k-means to k-ellipsoids Algorithm on Torus

Seungki Hong

Seoul National University

1 Introduction

Proteins are determined by the ordered sequences of amino acids. Even though the order of sequence is the same, the functional properties of the same proteins may be different, because of secondary, tertiary, and quaternary structures. These structures are determined by several dihedral angles. Specifically, the secondary structure of a protein determined by 2 diherdral angles, conventionally indicated with ϕ and ψ, which are on $[-\pi, \pi)$, or with rotation, $[0, 2\pi)$ respectively. These two angles are easily plotted with Ramachandran plot. Acutally, there also exist another diherdral angle ω, but it is ignorable, because the angle is naturally determined with π if the chain is the peptide bond. On the other hand, the side chain diherdral angles are conventionally described with χ_n for $n = 1, \cdots$. With these at least 2 angles, the functional property of a protein is determined. Thus, if we can cluster these angluar data, then the algorithm may provide biological insights to the researchers of protein structure.

These p-dimensional angles are on $\mathbb{T}^p$ which is the p-dimensional torus, and this angular space is not an Euclidean space. Actually, $\mathbb{T}^p$ may be embedded as an intrinsically p-dimensional manifold in $\mathbb{R}^{p+1}$ and can be cut-and-flattened as a cube $\mathbb{T}^p = [0, 2\pi)^p$ on $\mathbb{R}^p$. However, as already mentioned, since this space is not an Euclidean space, ordinary clustering methods may not work well.

Our purpose is to introduce relatively fast and easy-applicable clustering method on the torus, based on the conformal prediction approach. We will combine the k-means algorithm, which is sufficiently fast and simple to implement, and von Mises sine mixture model to construct the conformal prediction set as the union of spheres, or more generally, ellipsoids. In section 2, we will introduce some basic tools for toroidal space, extrinsic k-means algorithm for implementing k-means clustering on torus, and the description for conformal prediction framework. In section 3, we will construct the conformal prediction sets with k-means algorithm and prove the optimality of the combined algorithm. In section 4, we will observe some empirical problems which may be occured by high-dimension low-sample situation and suggest additional conditions to avoid such problems. In section 5, we will quote the method of selection of hyperparameters, which is already introduced by S. Jung, et al(2020). In section 6, we will observe the performance of the introduced method on $\mathbb{T}^2$ with artificially generated data sets.

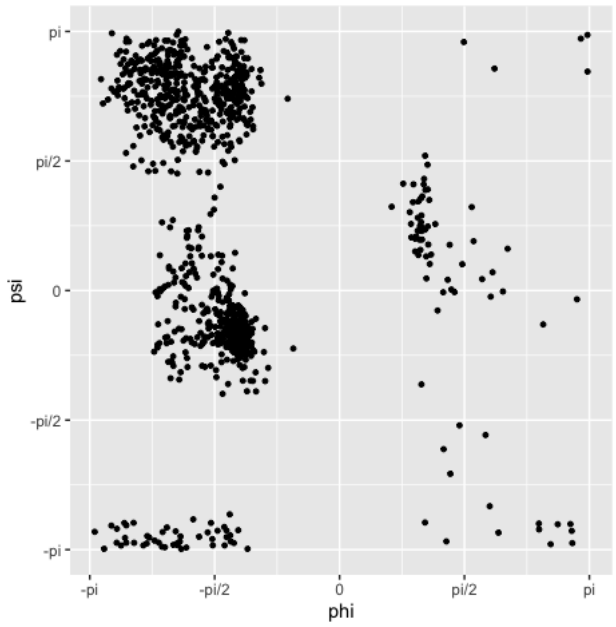

Figure 1: Ramachandran plot for SARS-CoV-2, which is well-known with COVID-19 virus.

2 Backgrounds

2.1 Definitions

Let $X, Y \in \mathbb{T}^p$ where $\mathbb{T}^p$ is the p-dimensional toroidal space. That is,

$$X = (\phi_{x1}, \cdots, \phi_{xp})^T, \ \phi_{xj} \in [0, 2\pi), \forall j = 1, \cdots, p$$

Then, as suggested in S. Jung, et al.(2020), define the angular subtraction $\ominus$ and the toroidal distance ρ as

$$X \ominus Y := \left(\arg(e^{i(\phi_{x1}-\phi_{y1})}), \cdots, \arg(e^{i(\phi_{xp}-\phi_{yp})})\right)^T \tag{1}$$

$$\begin{aligned}\rho(X, Y) := ||X \ominus Y||_2 &= \left[\sum_{j=1}^{p} \arg(e^{i(\phi_{xj}-\phi_{yj})})^2\right]^{\frac{1}{2}} \\ &= \sqrt{(X \ominus Y)^T (X \ominus Y)}\end{aligned} \tag{2}$$

Note that the toroidal space is not an Euclidean space. This implies that the "mean" point of various points in $\mathbb{T}^p$ is not the arithmetic mean of the given points, in general. Thus, we need to define the (sample) toroidal mean $\bar{X}$ and

the sum of squares S_{xx}.

$$\bar{X} := \arg\min_x \sum_{i=1}^{n} \rho^2(x, X_i) \tag{3}$$

$$S_{xx} := \sum_{i=1}^{n} \rho^2(X_i, \bar{X}) \tag{4}$$

By the property of toroidal space, $\bar{X}$ may not be unique. On the other hand, these definitions lead a crucial drawback to ordinary k-means clustering, because ordinary k-means algorithm works only on Euclidean space. Thus, to implement k-means algorithm on toroidal space, we need to transform the points to be on $\mathbb{R}^k$ for some k or redefine the distance.

2.2 Extrinsic k-means Algorithm on Torus

Here, we try to transform the points to be on $\mathbb{R}^{2p}$. Consider the mapping $f : \mathbb{T}^p \to \mathbb{R}^{2p}$ as

$$f(\phi_1, \cdots, \phi_p) = (\cos\phi_1, \cdots, \cos\phi_p, \sin\phi_1, \cdots, \sin\phi_p)$$

which is the simple Euclidean embedding and is injective. If $X_1, \cdots, X_n \in \mathbb{T}^p$, then the transformed points $\{f(X_i)\}_{i=1}^n \subset \mathbb{R}^{2p}$ are on Euclidean space and we can implement ordinary k-means algorithm. Then, if $c_1, \cdots, c_k \in \mathbb{R}^{2p}$ are the centroids of the result of k-means algorithm, transform these centroids to $f^{-1}(\frac{c_1}{||c_1||_2}), \cdots, f^{-1}(\frac{c_k}{||c_k||_2}) \in \mathbb{T}^p$ which have the role of centroids in toroidal space, and allocate the cluster memberships of $\{X_1, \cdots, X_n\}$ to be same with the transformed points. Precise algorithm is described below.

Algorithm 1 Extrinsic k-means algorithm

1: **procedure** EXTRINSIC K-MEANS($\{X_1, \cdots, X_n\}, k$)
2: $\{Y_1, \cdots, Y_n\} = \{f(X_1), \cdots, f(X_n)\}$ respectively
3: Implement ordinary k-means with $\{Y_1, \cdots, Y_n\}$ and k
4: $\{C_1, \cdots, C_k\} = \{f^{-1}(\frac{c_1}{||c_1||_2}), \cdots, f^{-1}(\frac{c_k}{||c_k||_2})\}$, where c_j's are the centroids from 3
5: Allocate the membership of Y_i to X_i for $i = 1, \cdots, n$
6: **end procedure**

2.3 Conformal Prediction

2.3.1 Conformal Prediction Set

Let $X_1, \cdots, X_n \sim P$, for an unknown distribution P and let $\mathbb{X}_n = \{X_1, \cdots, X_n\} \subset \Omega$. Suppose $x \in \Omega$ and $\mathbb{X}_{n+1}(x) = \mathbb{X}_n \cup \{x\}$. Now, consider the null hypothesis $H_0 : X_{n+1} = x$, where $X_{n+1} \sim P$. For given $\alpha \in [0, 1]$, our goal is to find a prediction set C_n^α which is the region that does not reject the null hypothesis with level α. To construct the prediction set, we will define the *conformity score*, which measures the similarity of a point to the given set. That is, the conformity score σ_i is

$$\sigma_i := g(X_i, \mathbb{X}_{n+1}), \ \forall i = 1, \cdots, n+1 \tag{5}$$

for some function g. Moreover, if $X_{(1)}, \cdots, X_{(n+1)}$ are ordered with $\sigma_{(1)} \leq \cdots \leq \sigma_{(n+1)}$ for $\sigma_{(i)} = g(X_{(i)}, \mathbb{X}_{n+1})$, then we may say that $X_{(n+1)}$ is the most similar point to $\mathbb{X}_{n+1}$.

Now, under H_0, consider the following distribution:

$$\pi(x) = \frac{1}{n+1} \sum_{i=1}^{n+1} I(\sigma_i(x) \leq \sigma_{n+1}(x)) \tag{6}$$

Then, since $\mathbb{X}_{n+1}(x)$ is exchangeable, the probability that $X_{n+1} = x$ ranks i-th smallest conformity score is uniformly distributed. This implies that $\pi(x)$ is uniformly distributed on $\{\frac{1}{n+1}, \cdots, 1\}$. Thus, we can construct the *conformal prediction set* as

$$C_n^\alpha = \{x : \pi(x) > \alpha\} \tag{7}$$

and thus

$$\mathbb{P}(X_{n+1} \in C_n^\alpha) = \mathbb{P}(\pi(X_{n+1}) > \alpha) \geq 1 - \alpha, \quad \forall n \tag{8}$$

On the other hand, by the definition of $\pi(x)$, the conformal prediction set can also be described as below.

$$\{x : \pi(x) > \alpha\} = \{x : \sigma(x) \geq \sigma_{(i_{n,\alpha})}\}, \ i_{n,\alpha} = \frac{\lfloor (n+1)\alpha \rfloor}{n+1} \tag{9}$$

Note that the probability distribution of X_i's is unknown. This implies that the conformal prediction set is available for any probabilistic structures and it only depends on the definition of conformity score. If conformity score is poorly chosen, then the conformal prediction set will become a trivial set which may envelop the data excessively. We need to choose the conformity score carefully to construct the prediction set which compactly covers the adequate data. That is, for a Borel measure μ of Ω, the ideal conformal prediction set C_n^α may satisfies

$$\mu(C_n^\alpha) = \min_{C \in \mathcal{C}^\alpha} \mu(C)$$

where $\mathcal{C}^\alpha$ is the collection of conformal prediction sets with level α.

2.3.2 Inductive Conformal Prediction

On the other hand, if there are sufficiently many data, then evaluating conformity score spends quite long time. By introducing *inductive conformal prediction set*, we can construct the conformal prediction set faster than original one by spliting the data into two sets. The validity of this method is proved by J. Lei, et al.(2012), J. Lei, et al(2013), thus precise description will be omiited in this section. The precise algorithm is described below:

3 k-means to k-ellipsoids

3.1 Homogeneous-spheres

Now, we will construct conformal prediction set with k-means algorithm. Let $X_1, \cdots, X_n \in \mathbb{T}^p$ and suppose that the number J of cluster is predetermined.

Algorithm 2 Inductive Conformal Prediction

procedure INDUCTIVE CONFORMAL PREDICTION($\{X_1, \cdots, X_n\}, \alpha, n_1 < n$)
  Split the data randomly as $\mathbb{X}_1 = \{X_1, \cdots, X_{n_1}\}$,$\mathbb{X}_2 = \{X_{n_1+1}, \cdots, X_n\}$.
  Construct σ with $\sigma(x) = g(x, \mathbb{X}_1)$ for some function g.
  Put $\sigma_i = g(X_{n_1+i}, \mathbb{X}_1)$ and order as $\sigma_{(1)} \leq \cdots \leq \sigma_{(n_2)}$.
  Construct $C_n^\alpha = \{x : \sigma(x) \geq \sigma_{(i_{n_2,\alpha})}\}$
end procedure

For using inductive conformal prediction framework, split the data as $\mathbb{X}_1 = \{X_1, \cdots, X_{n_1}\}$, and $\mathbb{X}_2 = \{X_{n_1+1}, \cdots, X_n\}$. Let $C_1, \cdots, C_J$ be the centroids which are evaluated by algorithm 1 with $\mathbb{X}_1$. Define the conformity score $\sigma(\cdot)$,

$$\sigma(x) := -\min_{1 \leq j \leq J} \rho^2(x, C_j) \tag{10}$$

Then, the inductive conformal prediction set C for level $\alpha \in [0, 1]$ is

$$C = \{x \in \mathbb{T}^p | \sigma(x) \geq \sigma_{(i_{n_2,\alpha})}\},\ i_{n_2,\alpha} = \frac{\lfloor (n_2+1)\alpha \rfloor}{n_2+1},\ n_2 = n - n_1$$

where $\sigma_i = \sigma(X_{n_1+i})$ and $\sigma_{(1)} \leq \cdots \leq \sigma_{(n_2)}$ is the ordered σ_i's. For simplicity, put $t_\alpha = \sigma_{(i_{n_2,\alpha})}$. Then,

$$\begin{aligned} C &= \{x \in \mathbb{T}^p | -\min_{1 \leq j \leq J} \rho^2(x, C_j) \geq t_\alpha\} \\ &= \{x \in \mathbb{T}^p | \min_{1 \leq j \leq J} \rho^2(x, C_j) \leq -t_\alpha\} \\ &= \bigcup_{j=1}^{J} \{x \in \mathbb{T}^p | \rho^2(x, C_j) \leq -t_\alpha\} \\ &= \bigcup_{j=1}^{J} B(c_j, \sqrt{-t_\alpha}) \end{aligned} \tag{11}$$

where $B(x, r)$ is a closed sphere centered at x with radius r. Thus, with centroids derived from k-means clustering, we can construct conformal prediction set which is the union of spheres. Note that there may be less than J clusters; if the spheres intersect, then they automatically merge and we will deal the merged set as a cluster. The precise algorithm is described below.

Algorithm 3 k-homogeneous spheres algorithm

procedure K-HOMOGENEOUS SPHERES($\{X_1, \cdots, X_n\}, J, \alpha$)
  Split the data into $\mathbb{X}_1$ and $\mathbb{X}_2$.
  Implement extrinsic k-means on $\mathbb{X}_1$ to get centers $c_1, \cdots, c_J$.
  For $\mathbb{X}_2$, compute the conformity scores $\sigma_i = -\min_j \rho^2(X_{n_1+i}, c_j)$
  Construct $C = \bigcup_{j=1}^{J} B(c_j, \sqrt{-\sigma_{(i_{n_2,\alpha})}})$
end procedure

3.2 Heterogeneous-spheres

In this case, we will modify the conformity score to make the form of conformal prediction set more flexible. Before modifying, let V_j be the j-th cluster of

extrinsic k-means algorithm, implemented with $\mathbb{X}_1$. Put $n_{1j} = \#(\mathbb{X}_1 \cap V_j)$ and $\bar{X}_j$ be the sample toroidal mean of $C_j := \mathbb{X}_1 \cap V_j$. Then, define the conformity score $\sigma(\cdot)$ as

$$\sigma(x) := -\min_j \left[\frac{\rho^2(x, c_j)}{\hat{\sigma}_j^2} + 2p \log \hat{\sigma}_j - 2 \log \hat{\pi}_j \right], \tag{12}$$

where $\hat{\pi}_j = \frac{n_{1j}}{n_1}, \hat{\sigma}_j^2 = \frac{1}{n_j p} \sum_{x \in C_j} \rho^2(x, \bar{X}_j)$ for $j = 1, \cdots, J$.

Now, as before, we can construct the inductive conformal prediction set as below:

$$\begin{aligned}
C &= \{x | \sigma(x) \geq \sigma_{(i_{n_2,\alpha})}\}, \quad t_\alpha = \sigma_{(i_{n_2,\alpha})} \\
&= \{x | -\min_j \left[\frac{\rho^2(x, c_j)}{\hat{\sigma}_j^2} + 2p \log \hat{\sigma}_j - 2 \log \hat{\pi}_j \right] \geq t_\alpha\} \\
&= \{x | \min_j \left[\frac{\rho^2(x, c_j)}{\hat{\sigma}_j^2} + 2p \log \hat{\sigma}_j - 2 \log \hat{\pi}_j \right] \leq -t_\alpha\} \\
&= \bigcup_{j=1}^{J} \{x | \frac{\rho^2(x, c_j)}{\hat{\sigma}_j^2} + 2p \log \hat{\sigma}_j - 2 \log \hat{\pi}_j \leq -t_\alpha\} \\
&= \bigcup_{j=1}^{J} \{x | \rho^2(x, c_j) \leq \hat{\sigma}_j^2 (e_j - t_\alpha)\}, \quad e_j = 2 \log \hat{\pi}_j - 2p \log \hat{\sigma}_j \\
&= \bigcup_{j=1}^{J} B(c_j, r_j), \quad r_j = \hat{\sigma}_j \sqrt{(e_j - t_\alpha)_+}
\end{aligned} \tag{13}$$

Note that the radii of spheres are not the same; there may be the case that $r_j = 0$. If the radius is 0, then the sphere vanishes automatically.

In this case, we newly introduce some parameters: π_j's and σ_j's. These parameters are automatically evaluated by the extrinsic k-means algorithm and thus, there is no difference between the algorithms for homogeneous and heterogeneous spheres, fundamentally. The only difference is that we just need to evaluate these new parameters with centroids, simultaneously.

3.3 General Case: k-ellipsoids

3.3.1 Approximating von Mises Sine Density

Before defining new conformity score, consider the p-variate CMS density, which is the approximated density for p-variate von Mises sine density by assuming sufficiently high concentrations and $\Sigma \succ 0$, introduced by K. V. Mardia, et al.(2012):

$$f^*(y; , \mu, \Sigma) = (2\pi)^{-p/2} |\Sigma|^{-1/2} \exp\{-\frac{1}{2}[\kappa^T (2 - 2c(y, \mu)) + s(y, \mu)^T \Lambda s(y, \mu)]\} \tag{14}$$

where $y = (y_1, \cdots, y_p)^T$, $\mu = (\mu_1, \cdots, \mu_p)^T$, $y, \mu \in \mathbb{T}^p$, for $1 \leq j, l \leq p$,

$\kappa_j > 0$, $-\infty < \lambda_{jl} < \infty$,

$$\begin{aligned} c(y,\mu) &= (\cos(y_1-\mu_1),\cdots,\cos(y_p-\mu_p))^T \\ s(y,\mu) &= (\sin(y_1-\mu_1),\cdots,\sin(y_p-\mu_p))^T \\ (\Lambda)_{jl} &= \lambda_{jl} = \lambda_{lj},\ j \neq l, \quad (\Lambda)_{jj} = \lambda_{jj} = 0 \\ (\Sigma^{-1})_{jl} &= \lambda_{jl},\ j \neq l, \quad (\Sigma^{-1})_{jj} = \kappa_j \end{aligned}$$

which is obtained by Taylor approximation of $\theta \approx \sin\theta$, $1 - \frac{\theta^2}{2} \approx \cos\theta$. Thus, for sufficiently large κ_j's, if $y_1, \cdots, y_n$ are the random samples from $f^*(\cdot;\mu,\Sigma)$, the approximate likelihood is

$$L^*(\mu,\Sigma) = (2\pi)^{-np/2}|\Sigma|^{-n/2}\exp\left[-\frac{n}{2}\operatorname{tr} S\Sigma^{-1}\right] \tag{15}$$

where

$$\begin{aligned} (S)_{jj} &= \frac{1}{n}\sum_{i=1}^{n}\{2-2\cos(y_{ij}-\mu_j)\}, \quad j = 1,\cdots,p \\ (S)_{jl} &= \frac{1}{n}\sum_{i=1}^{n}\sin(y_{ij}-\mu_j)\sin(y_{il}-\mu_l), \quad j \neq l \end{aligned}$$

In this case, the closed form of the MLE of μ does not exist. Thus, to evaluate the MLE, we need to use numerical method or approximated MLE. K. V. Mardia, et al.(2012) suggests an approximated MLE $\hat{\mu}^*$ of μ as below;

Let $\bar{U}_j = \sum_{i=1}^{n}\cos(y_{ij})/n$ and $\bar{V}_j = \sum_{i=1}^{n}\sin(y_{ij})/n$ for $j = 1,\cdots,p$. Then, for $\hat{\mu}^* = (\hat{\mu}_1^*,\cdots,\hat{\mu}_p^*)^T$,

$$\hat{\mu}_j^* = \arctan\frac{\bar{V}_j}{\bar{U}_j}, \quad j = 1,\cdots,p \tag{16}$$

On the other hand, we may further approximate the CMS density; since the approximation is derived by Taylor expansion, we may approximate the density by applying Taylor approximation as below:

$$\begin{aligned} f^*(y;,\mu,\Sigma) &= (2\pi)^{-p/2}|\Sigma|^{-1/2}\exp\{-\frac{1}{2}[\kappa^T(2-2c(y,\mu)) + s(y,\mu)^T\Lambda s(y,\mu)]\} \\ &\approx (2\pi)^{-p/2}|\Sigma|^{-1/2}\exp\{-\frac{1}{2}[(y\ominus\mu)^T\Sigma^{-1}(y\ominus\mu)]\} \end{aligned} \tag{17}$$

Then, for the $i.i.d$ case, the further approximate likelihood L' is

$$L'(\mu,\Sigma) = (2\pi)^{-np/2}|\Sigma|^{-n/2}\exp\left[-\frac{n}{2}\operatorname{tr} S'\Sigma^{-1}\right] \tag{18}$$

where $S' = \frac{1}{n}\sum_{i=1}^{n}(y_i\ominus\mu)(y_i\ominus\mu)^T$. Thus, if μ is known, $\hat{\Sigma} = S'$ maximizes L'. On the other hand, by the complexity of toroidal subtraction, evaluating the MLE of μ is still less tractable. To evaluate the MLE $\hat{\mu}'$, we may use numerical method to maximize

$$\log L'(\mu,S') = -\frac{n}{2}\log|S'| + constant \tag{19}$$

Alternatively, we may use the suggested approximation (7), $\hat{\mu}^*$.

3.3.2 Generalized Lloyd's Algorithm

Suppose $X_1, \cdots, X_n \in \mathbb{X}$ be given. Let Θ be the parameter space and let $\theta \in \Theta$. The residual, or distortion measure $R_i(\theta) := R(X_i; \theta)$ measures the dissimilarity or the distance of X_i and usually, the nonnegativity is usually assumed. The well-known Lloyd's algorithm minimizes total residual under the probabilistic structure of the space, which is precisely described by Y. Linde, et al.(1980). On the other hand, Generalized Lloyd's algorithm, which is named by J. Shin, et al.(2019), minimizes the total dissimilarity; that is, the algorithm minimizes the following function,

$$L(\theta) = \sum_{i=1}^{n} R_i(\theta) \tag{20}$$

K-means clustering uses Lloyd's algorithm by defining the residual as Euclidean distance between the given data and the centroids, where the centroids have the same role as θ.

Note that residual is completely the opposite concept of conformity score. Since conformity score measures the similarity, by multiplying -1 to the residual, we can simply get the conformity scores. This implies that generalized Lloyd's algorithm maximizes the total sum of conformity scores, equivalently.

Now, consider the conformity score with mixutre model. For $\{\theta_j\}_{j=1}^{J} \in \Theta$, let $f_j(\cdot) = f(\cdot; \theta_j)$ be a density function with j-th parameter. For $\theta = \{\theta_j\}_{j=1}^{J}$, the mixture density is defined as

$$p(x; \theta) := \sum_{j=1}^{J} \pi_j f_j(x), \quad \pi_j \geq 0, \pi_1 + \cdots + \pi_J = 1 \tag{21}$$

Define the conformity score $\sigma(\cdot)$ as below:

$$\sigma(x) := \max_j \left[\log \pi_j f_j(x)\right] \tag{22}$$

Then, generalized Lloyd's algorithm maximizes $l_M(\theta, \pi) = \sum_{i=1}^{n} \sigma(X_i)$. Precisely, suppose that the estimated parameters $\theta^{(t)}, \pi^{(t)}$ are given, which are evaluated by t-iterations of the algorithm. Define $w_{i,j}^{(t)}$ as

$$w_{i,j}^{(t)} = \begin{cases} 1, & \text{if } j = \arg\max_l \log[\pi_l^{(t)} f(X_i; \theta_l^{(t)})] \\ 0, & \text{otherwise} \end{cases}$$

and let

$$l(\theta, \pi | \theta^{(t)}, \pi^{(t)}) = \sum_{i=1}^{n} \sum_{j=1}^{J} w_{i,j}^{(t)} \log[\pi_j f(X_i; \theta_j)]$$

Note that $l_M(\theta^{(t)}, \pi^{(t)}) = l(\theta^{(t)}, \pi^{(t)} | \theta^{(t)}, \pi^{(t)})$, by the definition of $w_{i,j}^{(t)}$. On the other hand,

$$\begin{aligned} l(\theta, \pi | \theta^{(t)}, \pi^{(t)}) &= \sum_{i=1}^{n} \sum_{j=1}^{J} w_{i,j}^{(t)} \log[\pi_j f(X_i; \theta_j)] \\ &= \sum_{i=1}^{n} \sum_{j=1}^{J} w_{i,j}^{(t)} \log \pi_j + \sum_{i=1}^{n} \sum_{j=1}^{J} w_{i,j}^{(t)} \log f(X_i; \theta_j) \end{aligned}$$

Thus, $l(\theta, \pi|\theta^{(t)}, \pi^{(t)})$ is maximized where

$$\pi_j^{(t+1)} = \frac{1}{n}\sum_{i=1}^{n} w_{i,j}^{(t)}, \quad j = 1, \cdots, J$$

Moreover, assume that $\theta^{(t+1)}$ maximizes $l(\theta, \pi^{(t+1)}|\theta^{(t)}, \pi^{(t)})$. That is,

$$l(\theta^{(t+1)}, \pi^{(t+1)}|\theta^{(t)}, \pi^{(t)}) \geq l(\theta^{(t)}, \pi^{(t)}|\theta^{(t)}, \pi^{(t)})$$

Then, the following inequality satisfies:

$$\begin{aligned} l(\theta^{(t+1)}, \pi^{(t+1)}|\theta^{(t)}, \pi^{(t)}) &= \sum_{i=1}^{n}\sum_{j=1}^{J} w_{i,j}^{(t)} \log[\pi_j^{(t+1)} f(X_i; \theta_j^{(t+1)})] \\ &\leq \sum_{i=1}^{n} \max_j \log[\pi_j^{(t+1)} f(X_i; \theta_j^{(t+1)})] \\ &= l(\theta^{(t+1)}, \pi^{(t+1)}|\theta^{(t+1)}, \pi^{(t+1)}) \end{aligned}$$

Therefore, $l_M(\theta^{(t)}, \pi^{(t)}) \leq l_M(\theta^{(t+1)}, \pi^{(t+1)})$ and this implies that this iteration converges to local maximum of $l_M(\theta, \pi)$.

We need to know how to find $\theta^{(t+1)}$. Let $I_j^{(t)} = \{i \in \{1, \cdots, n\} | w_{i,j}^{(t)} = 1\}$, $j = 1, \cdots, J$. That is, $I_j^{(t)}$ is the index set of X_i's, which are "closest" to the j-th parameters. Then,

$$\begin{aligned} l(\theta, \pi|\theta^{(t)}, \pi^{(t)}) &= \sum_{i=1}^{n}\sum_{j=1}^{J} w_{i,j}^{(t)} \log[\pi_j f(X_i; \theta_j)] \\ &= \sum_{j=1}^{J}\sum_{i=1}^{n} w_{i,j}^{(t)} \log[\pi_j f(X_i; \theta_j)] \\ &= \sum_{j=1}^{J}\sum_{i \in I_j^{(t)}} \log[\pi_j f(X_i; \theta_j)] \end{aligned}$$

Hence, by evaluating MLEs of θ_j with $\{X_i\}_{i \in I_j^{(t)}}$ respectively, we can maximize $l(\theta, \pi|\theta^{(t)}, \pi^{(t)})$. Note that the optimizing procedure is quite similar to the that of the EM algorithm. The difference between EM and generalized Lloyd is that the generalized Lloyd optimizes $w_{i,j}$'s only with 0 or 1. This hard-threshold optimization may makes the convergence faster than EM.

3.3.3 Application to Approximated CMS Density

Now, consider the approximated CMS density (8) to the mixture model (12). Then, the conformity score is

$$\sigma_1(x) = \max_j \left[-\frac{1}{2}(x \ominus \mu_j)^T \Sigma_j^{-1} (x \ominus \mu_j) - \frac{1}{2}\log|\Sigma_j| - \frac{np}{2}\log(2\pi) + \log \pi_j \right]$$

which is equivalent to

$$\sigma(x) = -\min_j \left[(x \ominus \mu_j)^T \Sigma_j^{-1} (x \ominus \mu_j) + \log|\Sigma_j| - 2\log \pi_j \right]$$

Thus, as we saw in 2.3.2 and by the approximated MLE of (8), we can apply generalized Lloyd's algorithm directly. Note that if we use the approximated MLE $\hat{\mu}^*$of μ, then the algorithm may not converge to local maximum; nevertheless, this approximation works quite well. The detail procedures are described in Algorithm 4.

Algorithm 4 Generalized Lloyd's algorithm

1: **procedure** GLA($\{X_1, \cdots, X_n\}, J$)
2: Initialize $\theta_j = (\pi_j, \mu_j, \Sigma_j)$, $j = 1, \cdots, J$
3: set

$$w_{i,j} = \begin{cases} 1, & \text{if } j = \arg\max_l \left[-(x \ominus \mu_j)^T \Sigma^{-1}(x \ominus \mu_j) - \log|\Sigma| + 2\log \pi_j\right] \\ 0, & \text{otherwise} \end{cases}$$

$$I_j = \{i \in \{1, \cdots, n\} | w_{i,j} = 1\}$$

4: Update μ_j as (7) or by maximizing (10), with $\{X_i\}_{i \in I_j}$ for $j = 1, \cdots, J$
5: Update $\Sigma_j = \frac{1}{\sum_{i=1}^n w_{i,j}} \sum_{i=1}^n w_{i,j}(X_i \ominus \mu_j)(X_i \ominus \mu_j)^T$ for $j = 1, \cdots, J$
6: Update $\pi_j = \frac{1}{n} \sum_{i=1}^n w_{i,j}$ for $j = 1, \cdots, J$
7: Repeat step 3-6 until converge
8: **end procedure**

Now, we can construct inductive conformal prediction set as before. If the parameters are optimized by the generalized Lloyd's algorithm with the splited data $\mathbb{X}_1$, then the conformal prediction set is

$$\begin{aligned} C &= \{x | \sigma(x) \geq \sigma_{(i_{n_2,\alpha})}\}, \quad t_\alpha = \sigma_{(i_{n_2,\alpha})} \\ &= \{x | - \min_j \left[(x \ominus \hat{\mu}_j)^T \hat{\Sigma_j}^{-1} (x \ominus \hat{\mu}_j) + \log|\hat{\Sigma}_j| - 2\log \hat{\pi}_j\right] \geq t_\alpha\} \\ &= \bigcup_{j=1}^J \{x | (x \ominus \hat{\mu}_j)^T \hat{\Sigma_j}^{-1} (x \ominus \hat{\mu}_j) \leq e_j - t_\alpha\}, \quad e_j = 2\log \hat{\pi}_j - \log|\hat{\Sigma}_j| \end{aligned} \tag{23}$$

which has the form of unions of ellipsoids. Note that if $e_j \leq t_\alpha$, the ellipsoid automatically vanishes as we already observed in 3.2.

4 Empirical Problems

By choosing sufficiently large J (the number of ellipsoids) or by the approximation of centroids (or μ's in 3.3.3), the algorithms may not work. Specifically, by the definition of conformity score in 3.2, a cluster derived by extrinsic k-means may have only one point or empty. Then, we cannot evaluate $\hat{\sigma}^2$. For flexibility of the algorithms, we need to deal these cases numerically.

4.1 Heterogeneous Case

As already mentioned, the main problem is the evaluation of $\hat{\sigma}^2$ for a cluster which has only one point or empty. To solve this problem, consider the following:

$$\lim_{\hat{\sigma}_j \downarrow 0} \hat{\sigma}_j \sqrt{(e_j - t_\alpha)_+} = 0, \quad e_j = 2\log \hat{\pi}_j - 2p \log \hat{\sigma}_j$$

which is the radius of the j-th sphere. That is, since the cluster has only one point or empty, we may ignore the sphere generated by the cluster by setting $\hat{\sigma}_j^2$ sufficiently small.

4.2 General Case

In this case, the problem is quite similar as before, but more complicated. If $\#I_j \leq p$, where I_j is the index set defined in 3.3.2, then $rank(S') < p$ and thus the inverse matrix of S' does not exist. This nonexistence of inverse matrix leads error in algorithm 4; we cannot evaluate step 3. We may detect this phenomenon by calculating the determinant of S'; if $|S'|$ is less than sufficiently small positive number, say m, then we may regard S' as a singular matrix.

Consider that the volume of ellipsoid converges to 0 if $|S'| \downarrow 0$. Thus, we may circumvent this problem by allocating $S' = mI_p$ for sufficiently small positive number m, if $|S'| < m$. Then, the ellipsoid becomes a tiny sphere, and this may contain no points or fortunately only one point. As before, we may ignore this tiny sphere.

However, it seems that if the dimension p is sufficiently high so that the number of data is not much larger than pJ, where prespecified J is the number of ellipsoids, then the parameter fitting may not work well. This may be the fundamental drawback of optimizing with the generalized Lloyd's algorithm.

Hence, to overcome the curse of dimensionality, we alternatively suggest the following condition under the high-dimension low-sample situation:

If $2 \leq \#I_j \leq p$, then assume that $\Sigma_j = diag(\sigma_{jk}^2)_{k=1,\cdots,p}$ which only has positive diagonal entries. Under this restriction, the approximate MLE $\hat{\Sigma}_j$ of Σ_j is

$$\hat{\Sigma}_j = diag(\hat{\sigma}_{jk}^2), \quad \hat{\sigma}_{jk}^2 = \frac{1}{\#I_j} \sum_{i \in I_j} (X_{ik} \ominus \hat{\mu}_{jk})^2 \tag{24}$$

where $\hat{\mu_j} = (\hat{\mu}_{j1}, \cdots, \hat{\mu}_{jp})^T$ is the estimator of μ_j given by algorithm 4. With this additional assumption, if $\#I_j \geq 2$, the estimation of Σ_j may work well for any dimensionality. Note that if $\#I_j < 2$, then we may discard the cluster by letting $\hat{\sigma}_{jk}^2 \downarrow 0$, which is exactly the same logic as the heterogeneous case.

On the other hand, even if the additional assumption is quite restrictive, there still exists the singular matrix problem: if the data in the cluster are aligned on a axis, then one of $\hat{\sigma}_{jk}$'s must be vanished. To prevent this rare situation, we may assume the stronger condition:

If (24) is singular, that is, if $\hat{\sigma}_{jk}^2 = 0$ for some k, then furtherly assume that $\Sigma_j = \sigma_j^2 I$, where $\sigma_j^2 > 0$. Under this restriction, the approximate MLE $\hat{\Sigma}_j$ of Σ_j is

$$\hat{\Sigma}_j = \hat{\sigma}_j^2 I, \quad \hat{\sigma}_j^2 = \frac{1}{p\#I_j} \sum_{i \in I_j} \rho^2(X_i, \hat{\mu}_j) \tag{25}$$

However, this assumption may excessively envelop the points, that is, the generated sphere may be relatively large compared to the number of its elements. Hence, this hardly restrictive condition must be carefully assumed.

5 Hyperparameter Selection

In section 3, we assumed that the number J of ellipsoids is predetermined and the construction of prediction set is implemented. That is, the shape of conformal prediction set C_n^α also depend on J. From now, to clarify the dependency, denote the conformal prediction set as $C_n^{\alpha,J}$. By the distribution-free property of the conformal prediction, the coverage ratio of the conformal prediction set C_n^α is at least $1-\alpha$. To avoid the construction of trivial or nearly trivial prediction set, we need to choose α and J deliberately.

On the other hand, S. Jung, et al.(2020) suggests the following: Suppose α is given. Then the desired $\hat{J}$ is chosen with

$$\hat{J} = \arg\min_J \mu(C_n^{\alpha,J}) \tag{26}$$

Now, suppose that J is given. As α increases, the coverage of the prediction set also increases. However, if we just choose α minimizing the volume of the prediction set, α must be zero. To avoid this trick, S. Jung, et al.(2020) suggests

$$\hat{\alpha} = \arg\min_\alpha(\alpha + \mu(C_n^{\alpha,J})) \tag{27}$$

Finally, by combining these two methods, we may choose $\hat{J}$ and $\hat{\alpha}$ as

$$(\hat{\alpha}, \hat{J}) = \arg\min_{\alpha,J}(\alpha + \mu(C_n^{\alpha,J})) \tag{28}$$

On the other hand, J. Shin, et al.(2019) also suggests the method for choosing J, with hypothesis testing framework. This approach also seems applicable.

6 Clustering With Toy Data

We tested the proposed clustering algorithm with two artificial data on $\mathbb{T}^2$ which are already used by S. Jung, et al.(2020). The two artificial data are each sampled from the following models:

- Model I: The dataset of size $n = 270$ is sampled from a mixture of $K = 5$ clusters, where three clusters are sampled from bivariate normal distributions (with sizes 70, 50, 50) and the other two are each sampled from the uniform distribution on a rectangle defined on $\mathbb{R}^2$ (each with size 50), then wrapped onto the torus.
- Model II: The dataset of size $n = 500$ is sampled from a mixture of $K = 3$ clusters, where the first cluster is sampled from a spherical normal distribution with size $n_1 = 100$, the second cluster of size $n_2 = 350$ is from the uniform distribution on a large "L"-shaped region, and the third cluster of size 50 is sampled from the uniform distribution on the entire $\mathbb{T}^2$.

Data sets described above are generated for the estimation of prediction regions and clustering rules. These data sets are called training data. Independent sets of data from the same models are also generated for validation, and are called testing data. Clustering rules based on the mixture models with J, α chosen by the proposed criterion (28) result in $K = 5$ clusters for Model I

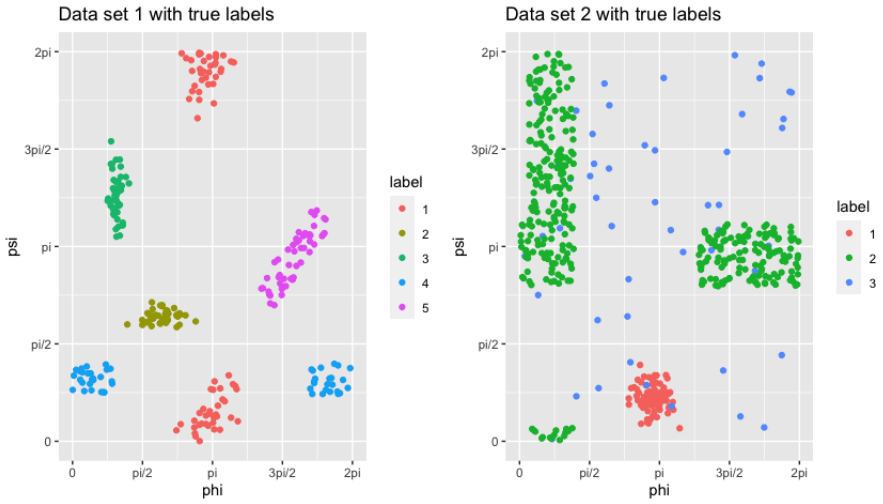

Figure 2: Simulated toroidal data sets with true cluster labels.

and $K = 2$ for Model II. We have compared (1) the naive K-means clustering (ignoring the angular constraint), (2) the extrinsic K-means clustering in the ambient space, and the proposed predictive clustering methods with membership assignment by (3) generating new cluster for the outliers. we will show the cluster results with some plots and evaluate the results with adjusted Rand index(ARI), which measure the correctness of the result cluster compared to the true data. ARI has the value of 1 when two indices match perfectly.

We now inspect the results for each model. Since Model I consists of five well-separated clusters, human eye is excellent in accurately clustering the data, with an understanding of the "cut-and-flattened" torus plots shown in Figure 2. The naive K-means algorithm does not consider the angular constraint of toroidal data, and results in a poor clustering (top left panel of figure 3). The extrinsic k-means seems much better than ordinary k-means, but the result is still unsatisfactory. It assigns one complete "true" cluster to the other cluster (bottom left panel of figure 3). The terrible results can be shown in the right panels of figure 3. In model II, ordinary k-means shows terrible result. It assigns the horizontal part of "L" to the "O" shape cluster. (Top right panel of figure 3). As in model I, the extrinsic k-means shows better results, but it also assign the top part of "L" cluster to the "O" cluster. Table 1 shows the results

Model	Ordinary k-means	Extrinsic k-means
Model I	0.65	0.76
Model II	0.09	0.55

Table 1: Adjusted Rand indices (ARI), evaluated for predicted cluster labels of synthetic toroidal data. The higher ARI, the better clustering membership prediction.

objectively. Ordinary k-means shows poor performance, especially in model II,

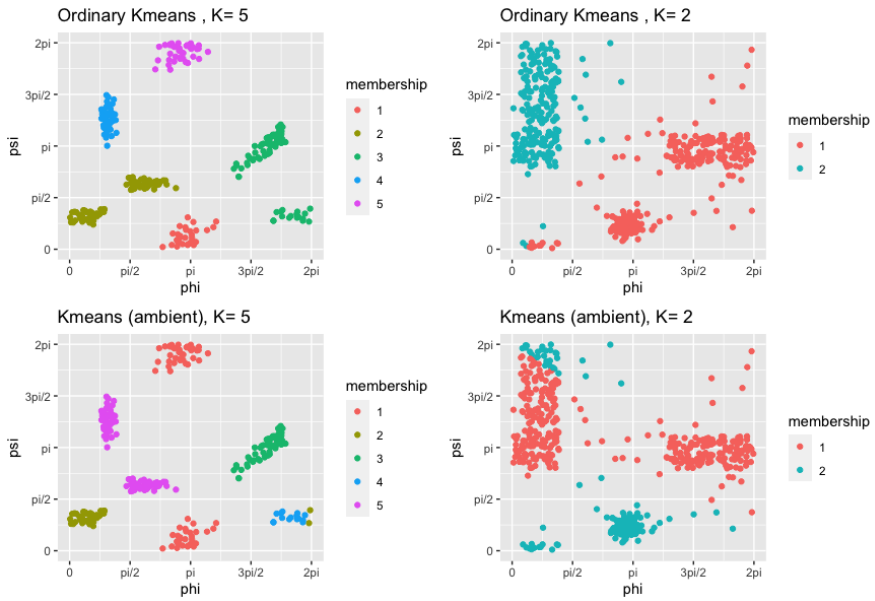

Figure 3: Clustering results for ordinary k-means and extrinsic k-means algorithm. Left column shows the results for model I, and the right column shows the results for model II.

and the extrinsic k-means is relatively superior to ordinary k-means, but the result is not satisfactory. Now, observe the results for the proposed models; homogeneous spheres, heterogeneous spheres, and general ellipsoids.

Figure 4 shows the prediction sets for various methods. As theoretically predicted, first row of figure 4 shows union of homogeneous spheres(circles). Similarly, second row shows the union of heterogeneous spheres(circles) and the last row shows the union of ellipsoids(ellipses). Moreover, as we can see in figure 5, the intersecting ellipsoids automatically construct a separate cluster. We can check that the results of left column panels in figure 4, 5 show the almost perfect clusters compared to the true model.

However, the extrinsic k-means also shows adequate performance for model I. The overwhelming performance of proposed models are demonstrated on model II. In figure 4, we can check that the prediction sets tightly fit the "L" shape cluster and "O" shape cluster. Also, in figure 5, we can see the clear "L" and "O", which means the well-assignment of clusters.

Model	homogeneous	heterogeneous	general-ellipsoids
Model I	0.91	0.92	0.93
Model II	0.89	0.84	0.90

Table 2: Adjusted Rand indices (ARI), evaluated for predicted cluster labels of synthetic toroidal data. The higher ARI, the better clustering membership prediction.

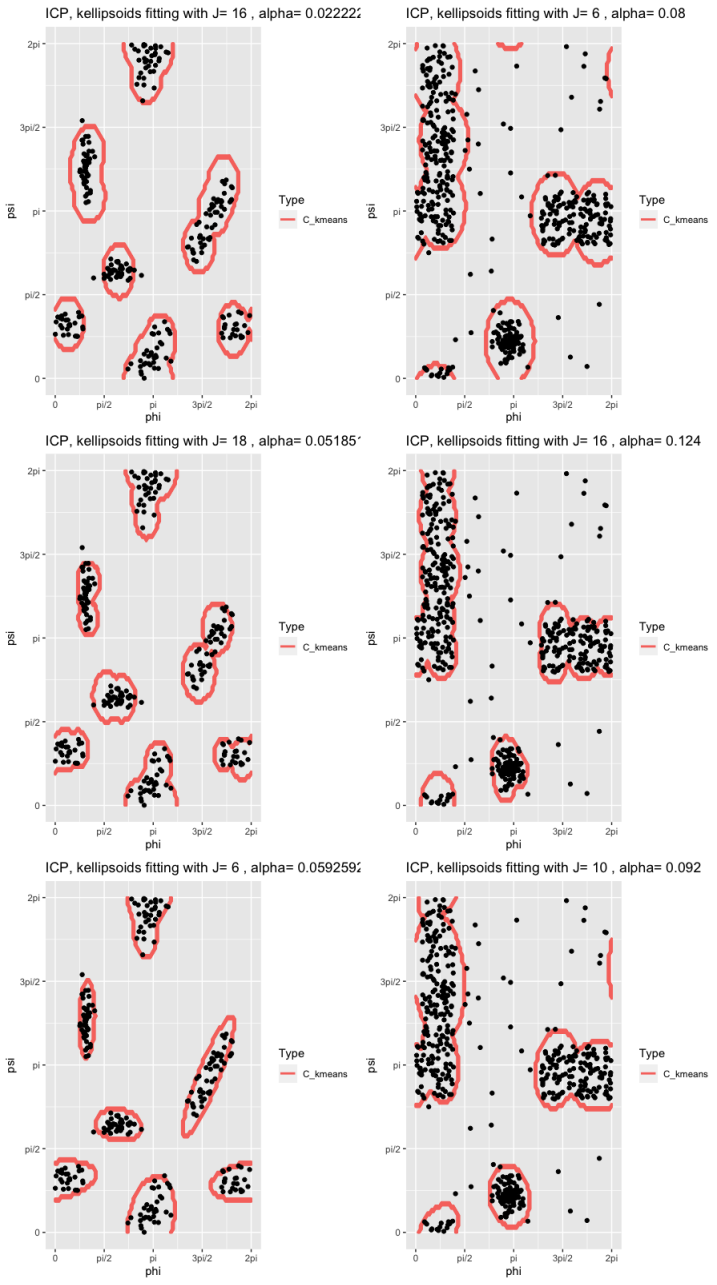

Figure 4: Boundary plot for the proposed models. Left column shows the results for model I, and the right column shows the results for model II. First row is for the homogeneous-spheres case, second row for the heterogeneous case, and the last row for the general ellipsoids.

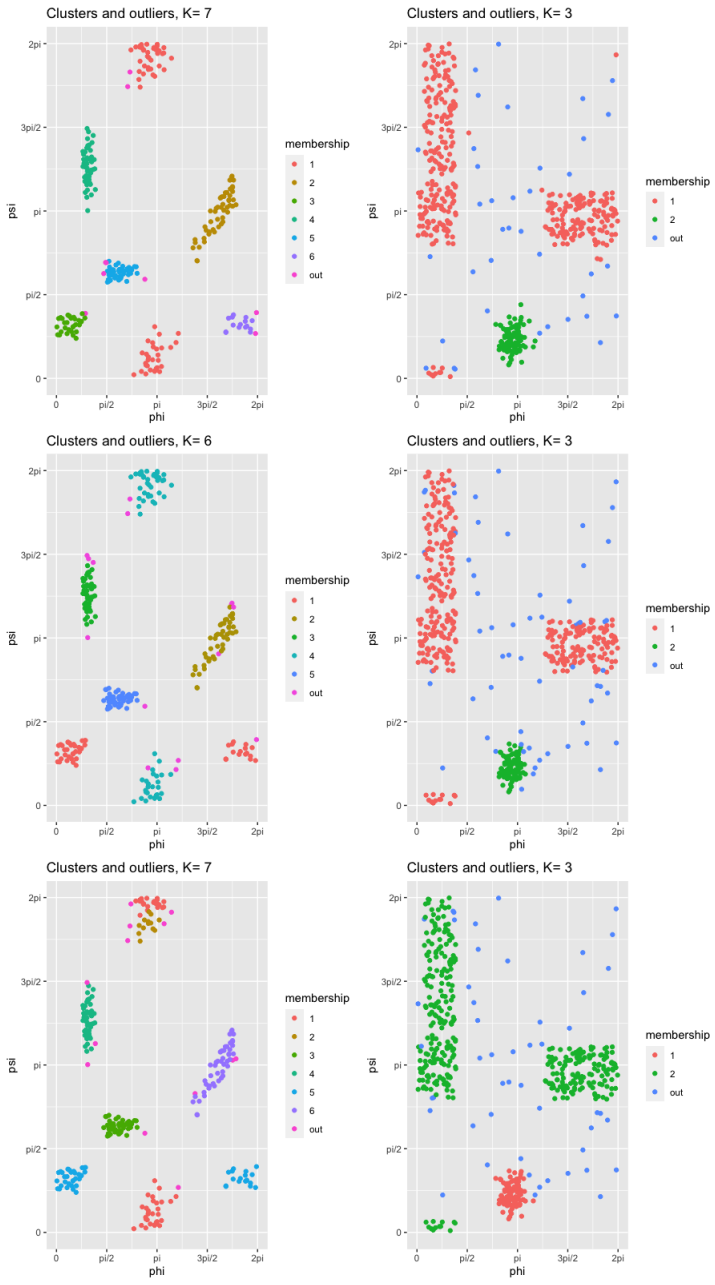

Figure 5: Clustering results for the proposed models. Left column shows the results for model I, and the right column shows the results for model II. First row is for the homogeneous-spheres case, second row for the heterogeneous case, and the last row for the general ellipsoids.

As we check in table 1, we evaluated the ARIs of proposed methods. Objectively, the performance of proposed models are sufficiently better than the extrinsic k-means. On the other hand, the performance of homogeneous case is impressive; homogeneous case is constructed with relatively simple theoretical statements and procedures. However, it performs similar, sometimes superior, to the other two proposed methods. Of course, we need to test these methods on higher space, but in $\mathbb{T}^2$ example, it seems that the methods work quite well.