연구주제 소개

Clustering in High dimensional data : Probabilistic Reduced K-means Clustering Model

응용통계학 전공 이승훈 지원자

요약

Cluster analysis는 비지도 학습이므로 어떤 그룹이 내포되어 있는지 사전 지식이 없을 경우에 군집을 발굴하고, 군집 간의 특성 차이와 군집 안에서의 유사성을 분석하고자 할 때 유용한 방법이다. Cluster analysis에서 가장 많이 쓰이는 K-means 기법은 데이터의 차원이 커질 때, 즉 변수의 개수가 많아질 때 잘 동작하지 않을 수 있으며, 군집에 대한 해석도 쉽지 않은 문제가 있다. 따라서 다차원 데이터의 경우 PCA와 같은 차원 축소 방법을 사용하여 변수의 개수를 줄인 후에, K-means를 사용하는 것이 일반적인 방법이다. 이를 Tandem-clustering이라 일컫는다.

하지만 PCA 와 같은 차원 축소 방법을 이용해서 찾아낸 feature가 반드시 cluster 구조를 반영할 것이라는 보장은 없다. 특히 cluster 구조와는 상관없는 변수들의 분산 또는 공분산이 클 때, PCA를 통해 차원 축소를 진행하면, 오히려 클러스터링 구조가 가려지게 될것이다. 이에 따라 clustering과 차원 축소를 동시에 진행하는 방법론들이 논의 되어왔다.

그중에서도 본 연구에서는 K-means clustering in a low-dimensional Euclidean space(Geert De Soete, J. Douglas Carroll, 1994)에서 제안한 방법론을 확률적인 모델로 바꿔 클러스터링을 진행하도록 한다. 시뮬레이션 결과 변수의 개수가 클러스터 당 표본 크기에 비해 큰 데이터에 대해 기존 방법론에 비해 우수한 성능을 확인했다. 또한, EM 알고리듬을 활용하므로 후에 결측이 포함된 데이터의 분석으로도 확장도 가능할 것이다.

배경 설명

Cluster Analysis

- 주어진 데이터의 특성(variable)을 고려하여 유사한 그룹으로 객체를 묶어 cluster를 정의하고, cluster의 대표적 특성을 찾아 분석하는 기법이다.
- Label이 필요없는 비지도 학습이므로 데이터에 어떤 그룹이 내포되어 있는지 사전지식이 없을 경우에 군집을 발굴하고, cluster간 특성 차이와 cluster 안에서의 유사성을 분석하고자 할 때 유용하다.(ex. K-means clustering)

Tandem Clustering

- K-means clustering 의 경우 variable의 개수(차원)가 커지는 경우 원활히 동작하지 않을 수 있다.(차원의 저주)
- 따라서 High dimensional 한 데이터에 대해서 clustering을 하고자 할 때, PCA 등과 같은 차원 축소 방법을 사용하여 variable의 개수를 줄인 후, K-means 를 사용하는 것이 일반적인 방법이다. (이를 Tandem Clustering이라 한다)
- 하지만 PCA 를 통해 찾아낸 subspace 가 반드시 cluster 구조를 반영할 것이라는 보장은 없다. (특히 노이즈 variable의 분산이 클 경우)

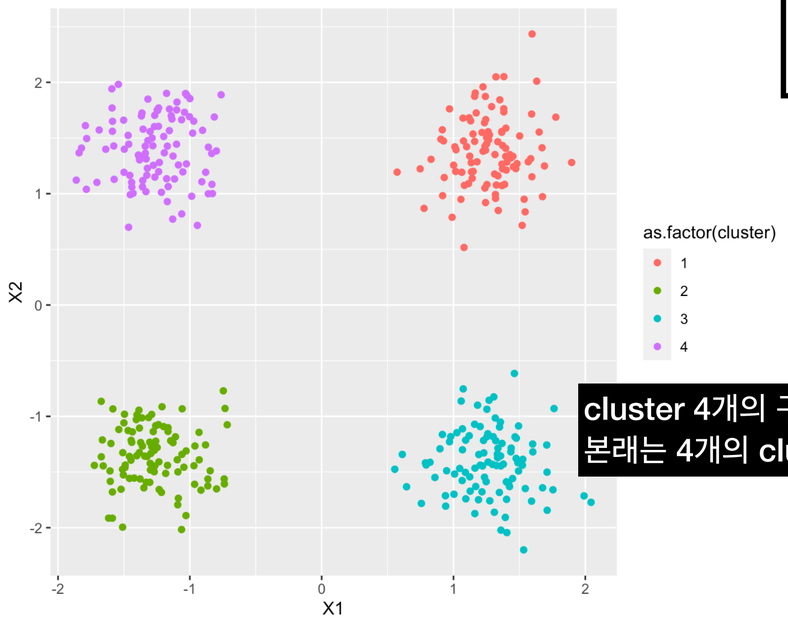
Tandem Clustering 의 한계 예시

- 예제 데이터

	Var1	...	Var100
1			
...			
400			

- Variable 의 개수 = 100
- Cluster 의 개수 = 4
- PSR = 0.1

True Cluster Structure

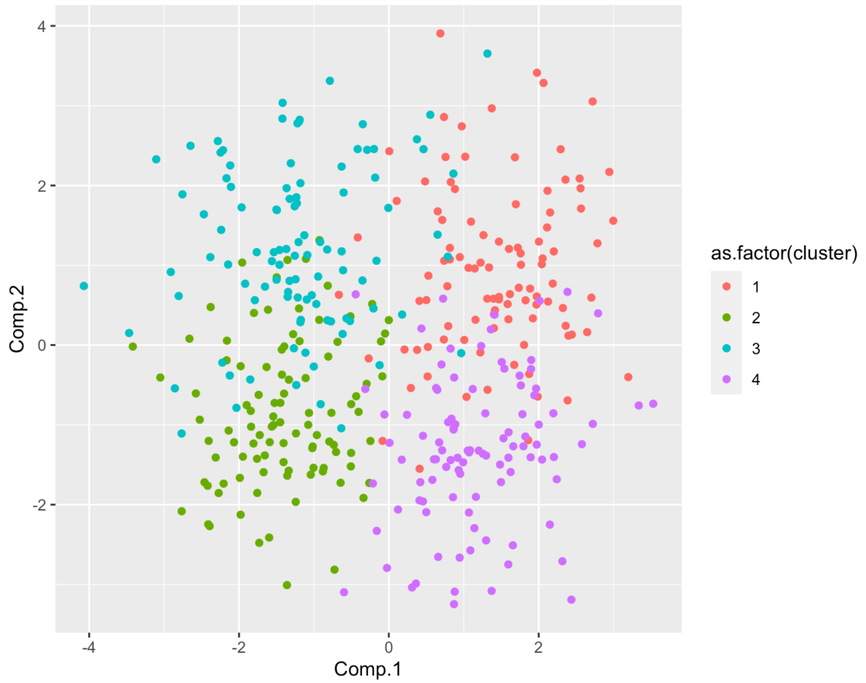


cluster 4개의 구조를 완전히 설명하는 True dimension 2개에 의해 본래는 4개의 cluster가 명확히 구분되는 데이터이다.

PCA를 통한 차원축소

데이터의 전체 분산은 Cluster 를 설명하는 차원에 있는 분산과
Cluster를 설명하는 차원과 **Orthogonal 한 차원에 있는 분산**으로 설명할 수 있다.
이 때, **전체 분산중 Cluster를 설명하는 차원 상의 분산 비율을 PSR 이라고 한다.**

PCA를 통해 축소된 차원 (2 dimension)



- PCA를 통해 차원축소를 진행할시, Cluster의 구조가 가려져 Clustering을 진행하기 어려워진다.
- Cluster 와 관련없는 차원의 분산이 클 때 더 심해지는 경향을 보인다.

이를 개선하기 위해,
Cluster Analysis 와 차원 축소를 동시에 진행하는 방법론이 고안되었다.

Reduced K-means(RKM)

(K-menas clustering in a low-dimensional Euclidean space, De soete & Douglas Caroll, 1994)

모형 : $\mathbf{F} = \|\mathbf{X} - \mathbf{EC}\|^2$, such that rank($\mathbf{C}$) $= Q$.

Data $\mathbf{X}(N \times p)$ Cluster Membership($N \times K$) Cluster Centroids($K \times P$) Original data의 차원 P를 축소하고자하는 차원 Q

설명 :

- 각 Object와 해당 Cluster centroids에 대한 거리의 합을 최소화 하도록 Cluster Centroids($\mathbf{C}$) 와 Cluster membership($\mathbf{E}$)를 업데이트 한다.
- 이때 Cluster Centroids 의 차원을 축소차원 $\mathbf{Q}$로 제약한다.
- ALS 알고리듬 이용한다. (Estimation을 위해 Cluster membership($\mathbf{E}$)를 찾는 단계와 Cluster Centroids($\mathbf{C}$)를 차원 축소하는 단계를 번갈아 가며 수렴할때 까지 반복한다.)
- **즉, 차원축소와 Clustering 을 독립적으로 실행하는 Tandem Analysis에서 벗어나, 차원축소와 Clustering을 동시에 진행한다.**

이런 RKM 모델을 통계적 모델로 전환시키는 시도를 했으며 Probabilistic Reduced K-means (PRKM) 이라 명명하도록 한다.

Probabilistic Reduced K-means

모델 가정

- 가정1 : 데이터 $\mathbf{X}$의 분포가 K개의 Gaussian Mixture Density 로 이루어져 있음

- 가정 2 : K개의 Gaussian component는 클러스터 구조를 설명하는 q개의 잠재 변수의 선형 가중 합. $\mathbf{x} = \mathbf{A}\mathbf{y}_k + \mathbf{v}$

모델

Gaussian Mixture Model(GMM) : Clustering

Linear Gaussian Model : 차원 축소

- $p(\mathbf{x}) = \sum_{k=1}^{K} p_k(z) p_k(\mathbf{x} \mid z)$ & $P_k(\mathbf{x} \mid Z) = \int_y p_k(\mathbf{x} \mid Z, \mathbf{y}) p(\mathbf{y}) d\mathbf{y} = N(\mathbf{A}\mu_k, \mathbf{A}\mathbf{A}^\top + \epsilon\mathbf{I})$

  - $\mathbf{A}(p \times q)$는 $\mathbf{X}$ 를 클러스터와 관련있는 부분공간(subspace)로 투영시키는 투영행렬

  - $\mathbf{y_k}$는 k 번째 가우지안 컴포넌트의 잠재변수를 뜻함 - Z 는 해당 데이터가 k번째 cluster에 해당하는가를 나타낸 지시행렬

    - $\mathbf{y}_k \sim N(\mu_k, \mathbf{I})$의 분포를 따른다.

  - 마지막으로 $\mathbf{v}$는 linear Gaussian Model의 오차항에 해당되며 $\mathbf{v} \sim N(0, \mathbf{R})$이다. 오차항의 공분산행렬 $R = \epsilon\mathbf{I}$로 제약을 줌.

EM Algorithm

Hidden cluster membership과 latent variable을 Missing 된 정보라고 간주하여 EM 알고리듬을 활용한다

- Step 1 : $\pi_1, \ldots, \pi_K, \mu_1, \ldots \mu_K, \mathbf{A}, \epsilon$의 초기값을 임의로 지정한다.
- Step 2(E-step) : 다음의 수식에 따라 k = 1,…,K 에 대해서 다음을 계산한다.

(a) $r(z_{ik})^{(t+1)} = p(z_{ik} = 1 \mid \mathbf{x}_i) = \dfrac{\hat{\pi}_k p(\mathbf{x}_i \mid z_{ik} = 1)}{\sum_{k=1}^{K} \hat{\pi}_k p(\mathbf{x}_i \mid z_{ik} = 1)}$

(b) $E(\mathbf{y} \mid \mathbf{x})_k^{(t+1)} = \mu_k^{(t)} + \mathbf{A}^{(t)\top}\mathbf{M}^{-1}(\mathbf{x} - \mathbf{A}^{(\mathbf{t})}\mu_k^{(t)}) = \mathbf{e}_{ki}^{(t+1)}$

(c) $E(\mathbf{y}\mathbf{y}^\top \mid \mathbf{x})_k^{(t+1)} = \left(\mathbf{I} - \mathbf{A}^{(\mathbf{t})\top}\mathbf{M}^{-1}\mathbf{A}^{(t)}\right) + E(\mathbf{y} \mid \mathbf{x})_k^{(t+1)} E(\mathbf{y}^\top \mid \mathbf{x})_k^{(t+1)} = \mathbf{V}_{ki}^{(t+1)}$

$\mathbf{M} = \left(\mathbf{A}^{(t)}\mathbf{A}^{(t)\top} + \epsilon^{(t)}\mathbf{I}\right)^{-1}$

- Step 3 (M-step) : 추정치를 아래와 같이 갱신한다.

(a) $\pi_k^{(t+1)} = \dfrac{\sum_{i=1}^{N} r(z_{ik})^{(t+1)}}{N}$

(b) $\mu_k^{(t+1)} = \dfrac{\sum_{i=1}^{N} r(z_{ik})^{(t+1)} \mathbf{e}_{ki}^{(t+1)}}{\sum_{i=1}^{N} r(z_{ik})^{(t+1)}}$ …

(c) $\mathbf{A}^{(t+1)} = \sum_{i=1}^{N}\sum_{k=1}^{K}\left(r(z_{ik})^{(t+1)}\mathbf{x}_i\mathbf{e}_{ki}^{(t+1)\top}\right)\left(\sum_{i=1}^{N}\sum_{k=1}^{K} r(z_{ik})^{(t+1)}\mathbf{V}_{ki}^{(t+1)}\right)^{-1}$

$\epsilon^{(t+1)} = \dfrac{\sum_{i=1}^{N}\sum_{k=1}^{K} r(z_{ik})^{(t+1)}\left(\mathbf{x}_i^\top\mathbf{x}_i - 2\mathbf{x}_i^\top\mathbf{A}^{(t+1)}\mathbf{e}_{ik}^{(t+1)} + tr(\mathbf{V}_{ik}^{(t+1)}\mathbf{A}^{(t+1)\top}\mathbf{A}^{(t+1)})\right)}{Np}$

- Step 4 : likelihood가 수렴할때 까지 위의 Step 2와 Step 3을 반복한다.
- Step 5 : 각 객체마다 사후확률이 가장 큰 클러스터로 할당한다.

참고

- EM 알고리듬은 매 Iteration마다 Likelihood 를 증가시킨다.
- 따라서 Convergence 를 보장할 수 있다.
- 그러나 Global optimum을 보장하지 않기 때문에 여러개의 initial 값을 주어 실행을 한 후, 가장 Likelihood가 큰 initial의 결과를 최종 결과로서 이용한다.

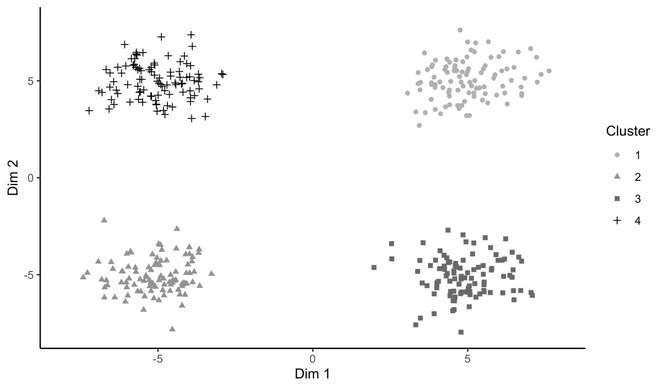
Simulation 결과 요약-1

각 조합마다 **10**개 데이터셋 생성하여 실험

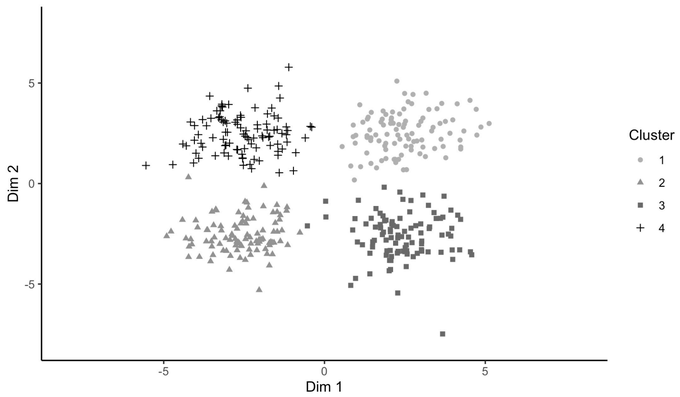
Simulation 데이터 형성

- Cluster의 개수 = 4 (고정)
- Variable의 개수 = {10, 50개, 100개} → **2차원으로 줄이려 한다**
- Overlapping 여부 = {Yes, No}
- Cluster 별 객체 수 = {50개, 100개}
- PSR = {0.05, 0.1 0.15} or {0.2,0.25,0.3}

Overlapping : No



Overlapping : Yes



clustering 평가 지표 : ARI(Adjusted Rand Index)

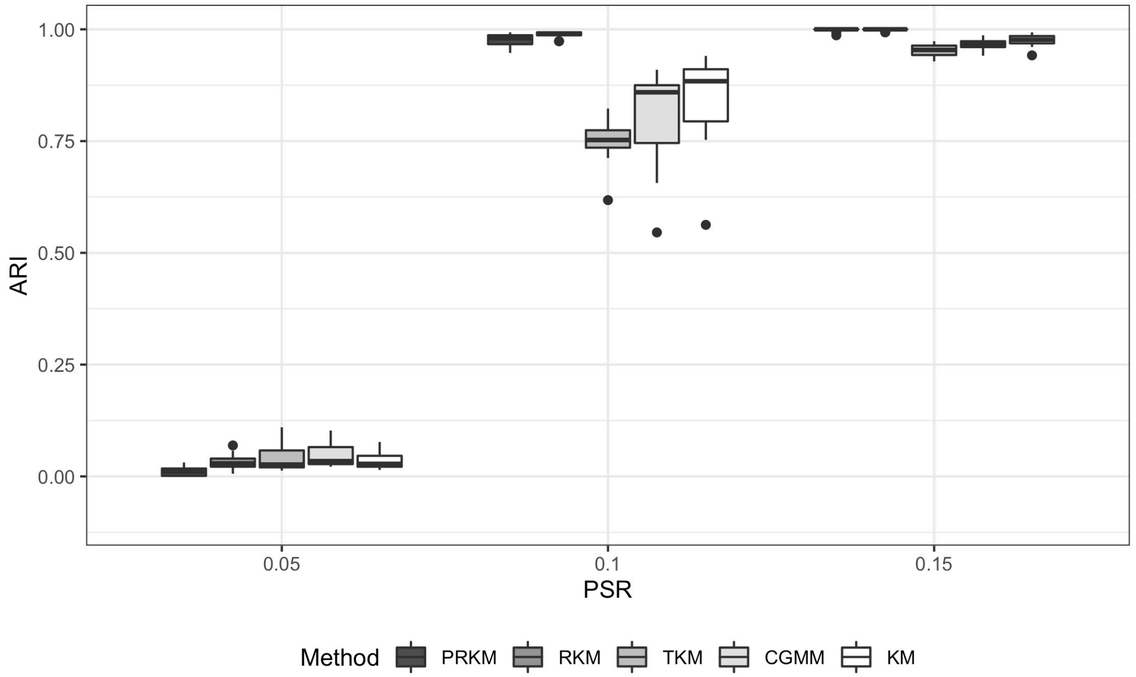
- ARI는 우연히 grouping 될 가능성에 대하여 보정한 clustering 평가지수
- 1에 가까울수록 완벽한 clustering
- 0일때, random하게 cluster 배정한 것과 동일한 성능
- 음수일 경우, random한 cluster 배정보다 못한 성능

비교 모델	표기
Probabilistic Reduced K-menas	**PRKM**
Reduced K-means	**RKM**
Tandem Clustering	**TKM**
Constrained Gaussian Mixture	**CGMM**
K-means	**KM**

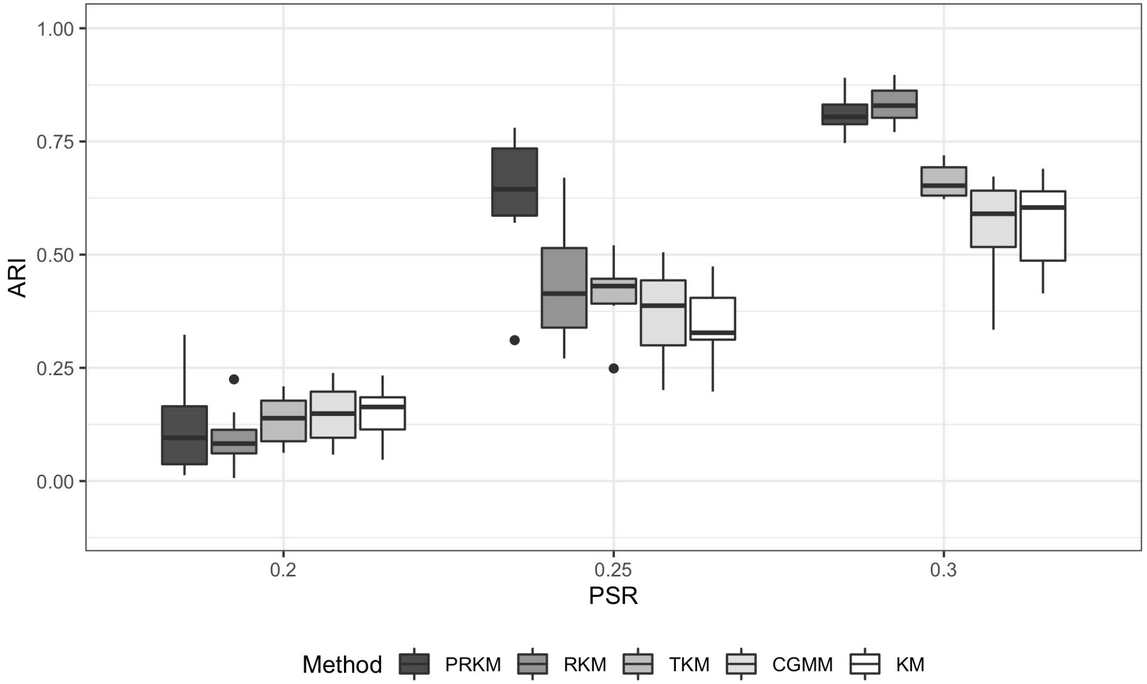
Simulation 결과 요약-2

Variable 개수: 100; cluster 별 개체 수 : 100 인 경우

Overlapping : No 인 경우, PSR 수준별 ARI



Overlapping : Yes 인 경우, PSR 수준별 ARI

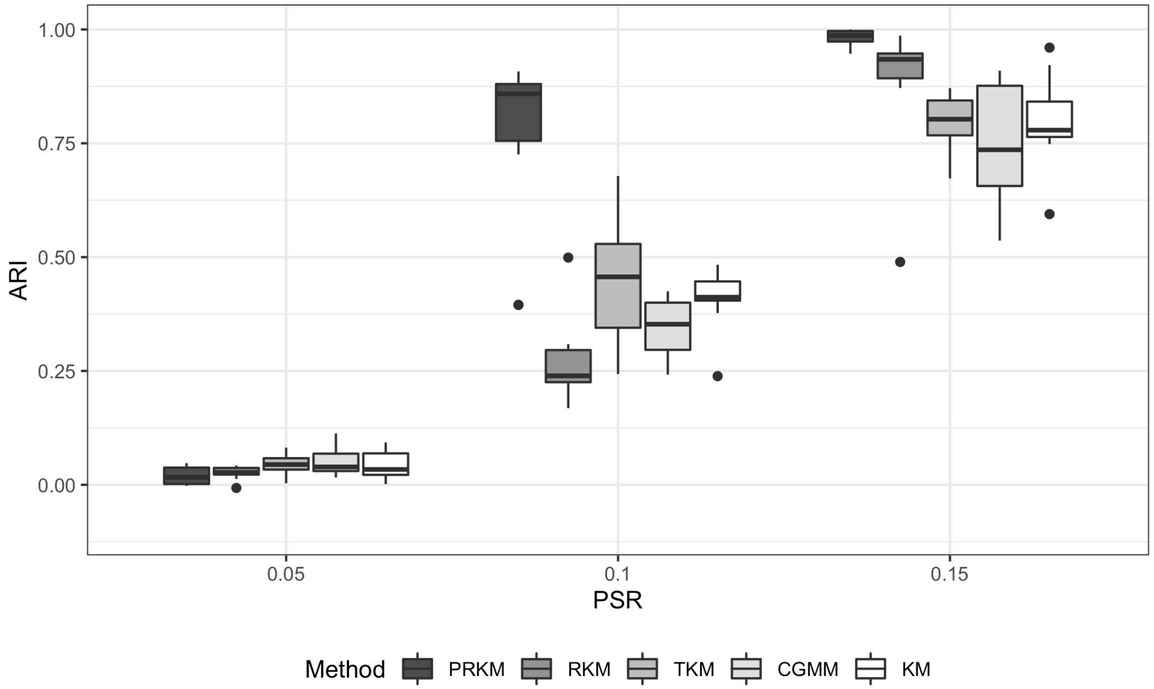


- Cluster의 Overlap이 존재하지 않는 경우 PSR 수준에 관계없이 항상 PRKM과 RKM의 성능이 기존의 Clustering 방법보다 우세
- Cluster의 Overlap이 존재하는 경우, PSR 수준이 낮을 때(0.25) PRKM 방법이 모든 방법론의 성능을 상회함

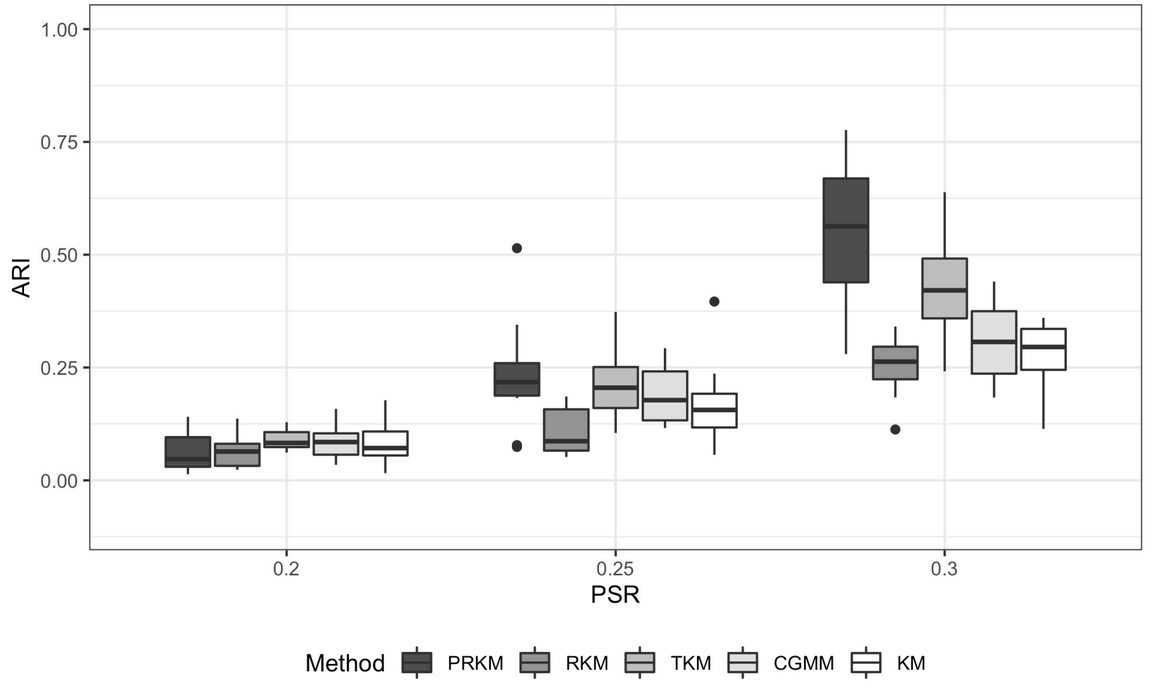
Simulation 결과 요약-3

Variable 개수: 100; cluster 별 개체 수 : 50 인 경우

Overlapping : No 인 경우, PSR 수준별 ARI



Overlapping : Yes 인 경우, PSR 수준별 ARI

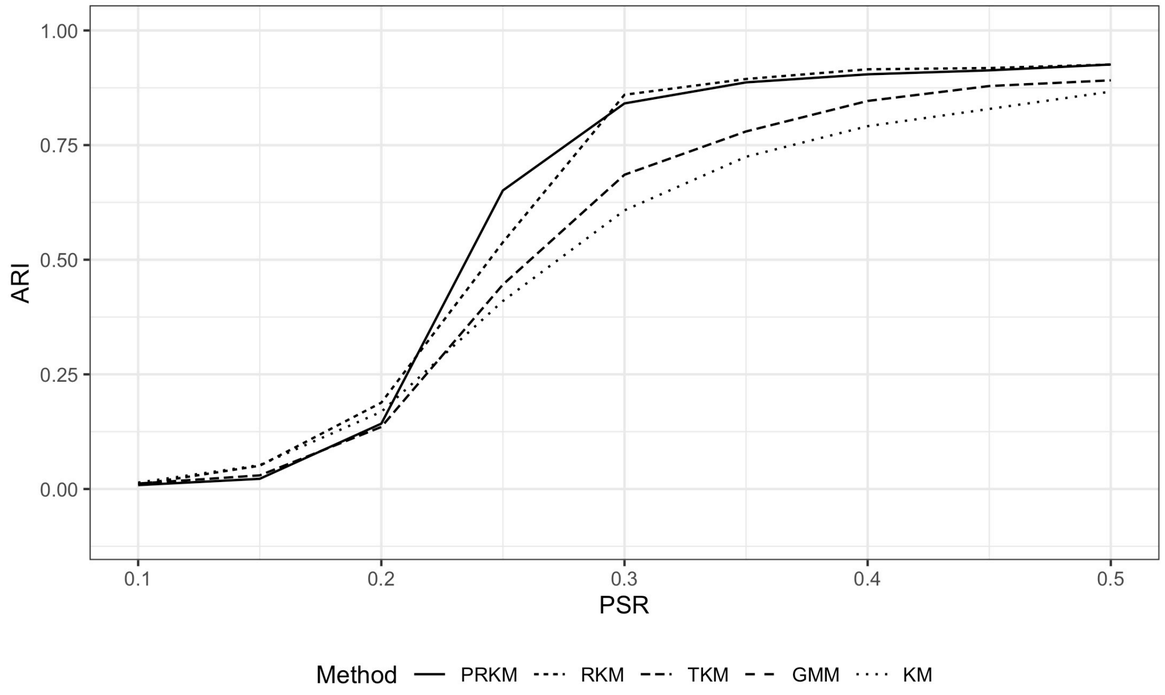


- Cluster의 Overlap이 존재여부에 관계없이, PRKM의 성능이 항상 기존의 Clustering 방법 뿐 아니라, RKM의 성능까지도 상회함

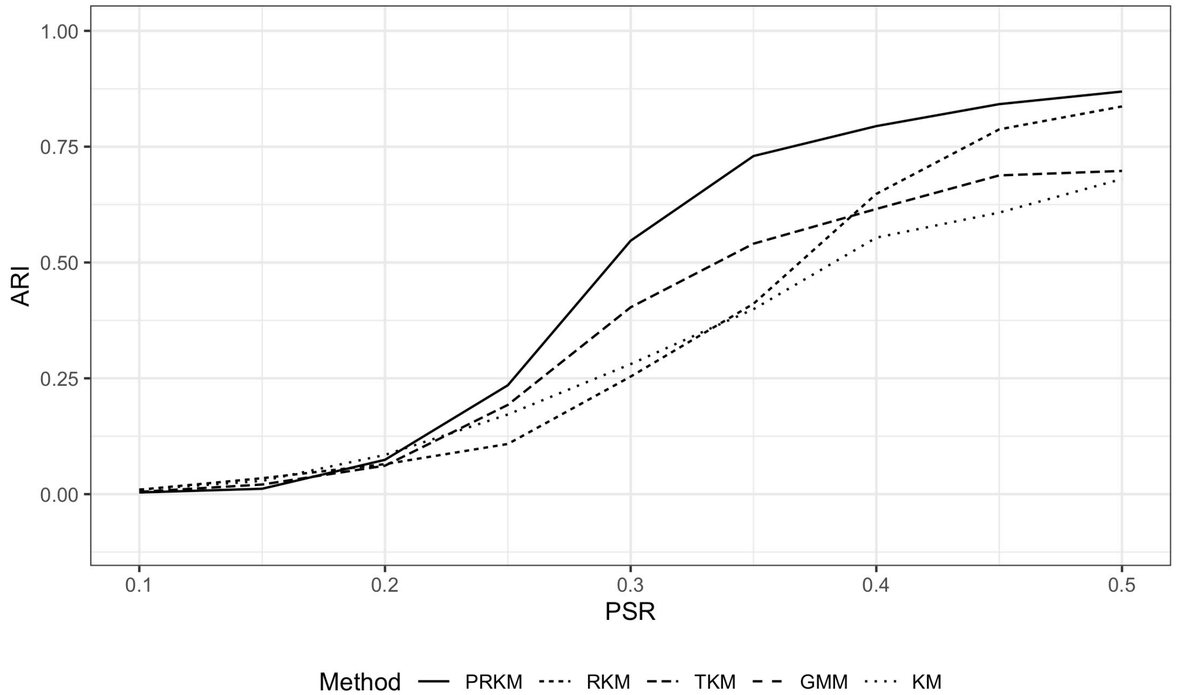
추가 Simulation

Overlapping : Yes; Cluster 별 개체 수 : 50; PSR = {0.1,0.15,…,0.45,0.5}

variable의 개수 50개인 경우, PSR 수준에 따른 ARI



variable의 개수 100개인 경우, PSR 수준에 따른 ARI



- 본 실험데이터(**Cluster overlapping이 존재하며, 클러스터의 객체 수(50개)가 variable의 개수보다 작거나(100개), 같은(50개) 상황**)에서는 대체적으로 **Probabilistic Reduced K-means가 가장 clustering 성능이 좋은 것을 확인 할 수 있었다.**

Simulation 결과 해석

- Cluster의 overlap의 존재여부와 관계없이, variable의 수가, cluster 별 객체의 수 보다 작을 수록 Clustering 방법론에 상관없이 낮은 PSR 수준에서도 좋은 clustering 성능을 보였다.

- Cluster의 overlap이 존재하지 않을 때, variable의 수가, cluster 별 객체의 수와 같다면 PSR수준에. 무관하게 RKM이 가장 좋은 성능을 보였으며, PRKM이 그 다음으로 성능이 좋았다. 다른 기존의 방법론의 경우 성능이 좋지 않음을 확인했다.

- Cluster의 overlap이 존재하면서도 variable의 수가, cluster 별 객체의 수와 같다면 중간 정도의 PSR수준(0.25)에서는 PRKM의 성능이 RKM의 성능을 상회했다.

- Cluster의 overlap 여부에 관계없이, variable의 수가, cluster 별 객체 수 보다 같다면, 너무 낮지 않은 PSR수준에서, 항상 PRKM의 성능이 가장 우수했다.

변수의 수가 많으면서도, 관찰치가 적은 데이터에서 Probabilistic Reduced K-means Clustering 이 유용할 것으로 기대함