Algorithms and Learning for Protein Science

From `k-means` to mixtures of von Mises in flat torii

Frederic.Cazals@inria.fr

Overview

- Theory/algorithms
  - `k-means` and seeding procedures
  - Gaussian mixtures – soft and hard
  - Model selection via Minimum message Length
- Protein science
  - Internal coordinates
  - Joint distributions for torsion angles

Algorithms

PART 1: Kmeans and EM
PART 2: Fitting complex mixtures in flat torii

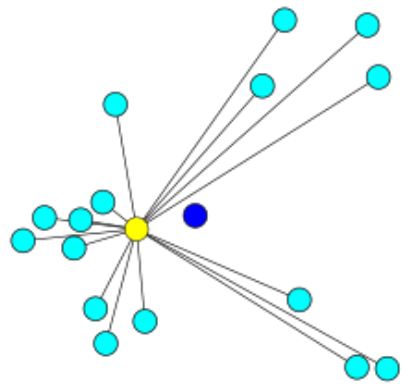
Algorithms

- `k-means`, `k-means++`: basics
- Seeding: local searches + multi-swaps and beyond
- Kmeans and co-clustering
- Comparing clustering using meta-clusters
- GMM: warmup
- Fitting GMM with SOFT EM
- Fitting GMM with HARD EM

Point set in Euclidean space: centroid, center of mass

Definition 1. For a point set $P = \{x_1, \ldots, X_n\}$ in $\mathbb{R}^d$:

- Center of mass/centroid: $x \in \mathbb{R}^d$ minimizing $\sum_{x_i \in P} \|x - x_i\|^2$
- Geometric median: $x \in \mathbb{R}^d$ minimizing $\sum_{x_i \in P} \|x - x_i\|$



Blue: center of mass; yellow: geometric median.

Lemma 2. The center of mass/ centroid μ of P minimizes the sum of squared distances to all points.

▷Ref: https://en.wikipedia.org/wiki/Geometric_median

The centroid minimizes the sum of squared distances

Denoting μ the center of mass:

$$\sum_{x_i \in P} \|x_i - x\|^2 = \sum_i \|x_i - \mu + \mu - x\|^2 \tag{1}$$

$$= \sum_i \langle x_i - \mu + \mu - x, x_i - \mu + \mu - x \rangle \tag{2}$$

$$= \sum_i \|x_i - \mu\|^2 + 2\langle \mu - x, \sum_i (x_i - \mu) \rangle + n \|\mu - x\|^2 \tag{3}$$

$$= \sum_i \|x_i - \mu\|^2 + n \|\mu - x\|^2 \tag{4}$$

since by definition of the centroid $\sum_i (x_i - \mu) = 0$.
Thus, the sum is minimized for $x = \mu$.

Optimization: the k-means criterion

▷ k-means criterion, for k clusters: find k centers $\{c_1, \ldots, c_k\}$ so as to minimize the following sum:

$$\Phi_K = \sum_{x_i \in P} \min_{c_j \in C} \left\| x_i - c_j \right\|^2 . \tag{5}$$

▷ Clustering induced:

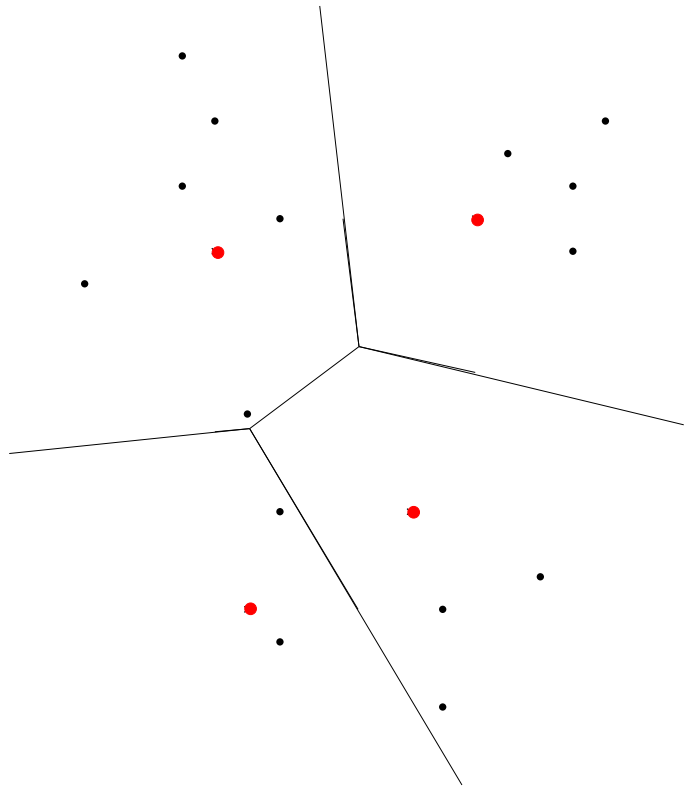
- For each sample x_i: distance to the nearest center c_j
- Clustering defined implicitly: induced by the Voronoi diagram – next slide

▷ Search space and hardness:

- When $k = 1$: the center sought is the center of mass
- The search space for centers is $(\mathbb{R}^d)^k$
- From the combinatorial standpoint, max number of assignment of points to clusters: Stirling number of second kind $\sim k^n/k!$
- For fixed k: problem can be solved in time $O(n^{(d+2)k+1})$ – enumerating all partitions
- In general – k is a function of n: the problem in NP-hard i.e. cannot be solved in polynomial time unless $P = NP$

k-means and Lloyd iterations

- Assign each point to its nearest center – Voronoi partition
- Replace each center by the center of mass of points in its Voronoi cell
- Iterate until convergence



▷ Nb: only $O(n^{dk})$ subsets of data points can be induced by Voronoi – not k^n.

k-means++: smart seeding procedure

- Idea: force the selection of seeds far away from those already chosen center $q \in C$
  - $D2 \overset{Def}{=} \text{Cost}(\{p\}, C) = \min_{q \in C} \|p - q\|^2$

procedure SmartSeeding(P, k)

Input: dataset P, num. of centers k
Uniformly sample $p \in C$ and set $C = \{p\}$
for $i = 2$ to k **do**
  Sample $p \in P \backslash C$ with proba. $\mathbb{P}[p] = \text{Cost}(\{p\}, C) / \sum_{q \in C} \text{Cost}(\{q\}, C)$
  $C \leftarrow C \cup \{p\}$
end for
Return C
end procedure

- Nb: squaring distances is enough; higher exponents may favor outliers.

k-means++ and smart seeding

- ▷ Usual problem: clusters are not well represented by the initial seeds
- ▷ Smart seeding:
  - force the initial seeds to stab/encounter the clusters
  - seed choosen with a probability proportional to the squared distances to already chosen seeds

Theorem 3. With k-means++, the expectation of the k-means functional satisfies

$$\frac{\mathbb{E}\left[\Phi_K\right]}{\Phi_{K,OPT}} \leq 8(\ln k + 2) = O(\ln k). \tag{6}$$

Theorem 4. The approximation factor is no better than $2 \ln k$.

▷Ref: k-means++: the advantages of smart seeding, Arthur and Vassilvitskii, ACM SODA 2007

Algorithms

`k-means`, `k-means++`: basics

Seeding: local searches + multi-swaps and beyond

Kmeans and co-clustering

Comparing clustering using meta-clusters

GMM: warmup

Fitting GMM with SOFT EM

Fitting GMM with HARD EM

k-means++-G: greedy k-means++

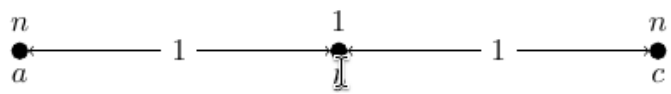
Quote from the k-means++ paper: "Also, experiments showed that k-means++ generally performed better if it selected several new centers during each iteration, and then greedily chose the one that decreased Φ_K as much as possible."

▷ Implemented in scikit-learn with $l = 2 + \log K$ candidates:

Theorem 5. (GR03) The approximation ratios of k-means++-G are:

- $\Omega(l^3 \log^3 k / \log^2(l \log k))$.
- $O(l^3 \log^3 k)$

▷ Pathological case: opt 2-clustering: $cost < 1$; if b is a center: $cost = \Omega(n)$



▷ Pool size and randomization:

- Increasing the pool size l jeopardizes randomization

▷Ref: Grunau et al, ACM SODA 2003

Local searches via single swaps and multi-swaps

- The single swap method from Arya et al:

procedure LocalSearchesExhaustive
  Input: dataset P, centers C
  if $\exists q \in C, \exists p \in P \backslash C$ such that $\text{Cost}(P, C - p + p) < \text{Cost}(P, C)$ **then**
    Pick the best swap of seeds p', q'
    $C \leftarrow C \backslash \{q'\} \cup \{p'\}$
  end if
  Return C
end procedure

- p multi-swaps: in the previous algorithm, seek replacements of size p.

Theorem 6. The p multi-swap method has a CAF $\leq (3 + 2/p)^2$. The bound is almost tight.

There exists configurations of points where the algorithm yields a $9 + \varepsilon$ approximation factor.

Ref: Mount et al, ACM SoCG 2004

k-means++-LS: combining ++ and local searches

▷ Idea: sample p – ~~exhaustive search~~ + search exhaustively substitutes $q \in C$

procedure LocalSearch++ k-means++-LS
  Input: dataset P, centers C
  Sample p with proba. $\text{Cost}(P)$
  $q' \leftarrow \arg\min_{q \in C} \text{Cost}(P, C - q + p)$
  if $\text{Cost}(P, C \backslash \{p\} \cup \{p\}) < \text{Cost}(P, C)$ **then**
    $C \leftarrow C \backslash \{q'\} \cup \{p'\}$
  end if
end procedure

procedure k-means++ with local search(P, k, Z)
  Initialize C via smart seeding
  for i=1,...,Z **do** $C \leftarrow LocalSearch++(C)$
  end for
  Return C
end procedure

Theorem 7. (LS19) With $Z \geq 10^5 k \log\log k$, one has the CFA $\mathbb{E}[\Phi_K]/\Phi_{K,OPT} \leq 509$. The algorithm runs in time $dnk^2 \log\log k$.

Theorem 8. (CA23) The single swap method achieves a CAF of < 26.64.

▷Ref: Lattanzi et Solher, PMLR 2019
▷Ref: Cohen-Added et al, Neurips 2023

Theory vs practice: k-means++-G vs k-means++-LS

▷ Theory: the approx. factor of k-means++ is better than that of k-means++-G, namely – $O(\log k)$ vs $O(l^3 \log^3 k)$

▷ But in practice: k-means++ used all over.

▷ Conclusion:

- Tight approximation ratios correspond to rather pathological cases that may not be met in practice
- Theoretical analysis based on more realistic data models?
- NB: see the Signal to Noise Ratio used to fit GMM (Chen and Zhang, Neurips 2024)

k-means++-MS: generalizing k-means++-LS

▷ Local search with p multi-swap:

- Opt out $p = O(1)$ seeds at a time – !!! $\binom{k+p}{p}$ candidate swaps !!!
- Perform $Z = O(ndk^{p-1})$ iterations

Theorem 9. (CA23) k-means++-MS achieves a constant factor approximation of CAF < 10.48

▷ Greedy variant for one multi-swap iteration: starting with $k + p$ seeds; iteratively removes the seed minimizing the increases in the SSE function – i.e. the seed least useful one to represent the data

▷ Practice: (finally!) outperforms k-means++-G

▷Ref: Cohen-added et al, Neurips 2023

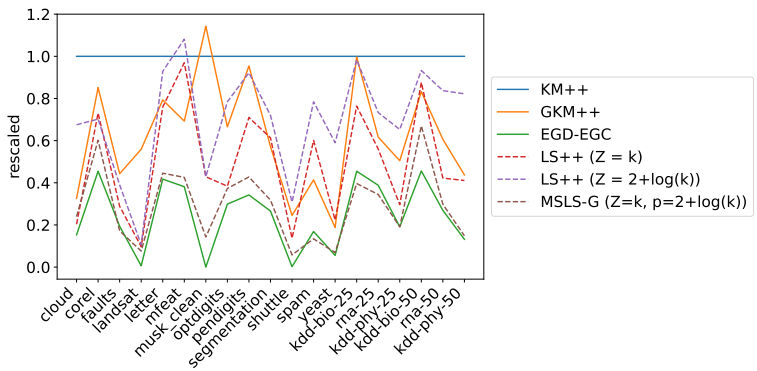
Seeding strategies: limitations

- Seeding methods use distances between points; Φ_K uses distances to centroids
- Greedy seeding incurs a variance drop off along the choice of seeds
- First seed: maximum variance for pairwise distances
- Last seed: much less

Our contribution: a novel line of seeding methods

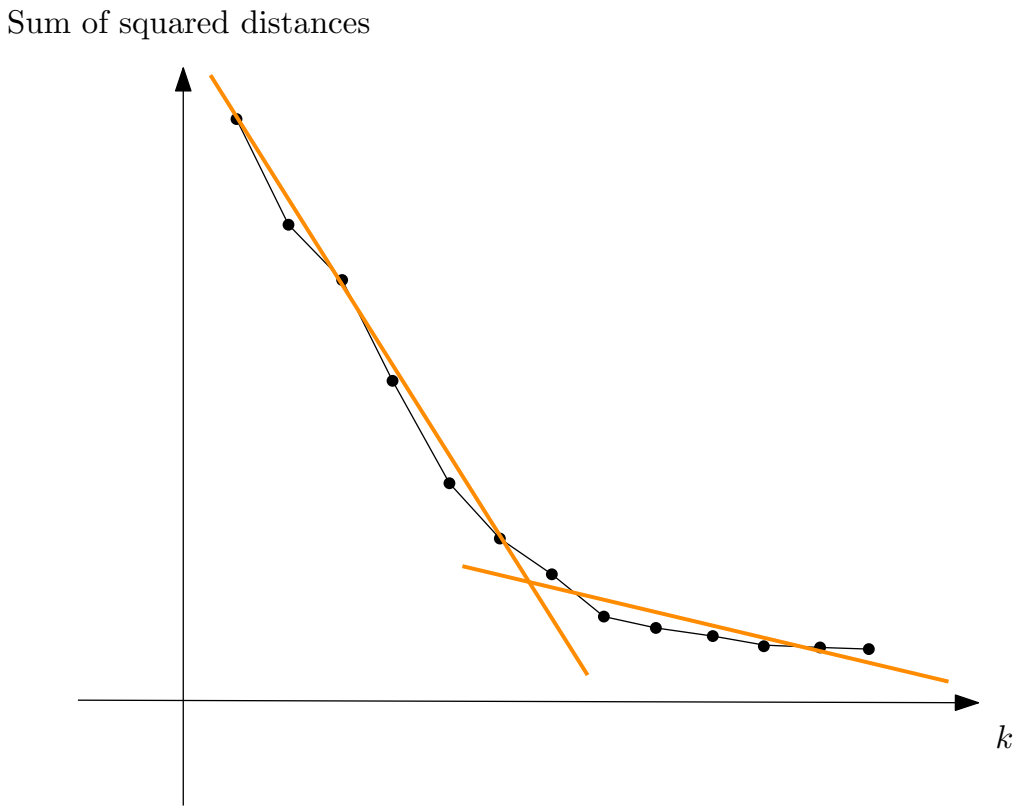
▷ Observations and fixes:

- Seeding methods use distances between points; Φ_K uses distances to centroids
  - $\rightarrow$ look ahead and rank candidates using distances to centroids induced by seeds
- Greedy seeding incurs a variance drop off along the choice of seeds
  - $\rightarrow$ order the seeds being challenged for local search



▷Ref: Carrière and Cazals, 2025

On the number of cluster : the elbow criterion



▷ Nb: cf also the scree plot for PCA: $g(k) = \sum_k \lambda_i$

▷ Exercise: use dynamic programming to fit a bilinear least square model – or more generally a piecewise linear model with k pieces

Algorithms

`k-means`, `k-means++`: basics

Seeding: local searches + multi-swaps and beyond

Kmeans and co-clustering

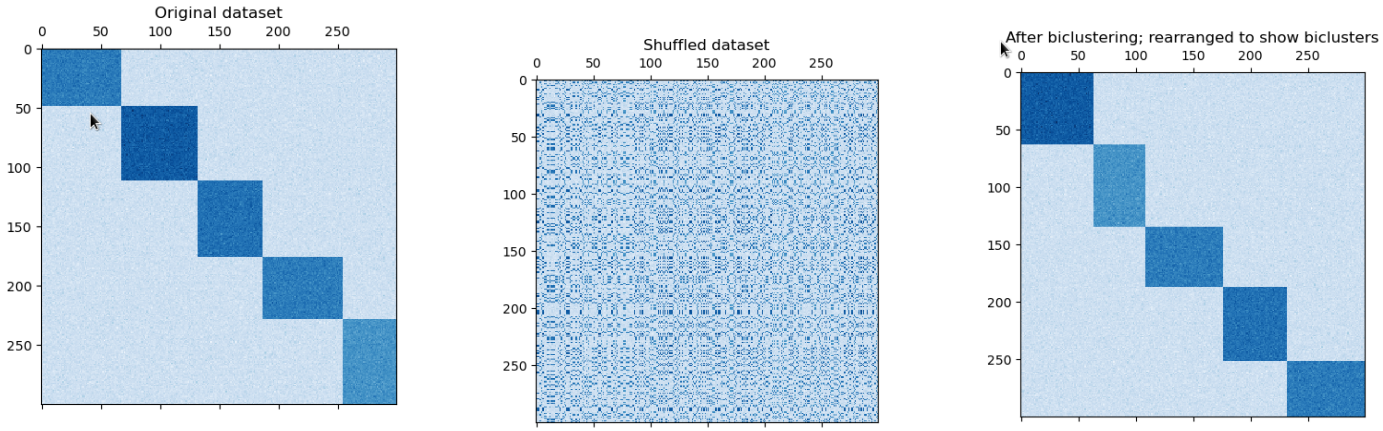
Comparing clustering using meta-clusters

GMM: warmup

Fitting GMM with SOFT EM

Fitting GMM with HARD EM

Co-clustering



https://scikit-learn.org/stable/auto_examples/bicluster/plot_spectral_coclustering.html

k-means and 2-factor Non Negative Matrix Factorization (NMTF)

▷ Data matrix approximation for clustering: approximate the $n \times p$ data matrix X:

$$X \approx GF \tag{7}$$

$$\text{Dimension-wise:}(n \times p) : (n \times K)(K \times p), \tag{8}$$

with

- Cluster indicator matrix $G \in \mathbb{R}_+^{n \times K}$
- Centroids matrix $F \in \mathbb{R}^{K \times p}$ – centroids of the clusters

▷ The relationship between k-means and NMTF is given by

Theorem 10. Orthogonal NMF, which solves

$$\min_{F \geq 0, G \geq 0} \|X - GF\|_F^2 \text{ such that } G^T G = I_K, \tag{9}$$

is equivalent to k-means clustering.

▷ Benefit of using matrix factorization: ability to handle several decompositions

▷Ref: Ding et al, Orthogonal nonnegative matrix tri-factorizations for clustering, 2006

2-factor NMTF: illustration

- Matrix approximation:

$$X \approx GF \tag{10}$$

$$\text{Dimension-wise:}(n \times p) : (n \times K)(K \times p) \tag{11}$$

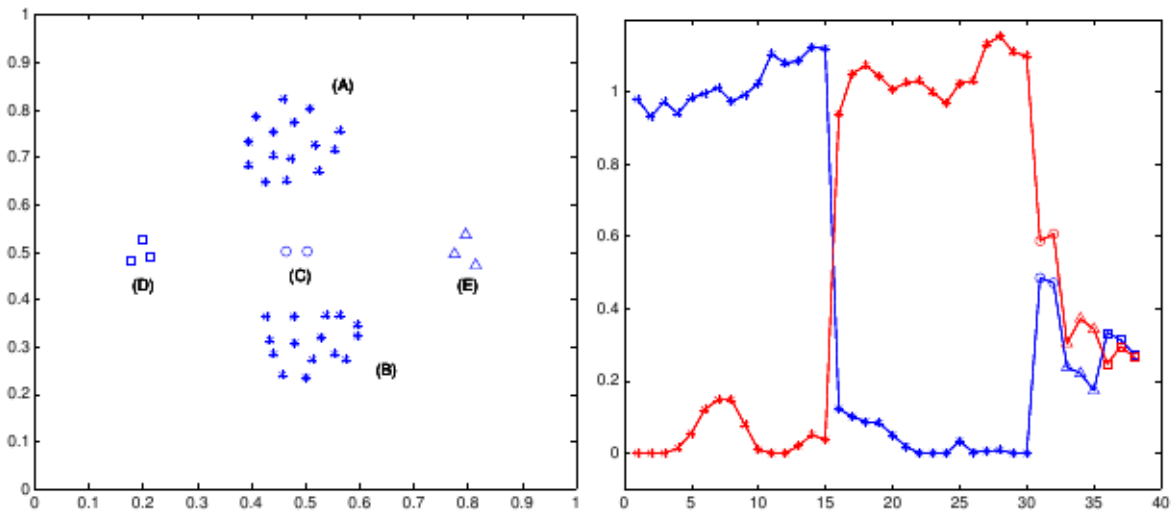


Figure 3: Left: A 2D dataset of 38 data points. Right: Their $H = (\mathbf{h}_1, \mathbf{h}_2)$ values are shown as blue and red curves. Datapoints are ordered by regions $\{B, A, C, E, D\}$, where $B = \{x_1, \cdots, x_{15}\}$, $A = \{x_{16}, \cdots, x_{30}\}$, $C = \{x_{31}, x_{32}\}$, $E = \{x_{33}, x_{34}, x_{35}\}$, $D = \{x_{36}, x_{37}, x_{38}\}$.) H values for points in regions $\{C, E, D\}$ indicate they are fractionally assigned to clusters.

NMF: illustration. From [?]. Nb: on this example, the orthogonality of matrix G has not been enforced. This is soft rather than hard clustering.

Algorithms

`k-means`, `k-means++`: basics

Seeding: local searches + multi-swaps and beyond

Kmeans and co-clustering

Comparing clustering using meta-clusters

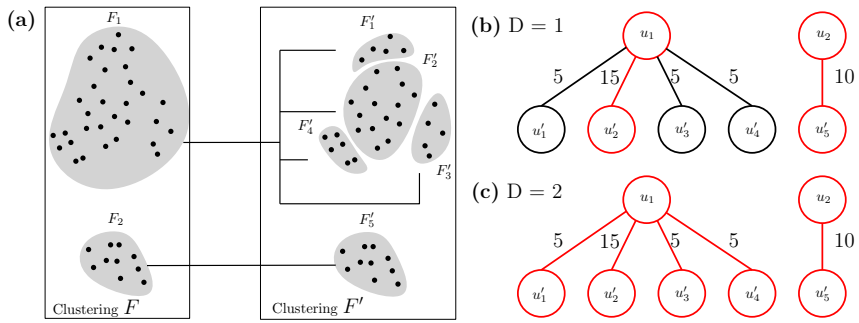
GMM: warmup

Fitting GMM with SOFT EM

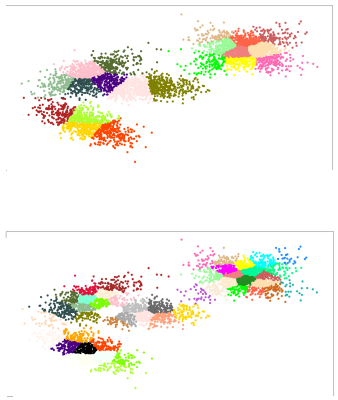
Fitting GMM with HARD EM

Comparing two clusterings using matchings between clusters of clusters

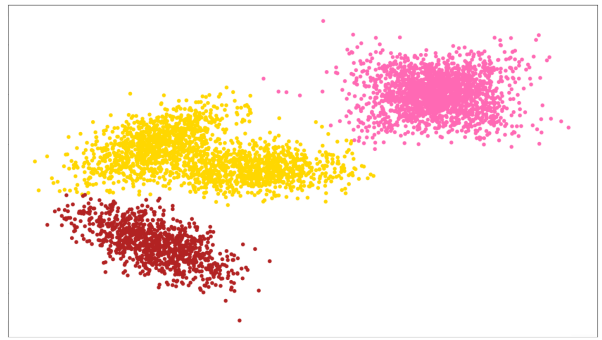
F. Cazals, D. Mazauric, R. Tetley, and R. Watrigant
ACM Trans. Exp. Algorithms, 2019
`https://sbl.inria.fr/doc/D_family_matching-user-manual.html`



Merging clusters: a matter of scale



(A) Two clusterings (kmeans++, Tomato, etc)



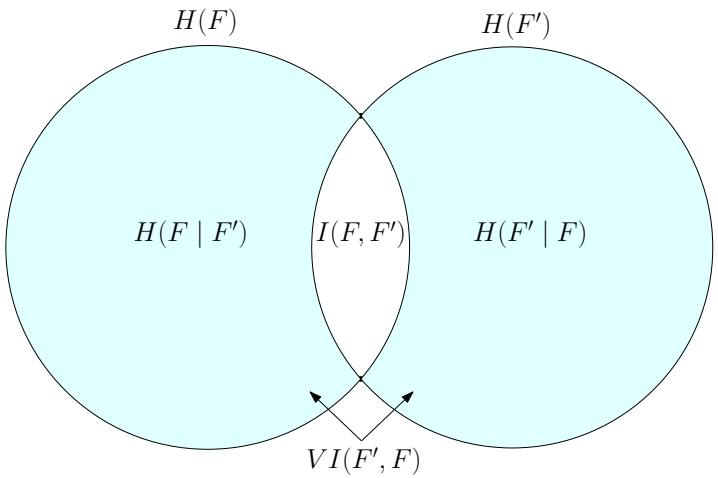
(B) Meta-clusters as union of clusters

Comparing clusterings: the Variation of Information

- A set Z of t items
- A clustering F of size r for Z: $F = \{F_1, \ldots, F_r\}$; $n_k = |F_k|$; $p_k = n_k/t$.
- A clustering F of size r' for Z: $F = \{F_1, \ldots, F_r\}$; $n'_k = |F'_{k'}|$;
- Overlap between two clusters: $p(k, k') = |F_k \cap F'_{k'}|/t$.
- Entropy of clustering: $H(F) = -\sum_{k=1,\ldots,r} p(k) \ln p(k)$

- Mutual information between F and F':

$$I(F, F') = \sum_k \sum_{k'} p(k, k') \ln \frac{p(k, k')}{p(k)p(k')}.$$



- Variation of information (VI):

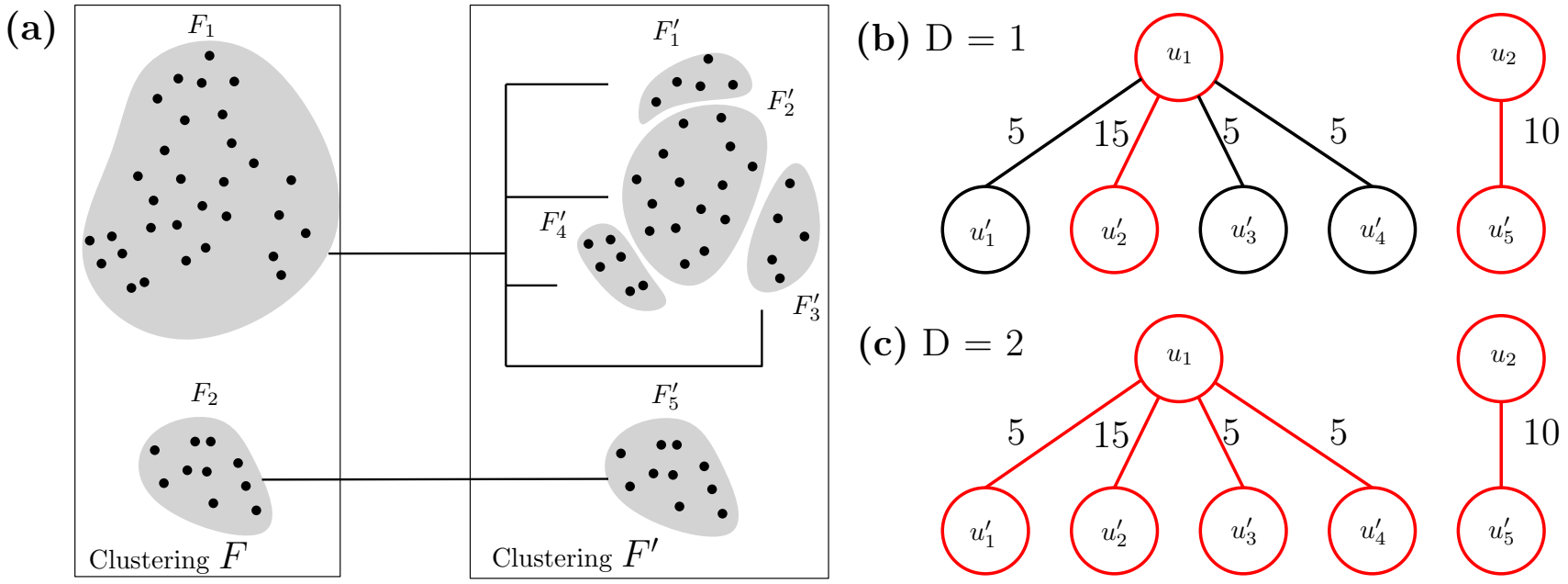
$$VI(F, F') = H(F) + H(F') - 2I(F, F').$$

- Main properties:
  - VI is a metric
  - $VI(F, F') \leq \ln t$

▷Ref: M. Meila, Journal of Multivariate Analysis, 2007

Grouping clusters into **metaclusters:** problem formalization in terms of intersection graph

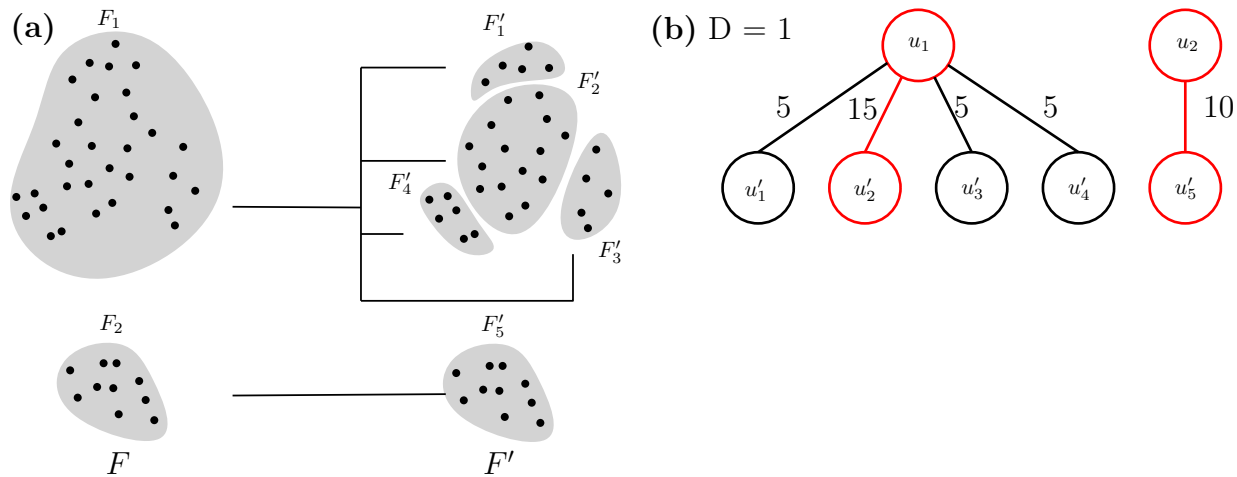
- ▷ Goal: recovering some coherence between groups of clusters
  - as a function of a scale parameter D



- ▷ Rationale: many-to-many
  - Aggregating many clusters, map to many clusters
  - Characterize the scale at which clusters merge

Comparing clusterings: previous work

- ▷ 1-1 mapping of clusters: equivalent to the problem of computing a maximum weighted matching in weighted bipartite graph.
- ▷ Solution: solved in $O(n^2 \log n + nm)$
- ▷ Particular case of the D-family-matching problem for $D = 1$ – see later



Intersection graph

- Notations:
  - Data: $Z = \{z_1, \ldots, z_t\}$
  - Clustering F of size r: $F = \{F_1, \ldots, F_r\}$

    $$F_i \subseteq Z, F_i \neq \emptyset \text{ and } F_i \cap F_j = \emptyset \text{ for every } i, j \in \{1, \ldots, r\}, i \neq j.$$

  - Clustering F' of size r': $F' = \{F'_1, \ldots, F'_{r'}\}$

    $$F'_i \subseteq Z, F'_i \neq \emptyset, \text{ and } F'_i \cap F'_j = \emptyset \text{ for every } i, j \in \{1, \ldots, r'\}, i \neq j.$$

NB: a clustering may not contain all t items

Definition 11 (Intersection graph $G = (U, U', E, w)$ for F and F').

The set $U = \{u_1, \ldots, u_r\}$: vertices of F
The set $U' = \{u'_1, \ldots, u'_{r'}\}$: vertices of F'
Edges $E = \{\{u_i, u'_j\} \mid F_i \cap F'_j \neq \emptyset, 1 \leq i \leq r, 1 \leq j \leq r'\}$.
Edge weight of edge $e = \{u_i, u'_j\} \in E$ is $w_e = |F_i \cap F'_j|$.

D-family matching

▷ Let $D \in \mathbb{N}^+$: a constraint on the diameter of certain subgraph of the intersection graph

Definition 12. [D-family-matching for an intersection graph]
A family $\mathcal{S} = \{S_1, \ldots, S_k\}$, $k \geq 1$, such that

- for every $i, j \in \{1, \ldots, k\}$, if $i \neq j$, then: $S_i \subseteq V$, $S_i \neq \emptyset$, $S_i \cap S_j = \emptyset$,
- and the graph $G[S_i]$ induced by the set of nodes S_i has diameter at most D.

▷ Comments:

- $D = 1$: matching
- $D = 2$: clusters as stars

▷ Notations:

- Set of all D-family matchings of a graph G: $\mathcal{S}_D(G)$

D-family matching problem

- Score $\Phi(\mathcal{S})$ of a D-family-matching $\mathcal{S}$:

$$\Phi(\mathcal{S}) = \sum_{i=1}^{k} \sum_{e \in E(G[S_i])} w_e. \tag{12}$$

- Remarks:
  - The sum runs over **all** edges of a connected component. (Later: see algorithms based on spanning trees.)
  - We wish to compute a D-family-matching which minimizes the inconsistencies.

Definition 13 (D-family-matching problem). Let $D \in \mathbb{N}^+$. Given an intersection graph G, the D-family-matching problem consists in computing

$$\text{(Opt score for a given } D) \quad \Phi_D(G) = \max_{\mathcal{S} \in \mathcal{S}_D(G)} \Phi(\mathcal{S}). \tag{13}$$

NB: Score with the diameter D stressed: $\Phi(\mathcal{S}^{D=d})$

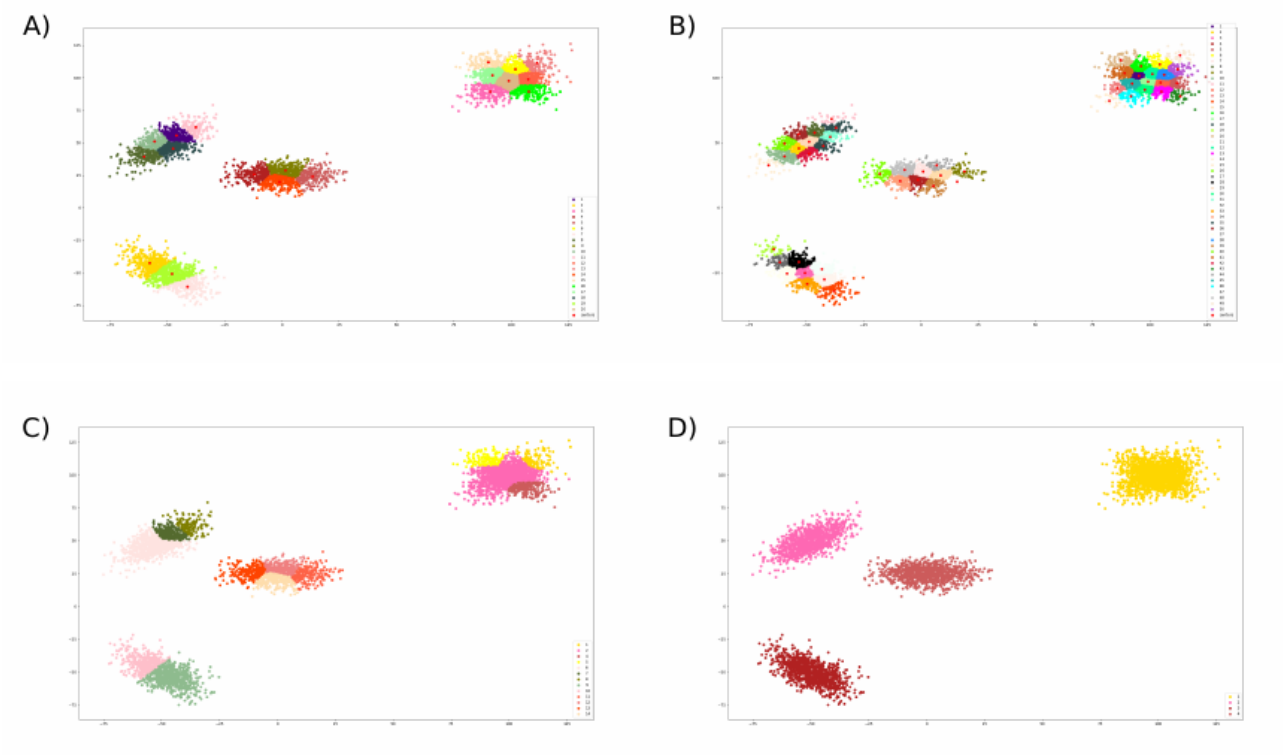
Comparing clusterings: at which scale do clusters merge?

What is the right number of clusters?

- ▷ Example:
  - Using `k-means++` to cluster 5000 samples from five Gaussian blobs
  - Using D-family matching to infer the right/natural # of clusters

(A) k-means++, $k = 20$

(B) k-means++, $k = 50$



(C) $D = 3$, 17 meta clusters, $\Phi_D(G) = 4068$

(D) $D = 4$, 4 meta clusters, $\Phi_D(G) = 5000$

Algorithms

`k-means`, `k-means++`: basics

Seeding: local searches + multi-swaps and beyond

Kmeans and co-clustering

Comparing clustering using meta-clusters

GMM: warmup

Fitting GMM with SOFT EM

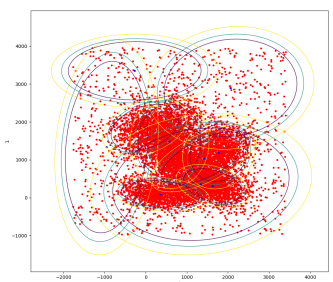
Fitting GMM with HARD EM

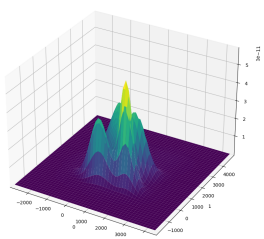
Gaussian Mixture Models

- Point set: $\{x_1, \ldots, x_n\} \subset \mathbb{R}^d$.
- Gaussian mixture model:

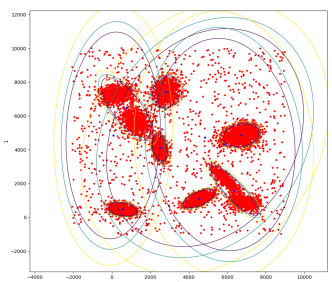
$$g(x) = \sum_{k=1,\ldots,K} w_k \mathcal{N}(x_i \mid \mu_k, \Sigma_k)), \sum_k w_k = 1. \tag{14}$$

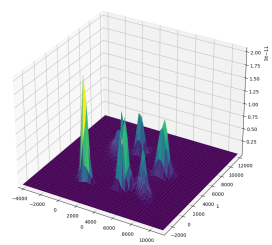
- Question: design a GMM fitting X
- Parameters: $\Theta = \{\mu_k, \Sigma_k\}, k = 1, \ldots, K$; $\Theta' = \Theta \cup \{w_k\}, k = 1, \ldots, K$





Low separation





Clear separation

From a data partition to the initial GMM

▷ Find representative centers: `k-means++`

▷ `Means2GMM` algorithm:

1: **procedure** `Means2GMM`$(X, \mu_1, ..., \mu_K)$

2: Partition $C_1, ..., C_K \overset{Def}{=}$ assign each $x_i \in X$ to its closest mean
3: //Build GMM components
4: **for** $k \leftarrow 1$ to K **do**
5: $\mu_k = 1/|C_k| \sum_{x \in C_k} x$
6: $w_k = |C_k| \ / \ |X|$
7: $\Sigma_k = 1/|C_k| \sum_{x \in C_k} (x - \mu_k)(x - \mu_k)^T$
8: If Σ_k is not positive definite, take $\Sigma_k = 1/(d|C_k|) \sum_{x \in C_k} \|x - \mu_k\|^2 \mathbf{I}_d)$
9: If Σ_k is still not positive definite, take $\Sigma_k = \mathbf{I}_d$
10: **end for**
11: **end procedure**

▷ `Means2SphGMM`: change the full anisotropic estimation of line 7 by the isotropic estimation of line 8.

▷`Ref: Blomer et al, 2016`

Algorithms

`k-means`, `k-means++`: basics

Seeding: local searches + multi-swaps and beyond

Kmeans and co-clustering

Comparing clustering using meta-clusters

GMM: warmup

Fitting GMM with SOFT EM

Fitting GMM with HARD EM

The multivariate Gaussian and the Mahalanobis distance

▷ The d-dimensional normal multivariate distribution:

$$\mathcal{N}(x \mid \mu, \Sigma) = \frac{1}{\det(2\pi\Sigma)^{1/2}} \exp(-\frac{1}{2}(x-\mu)^{\mathsf{T}}\Sigma^{-1}(x-\mu)). \tag{15}$$

Definition 14. Given a probability distribution Q with mean μ and positive semi-definite covariance matrix Σ, the Mahalanobis distance is

$$d_M(x, Q) = \sqrt{(x-\mu)^{\mathsf{T}}\Sigma^{-1}(x-\mu)}$$
$$d_M(x, y) = \sqrt{(x-y)^{\mathsf{T}}\Sigma^{-1}(x-y)}$$

▷ Mahalanobis distance as a Euclidean norm:
Assuming one can write $\Sigma^{-1} = B^{-\mathsf{T}}B^{-1}$:

$$d_M(v) = \sqrt{v^{\mathsf{T}}B^{-\mathsf{T}}B^{-1}v} = \sqrt{(B^{-1}v)^{\mathsf{T}}B^{-1}v} = \left\|B^{-1}v\right\|. \tag{16}$$

▷Ref: `Kessy et al, Opt whitening and decorrelation, 2018`

Whitening:
Transforming RV to orthogonality

- ▷ Whitening: transforming random variables to orthogonality
- ▷ Gaussian distribution:

Theorem 15. Let $X \sim \mathcal{N}(\mu, \Sigma)$. Define $Z = B^{-1}(X - \mu)$ with B defined below. Then $Z \sim \mathcal{N}(0, \mathbf{I}_d)$.

▷ Proof sketch: 1. Decomposition of the covariance matrix Using the spectral theorem:

$$\Sigma = PDP^{\mathsf{T}} = PD^{1/2}(PD^{1/2})^{\mathsf{T}} \overset{Def}{=} BB^{\mathsf{T}}. \tag{17}$$

NB: expression for Σ^{-1}:

$$\Sigma\Sigma^{-1} = BB^{\mathsf{T}}\Sigma = I \Rightarrow \Sigma^{-1} = B^{-\mathsf{T}}B^{-1} \text{ or } \Sigma^{-1/2} = B^{-1} \tag{18}$$

2. Change of variables:

$$f_Z(z) = f_X(x) \mid \det(Jacobian) \mid .$$

▷Ref: Kessy et al, Optimal whitening and decorrelation, The American Statistician, 2018

The multivariate Gaussian: posterior

Consider a mixture of K Gaussians

$$g(x) = \sum_{k=1,\ldots,K} w_k \mathcal{N}(x_i \mid \mu_k, \Sigma_k)), \sum_k w_k = 1. \tag{19}$$

Given a point x, let us investigate the probability to have x generated by the k-th component. To model, this process, consider K boolean latent variables, with $z_k = 1$ iff the k-th component generated x. From which we define:

$$\begin{cases} \text{prior probability:} & \mathcal{N}(x \mid \mu_k, \Sigma_k) \\ \text{posterior probability:} & \gamma_{xk} = \mathbb{P}[z_k = 1 \mid x] \end{cases} \tag{20}$$

Using Bayes' rule, we obtain:

$$\gamma_{xk} = \mathbb{P}[z_k = 1 \mid x] = \frac{\mathbb{P}[x|z_k = 1]\,\mathbb{P}[z_k = 1]}{\mathbb{P}[x]} = \frac{w_k \mathcal{N}(x \mid \mu_k, \Sigma_k)}{\sum_k w_k \mathcal{N}(x \mid \mu_k, \Sigma_k)}. \tag{21}$$

Maximum Likelihood: problem

- Likelihood for the n samples:

$$\mathbb{P}\left[X \mid \Theta'\right] = \prod_{i=1,\ldots,n} g(x). \tag{22}$$

- Associated log likelihood for the n samples:

$$\mathcal{LL}(X \mid \Theta') = \ln \mathcal{N}(X \mid \Theta') = \sum_{i=1,\ldots,n} \ln\Big(\sum_k w_k \mathcal{N}(x_i \mid \mu_k, \Sigma_k)\Big). \tag{23}$$

- Goal: maximize the likelihood.

Maximum Likelihood: solutions

▷ Maximization wrt to the μ_k: derivative of the LL (Eq. 23) wrt the μ_k yields

$$-\sum_k \gamma_{ik}\Sigma_k(x_i - \mu_k) = 0 \tag{24}$$

from which we get

$$\begin{cases} \mu_k = \frac{1}{N_k}\sum_i \gamma_{ik}x_i, \\ N_k = \sum_i \gamma_{ik}. \end{cases} \tag{25}$$

NB: μ_k is a weighted center of mass, with weights equal to the posterior probabilities.

▷ Maximization wrt to the Σ_k:

$$\Sigma_k = \frac{1}{N_k}\sum_i \gamma_{ik}(x_i - \mu_k)(x_i - \mu_k)^\mathsf{T}. \tag{26}$$

▷ Maximization wrt to the mixing coefficients π_k: This is done using Lagrange multipliers:

$$\sum_{i=1,\ldots,n} \ln\Big(\sum_k w_k \mathcal{N}(x_i \mid \mu_k, \Sigma_k)\Big) + \lambda\Big(\sum_k w_k - 1\Big). \tag{27}$$

and the calculation yields

$$w_k = {}^{N_k}/_{N}. \tag{28}$$

▷Ref: `Bishop, Patter recognition and machine learning, 2006`

Log likelihood and Gaussians:

singularities - over-fitting

▷ Pb with singularities: assume that

- the covariance matrices satisfy $\Sigma_k = \sigma_k^2 \mathbf{I}_d$.
- some sample point matches one mean, that is $\mu_k = x_i$, for some indices k and i.

We get the probability

$$\mathcal{N}(x_i \mid \mu_k, \Sigma_k) = \frac{1}{\sqrt{|2\pi\sigma_k^2 \mathbf{I}_d|}} = \frac{1}{(2\pi)^{d/2}} \frac{1}{\sigma_k^d}. \tag{29}$$

- This terms tends to infinity, and so does the LL.
- When fitting a GMM, if a components specializes to one point, its variances goes to zero, and the LL goes to infinity. Thus, need to identify singular components and process them accordingly.

Algorithms

`k-means`, `k-means++`: basics

Seeding: local searches + multi-swaps and beyond

Kmeans and co-clustering

Comparing clustering using meta-clusters

GMM: warmup

Fitting GMM with SOFT EM

Fitting GMM with HARD EM

Problem statement and hardness

▷ Goal: for a point cloud generated by a k-GMM: identify the generator of each sample

$$\text{Find: } z^* = (z_j^*)^\mathsf{T} \in [k]^n \tag{30}$$

▷ General sampling model:

$$Y_j = \Theta^*_{z_j^*} + \varepsilon_j, \text{ with } \varepsilon_j \sim \mathcal{N}(0, \Sigma_j^*) \tag{31}$$

- Model 1: different centers $\Theta^*_{z_j^*}$; global covariance matrix Σ^*
- Model 2: different centers $\Theta^*_{z_j^*}$; different covariance matrices Σ_j^*

▷ NB: model 1 with Σ^* is known: applying whitening, problem converted into an isotropic GMM via the transform $\Sigma^{*-\frac{1}{2}} Y_j$

▷ Loss function: need to find the proper permutation of labels:

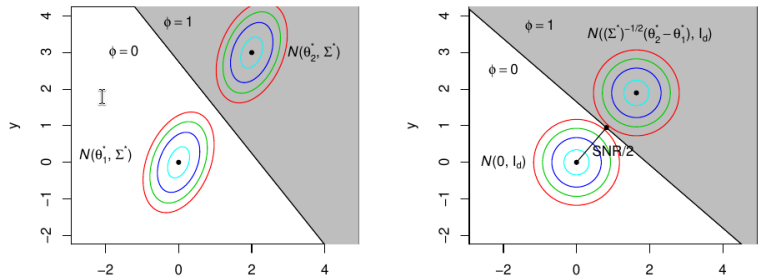
$$\text{For any } z, z^* \in [k]^n : h(z, z^*) \min_{\psi \in \Psi} \frac{1}{n} \sum_j \mathbf{1}(\psi(z_j) \neq z_j^*) \tag{32}$$

▷Ref: `Chen and Zhang, NeurIPS 2024`

Hardness via Signal to Noise Ratio

- Difficulty of clustering: separation between the Gaussians
- For isotropic Gaussians: Signal to Noise ratio using the Mahalanobis distance

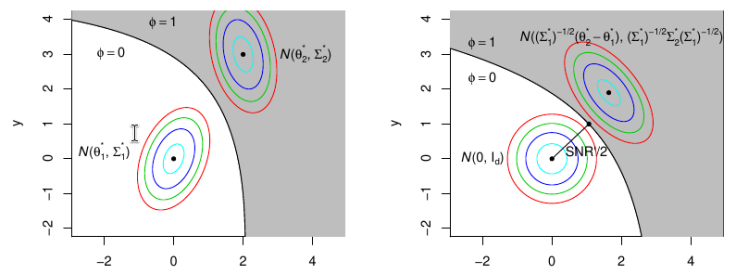
$$SNR = \min_{a,b\in[k];a\neq b} \left\| \Sigma^{*-\frac{1}{2}} (\Theta_a^* - \Theta_b^*) \right\|. \tag{33}$$



NB:

$$\Sigma^* = \sigma^2 \mathbf{I}_d \Rightarrow \text{SNR} = \frac{\Delta}{\sigma}, \text{ with } \Delta = \min_{a,b\in[k],a\neq b} \|\Theta_a^* - \Theta_b^*\|. \tag{34}$$

- Non isotropic Gaussians: no closed form



- Ref: Chen and Zhang, NeurIPS 2024

EM with hard clustering

Algorithm 2: Adjusted Lloyd's Algorithm for Model 2.

Input: Data Y, number of clusters k, an initialization $z^{(0)}$, number of iterations T.
Output: $z^{(T)}$

1 **for** $t = 1, \ldots, T$ **do**

2 Update the centers:

$$\theta_a^{(t)} = \frac{\sum_{j\in[n]} Y_j \mathbb{I}\left\{z_j^{(t-1)} = a\right\}}{\sum_{j\in[n]} \mathbb{I}\left\{z_j^{(t-1)} = a\right\}}, \quad \forall a \in [k]. \tag{12}$$

3 Update the covariance matrices:

$$\Sigma_a^{(t)} = \frac{\sum_{j\in[n]} (Y_j - \theta_a^{(t)})(Y_j - \theta_a^{(t)})^T \mathbb{I}\left\{z_j^{(t-1)} = a\right\}}{\sum_{j\in[n]} \mathbb{I}\left\{z_j^{(t-1)} = a\right\}}, \quad \forall a \in [k]. \tag{13}$$

4 Update the cluster assignment vector:

$$z_j^{(t)} = \underset{a\in[k]}{\operatorname{argmin}} (Y_j - \theta_a^{(t)})^T (\Sigma_a^{(t)})^{-1} (Y_j - \theta_a^{(t)}) + \log|\Sigma_a^{(t)}|, \quad \forall j \in [n]. \tag{14}$$

▷Ref: Chen and Zhang, NeurIPS 2024

Consistent clustering and minimax lower bounds

Performing best in the worst case

Model 1

Theorem 2.1. *Under the assumption* $\frac{SNR}{\sqrt{\log k}} \to \infty$, *we have*

$$\inf_{\hat{z}} \sup_{z^* \in [k]^n} \mathbb{E} h(\hat{z}, z^*) \geq \exp\left(-(1+o(1)) \frac{SNR^2}{8} \right). \tag{6}$$

If $SNR = O(1)$ *instead, we have* $\inf_{\hat{z}} \sup_{z^* \in [k]^n} \mathbb{E} h(\hat{z}, z^*) \geq c$ *for some constant* $c > 0$.

Model 2

Theorem 3.1. *Assume* $d = O(1)$ *and* $\max_{a,b \in [k]} \lambda_d(\Sigma_a^*)/\lambda_1(\Sigma_b^*) = O(1)$. *Under the assumption* $\frac{SNR'}{\sqrt{\log k}} \to \infty$, *we have*

$$\inf_{\hat{z}} \sup_{z^* \in [k]^n} \mathbb{E} h(\hat{z}, z^*) \geq \exp\left(-(1+o(1)) \frac{SNR'^2}{8} \right).$$

If $SNR' = O(1)$ *instead, we have* $\inf_{\hat{z}} \sup_{z^* \in [k]^n} \mathbb{E} h(\hat{z}, z^*) \geq c$ *for some constant* $c > 0$.

▷ Comments:

- Parameter space only for z^*; centers and covariances are fixed
- If k is fixed: $SNR \to \infty$ is a sufficient condition for consistent clustering

▷Ref: Chen and Zhang, NeurIPS 2024

Algorithms

PART 1: Kmeans and EM
PART 2: Fitting complex mixtures in flat torii

Algorithms

Model selection: warmup

Modeling joint distribution on flat torii

The Minimum Message Length approach to model selection

Model selection: goal and notations

- Goal: given a set of iid observations, select a model that
  - fits/explains the data,
  - and/or possibly predicts new outcomes – generative model

- Notations:
  - Θ: some space defining a statistical model with d-dimensional parameters
  - Prior on the parameters: $h(\theta), \theta \in \Theta$
  - $x^{(n)}$: a sequence of n iid observations of some unknown random variable X. Abusing notations: denoted x
  - $\log f(x;\theta)$ the log likelihood of data with respect to a statistical model / hypothesis

- Example / Bernoulli: with n coin tosses $x = \{x_1, \ldots, x_n\}$ and k Head, estimate $\hat{p}$ to get Head:

$$f(x;\theta) = \binom{n}{k}\theta^k(1-\theta)^{n-k}; \log f(x;\theta) = C + k\log\theta + (n-k)\log(1-\theta) \qquad (35)$$

$\Rightarrow \max_\theta \log f(x;\theta)$ yields $\hat{\theta} = k/n$.

`https://en.wikipedia.org/wiki/Model_selection`

Model selection: classical strategies

▷ Maximum likelihood estimator $\hat{\theta}_{\mathrm{ML}}$:

$$\hat{\theta}_{MLE} = \arg\max_{\theta} \log \mathbb{P}[x|\theta]\,. \tag{36}$$

- No a priori on the statistical model, model complexity ignored

▷ Maximum a Posteriori $\hat{\theta}_{\mathrm{MAP}}$:

$$\arg\max_{\theta} \mathbb{P}[\theta|x] = \arg\max_{\theta} \frac{\mathbb{P}[x|\theta]\,\mathbb{P}[\theta]}{\cancel{\mathbb{P}[x]}}\,. \tag{37}$$

- uses a prior belief on θ; ignores $\mathbb{P}[x]$

▷ Bayesian: also using a prior on θ, estimate the full posterior

$$\mathbb{P}[\theta|x] = \frac{\mathbb{P}[x|\theta]\,\mathbb{P}[\theta]}{\mathbb{P}[x]}\,. \tag{38}$$

- Difficulty: computing $\mathbb{P}[x] = \int_{\theta} \mathbb{P}[x|\theta]\, d\mathbb{P}[\theta]$

▷ NB: posterior = likelihood $\times$ prior / evidence

▷ Minimum Description/Message Length: $\underline{\text{likelihood}} + \underline{\text{model complexity}} \Rightarrow$ of interest for mixtures

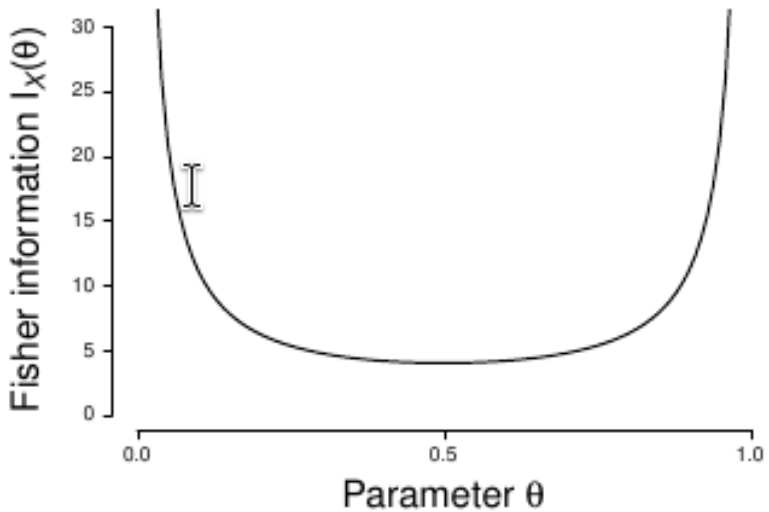
(Univariate) Fisher information - definition

- ▷ Goal: assess the overall sensitivity of a statistical model to its param. θ

Definition 16. (Unit Fisher information)

$$I_\theta = \begin{cases} \sum_{x \in \mathcal{X}} [\frac{d}{d\theta} \log f(x|\theta)]^2 p_\theta(x) (\text{discrete}) \\ \int_x [\frac{d}{d\theta} \log f(x|\theta)]^2 f(x \mid \theta) dx (\text{continuous}) \end{cases} \tag{39}$$

- ▷ Example: Bernoulli / coin toss with parameter θ : $I_\theta = \frac{1}{\theta(1-\theta)}$



- ▷ NB: for n iid trials: $I_{X^n;\theta} = nI_\theta$
- ▷Ref: Ly et al, J. of Mathematical Psychology, 2017

Fisher matrix: multivariate case

Definition 17. Matrix

$$I_\theta = \left(I_{ij}(\theta)\right), \tag{40}$$

with

$$I_{ij}(\theta) = -\mathbb{E}_{\mathcal{X}}\left[\frac{\partial^2 \ln L}{\partial\theta_i \partial\theta_j}\right] \tag{41}$$

- NB: Fisher's matrix is positive semidefinite.

Priors and Jeffreys' prior

- Uniform prior: assigns the same probability to every model set of the same volume
- Caveats:
  - Without having seen any datum: any region in model space equally likely
  - Posterior is not invariant by reparameterization of the model – see Example below
- One substitute:

Definition 18. (Jeffreys' prior)

$$g_J(\theta) = \frac{\sqrt{I_\theta}}{V}, \text{ with } V = \int_\theta \sqrt{I_\theta} d\theta. \tag{42}$$

- Example: Jeffreys' prior for the Bernoulli experiment: $g_J(\theta) = \frac{1}{\pi\sqrt{\theta(1-\theta)}}$

Model selection and priors: illustration (I)

- Bernoulli experiments with $n_{heads} = 7$ out of 10 tosses: $f(x^n_{obs}) = \theta^7(1-\theta)^3$
- Posterior associated with the uniform prior

$$\mathbb{P}[\theta|x^n] = 1320 * \theta^7(1-\theta)^3 \tag{43}$$

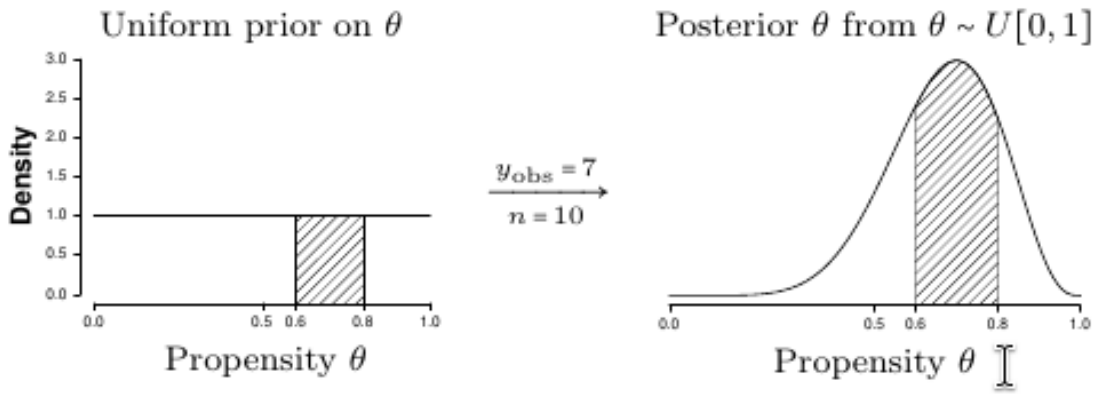
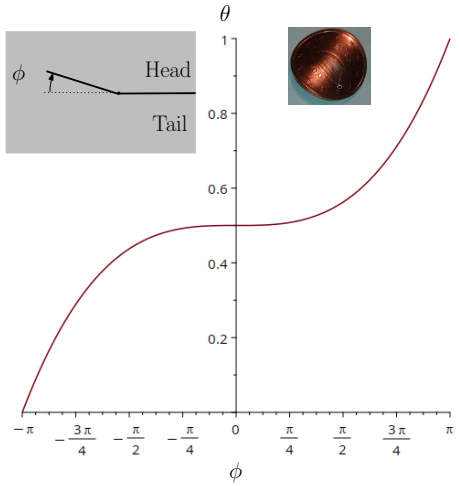


FIG 3. *Bayesian updating based on observations x^n_{obs} with $y_{obs} = 7$ heads out of $n = 10$ tosses. In the left panel, the uniform prior distribution assigns equal probability to every possible value of the coin's propensity θ. In the right panel, the posterior distribution is a compromise between the prior and the observed data.*

▷Ref: Ly et al, J. of Mathematical Psychology, 2017

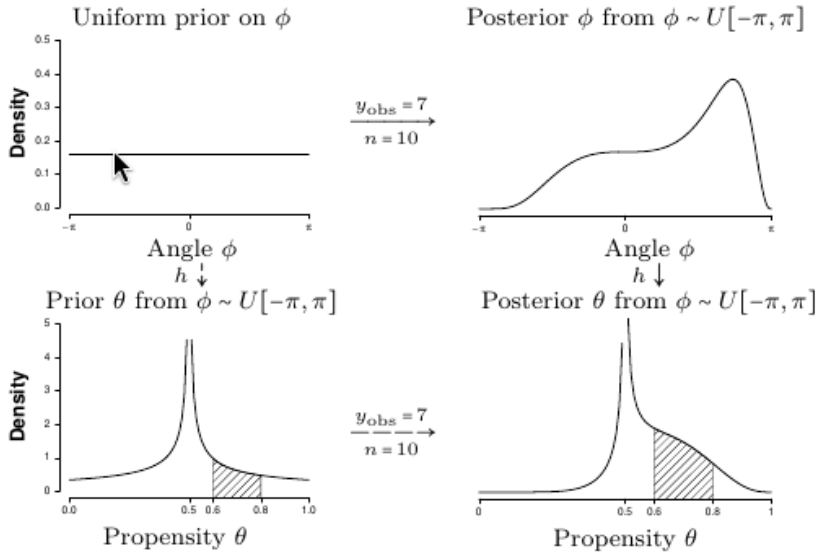
Model selection and priors: example (II)



▷ Reparameterization: propensity θ assuming the coin is bent with angle ϕ

$$\theta = h(\phi)\frac{1}{2} + \frac{1}{2}(\frac{\phi}{3})^3.$$

▷ Densities in param. space and posteriors:



Posterior very different from that with the uniform prior

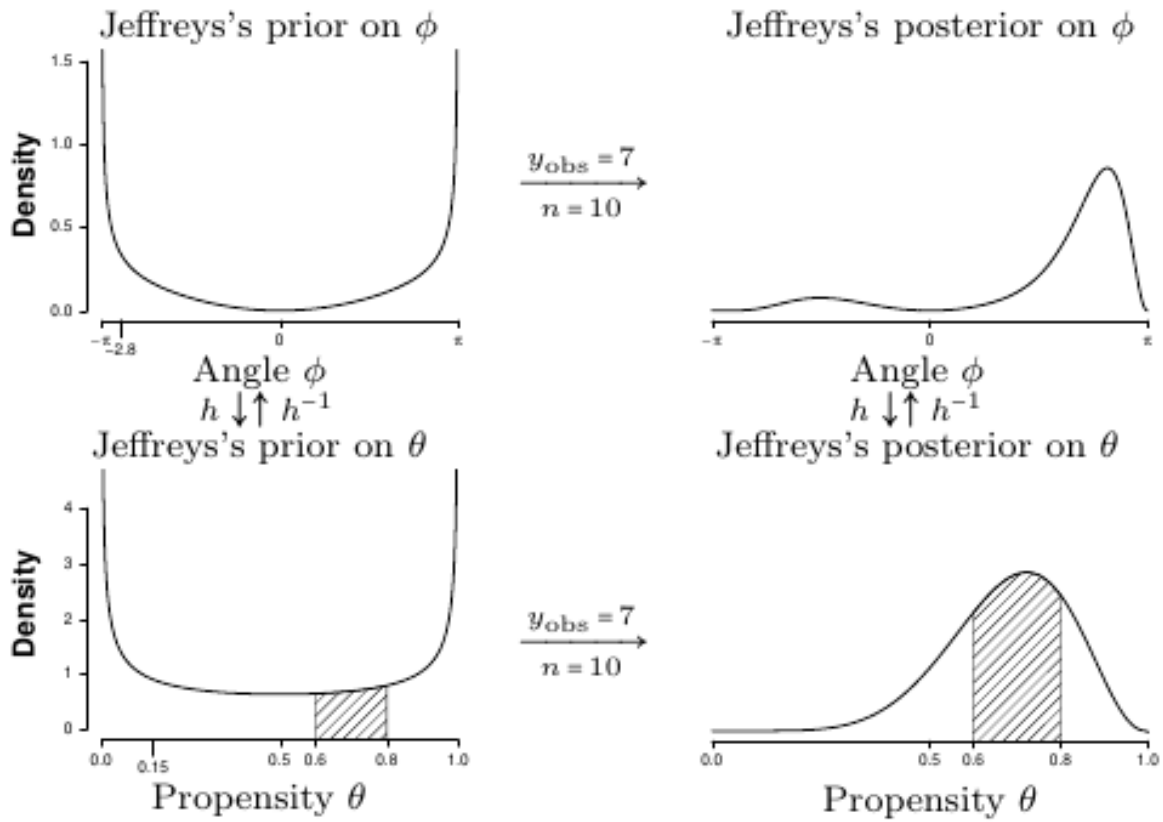
▷Ref: Ly et al, J. of Mathematical Psychology, 2017

Model selection and priors: example (III)

- Using Jeffrey's prior on ϕ or θ yield the same posterior

$$g_J(\phi) = \frac{3\phi^2}{\pi\sqrt{\pi^6 - \phi^6}} \tag{44}$$

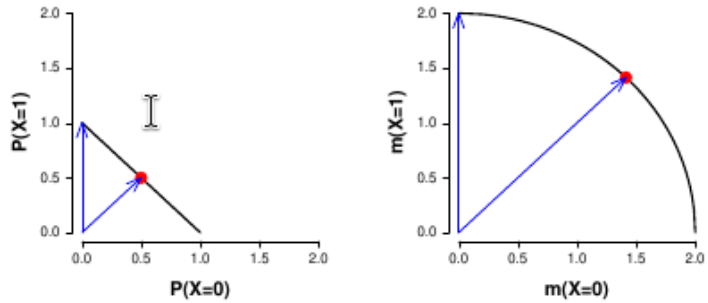
$$g_J(\theta) = \frac{1}{\pi\sqrt{\theta(1-\theta)}}. \tag{45}$$



Ref: Ly et al, J. of Mathematical Psychology, 2017

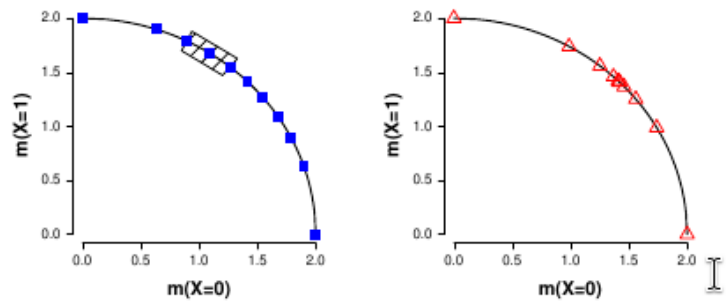
Model selection and priors: example (IV)
Distribution in model space $\mathcal{M}$

▷ Model encoding and model space: $m_\theta = [\mathbb{P}[0], \mathbb{P}[1]]$ vs $m_\theta = [2\sqrt{\mathbb{P}[0]}, 2\sqrt{\mathbb{P}[1]}]$



NB: latter representation preferred since all models have the length

▷ Distribution of models in $\mathcal{M}$ using the θ and ϕ parameterization–spacings 0.1:



▷ Uniform distribution in model space:

$$V = \int_{\mathcal{M}_\theta} 1 dm_\theta(X) = \int_\Theta \sqrt{I_\theta} d\theta. \tag{46}$$

▷Ref: Bickel et al, Efficient and adaptive estim. for semiparametric models, J. Hopkins Univ. Press, 1993

Fisher's info: two expressions

▷ NB: Likelihood: $L = L(x \mid \theta) = f(x \mid \theta)$; $\int_{\mathcal{X}} L dx = 1$.

Lemma 19. Under suitable conditions – see Amari et al:

$$I_\theta = \mathbb{E}_{\mathcal{X}}\left[(\frac{\partial \ln L}{\partial \theta})^2\right] = -\mathbb{E}_{\mathcal{X}}\left[\frac{\partial^2 \ln L}{\partial \theta^2}\right] \tag{47}$$

▷ Proof sketch:

First note the following:

$$(\frac{\partial \ln L}{\partial \theta})^2 = \frac{\partial \ln L}{\partial \theta}\frac{1}{L}\frac{\partial L}{\partial \theta} \tag{48}$$

$$\frac{\partial}{\partial \theta}(L\frac{\partial \ln L}{\partial \theta}) = L\frac{\partial^2 \ln L}{\partial \theta^2} + \frac{\partial L}{\partial \theta}\frac{\partial \ln L}{\partial \theta} \tag{49}$$

Then – subscript $\mathcal{X}$ omitted in expectations:

$$\mathbb{E}\left[(\frac{\partial \ln L}{\partial \theta})^2\right] = \int L(\frac{\partial \ln L}{\partial \theta})^2 dx = \int \frac{\partial \ln L}{\partial \theta}\frac{\partial L}{\partial \theta} dx \text{ – with Eq.(48)} \tag{50}$$

$$= \int \frac{\partial}{\partial \theta}(L\frac{\partial \ln L}{\partial \theta}) - L\frac{\partial^2 \ln L}{\partial \theta^2} dx = \frac{\partial}{\partial \theta}\int L\frac{\partial \ln L}{\partial \theta} dx - \mathbb{E}\left[\frac{\partial^2 \ln L}{\partial \theta^2}\right] \tag{51}$$

$$= \frac{\partial}{\partial \theta}\int \frac{\partial L}{\partial \theta} dx - \mathbb{E}\left[\frac{\partial^2 \ln L}{\partial \theta^2}\right] = \frac{\partial^2}{\partial \theta^2}\int L dx - \mathbb{E}\left[\frac{\partial^2 \ln L}{\partial \theta^2}\right] \tag{52}$$

$$= 0 - \mathbb{E}\left[\frac{\partial^2 \ln L}{\partial \theta^2}\right] \text{ – since } \int L dx = 1. \tag{53}$$

▷Ref: Amari and Nagaoka, Methods of Information Geometry. Oxford Univ. Press, 2000

Invariance of the posterior using Jeffreys' prior

- Generic expression of the posterior pdf using Bayes' rule and Lemma 19:

$$\mathbb{P}[\theta \mid x] = \mathbb{P}[x \mid \theta]\, g_J(\theta) \propto L(x \mid \theta)\sqrt{-\mathbb{E}\left[\frac{\partial^2 \ln L}{\partial \theta^2}\right]} = L(x \mid \theta)\sqrt{\mathbb{E}\left[(\frac{\partial \ln L}{\partial \theta})^2\right]} \quad (54)$$

- Posterior: pdf using new parameter η and dependency $\theta(\eta)$

$$\mathbb{P}[\eta \mid x] = \mathbb{P}[\theta(\eta) \mid x] \mid \frac{\partial \theta}{\partial \eta} \mid = L(x \mid \theta(\eta))\sqrt{\mathbb{E}\left[(\frac{\partial \ln L}{\partial \theta(\eta)})^2\right]} \mid \frac{\partial \theta}{\partial \eta} \mid \quad (55)$$

$$= L(x \mid \theta(\eta))\sqrt{\mathbb{E}\left[(\frac{\partial \ln L}{\partial \theta(\eta)}\frac{\partial \theta}{\partial \eta})^2\right]} \quad (56)$$

- Posterior: direct calculation using the parameter η and the function $\theta(\eta)$

$$\mathbb{P}[\eta \mid x] = L(x \mid \eta)\sqrt{\mathbb{E}\left[(\frac{\partial \ln L}{\partial \eta})^2\right]} = L(x \mid \eta)\sqrt{\mathbb{E}\left[(\frac{\partial \ln L}{\partial \theta}\frac{\partial \theta}{\partial \eta})^2\right]} \quad (57)$$

Model selection strategies

- Comparing various models θ_i: components I_θ
  - Goodness of fit via MLE estimate
  - Dimension i.e. number of free parameters
  - Geometric complexity: volume of model space

- Main strategies
  - Akaike information criterion (AIC)
  - Bayesian information criterion (BIC)
  - Fisher information approximation (FIAT)

$$AIC = -2 \log f_j(x_{\text{obs}}^{(n)} \mid \hat{\theta}_j(x_{\text{obs}}^{(n)})) + 2d_j$$

$$BIC = -2 \log f_j(x_{\text{obs}}^{(n)} \mid \hat{\theta}_j(x_{\text{obs}}^{(n)})) + d_j \log n$$

$$FIAT = \log f_j(x_{\text{obs}}^{(n)} \mid \hat{\theta}_j(x_{\text{obs}}^{(n)})) + \frac{d_j}{2} \log \frac{n}{2\pi} + \log\Big(\int_\Theta \sqrt{\det I_\theta}\, d\theta_j\Big)$$

▷Ref: Grünwald, The minimum description length principle, MIT press, 2007

▷Ref: Ly et al, J. of Mathematical Psychology, 2017

Algorithms

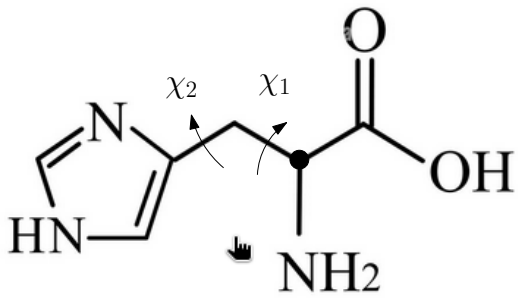
Model selection: warmup

Modeling joint distribution on flat torii

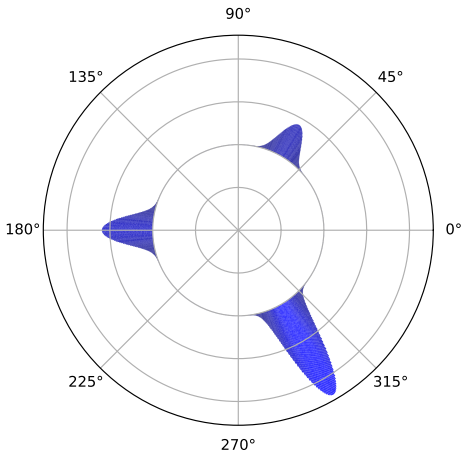
The Minimum Message Length approach to model selection

Side chains and rotamers

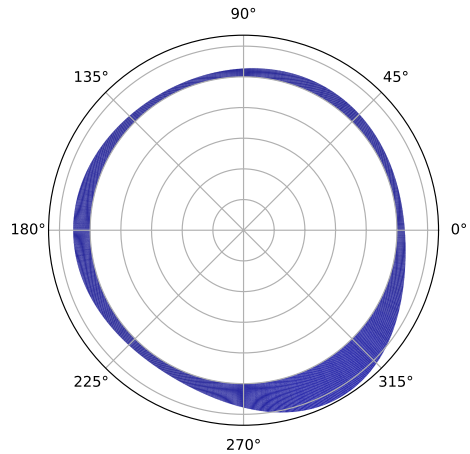
- The χ angles of Histidine:



- Rotameric vs non rotameric χ angles:



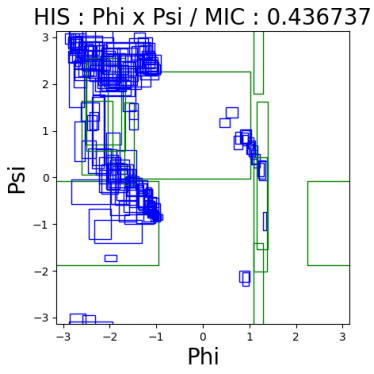
Rotameric / discrete

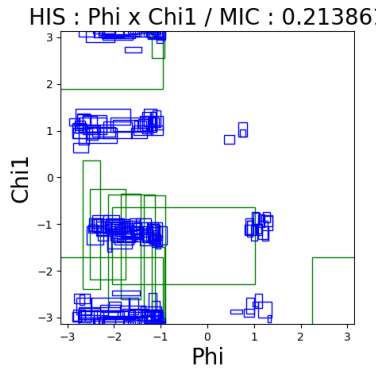


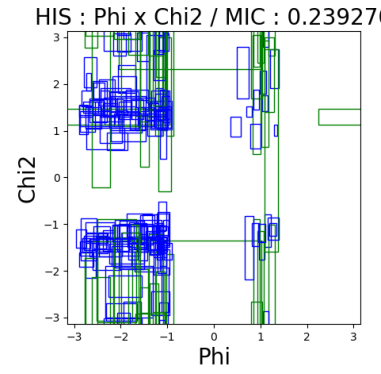
Non rotameric

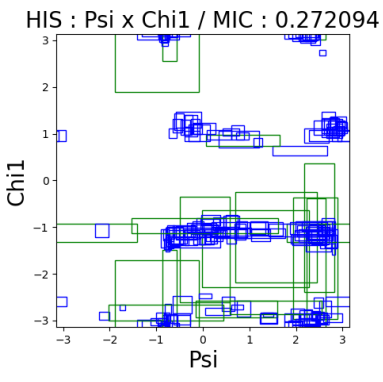
Backbone dependent χ angles

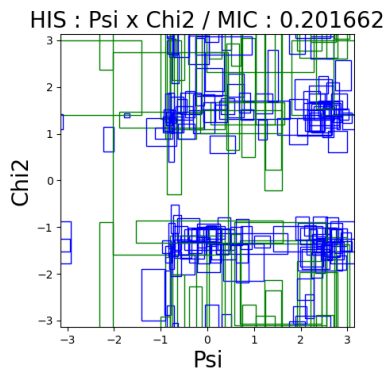
- Example, HIS: $(\phi, \psi, \chi 1, \chi 2)$

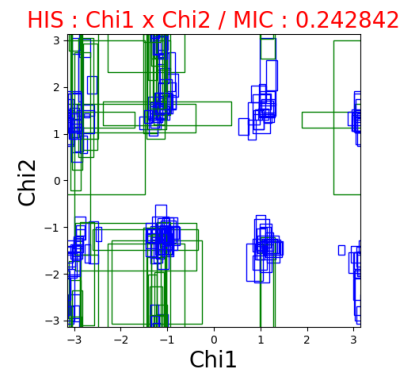












- Limitations of these distributions:
  - ...
  - ...

Phisical – $\phi\psi\chi$al: mixture of von Mises

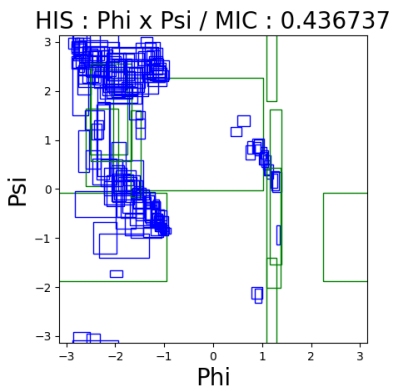
▷ Goal: per amino-acid, model the joint density $(\phi, \psi, \chi_1, \ldots, \chi_n)$
NB: $X = (x_1, \ldots, x_d)$ with $x_i \in [0, 2\pi)$; that is, X is a point on the flat torus $\mathbb{T}^d$.

▷ Mixture model

- Mixture component: product of univariate von Mises

$$f_{\Theta_i}(X) = \prod_{i=1,\ldots,d} \exp^{\kappa_i \cos(x_i - \mu_i)} . \tag{58}$$

- Mixture model: $F(X) = \sum_{i=1,\ldots,M} w_i f_{\Theta_i}(X)$
- Num. parameters: num components: 1; params of the components: $M(1 + 2d)$.



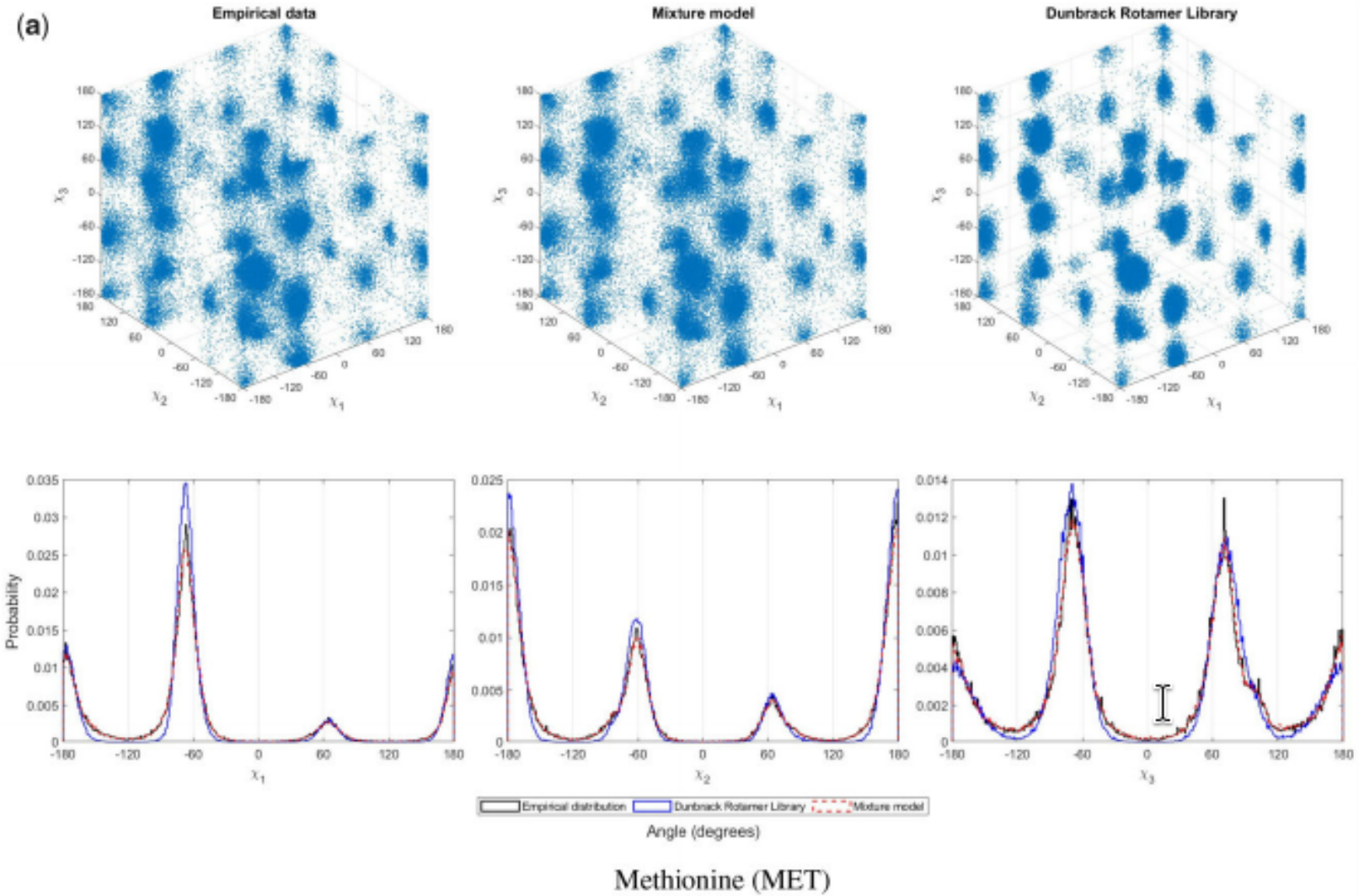
▷ Question: provide two limitations of these distributions

- ...
- ...

▷Ref: Konagurthy et al, Bioinformatics 39, 2023

Example 1: rotameric side chain – MET

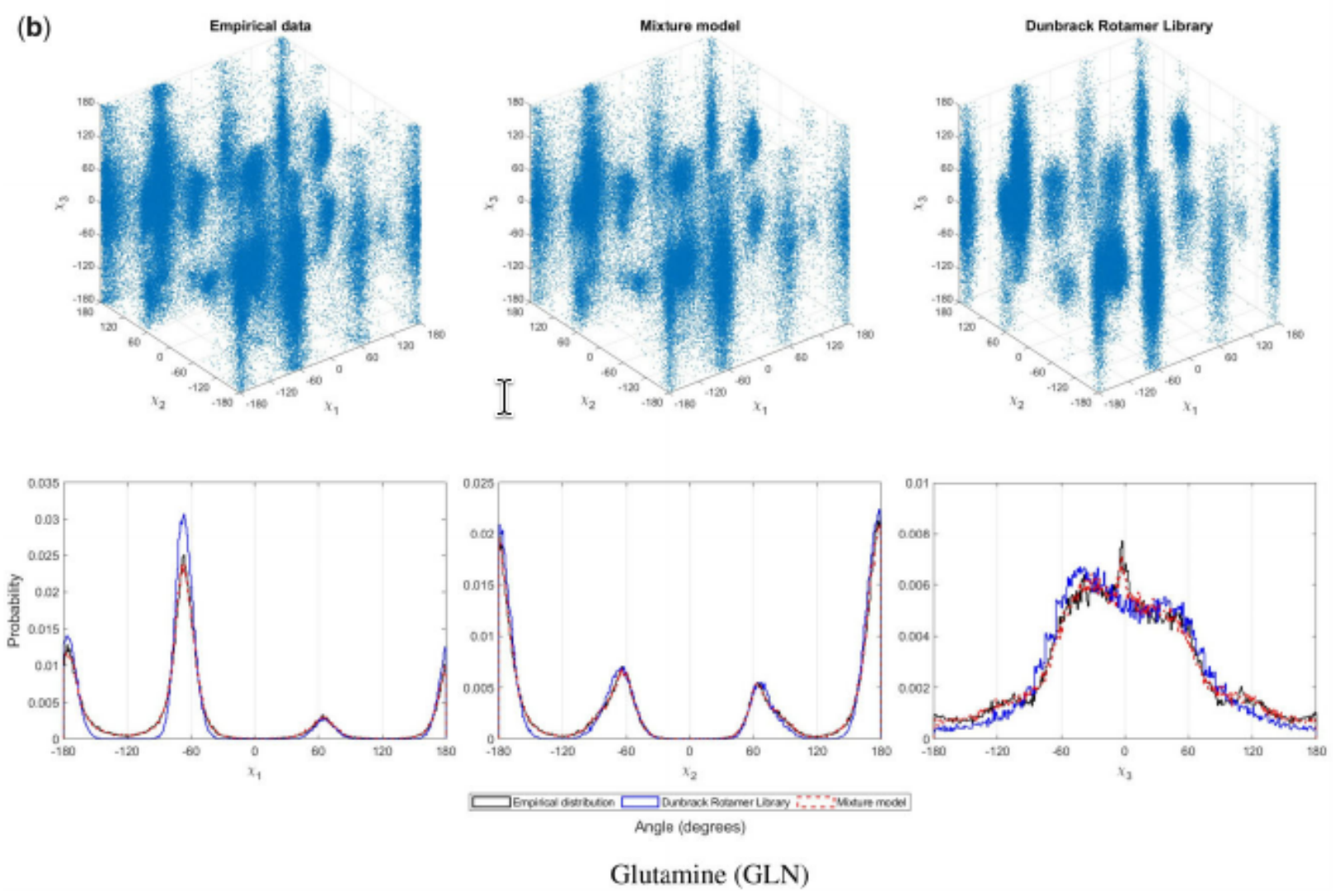
Projection into (χ_1, χ_2, χ_3) and comparison against Dunbrack's lib.



▷Ref: Konagurthy et al, Bioinformatics 39, 2023

Example 2: non rotameric side chain – GLN

Projection into (χ_1, χ_2, χ_3) and comparison against Dunbrack's lib.



▷Ref: Konagurthy et al, Bioinformatics 39, 2023

Phisical: overview

Table 2. Quantitative comparison between the MML-inferred mixture model ($\mathcal{M}^{(aa)}$) and that of the Dunbrack rotamer library ($\mathcal{D}^{(aa)}_{rotamer}$).

		MML mixture model ($\mathcal{M}^{(aa)}$) message length statistics in bits (rounded)					Dunbrack rotamer library ($\mathcal{D}^{(aa)}_{rotamer}$) message length statistics in bits (rounded)					Null model (raw) in bits	
(aa)	$N^{(aa)}$	($\lvert\mathcal{M}^{(aa)}\rvert$; $\lvert A^{(aa)}\rvert$)	First part (complexity)	Second part (fit)	Total (complexity + fit)	$\frac{\text{Total}}{N^{(aa)}}$	($\lvert\mathcal{D}^{(aa)}_{rotamer}\rvert$; #Params)	First part (complexity)	Second part (fit)	Total (complexity + fit)	$\frac{\text{Total}}{N^{(aa)}}$	Null($X^{(aa)}$)	$\frac{\text{Null}(X^{(aa)})}{N^{(aa)}}$
LEU	2,171,630	(165; 1484)	7017	34,540,650	34,547,667	15.9	(11,664; 57,024)	1,079,722	46,109,408	47,189,130	21.7	53,595,177	24.7
ALA	1,861,359	(25; 124)	701	14,847,660	14,848,361	8.0	(N/A; N/A)	N/A	N/A	N/A	N/A	22,968,891	12.3
VAL	1,601,058	(96; 671)	3389	18,795,871	18,799,260	11.7	(3888; 10,368)	217,209	26,750,651	26,967,860	16.8	29,635,223	18.5
GLY	1,588,115	(30; 149)	746	15,965,309	15,966,055	10.1	(N/A; N/A)	N/A	N/A	N/A	N/A	19,597,101	12.3
GLU	1,446,860	(262; 2881)	12,205	33,234,644	33,246,849	23.0	(69,984; 488,592)	9,696,033	39,933,578	49,629,612	34.3	44,635,088	30.8
SER	1,337,273	(114; 797)	3825	18,289,465	18,293,291	13.7	(3888; 10,368)	210,730	23,624,303	23,835,033	17.8	24,752,622	18.5
ILE	1,333,508	(172; 1547)	7356	20,475,170	20,482,526	15.4	(11,664; 57,024)	964,619	27,688,670	28,653,289	21.5	32,910,577	24.7
ASP	1,279,567	(170; 1529)	6524	23,223,302	23,229,826	18.2	(23,328; 115,344)	2,336,817	27,634,793	29,971,610	23.4	31,579,330	24.7
THR	1,221,604	(90; 629)	3057	15,740,512	15,743,569	12.9	(3888; 10,368)	211,687	20,733,566	20,945,253	17.1	22,611,615	18.5
LYS	1,176,395	(266; 3457)	13,691	32,006,948	32,020,639	27.2	(104,976; 943,488)	14,337,386	37,818,245	52,155,632	44.3	43,549,614	37.0
ARG	1,130,448	(250; 3749)	15,898	32,987,603	33,003,501	29.2	(104,976; 943,488)	15,442,702	37,252,663	52,695,365	46.6	48,823,456	43.2
PRO	1,004,859	(231; 2078)	13,779	11,810,146	11,823,926	11.8	(2592; 11,664)	254,495	18,318,268	18,572,763	18.5	24,799,619	24.7
ASN	948,274	(180; 1619)	6793	17,855,829	17,862,622	18.8	(46,656; 231,984)	4,586,850	21,232,141	25,818,991	27.2	23,403,118	24.7
PHE	927,298	(226; 2033)	9365	15,950,596	15,959,961	17.2	(23,328; 115,344)	2,216,337	19,089,817	21,306,154	23.0	22,885,436	24.7
GLN	820,871	(239; 2628)	10,868	18,921,120	18,931,988	23.1	(139,968; 978,480)	18,417,683	23,167,291	41,584,974	50.7	25,323,563	30.8
TYR	788,176	(192; 1727)	7830	13,596,728	13,604,557	17.3	(23,328; 115,344)	2,248,951	16,184,209	18,433,160	23.4	19,451,947	24.7
HIS	515,611	(163; 1466)	6227	9,602,801	9,609,028	18.6	(46,656; 231,984)	4,373,651	11,419,682	15,793,334	30.6	12,725,125	24.7
MET	417,170	(270; 2969)	12,440	9,306,924	9,319,365	22.3	(34,992; 243,648)	4,222,664	11,504,102	15,726,767	37.7	12,869,538	30.8
TRP	310,470	(212; 1907)	8591	5,397,385	5,405,976	17.4	(46,656; 231,984)	4,062,897	6,659,922	10,722,819	34.5	7,662,306	24.7
CYS	296,547	(96; 671)	3148	3,943,308	3,946,457	13.3	(3,888; 10,368)	190,183	5,025,548	5,215,731	17.6	5,489,018	18.5

For each of the 20 naturally occurring amino acids (aa), $N^{(aa)}$ gives the size of the input set ($X^{(aa)}$) on which the comparison is based. $\lvert\mathcal{M}^{(aa)}\rvert$ gives the number of components of the mixture model, and $\lvert A^{(aa)}\rvert$ gives the number of parameters across all components of the mixture model, inferred unsupervised. $\lvert\mathcal{D}^{(aa)}_{rotamer}\rvert$ is the cumulative sum of all components described by the Dunbrack rotamer library, whereas #Params gives the corresponding total number of parameters implicit in their library. Across both models, the complexity (first part length in bits), fidelity (second part length in bits), and their two-part total are shown. The number of bits-per-residue for each of the models is also shown (the respective total message length by $N^{(aa)}$). Finally, to measure the extent of lossless compression each model provides, the null model message length of stating the vector of dihedral angles encoded under a uniform distribution is shown as a bottom-line. Note the 'N/A' terms across alanine (ALA) and glycine (GLY) arise because those amino acids do not have sidechain dihedral angles. While we model the joint distributions of dihedral including the backbone, Dunbrack on the other hand only provide sidechain distributions conditional on the backbone. Hence for ALA and GLY, Dunbrack library estimates are necessarily empty.

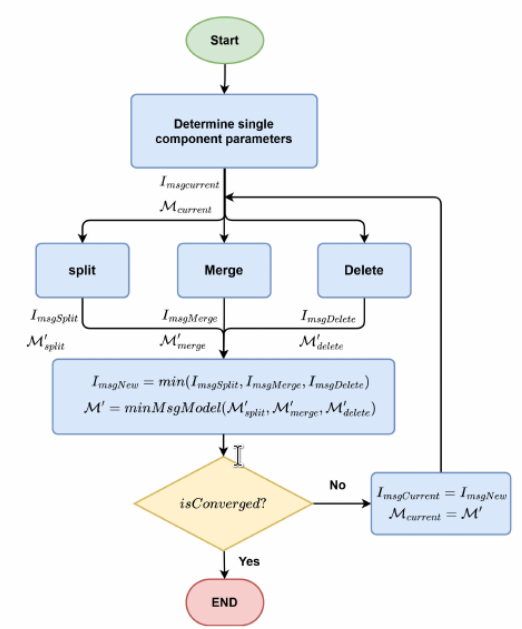
▷Ref: Konagurthy et al, Bioinformatics 39, 2023

$\phi\psi\chi$al: method and limitations

- Methods for fitting mixtures:
  - Dirichlet processes
  - EM + regularization (AIC, BIC)
  - Minimum Message Length / Minimum Description Length

- From coding theory: turn the proba. $\mathbb{P}[\theta, x] = \mathbb{P}[\theta]\,\mathbb{P}[x|\theta]$ into a message length

$$ml(\theta, x) = \underbrace{-\log_2 \mathbb{P}[\theta]}_{\textit{Model}} \underbrace{-\log_2 \mathbb{P}[x|\theta]}_{\textit{Data / likelihood}}. \tag{59}$$



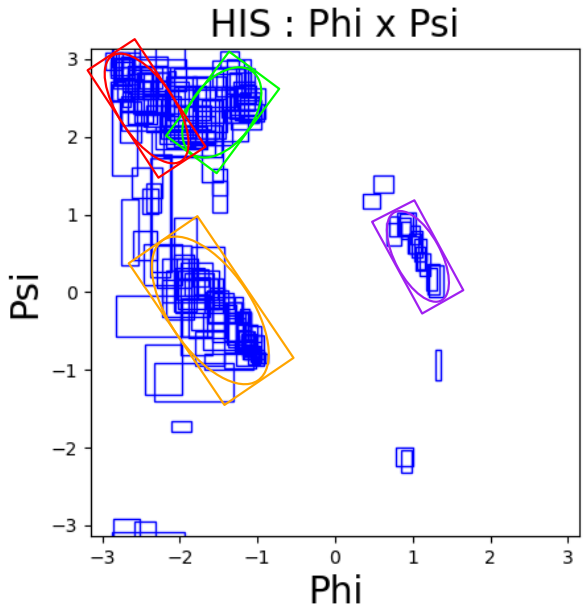
▷Ref: A. Dempster, N. Laird, D. Rubin, ML from incomplete data via the EM algorithm, J. Royal Stat. Society, 1977

▷Ref: P. Grünwald, The minimum description length principle, MIT, 2007

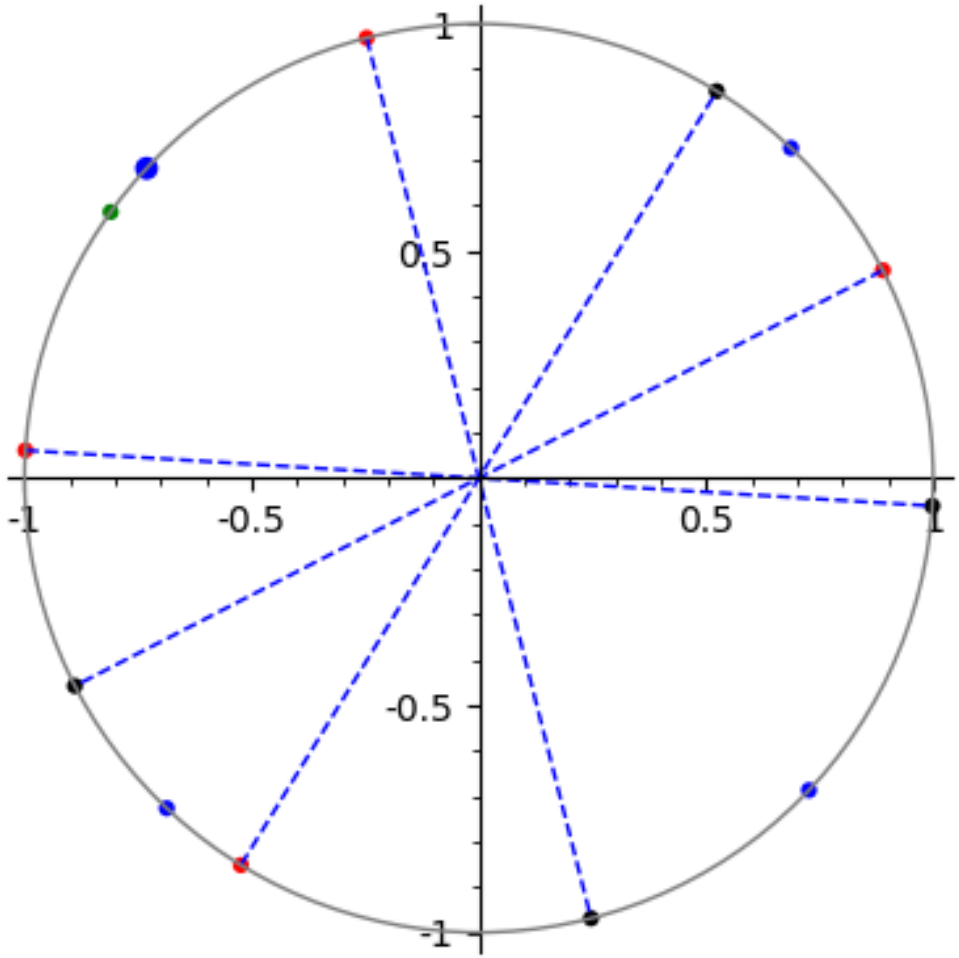
▷Ref: Konagurthy et al, Bioinformatics 39, 2023

Designing complex mixtures: fundamental questions

- Classical mixture components in $\mathbb{R}^d$:
  - product of d univariate functions – cf Phisical
  - fully dimensional function – von Mises / Gaussian
- Questions:
  - Clusters: which dimension / which shape ?
  - Mixture components: which functional form ?
  - Cluster versus mixture components: coherence ?



Fréchet mean and p-mean on the unit circle



- Authors:
  - Frédéric Cazals
  - Timothée O'Donnell

Data centering on S^1: p-mean and Fréchet mean

- Input
  - n (rational) angles $\Theta_0 = \{\theta_i\}_{i=1,\ldots,n}$
  - Associated non-negative weights: $\{w_i\}_{i=1,\ldots,n}$
- Circular distance

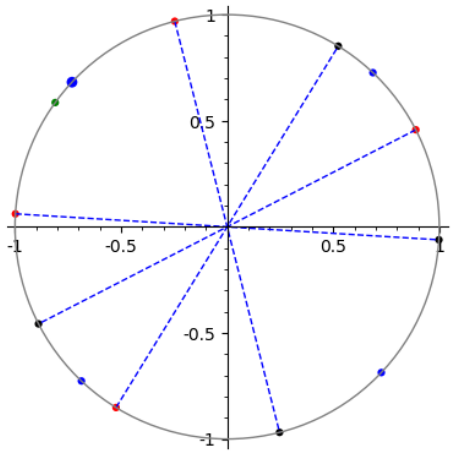
$$d(\theta, \theta_i) = \min(|\ \theta - \theta_i\ |, 2\pi - |\ \theta - \theta_i\ |) \tag{60}$$

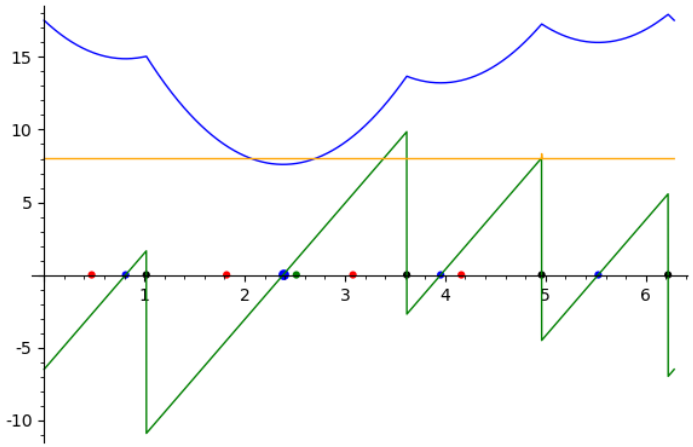
- p-mean functional, for an integer $p \geq 1$

$$F_P(\theta) = \sum_{i=1,\ldots,n} w_i f_i(\theta), \text{ with } f_i(\theta) = d^p(\theta, \theta_i). \tag{61}$$

- p-mean – Fréchet mean for $p = 2$

$$\theta^* = \arg \min_{\theta \in [0, 2\pi)} F_P(\theta). \tag{62}$$





Algorithms

Model selection: warmup

Modeling joint distribution on flat torii

The Minimum Message Length approach to model selection

Estimation and inference by compact coding: Quadratic Mimimum Message Length

- C. Wallace and P. Freeman, Estimation and inference by compact coding, J. of the Royal Statistical Society Series B, 1987
- C. Wallace, Statistical and inductive inference by minimum message length, Springer, 2005.
- P. Grünwald, The mnimum Description length principle, MIT press, 2007.

"In order to understand the world, we must first understand how information is transmitted and received." Claude Shannon

ApresT 15/04/2024, Frederic.Cazals@inria.fr

Model selection: coding with the Minimum Message Length

The 1d case

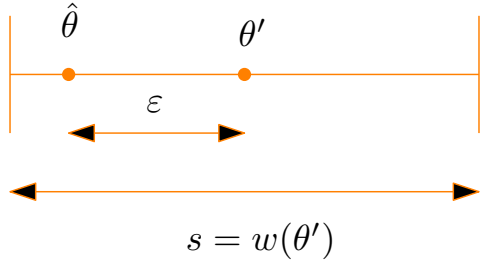
Consider the joint probability:

$$\mathbb{P}[\theta, x] = \mathbb{P}[\theta]\,\mathbb{P}[x|\theta] \tag{63}$$

In coding theory, that yields a message length

$$ml(\theta, x) = \underbrace{-\log_2 \mathbb{P}[\theta]}_{\textit{Model}} \underbrace{-\log_2 \mathbb{P}[x|\theta]}_{\textit{Data / likelihood}}. \tag{64}$$

- (Strict) Minimum Message Length: a two-step selection process in tandem:
  - Constraint: quantized parameter set $\Theta = \{\hat{\theta}_j, j = 1, \dots\}$.
    $w(\hat{\theta}_j)$: the **width** of the interval associated to $\hat{\theta}_j$
  - Alice, statistician, chooses: the generic model, $w(\cdot)$, the model $\theta' \in \Theta$



  - Bob, coding specialist: choses $\hat{\theta}$ nearest to θ', encodes, and sends the msg

MML strategy: cont'd

▷ Hypothesis 1: given a prior $h(\theta)$, total probability for the interval $[\hat{\theta} - w(\hat{\theta})/2, \hat{\theta} + w(\hat{\theta})/2]$

$$\mathbb{P}\left[\hat{\theta}_j\right] \sim w(\hat{\theta}_j) h(\hat{\theta}_j) \tag{65}$$

NB: $\sum_j \mathbb{P}\left[\hat{\theta}_j\right]$ may not add up to one ... but worse approximations ahead !

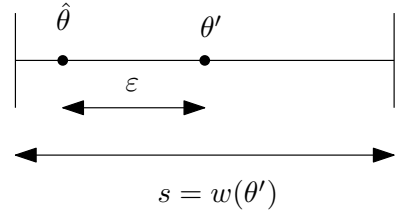
▷ Msg length for model + likelihood:

$$I_1(x) = \underbrace{-\log(w(\hat{\theta}_j) h(\hat{\theta}_j))}_{\textit{Model}} \underbrace{-\log f(x; \hat{\theta}_j)}_{\textit{Likelihood}} . \tag{66}$$

▷ Challenge to minimize $I_1(x)$: Alice must design the spacings $w(\cdot)$ – without the knowledge of x, and choose θ' that will get converted to $\hat{\theta}$ by Bob to send the msg

Quadratic MML: Alice chooses the spacing $w(\cdot)$ (I)

▷ Hypothesis 2 on the spacings $w(\cdot)$:
$\varepsilon = \hat{\theta} - \theta' \leq \pm w(\theta')/2$



Hypothesis on the moments of $\hat{\theta}$ in this interval

$$\mathbb{E}\left[\hat{\theta} - \theta'\right] = 0; \mathbb{E}\left[(\hat{\theta} - \theta')^2\right] = \int_{-w(\theta')/2}^{w(\theta')/2} \frac{1}{w(\theta')} x^2 dx = w(\theta')^2/12. \quad (67)$$

▷ Msg length I_1: Taylor expansion of $\log f(x;\hat{\theta})$ at $\hat{\theta} = \theta'$:

$$I_1(x) = -\log(w(\hat{\theta})h(\hat{\theta})) - \log f(x;\hat{\theta}) \quad (68)$$

$$\approx -\log(w(\theta')h(\theta')) \quad (69)$$

$$-\log f(x;\theta') - (\hat{\theta} - \theta')\frac{\partial}{\partial\theta'}\log f(x;\theta') - \frac{1}{2}(\hat{\theta} - \theta')^2\frac{\partial^2}{\partial\theta'^2}\log f(x;\theta') \quad (70)$$

Quadratic MML: Alice chooses the spacing $w(\cdot)$ (II)

Taking expections with respect to the quantization – shorthand $s = w(\theta')$:

$$\mathbb{E}_c\left[\hat{\theta} - \theta'\right] = 0, \mathbb{E}_c\left[(\hat{\theta} - \theta')^2\right] = s^2/12$$

yields

$$\mathbb{E}_c\left[I_1(x)\right] \approx -\log(sh(\theta')) - \log f(x;\theta') - \frac{1}{24}s^2\frac{\partial^2}{\partial\theta'^2}\log f(x;\theta') \tag{71}$$

Which is minimized by setting

$$s^2 = w(\theta')^2 = -12/[\frac{\partial^2}{\partial\theta'^2}\log f(x;\theta')] \tag{72}$$

▷ Corresponding msg length:

$$I_1(x) = \underbrace{-\log h(\theta') + \frac{1}{2}\log\frac{I_{x,\theta'}}{12}}_{Model}\underbrace{-\log f(x;\theta') + \frac{1}{2}}_{Data}. \tag{73}$$

Definition 20. The MML estimate is the value of θ' minimizing Eq. (73)

▷ Problem: $w(\cdot)$ is choosen in advanced by Alice – without the knowledge of $x \Rightarrow$ previous def. is useless in practice

Fisher information

▷ Fisher information: expectation of the 2nd derivative

$$F(\theta', x) = \frac{\partial^2}{\partial \theta'^2} \log f(x; \theta'), \tag{74}$$

$$I_{\theta'} = -\mathbb{E}_{X \sim \theta'} \left[F(\theta', x) \right]. \tag{75}$$

$\Rightarrow$ acts a condition number: sensitivity of the model f when the parameter θ changes.

▷ Final msg length: using $\hat{\theta}$ chosen via spacing function with $w(\theta) = \sqrt{12/I_\theta}$

$$I_1(x) \approx [-\log \frac{h(\theta')}{\sqrt{I_{\theta'}/12}}] + [-\log f(x; \theta')] + [\frac{1}{2} \frac{F(\theta', x)}{I_{\theta'}}]. \tag{76}$$

- volume/proba of the region $\ni \theta'$
- data encoding i.e. likelihood
- penalty due to the replacement of θ' by $\hat{\theta}$

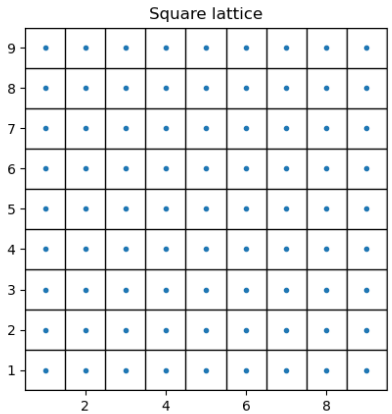
NB: last term can be taken constant if $F(\theta', x)/I_{\theta'}$ is upper bounded.

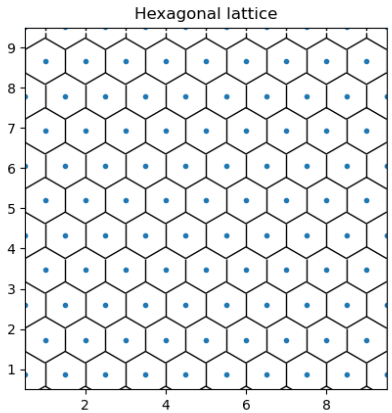
Possible caveats: assumptions on the quadratic MML

- $\forall x \in X$, the function $f(x;\theta)$ has approx. quadratic dependence on θ near its maximum – cf Taylor expansion.
- The space Θ has a locally Euclidean metric for the nearest rounding process to make sense.
- The Fisher information I_θ is defined everywhere
- The prior and I_θ vary little over distances in Θ of the order $1/\sqrt{I_\theta}$

Coding and Gersho's conjecture

▷ Coding a point using a lattice: replace x by the center of mass of the lattice Voronoi cell





The following quantities are of interest–with P a congruent Voronoi polytope:

- Volume of P: $\texttt{Volume}(P) = \int_P dx$
- The second moment with respect to the centroid of P: $U(P) = \int_P \|x - \hat{x}\|^2 \, dx$, with $\hat{x}$ the centroid of P.
- The normalized / expected second moment: $I(P) = U(p)/\texttt{Volume}(P)$.

Define:

$$q(P) = \frac{1}{d} \frac{I(P)}{\texttt{Volume}(P)^{2/d}} = \frac{1}{d} \frac{\int_P \|x - \hat{x}\|^2 \, dx}{\texttt{Volume}(P)^{1+2/d}} \quad (77)$$

▷ Conjecture: ≪For the distorsion minimizing encoder, the regions are congruent to the polytope P of a lattice:≫

$$q_d = \min_P q(P). \quad (78)$$

▷Ref: Gersho, IEEE Trans. Info. Theory, 1979

▷Ref: Conway - Sloane, IEEE Trans. Info. Theory, 1982

Quadratic MML: the general case

▷ Param: vector $\theta = (\theta_1, \ldots, \theta_d)^{\mathsf{T}}$

▷ Model term, $\mathbb{P}[\theta]$: obtained by multiplying the volume of the uncertainty region in which θ is centered, with the probability (assumed to be constant) in that region.

$$\mathbb{P}[\theta] = V * h(\theta), \text{ with } V = \frac{q_d^{-d/2}}{\sqrt{I_\theta}} \tag{79}$$

The MML becomes:

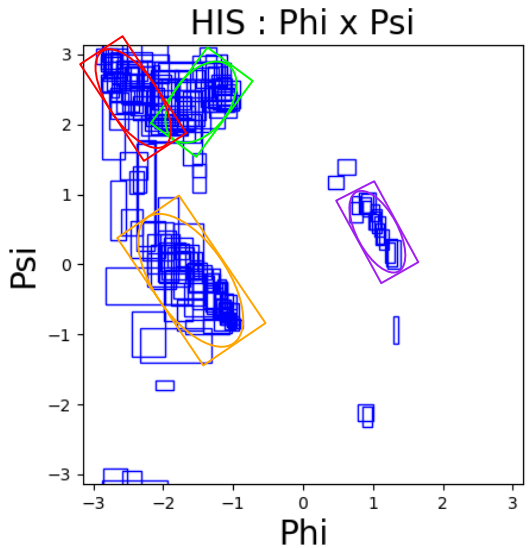
$$I_1(\theta, x) = \underbrace{-\log\left(\frac{h(\theta)}{\sqrt{I_\theta}} q_d^{-d/2}\right)}_{Model} \underbrace{-\log f(X;\theta) + \frac{d}{2}}_{Data} \tag{80}$$

▷ Data term: the negative log likelihood, penalized by the complex cost.

▷Ref: Wallace and Freeman, Estimation and inference by compact coding, J. of the Royal Statistical Society Series B, 1987

Application: designing mixtures in flat torii

- (ϕ, ψ) torsion angles in proteins:



Modeling the density of $\phi\psi\chi$al: couplings illustrated on (ϕ, ψ) for HIS.

- Ref: $\phi\psi\chi$al, Bioinformatics, 2023

- Contenders under scrutiny:
  - ~~(Dirichlet) processes~~
  - Improved versions of EM
  - Mixtures using multivariate Gaussians / von Mises distributions