta) = \max_k \underbrace{\max_{i,j \in C_k} \|x_i - x_j\|}_{\text{diameter}}$

- ▶ Minimize the diameter of the clusters
- ▶ Optimizing this cost is NP-hard

▶ **Algorithms**

- ▶ **Fastest First Traversal** [?] – a factor 2 approximation for the min cost

For every \mathcal{D} , FFT produces a Δ so that

$$\mathcal{L}^{opt} \leq \mathcal{L}(\Delta) \leq 2\mathcal{L}^{opt}$$

- ▶ rediscovered many times

Algorithm Fastest First Traversal

Input Data $\mathcal{D} = \{x_i\}_{i=1:n}$, number clusters K

defines **centers** $\mu_{1:K} \in \mathcal{D}$

(many other clustering algorithms use centers)

1. pick μ_1 at random from \mathcal{D}
2. for $k = 2 : K$
$$\mu_k \leftarrow \underset{\mathcal{D}}{\operatorname{argmax}} \operatorname{distance}(x_i, \{\mu_{1:k-1}\})$$
3. for $i = 1 : n$ (assign points to centers)
 $k(i) = k$ if μ_k is the nearest center to x_i

K-means clustering

Algorithm K-Means[?]

Input Data $\mathcal{D} = \{x_i\}_{i=1:n}$, number clusters K
Initialize centers $\mu_1, \mu_2, \dots, \mu_K \in \mathbb{R}^d$ at random
Iterate until convergence

1. for $i = 1 : n$ (assign points to clusters \Rightarrow new clustering)

$$k(i) = \underset{k}{\operatorname{argmin}} \|x_i - \mu_k\|$$

2. for $k = 1 : K$ (recalculate centers)

$$\mu_k = \frac{1}{|C_k|} \sum_{i \in C_k} x_i \quad (1)$$

► Convergence

- if Δ doesn't change at iteration m it will never change after that
- convergence in finite number of steps to **local optimum** of cost \mathcal{L} (defined next)
- therefore, initialization will matter

The K-means cost

$$\mathcal{L}(\Delta) = \sum_{k=1}^K \sum_{i \in C_k} \|x_i - \mu_k\|^2 \quad (2)$$

- ▶ K-means solves a **least-squares** problem
- ▶ the cost \mathcal{L} is called **quadratic distortion**

Proposition The K-means algorithm decreases $\mathcal{L}(\Delta)$ at every step.

Sketch of proof

- ▶ step 1: reassigning the labels can only decrease \mathcal{L}
- ▶ step 2: reassigning the centers μ_k can only decrease \mathcal{L} because μ_k as given by (1) is the solution to

$$\mu_k = \min_{\mu \in \mathbb{R}^d} \sum_{i \in C_k} \|x_i - \mu\|^2 \quad (3)$$

Equivalent and similar cost functions

- ▶ The distortion can also be expressed using intracluster distances

$$\mathcal{L}(\Delta) = \sum_{k=1}^K \frac{1}{n_k} \sum_{i,j \in C_k} \|x_i - x_j\|^2 \quad (4)$$

- ▶ **Correlation clustering** is defined as optimizing the related criterion

$$\mathcal{L}(\Delta) = \sum_{k=1}^K \sum_{i,j \in C_k} \|x_i - x_j\|^2$$

- ▶ This cost is equivalent to the (negative) sum of (squared) intercluster distances

$$\mathcal{L}(\Delta) = - \sum_{k=1}^K \sum_{i \in C_k} \sum_{j \notin C_k} \|x_i - x_j\|^2 + \text{constant} \quad (5)$$

Proof of (6) Replace μ_k as expressed in (1) in the expression of \mathcal{L} , then rearrange the terms

Proof of (5) $\sum_k \sum_{i,j \in C_k} \|x_i - x_j\|^2 = \underbrace{\sum_{i=1}^n \sum_{j=1}^n \|x_i - x_j\|^2}_{\text{independent of } \Delta} - \sum_k \sum_{i \in C_k} \sum_{j \notin C_k} \|x_i - x_j\|^2$

The K-means cost in matrix form – the assignment matrix

- \mathcal{L} as sum of squared **intracluster** distances

$$\mathcal{L}(\Delta) = \sum_{k=1}^K \frac{1}{|C_k|} \sum_{i,j \in C_k} \|x_i - x_j\|^2 \quad (6)$$

-
- Define the **assignment matrix** associated with Δ by $Z(\Delta)$
Let $\Delta = \{C_1 = \{1, 2, 3\}, C_2 = \{4, 5\}\}$

$$Z^{\text{unnorm}}(\Delta) = \begin{matrix} & \begin{matrix} C_1 & C_2 \end{matrix} \\ \begin{matrix} \text{point } i \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix} \end{matrix} \quad Z(\Delta) = \begin{matrix} & \begin{matrix} C_1 & C_2 \end{matrix} \\ \begin{matrix} \text{point } i \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} 1/\sqrt{3} & 0 \\ 1/\sqrt{3} & 0 \\ 1/\sqrt{3} & 0 \\ 0 & 1/\sqrt{2} \\ 0 & 1/\sqrt{2} \end{bmatrix} \end{matrix}$$

Then Z is an orthogonal matrix (columns are orthonormal) and

$$\mathcal{L}(\Delta) = \text{trace } Z^T D Z \quad \text{with } D_{ij} = \|x_i - x_j\|^2 \quad (7)$$

$$\text{Let } \mathcal{Z} = \{Z \in \mathbb{R}^{n \times K}, K \text{ orthonormal}\}$$

Proof of (7) Start from (2) and note that $\text{trace } Z^T A Z = \sum_k \sum_{i,j \in C_k} Z_{ik} Z_{jk} A_{ij} = \sum_k \sum_{i,j \in C_k} \frac{1}{|C_k|} A_{ij}$

The K-means cost in matrix form – the co-occurrence matrix

$$n = 5, \Delta = (1, 1, 1, 2, 2),$$

$$X(\Delta) = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

1. $X(\Delta)$ is symmetric, positive definite, ≥ 0 elements
2. $X(\Delta)$ has row sums equal to 1
3. $\text{trace } X(\Delta) = K$

$$\|X(\Delta)\|_F^2 = \langle X, X \rangle = K$$

$$X(\Delta) = Z(\Delta)Z^T(\Delta)$$

$$2\mathcal{L}(\Delta) = \sum_{k=1}^K \frac{1}{|C_k|} \sum_{i,j \in C_k} \|x_i - x_j\|^2 = \frac{1}{2} \langle D, X(\Delta) \rangle$$

with $D_{ij} = \|x_i - x_j\|^2$

Spectral and convex relaxations

$$\begin{aligned} \mathcal{L}(\Delta) &= \frac{1}{2} \langle D, X(\Delta) \rangle, \quad D = \text{squared distance matrix} \in \mathbb{R}^{n \times n} \\ \mathcal{X} &= \{ X \in \mathbb{R}^{n \times n}, X \succeq 0, X_{ij} \geq 0, \text{trace } X = K, X\mathbf{1} = \mathbf{1} \} \\ \mathcal{Z} &= \{ Z \in \mathbb{R}^{n \times K}, K \text{ orthonormal} \} \end{aligned}$$

Spectral relaxation of the K-means problem

$$\min_{Z \in \mathcal{Z}} \text{trace } Z^T D Z$$

This is solved by an **eigendecomposition** $Z^* =$ top K eigenvectors of D

Convex relaxation of the K-means problem

$$\min_{X \in \mathcal{X}} \langle D, X \rangle$$

This is a **Semi-Definite Program (SDP)**

Minimizing \mathcal{L}

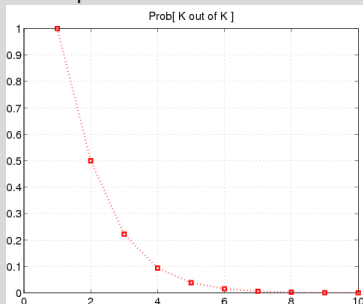
- ▶ By K-means – clustering Δ , **local optima**
- ▶ By convex/spectral relaxation – matrix Z, X , **global optimum**

Symmetries between costs

- ▶ K-means cost $\mathcal{L}(\Delta) = \min_{\mu_{1:K}} \sum_k \sum_{i \in C_k} \|x_i - \mu_k\|^2$
- ▶ K-medians cost $\mathcal{L}(\Delta) = \min_{\mu_{1:K}} \sum_k \sum_{i \in C_k} \|x_i - \mu_k\|$
- ▶ Correlation clustering cost $\mathcal{L}(\Delta) = \sum_k \sum_{i,j \in C_k} \|x_i - x_j\|^2$
- ▶ min Diameter cost $\mathcal{L}^2(\Delta) = \max_k \max_{i,j \in C_k} \|x_i - x_j\|^2$

Initialization of the centroids $\mu_{1:K}$

- ▶ Idea 1: start with K points at random
 - ▶ Idea 2: start with K data points at random
- What's wrong with choosing K data points at random?



The probability of hitting all K clusters with K samples approaches 0 when $K > 5$

- ▶ Idea 3: start with K data points using **Fastest First Traversal** [] (greedy simple approach to spread out centers)
- ▶ Idea 4: **k-means++** [] (randomized, theoretically backed approach to spread out centers)
- ▶ Idea 5: **"K-logK" Initialization** (start with enough centers to hit all clusters, then prune down to K)

For EM Algorithm [], for K-means [?]

The “K-logK” initialization

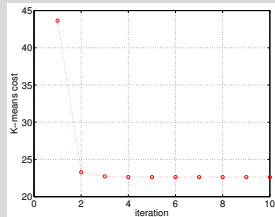
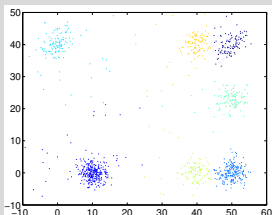
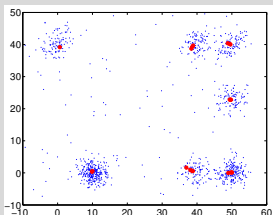
The **K-logK Initialization** (see also [?])

1. pick $\mu_{1:K'}^0$ at random from data set, where $K' = O(K \log K)$
(this assures that each cluster has at least 1 center w.h.p)
2. run 1 step of K-means
3. remove all centers μ_k^0 that have few points, e.g $|C_k| < \frac{n}{eK'}$
4. from the remaining centers select K centers by **Fastest First Traversal**
 - 4.1 pick μ_1 at random from the remaining $\{\mu_{1:K'}^0\}$
 - 4.2 for $k = 2 : K$, $\mu_k \leftarrow \underset{\mu_{k'}^0}{\operatorname{argmax}} \min_{j=1:k-1} \|\mu_{k'}^0 - \mu_j\|$, i.e next μ_k is furthest away from the already chosen centers
5. continue with the standard **K-means** algorithm

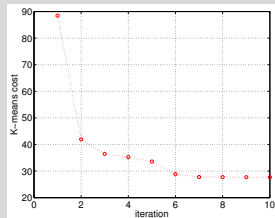
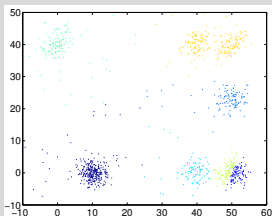
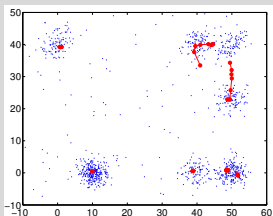
K-means clustering with K-logK Initialization

Example using a mixture of 7 Normal distributions with 100 outliers sampled uniformly

K-LOGK $K = 7$, $T = 100$, $n = 1100$, $c = 1$

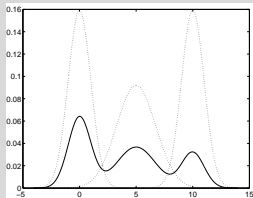


NAIVE $K = 7$ $T = 100$, $n = 1100$



Model based clustering: Mixture models

Mixture in 1D



- ▶ The **mixture density**

$$f(x) = \sum_{k=1}^K \pi_k f_k(x)$$

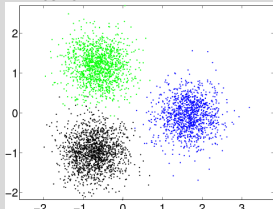
- ▶ $f_k(x)$ = the **components** of the mixture
 - ▶ each is a density
 - ▶ f called **mixture of Gaussians** if $f_k = \text{Normal}_{\mu_k, \Sigma_k}$
- ▶ π_k = the **mixing proportions**,
 $\sum_k = 1^K \pi_k = 1, \pi_k \geq 0.$
- ▶ **model parameters** $\theta = (\pi_{1:K}, \mu_{1:K}, \Sigma_{1:K})$

- ▶ The **degree of membership** of point i to cluster k

$$\gamma_{ki} \stackrel{\text{def}}{=} P[x_i \in C_k] = \frac{\pi_k f_k(x)}{f(x)} \text{ for } i = 1 : n, k = 1 : K \quad (8)$$

- ▶ depends on x_i and on the model parameters

Mixture in 2D



Criterion for clustering: Max likelihood

- ▶ denote $\theta = (\pi_{1:K}, \mu_{1:K}, \Sigma_{1:K})$ (the parameters of the mixture model)
- ▶ Define **likelihood** $P[\mathcal{D}|\theta] = \prod_{i=1}^n f(x_i)$
- ▶ Typically, we use the **log likelihood**

$$l(\theta) = \ln \prod_{i=1}^n f(x_i) = \sum_{i=1}^n \ln \sum_k \pi_k f_k(x_i) \quad (9)$$

- ▶ denote $\theta^{ML} = \underset{\theta}{\operatorname{argmax}} l(\theta)$
- ▶ θ^{ML} determines a soft clustering γ by (8)
- ▶ a soft clustering γ determines a θ (see later)
- ▶ Therefore we can write

$$\mathcal{L}(\gamma) = -l(\theta(\gamma))$$

Algorithms for model-based clustering

Maximize the (log-)likelihood w.r.t θ

- ▶ directly - (e.g by gradient ascent in θ)
- ▶ by the EM algorithm (very popular!)
- ▶ indirectly, w.h.p. by "computer science" algorithms

w.h.p = with high probability (over data sets)

The Expectation-Maximization (EM) Algorithm

Algorithm Expectation-Maximization (EM)

Input Data $\mathcal{D} = \{x_i\}_{i=1:n}$, number clusters K
Initialize parameters $\pi_{1:K} \in \mathbb{R}$, $\mu_{1:K} \in \mathbb{R}^d$, $\Sigma_{1:K} \in \mathbb{R}^{d \times d}$ at random¹
Iterate until convergence

E step (Optimize clustering) for $i = 1 : n$, $k = 1 : K$

$$\gamma_{ki} = \frac{\pi_k f_k(x)}{f(x)}$$

M step (Optimize parameters) set $\Gamma_k = \sum_{i=1}^n \gamma_{ki}$, $k = 1 : K$ (number of points in cluster k)

$$\pi_k = \frac{\Gamma_k}{n}, \quad k = 1 : K$$

$$\mu_k = \sum_{i=1}^n \frac{\gamma_{ki}}{\Gamma_k} x_i$$

$$\Sigma_k = \frac{\sum_{i=1}^n \gamma_{ki} (x_i - \mu_k)(x_i - \mu_k)^T}{\Gamma_k}$$

- ▶ $\pi_{1:K}, \mu_{1:K}, \Sigma_{1:K}$ are the maximizers of $l_c(\theta)$ in (13)
- ▶ $\sum_k \Gamma_k = n$

¹ Σ_k need to be symmetric, positive definite matrices

The EM Algorithm – Motivation

- Define the **indicator variables**

$$z_{ik} = \begin{cases} 1 & \text{if } i \in C_k \\ 0 & \text{if } i \notin C_k \end{cases} \quad (10)$$

denote $\bar{z} = \{z_{ki}\}_{k=1:K}^{i=1:n}$

- Define the **complete log-likelihood**

$$l_c(\theta, \bar{z}) = \sum_{i=1}^n \sum_{k=1}^K z_{ki} \ln \pi_k f_k(x_i) \quad (11)$$

- $E[z_{ki}] = \gamma_{ki}$
- Then

$$E[l_c(\theta, \bar{z})] = \sum_{i=1}^n \sum_{k=1}^K E[z_{ki}] [\ln \pi_k + \ln f_k(x_i)] \quad (12)$$

$$= \sum_{i=1}^n \sum_{k=1}^K \gamma_{ki} \ln \pi_k + \sum_{i=1}^n \sum_{k=1}^K \gamma_{ki} \ln f_k(x_i) \quad (13)$$

- ▶ If θ known, γ_{ki} can be obtained by (8)
(Expectation)
- ▶ If γ_{ki} known, π_k, μ_k, Σ_k can be obtained by separately maximizing the terms of $E[l_c]$
(Maximization)

Brief analysis of EM

$$Q(\theta, \gamma) = \sum_{i=1}^n \sum_{k=1}^K \gamma_{ki} \ln \underbrace{\pi_k f_k(x_i)}_{\theta}$$

- ▶ each step of EM increases $Q(\theta, \gamma)$
- ▶ Q converges to a local maximum
- ▶ at every local maxi of Q , $\theta \leftrightarrow \gamma$ are fixed point
- ▶ $Q(\theta^*, \gamma^*)$ local max for $Q \Rightarrow l(\theta^*)$ local max for $l(\theta)$
- ▶ under certain regularity conditions $\theta \rightarrow \theta^{ML}$ [?]
- ▶ the E and M steps can be seen as projections [?]

- ▶ Exact maximization in **M step** is not essential.
Sufficient to increase Q .
This is called **Generalized EM**

Probabilistic alternate projection view of EM[?]

- ▶ let z_i = which gaussian generated i ? (random variable), $X = (x_{1:n})$, $Z = (z_{1:n})$
- ▶ Redefine Q

$$Q(\tilde{P}, \theta) = \mathcal{L}(\theta) - KL(\tilde{P} || P(Z|X, \theta))$$

where $P(X, Z|\theta) = \prod_i \prod_k P[z_i = k]P[x_i|\theta_k]$

$\tilde{P}(Z)$ is any distribution over Z ,

$KL(P(w)||Q(w)) = \sum_w P(w) \ln \frac{P(w)}{Q(w)}$ the **Kullback-Leibler divergence**

Then,

- ▶ **E step** $\max_{\tilde{P}} Q \Leftrightarrow KL(\tilde{P} || P(Z|X, \theta))$
- ▶ **M step** $\max_{\theta} Q \Leftrightarrow KL(P(X|Z, \theta^{old}) || P(X|\theta))$
- ▶ Interpretation: KL is “distance”, “shortest distance” = **projection**

The M step in special cases

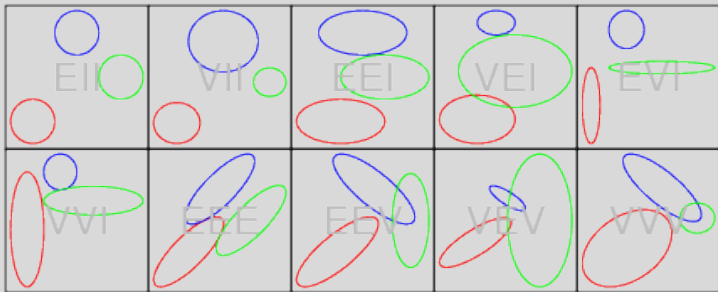
- ▶ Note that the expressions for $\mu_k, \Sigma_k =$ expressions for μ, Σ in the normal distribution, with data points x_i **weighted** by $\frac{\gamma_{ki}}{\Gamma_k}$

M step

general case	$\Sigma_k = \sum_{i=1}^n \frac{\gamma_{ki}}{\Gamma_k} (x_i - \mu_k)(x_i - \mu_k)^T$
$\Sigma_k = \Sigma$ "same shape & size" clusters	$\Sigma \leftarrow \frac{\sum_{i=1}^n \sum_{k=1}^K \gamma_{ki} (x_i - \mu_k)(x_i - \mu_k)^T}{n}$
$\Sigma_k = \sigma_k^2 I_d$ "round" clusters	$\sigma_k^2 \leftarrow \frac{\sum_{i=1}^n \gamma_{ki} \ x_i - \mu_k\ ^2}{d \Gamma_k}$
$\Sigma_k = \sigma^2 I_d$ "round, same size" clusters	$\sigma^2 \leftarrow \frac{\sum_{i=1}^n \sum_{k=1}^K \gamma_{ki} \ x_i - \mu_k\ ^2}{nd}$

Exercise Prove the formulas above

- ▶ Note also that **K-means** is **EM** with $\Sigma_k = \sigma^2 I_d, \sigma^2 \rightarrow 0$ **Exercise** Prove it



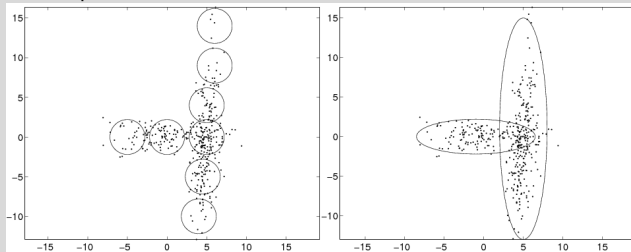
More special cases [?] introduce the following description for a covariance matrix in terms of *volume*, *shape*, *alignment with axes* (=determinant, trace, e-vectors). The letters below mean: I=unitary (shape, axes), E=equal (for all k), V=unequal

- ▶ EII: equal volume, round shape (spherical covariance)
- ▶ VII: varying volume, round shape (spherical covariance)
- ▶ EEI: equal volume, equal shape, axis parallel orientation (diagonal covariance)
- ▶ VEI: varying volume, equal shape, axis parallel orientation (diagonal covariance)
- ▶ EVI: equal volume, varying shape, axis parallel orientation (diagonal covariance)
- ▶ VVI: varying volume, varying shape, equal orientation (diagonal covariance)
- ▶ EEE: equal volume, equal shape, equal orientation (ellipsoidal covariance)
- ▶ EEV: equal volume, equal shape, varying orientation (ellipsoidal covariance)
- ▶ VEV: varying volume, equal shape, varying orientation (ellipsoidal covariance)
- ▶ VVV: varying volume, varying shape, varying orientation (ellipsoidal covariance)

(from [?])

EM versus K-means

- ▶ Alternates between cluster assignments and parameter estimation
- ▶ Cluster assignments γ_{ki} are probabilistic
- ▶ Cluster parametrization more flexible



- ▶ Converges to local optimum of **log-likelihood**
Initialization recommended by **K-logK** method []
- ▶ **Modern algorithms with guarantees** (for e.g. mixtures of Gaussians)
 - ▶ Random projections
 - ▶ Projection on principal subspace [?]
 - ▶ **Two step EM** (=K-logK initialization + one more EM iteration) []

"Computer science" algorithms for mixture models

- ▶ Assume clusters well-separated (S)
 - ▶ e.g. $\|\mu_k - \mu_l\| \geq C \max(\sigma_k, \sigma_l)$
 - ▶ with $\sigma_k^2 = \max \text{eigenvalue}(\Sigma_k)$
- ▶ true distribution is mixture
 - ▶ of Gaussians
 - ▶ of **log-concave** f_k 's (i.e. $\ln f_k$ is concave function)
- ▶ then, w.h.p. (n, K, d, C)
 - ▶ we can label all data points correctly
 - ▶ \Rightarrow we can find good estimate for θ

Even with (S) this is not an easy task in high dimensions

Because $f_k(\mu_k) \rightarrow 0$ in high dimensions (i.e there are few points from Gaussian k near μ_k)

Other "CS" algorithms

- ▶ [?] round, equal sized Gaussian, random projection
- ▶ [?] arbitrary shaped Gaussian, distances
- ▶ [?] log-concave, principal subspace projection

Example Theorem (Achlioptas & McSherry, 2005) If data come from K Gaussians, $n \gg K(d + \log K)/\pi_{\min}$, and

$$\|\mu_k - \mu_l\| \geq 4\sigma_k \sqrt{1/\pi_k + 1/\pi_l} + 4\sigma_k \sqrt{K \log nK + K^2}$$

then, w.h.p. $1 - \delta(d, K, n)$, their algorithm finds true labels

Good

- ▶ theoretical guarantees
- ▶ no local optima
- ▶ suggest heuristics for EM K-means
 - ▶ project data on principal subspace (when $d \gg K$)

But

- ▶ strong assumptions: large separation (unrealistic), concentration of f_k 's (or f_k known), K known
- ▶ try to find perfect solution (too ambitious)

A fundamental result

The Johnson-Lindenstrauss Lemma For any $\varepsilon \in (0, 1]$ and any integer n , let d' be a positive integer such that $d' \geq 4(\varepsilon^2/2 - \varepsilon^3/3)^{-1} \ln n$. Then for any set \mathcal{D} of n points in \mathbb{R}^d , there is a map $f : \mathbb{R}^d \rightarrow \mathbb{R}^{d'}$ such that for all $u, v \in V$,

$$(1 - \varepsilon)\|u - v\|^2 \leq \|f(u) - f(v)\|^2 \leq (1 + \varepsilon)\|u - v\|^2 \quad (14)$$

Furthermore, this map can be found in randomized polynomial time.

- ▶ note that the **embedding dimension** d' does **not** depend on the original dimension d , but depends on n, ε
- ▶ [?] show that: the mapping f is linear and that w.p. $1 - \frac{1}{n}$ a **random projection (rescaled)** has this property
- ▶ **their proof is elementary** Projecting a fixed vector v on a a random subspace is the same as projecting a random vector v on a fixed subspace. Assume $v = [v_1, \dots, v_d]$ with $v \sim$ i.i.d. and let \tilde{v} = projection of v on axes $1 : d'$. Then $E[\|\tilde{v}\|^2] = d' E[v_j^2] = \frac{d'}{d} E[\|v\|^2]$. The next step is to show that the variance of $\|\tilde{v}\|^2$ is very small when d' is sufficiently large.

A two-step EM algorithm [?]

Assumes K spherical gaussians, separation $\|\mu_k^{true} - \mu_{k'}^{true}\| \geq C\sqrt{d}\sigma_k$

1. Pick $K' = \mathcal{O}(K \ln K)$ centers μ_k^0 at random from the data
2. Set $\sigma_k^0 = \frac{d}{2} \min_{k \neq k'} \|\mu_k^0 - \mu_{k'}^0\|^2$, $\pi_k^0 = 1/K'$
3. Run one E step and one M step $\implies \{\pi_k^1, \mu_k^1, \sigma_k^1\}_{k=1:K'}$
4. Compute “distances” $d(\mu_k^1, \mu_{k'}^1) = \frac{\|\mu_k^1 - \mu_{k'}^1\|}{\sigma_k^1 - \sigma_{k'}^1}$
5. Prune all clusters with $\pi_k^1 \leq 1/4K'$
6. Run **Fastest First Traversal** with distances $d(\mu_k^1, \mu_{k'}^1)$ to select K of the remaining centers.
Set $\pi_k^1 = 1/K$.
7. Run one E step and one M step $\implies \{\pi_k^2, \mu_k^2, \sigma_k^2\}_{k=1:K}$

theorem For any $\delta, \varepsilon > 0$ if d large, n large enough, separation $C \geq d^{1/4}$ the **Two step EM** algorithm obtains centers μ_k so that

$$\|\mu_k - \mu_k^{true}\| \leq \|\text{mean}(C_k^{true}) - \mu_k^{true}\| + \varepsilon\sigma_k\sqrt{d}$$

Selecting K for mixture models

The BIC (Bayesian Information) Criterion

- ▶ let $\theta_K =$ parameters for γ_K
- ▶ let $\#\theta_K =$ number independent parameters in θ_K
 - ▶ e.g for mixture of Gaussians with full Σ_k 's in d dimensions

$$\#\theta_K = \underbrace{K - 1}_{\pi_{1:K}} + \underbrace{Kd}_{\mu_{1:K}} + \underbrace{Kd(d - 1)/2}_{\Sigma_{1:K}}$$

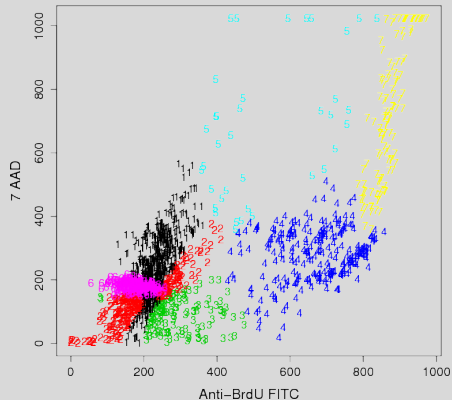
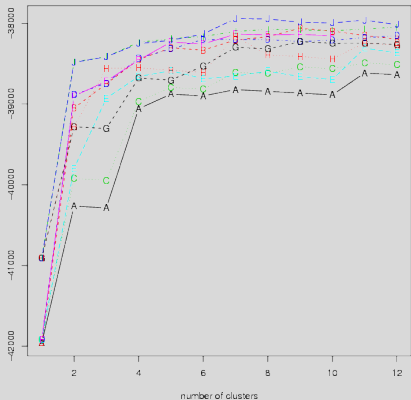
- ▶ define

$$BIC(\theta_K) = l(\theta_K) - \frac{\#\theta_K}{2} \ln n$$

- ▶ **Select K that maximizes $BIC(\theta_K)$**
- ▶ selects true K for $n \rightarrow \infty$ and other technical conditions (e.g parameters in compact set)
- ▶ but theoretically not justified (and overpenalizing) for finite n

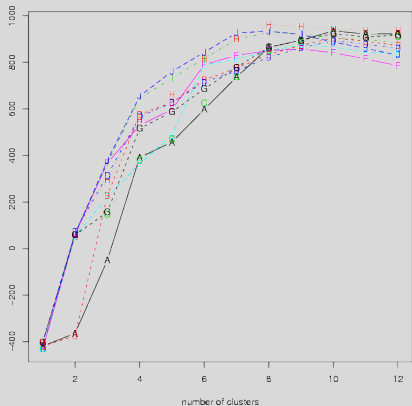
Number of Clusters vs. BIC EII (A), VII (B), EEI (C), VEI (D),
 EVI (E), VVI (F), EEE (G), EEV (H), VEV (I), VVV (J)

EEV, 8 Cluster Solution



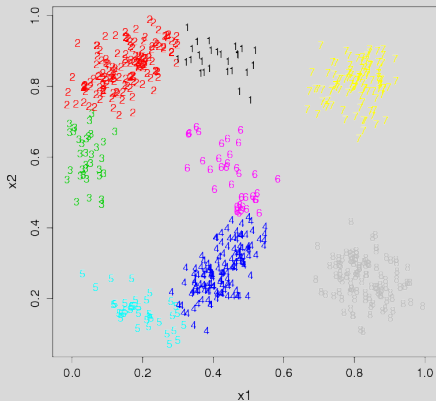
(from [?])

Number of Clusters vs. BIC EII (A), VII (B), EEI (C), VEI (D),
 EVI (E), VVI (F), EEE (G), EEV (H), VEV (I), VVV (J)



(from [?])

EEV, 8 Cluster Solution



Selecting K for hard clusterings

- ▶ based on statistical testing: the **gap** statistic (Tibshirani, Walther, Hastie, 2000)
- ▶ **X-means** [?] heuristic: splits/merges clusters based on statistical tests of Gaussianity
- ▶ Stability methods
 - ▶ Empirical – prove instability
 - ▶ Optimization based – prove stability

Empirical Stability methods for choosing K

- ▶ like bootstrap, or crossvalidation
- ▶ **Idea** (implemented by [?])

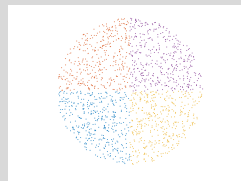
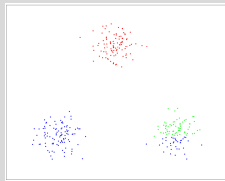
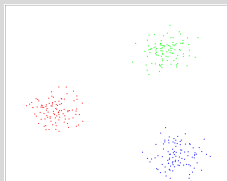
for each K

1. perturb data $\mathcal{D} \rightarrow \mathcal{D}'$
2. cluster $\mathcal{D}' \rightarrow \Delta'_K$
3. compare Δ_K, Δ'_K . Are they similar?
If yes, we say Δ_K is **stable to perturbations**

Fundamental assumption If Δ_K is **stable to perturbations** then K is the correct number of clusters

- ▶ these methods are supported by experiments (not extensive)
- ▶ **not directly supported by theory** . . . see [?] for a summary of the area

Is this clustering approximately correct?



SS method

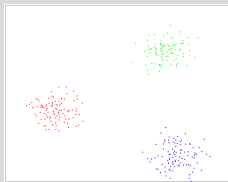
Yes, $OI=1e^{-4}$

Don't know

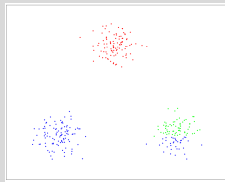
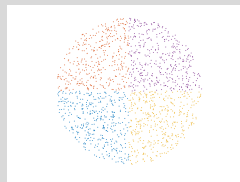
Don't know

- ▶ Given data \mathcal{D} , clustering Δ
- ▶ $\mathcal{L}(\text{data, clustering})$ (e.g. K-means)
- ▶ “correct”
 - ▶ = the “only” “good” clustering supported by \mathcal{D}
 - ▶ \Leftrightarrow any other Δ' with smaller \mathcal{L} is ϵ -close to Δ

Is this clustering approximately correct?



SS method

Yes, $OI=1e^{-4}$
good, stableDon't know
badDon't know
unstable

- ▶ Given data \mathcal{D} , clustering Δ
- ▶ $\mathcal{L}(\text{data}, \text{clustering})$
- ▶ “correct”

= the “only” “good” clustering supported by \mathcal{D}

\Leftrightarrow any other Δ' with smaller \mathcal{L} is ε -close to Δ

(e.g. K-means)

What is an **Optimality Interval (OI)**?

Theorem (Meta-theorem)

If Δ fits the data \mathcal{D} well, then we shall prove that any other clustering Δ' that also fits \mathcal{D} well will be a *small perturbation* of Δ .

- ▶ Δ' is **good** if

$$\mathcal{L}(\Delta') \leq \mathcal{L}(\Delta) + \alpha.$$

- ▶ δ is **OI**: for all good Δ' ,

$$d_{ME}(\Delta', \Delta) \leq \delta$$

where d_{ME} is the **misclassification error/earth mover distance**

- ▶ if OI exists we say Δ is **stable**

How? 1. Mapping a clustering to a matrix

$$n = 5, \Delta = (1, 1, 1, 2, 2),$$

$$X(\Delta) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

1. $X(\Delta)$ is symmetric, positive definite, ≥ 0 elements
2. $X(\Delta)$ has row sums equal to 1
3. $\text{trace } X(\Delta) = K$

$$\|X(\Delta)\|_F^2 = K$$

Let \mathcal{X} be the space $n \times n$ of matrices with Properties 1, 2, 3 above

- ▶ \mathcal{X} is convex
- ▶ $X(C)$ are extreme points of \mathcal{X}

How? 2. Convex relaxations

Original clustering problem Given data \mathcal{D} , K , $\mathcal{L}()$

$$\text{minimize}_{\Delta} \quad \mathcal{L}(\mathcal{D}, \Delta) \quad \text{with solution } \Delta^{\text{opt}}$$

Convex relaxation

- ▶ map clustering $\Delta \rightarrow$ matrix $X(\Delta) \in \mathcal{X}$
- ▶ so that $\mathcal{L}(X)$ convex in X
- ▶ Relaxed problem

$$L^* = \min_{X \in \mathcal{X}} \mathcal{L}(X), \quad \text{with solution } X^* \tag{15}$$

The Sublevel Set (SS) method

Network Given data, L , convex relaxation

Step 0 Cluster data, obtain a clustering Δ .

Step 1 Use convex relaxation to define new optimization problem

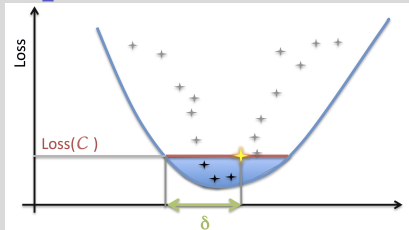
$$\text{SS } \delta = \max_{X' \in \mathcal{X}} \|X(\Delta) - X'\|_F, \quad \text{s.t. } \mathcal{L}(X') \leq \mathcal{L}(\Delta).$$

Step 2 Prove that $\| \|_F \leq \delta \Rightarrow d_{ME}(\cdot) \leq \epsilon$

M, MLJ 2012

Done: ϵ is a **Optimality Interval (OI)** for Δ .

$\mathcal{X}_{\mathcal{L} \leq c} = \{X \in \mathcal{X}, \mathcal{L}(X) \leq c\}$ is **sublevel set** of L



Two technical bits

1. SS is **convex** only if $\|X' - X(\Delta)\|$ **concave**

- ▶ Use $\|\cdot\|_F$ Frobenius norm. $\|X(\Delta)\|_F^2 = K$ for any clustering.

2. Relating $\|\cdot\|_F$ to distance between clusterings.

$$\|X(\Delta) - X(\Delta')\|_F^2 \leq \delta \quad \Rightarrow \quad d_{ME}(\Delta, \Delta') \leq \epsilon$$

distance between matrices “misclassification error” metric
between clusterings

- ▶ Theorem proved in M, *Machine Learning Journal*, 2012 with $\epsilon = 2\delta p_{\max}$.
- ▶ The tightest result known. Upper/lower bounds between d_{ME} , $\|\cdot\|_F$ and Rand Index
- ▶ Proofs use geometry of convex sets + refined analysis for small distances
- ▶ Example from Wan, M NIPS16 OI by existing results Rohe et al. 2011 $\sim 10^2$ OI by our method

Relation with other work

▶ Previous ideas on OI

- ▶ Spectral bounds for Spectral Clustering M, Shortreed, Xu AISTATS05
- ▶ Spectral bounds for K-means, NCut and other quadratic costs M, ICML06 and JMVA 2018
- ▶ Spectral bounds for networks model based clustering: Stochastic Block Model and Preference Frame Model Wan, M NIPS2016

▶ Previous work we build on

- ▶ Convex relaxations for clustering MANY! here we use SDP for K-means Peng, Wei 2007
- ▶ Transforming bound on $\|X - X'\|_F$ into bound on d_{ME} M MLJ 2012
- ▶ **Contrast with** work on Clusterability and resilience, e.g. Ben-David, 2015, Bilu, Linial 2009
 - ▶ “Their” work: assume \exists stable Δ , prove it can be found efficiently
 - ▶ This work: given Δ , prove it is stable

For what clustering paradigms can we obtain OI's?

“All” ways to map Δ to a matrix

space	matrix	definition	size
\mathcal{X}	$X(\Delta)$	$X_{ij} = 1/n_k$ iff $i, j \in C_k$	$n \times n$, block-diagonal
$\tilde{\mathcal{X}}$	$\tilde{X}(\Delta)$	$\tilde{X}_{ij} = 1$ iff $i, j \in C_k$	$n \times n$, block-diagonal
\mathcal{Z}	$Z(\Delta)$	$Z_{ik} = 1/\sqrt{n_k}$ iff $i \in C_k$	$n \times K$, orthogonal

Theorem

M NeurIPS 2018 If L has a convex relaxation involving one of $\mathcal{X}, \tilde{\mathcal{X}}, \mathcal{Z}$, then

(1) There exists a convex SS problem

$$(SS) \quad \delta = \min_{X' \in \mathcal{X}, X' \leq I} \langle X(\Delta), X' \rangle \quad (\text{similarly for } \tilde{\mathcal{X}}, \mathcal{Z}).$$

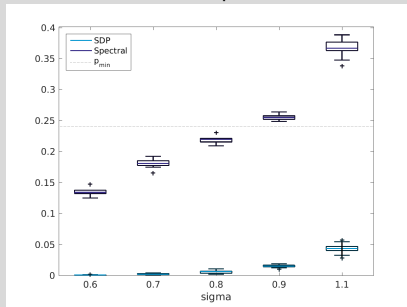
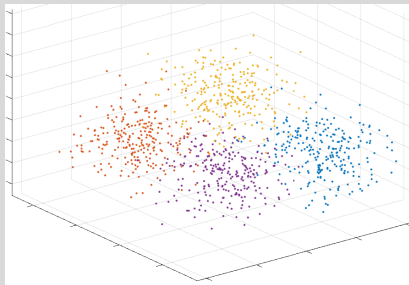
(2) From optimal δ an OI ε can be obtained, valid when $\varepsilon \leq p_{\min}$.

$$\begin{aligned} \mathcal{X} : X_{ij} = 1/n_k \text{ iff } i, j \in C_k & \quad \varepsilon = (K - \delta)p_{\max} \\ \tilde{\mathcal{X}} : \tilde{X}_{ij} = 1 \text{ iff } i, j \in C_k & \quad \varepsilon = \frac{\sum_{k \in [K]} n_k^2 + (n - K + 1)^2 + (K - 1) - 2\delta}{2p_{\min}} \\ \mathcal{Z} : Z_{ik} = 1/\sqrt{n_k} \text{ iff } i \in C_k & \quad \varepsilon = (K - \delta^2/2)p_{\max} \end{aligned}$$

Existence of guarantee depends only on space of convex relaxation.

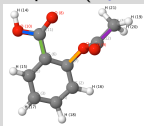
Results for K-means clusterings

$K = 4$ equal Gaussian clusters, $n = 1024$, $\|\mu_k - \mu_l\| = 4\sqrt{2} \approx 5.67$
 data for $\sigma = 0.9$ Values of ϵ vs cluster spread σ



Spectral=M ICML06, SDP=M NeurIPS 2018

Aspirin ($C_9O_4H_8$) molecular simulation data Chmiela et al. 2017

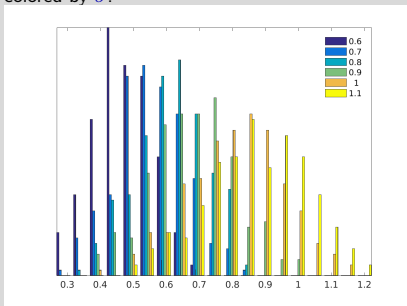


$K = 2$
 $\rho_{\min} = .26$
 $\rho_{\max} = .74$

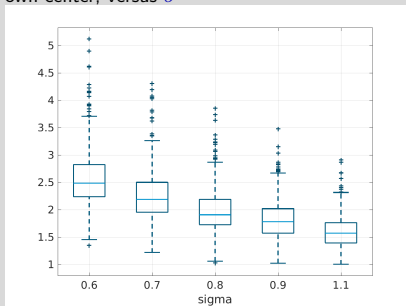
$n = 2118$ $\epsilon = 0.065$

Separation statistics

distance to own center over min center separation, colored by σ .



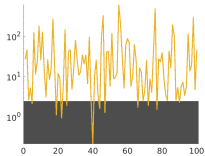
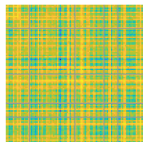
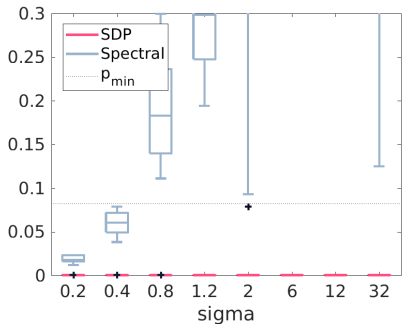
distance to second closest center over distance to own center, versus σ



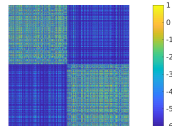
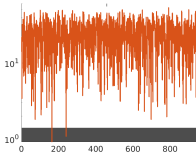
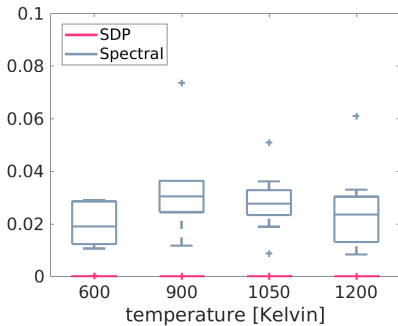
Results for Spectral Clustering by Normalized Cut

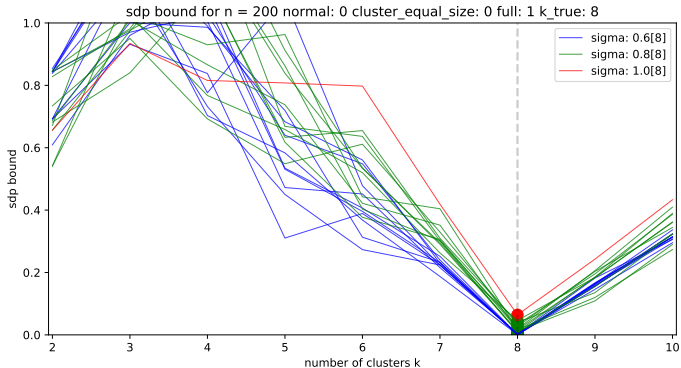
Spectral=M AISTATS05, SDP=M NeurIPS 2018

Synthetic S , $n = 100$



Chemical reaction data, $n \approx 1000$



Stability and the selection of K Cheng, M, Harchaoui (in preparation)

Summary of SS method

1. Cluster data
2. Set up and solve SS problem
3. If solution δ small enough, **guarantee** Δ is approximately optimal and all other good clusterings are near it
 - ▶ without any model assumptions, practically applicable
 - ▶ not all Δ can have guarantees

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 [?]