결측을 포함한 클러스터링 방법 비교

결측자료분석 기말고사 프로젝트 -2020020345 이승훈-

1. 프로젝트 배경

결측을 포함하고 있는 데이터를 이용하여 클러스터링 분석을 할 때, 가장 쉽게 생각할 수 있는 방법은 결측대체를 먼저 시행한 다음에, 결측 대체가 된 데이터 셋에 대해서 완전한 자료라고 생각하고 클러스터링 분석을 진행하는 것이다. 그러나 결측 대체된 값은 불안정하고 그 신뢰성에 문제가 있을 수도 있다고 생각했다.

따라서 본 프로젝트에서는 사전에 결측 대체를 진행하지 않고 클러스터링을 진행하는 방법에 대해 탐구하고, 결측 대체를 진행한 다음 클러스터링 분석을 하는 방법과 성능 비교 논의를 해 보고자 한다.

2. 결측대체가 필요하지 않은 방법론과 평가지표 요약

2-1. K-means with soft constraint (KSC) [1]

KSC의 핵심 아이디어는 결측 자료가 포함 된 데이터를 모든 개체에 대해서 관측된 변수(F_o)와 결측이 존재하는 변수(F_m) 으로 나눈 다음, F_o 에 대해서 k-means 를 진행 할 때 F_m 중 완전히 관찰 된 객체 만으로 soft constraint 를 주는 것이다. Soft constraint 는 다음과 같이 정의된다.

$$s_{i,j} = -\sqrt{\Sigma_{f\in F_m}(d_i.f - d_j.f)^2}$$

즉, 어떤 객체들이 같은 클러스터로 분류될 경우, 부분적으로 결측이 발생한 변수를 이용하여 거리를 계산하고, 이 거리를 패널티로서 활용하는 것이다. 이 때, soft constraint 는 F_m에서 완전히 관찰된 객체(i, j) 사이에서만 정의가 된다. Soft constraint 를 이용한 KSC 알고리듬은 다음의 목적함수를 최소화 하는 방향으로 객체들을 클러스터에 할당한다. 이때 w는 하이퍼파라미터이며, CV는 d 에 대한 Soft Constraint의 합, V_{max} 와 CV_{max} 는 표준화를 위한 값이다.

$$C := \text{argmin}_{C_i}\left((1-w)\frac{dist(d,C_i)^2}{V_{max}} + w\frac{CV_d}{CV_{max}}\right)$$

2-2. K-Pod [2] 과 K-means with build-in imputation (ClustImpute) [3]

두 알고리듬은 결측대체단계와 K-means 클러스터링 단계를 번갈아가며 여러번 반복 한다는 점에서 공통점을 가지고 있으나, 결측대체 방법에서 차이를 보인다. 즉 K-Pod 의 경우에는 객체가 할당된 클러스터의 Centroid 로 결측대체를 진행하나, K-means with build-in impuation, (편의를 위해 ClustImpute라 명하겠다)의 경우에는 객체가 할당된 클러스터의 멤버의 값 중 하나를 무작위로 추출하여 결측대체를 진행한다는 점에서 다르다. 두 알고리듬은 각각 R의 kpodclustr 과 ClustImpute 패키지에서 이용 가능하다.

2-3 클러스터링 평가지표

클러스터링 평가지표로는 클러스터링 정확도를 비교하기 위한 Adjusted Rand Index와, 클러스터 내부를 평가하기 위한 dunn index 를 활용했다. 세부내용은 다음과 같다.

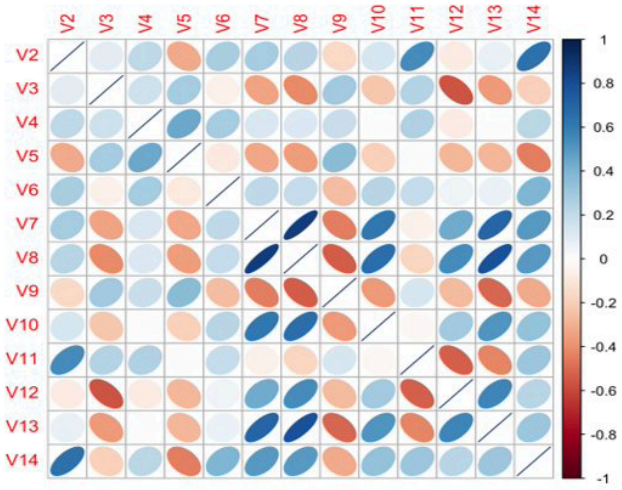
- Adjusted Rand index는 Rand index 를 우연에 대해서 보정한 인덱스로 0은 무작위 할당과 동일한 성능을 뜻하며, 1에 가까울 수록 완벽에 가까운 분류라고 할 수 있다.
- Dunn index 는 클러스터 간의 최소 거리를 클러스터 안의 최대 거리로 나눈 값이다. 즉, 값이 높을 수록, 클러스터 끼리는 거리가 멀고, 클러스터안 개체의 유사성은 높다는 것을 뜻한다.

3. 클러스터링 성능 비교를 위한 데이터 소개

3-1 Wine 데이터 소개

클러스터링 예제에 흔히 사용되는 Wine Data Set(UCI Machine Learning Repository) 를 사용했다. V1은 label에 대한 정보를 담고있으며, 나머지 변수들은 다음과 같다.

[그림 1 : WINE 데이터의 상관계수 플롯]



[표 1 : WINE 데이터의 변수와 표준편차]

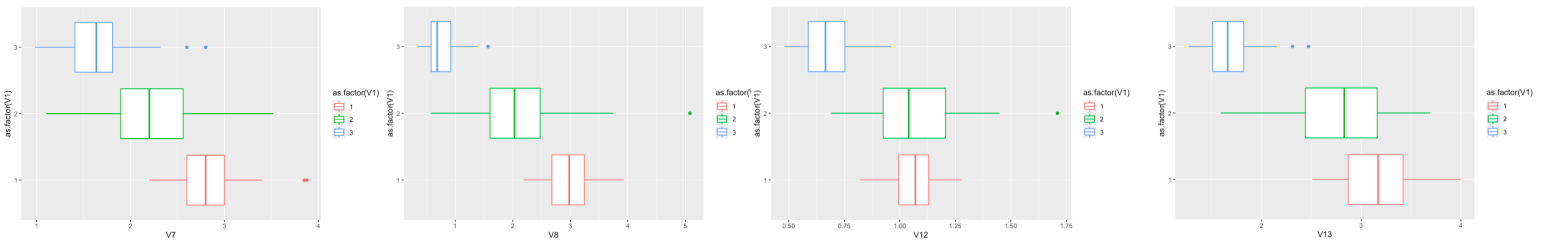
변수		SD
V2	Alcohol	0.812
V3	Malic acid	1.117
V4	Ash	0.274
V5	Alcalinity of ash	3.340
V6	Magnesium	14.282
V7	Total phenols	0.626
V8	Flavanoids	0.999
V9	Nonflavanoid phenols	0.124
V10	Proanthocyanins	0.572
V11	Color intensity	2.318
V12	Hue	0.229
V13	OD280/OD315 of diluted wines	0.710
V14	Proline	314.907

변수간의 피어슨의 상관계수가 높은 변수들이 몇몇 있다는 것을 확인했다. 또한 원 데이터의 변수들은 표준편차의 차이가 심하게 나는 편이었다. K-means 알고리듬의 특성상, 척도를 맞춰줄 필요가 있으므로 사전에 표준화 작업을 거쳤다. 추가적으로 모든 변수가 노말 분포를 만족하지 못하는 것을 확인하여 로그 변환을 시도했으나, 개선이 되지 않아, 표준화만 된 변수로 클러스터링 분석을 진행했다.

3-2 결측 데이터 발생 과정

- 결측을 발생 시킬 때, 와인 종류에 따라서 결측 발생 비율이 달라진다고 가정을 했다. 즉 3번, 2번, 1번 클러스터 순으로 결측이 많이 발생한다고 가정했다.
- 결측이 발생된 변수는 클러스터마다 분포의 차이를 많이 보이는 변수 V7, V8, V12, V13을 선정했다.

[그림 2 : 클러스터에 따른 결측발생 변수 박스플롯]



V7 **V8** **V12** **V13**

- 클러스터 레이블의 숫자가 높을 수록 더 많이 결측을 생성하는 것이 목적이므로, 전체 데이터를 PCA 를 진행하여, 클러스터 레이블과 가장 연관이 깊었던 10 번째 PC를 활용하여 결측을 생성했다. 결측을 생성할 때, 결측 비율에 따른 클러스터링 성능의 변화를 보고자 하는 목적으로 결측 비율이 13% 와 27% 가 되도록 결측을 발생시켰다.

$$10th\ PC = 0.728\ Cluster + 0.193\ V2 + 0.209\ V4 - 0.209\ V5 + 0.173\ V10 - 0.444\ V11 + 0.311\ V14$$

  - 10 번째 PC의 상위 15%의 점수를 가진 객체 중에서 무작위로 85% 의 결측을 발생시켰다.

- 10 번째 PC의 상위 30%의 점수를 가진 객체 중에서 무작위로 85% 의 결측을 발생시켰다.

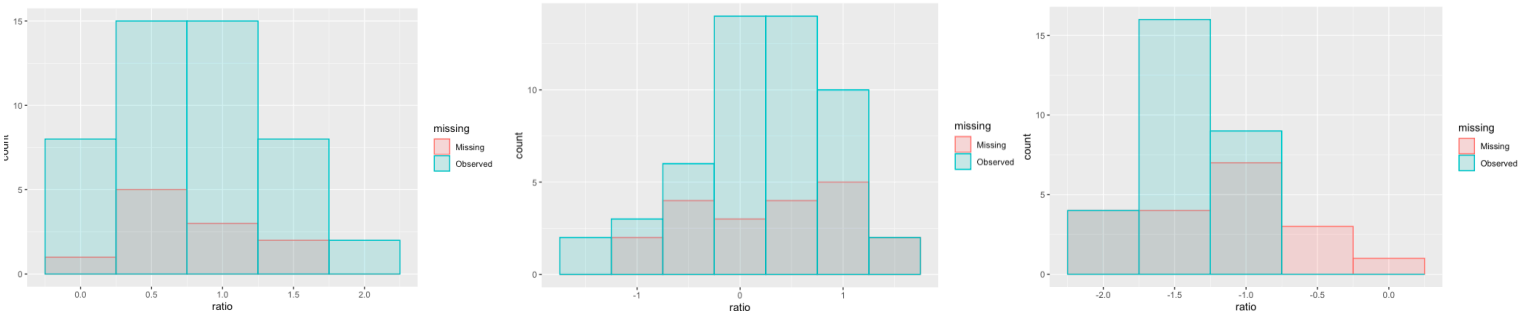
- 결측이 발생하는데 있어 클러스터에 대한 정보 뿐 아니라 다른 변수도 영향을 주기 때문에 MAR이라 볼 수도 있지만, 주로 어떤 클러스터에 속해있는지가 결측에 많은 영향을 미쳤기 때문에 MNAR 에 가깝다고 할 수 있을 것이다.

3-3 결측 데이터 발생 결과

General Missing 패턴이 만들어졌으며, 변수당 평균 13% 결측이 발생한 데이터 셋의 경우에는 1번, 2번, 3번 클러스터 각각 차례로 8.47%, 11.27%, 29.17% 의 미싱비율이 만들어졌으며, 변수당 평균 27% 결측이 발생한 경우에는 18.64%, 29.58%, 45.83% 결측이 발생했다.

또한 최소한 MCAR 가정은 아님을 확인 하기 위하여 변수별로 결측으로 처리한 객체와 관찰된 객체의 히스토그램을 그렸다. 이때 주의 할 점은 클러스터링을 목적으로 하기 때문에 클러스터를 고려하여 데이터 분포를 살펴봐야 한다는 것이다. 클러스터 구조를 고려하지 않는다면 오히려 MCAR 가정이 충족되는 것처럼 보일 수 있다. [그림] 에서 볼 수 있듯이 관찰된 객체와 결측된 객체의 분포는 3번, 2번 , 1번 순으로 상이 한 것을 확인했으며, 최소한 MCAR 가정은 아니라는 것을 확인 할 수 있었다.

[그림 3 : 27% 결측일 경우 V13 변수의 결측된 객체와 관찰된 객체의 분포 비교 히스토그램]



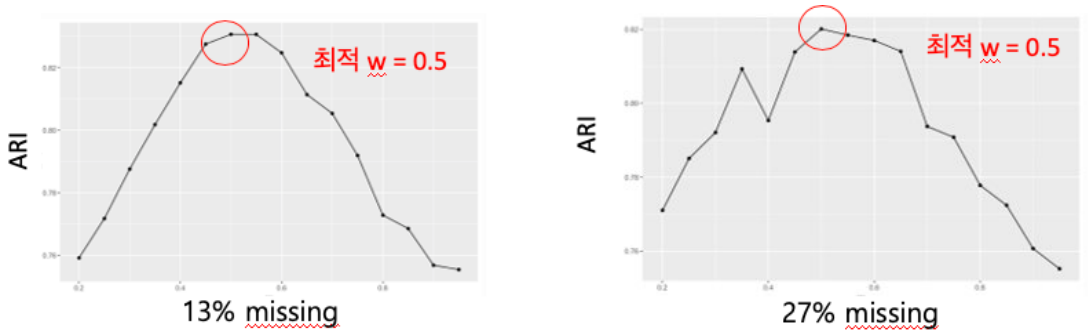
1번 클러스터 **2번 클러스터** **3번 클러스터**

4. 클러스터링 성능 비교

4-1 KSC의 w 하이퍼파라미터 결정

KSC 를 이용할 경우 가중치 하이퍼파라미터 w를 결정해 줘야 하는 어려움이 있다. 관련 논문에서는 결측이 발생한 변수가 클러스터링에 중요한 역할을 할수록 w 를 크게 줘야 한다고 언급하고 있다. 또한 중요도에 대한 사전 정보가 없다면, 클러스터에 대한 레이벨이 담긴 일부 자료에 대해서 KSC 를 적용하여 최적의 w를 찾을수 있다. 본 프로젝트에서는 전체 데이터를 가지고 최적의 파라미터를 찾은 후, 최적의 파라미터 값을 안다고 가정한다.

[그림 4 : K-MEANS WITH SOFT CONSTRAINT 의 w 결정]



결과적으로, 결측 비율을 달리한 두 자료 모두 0.5가 최적의 하이퍼 파라미터임을 알아낼 수 있었다.

4-2 결측대체를 진행하지 않는 클러스터링 방법과 기존 방법간의 성능 비교

2 절에서 소개한 방법론과 결측대체를 진행한 후에 K-means 방법을 적용하는 기존의 방법들을 비교하도록 한다. 결측대체 자료에 K-means 를 적용할 때는 10개의 초깃값을 주어 K-means 알고리듬이 초기치에 민감한 단점을 보완하고자 했다. 또한 Multiple Imputation 을 이용해 결측대체를 한 경우, 각 대체자료 셋에 K-means 를 적용 후, 결과를 voting 하는 방식을 택했다.

결측이 없는 Wine 데이터의 경우 ARI는 0.8975, Dunn index 는 0.2323이 얻었다. 또한 결측이 없었던 변수만으로 K-means 를 할 경우는 ARI 0.7310과 Dunn index 0.1769 를 얻었으므로, 의미있는 결과의 하한선이라 생각할 수 있다.

[표 2 : 클러스터링 성능 비교]

	Adjusted Rand Index (13%/27%)		Dunn Index (13%/27%)	
KSC	0.8351	0.8351	0.1769	0.1548
K-Pod	0.7921	0.7787	0.1601	0.1621
ClustImpute	0.8486	0.8323	0.1621	0.1620
평균대체	0.7763	0.7787	0.1520	0.1606
핫덱 대체	0.7517	0.7417	0.1619	0.1220
KNN 대체 (k=1)	0.8651	0.7440	0.1830	0.1325
KNN 대체 (k=15)	0.8551	0.8355	0.1830	0.1772
MI - pmm	0.8486	0.8483	0.2033	0.2323
MI - predict	0.8803	0.7550	0.2323	0.1539
MI - norm.nob	0.8487	0.7709	0.2033	0.1682
MI - RF	0.8806	0.8342	0.2323	0.1395

ARI 를 기준으로 KSC와 ClustImpute 의 경우에는 준수한 성능을 보임을 알 수 있지만 K-Pod 의 경우는 좋지 않았다. K-Pod 같은 경우에는 ClusterImpute와는 다르게, 한 번 잘못된 centroid 에 빠지면, 결측대체도 잘못된 방향으로 진행이 되고, 이 과정이 순환되면서 결과가 악화되는 것이라 예상해본다.

흥미로운 점은, 결측 비율이 작은 경우에 regression 방법을 이용한 방법론이 ARI와 dunn index 모두에서 성능을 높이는데 많은 도움을 주지만, 결측비율이 늘어나자 평균대체보다도 안 좋은 성능을 냈다는 점이다. 이는 클러스터마다 데이터의 특성이 다르고 결측 비율이 늘어날수록 3번 클러스터의 특성을 대변할 개체가 사라지는것이 원인이라 생각한다.

ARI 나 dunn index 모두에서 결측 비율이 작을 때는 k 파라미터를 1로 잡아도 준수한 성능을 보였으나, 결측비율이 늘어감에 따라서 k 를 크게 잡아줘야, 좀 더 좋은 성능을 기대할 수 있었다. 이는 결측 비율이 늘어날수록 3번 클러스터에 대응되는 객체가 줄어들기 때문이다.

4-3 결측대체 방법에 따른 bias, variance 비교

결측대체 후에 K-means 를 적용한 경우, 결측대체가 얼마나 적절하게 되었는지에 따라서 클러스터링 성능이 좌우 될 수 있다. 주의할 점은 클러스터링 구조를 고려해야 한다는 것이다. (클러스터를 무시하고 mean 의 bias 를 측정할 경우 평균대체에서 bias 가 0이 나오며, 이는 부적절하다). 따라서 클러스터에 따른 평균, 즉 클러스터의 centroid에 대한 bias와 within cluster variance의 예시로 V7 변수만 보도록 한다.

[표 3 : 결측 대체 방법에 따른 클러스터 중심에서의 MAE]

	Mean	KNN (k=1)	MI-PMM
Cluster1	0.082/0.175	0.027/0.052	0.004/0.029
Cluster2	0.099/0.096	0.08/0.081	0.004/0.007

	Mean	KNN (k=1)	MI-PMM
Cluster3	0.249/0.357	0.097/0.078	0.02/0.020

세번째 클러스터에서 평균대체 방법의 bias 가 특히 두드러지는 것을 확인 할 수 있었다. 예상대로 centroid 를 가장 잘 복원하는 결측대체법을 활용해야 클러스터링 성능도 개선될 수 있다는 사실을 알 수 있다.

[표 4 : 결측 대체 방법에 따른 클러스터 내 분산 추정]

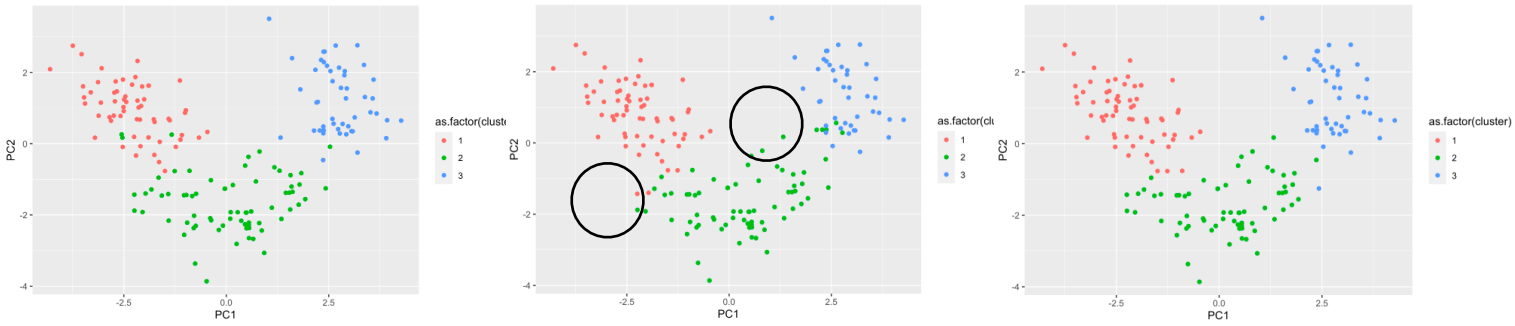
	TRUE	Mean	KNN (k=1)	MI-PMM
Cluster1	0.2933	0.3312/0.3513	0.3064/0.3520	0.3319/0.3503
Cluster2	0.7593	0.7062/0.5739	0.8232/0.8435	0.8879/0.8190
Cluster3	0.3253	0.4408/0.4531	0.4039/0.4198	0.3136/0.3155

모든 방법론에서 within cluster variance 가 과평가 된 면이 있었었다. 두 번째 클러스터에 대해서는 KNN 과 MI-PMM 방법이 가장 과평가 되었고, 세 번째 클러스터에 대해서는 평균 대체법이 가장 과평가 되었다.

4-4 클러스터링 결과 시각화

클러스터링 결과는 PCA 를 통해 변수를 차원축소 함으로서 2차원 평면에 나타낼 수 있다. [그림 5] 에서 볼 수 있듯, 평균 대체의 경우, 클러스터의 경계면을 잘 구분하지 못하는 것을 확인 할 수 있다.

[그림 5 : 클러스터링 시각화 예시]



실제 클러스터 **평균 대체** **MI - UMM**

5. 결론

본 프로젝트에서는 클러스터 자체가 결측에 영향을 주는 데이터 셋에 한하여 실험을 진행했다. 그 결과, PMM 을 이용한 결측대체 방법을 이용한 후, 후에 voting 방식으로 클러스터 레이벨을 추론하는 것을 가장 추천한다. ARI와 dunn index 모두에서 결측비율 상관없이 가장 성능이 좋은 편에 속했기 때문이다. 또한 KNN 을 이용한 결측 대체 방법도 추천할만 하다. 다만 결측비율이 늘어감에 따라서 k 하이퍼파라미터를 상향해줘야 할것이다. PMM 과 KNN 모두 가장 가까운 개체에 의존하여 대체값을 찾는 방법이란 측면에서, K-means에서의 유사한 객체끼리 클러스터링 되는 방법과 비슷한 측면이 있어 성능이 좋았을 것이라 예상한다.

또한 결측대체를 사전에 진행하지 않는 방법에 대해서는 ClustImpute 는 추천하나 KSC는 추천하지 않는다. w라는 하이퍼 파라미터를 결정하는 문제가 결코 쉽지 않기 때문이다. 레이벨이 있는 일부 데이터를 가지고 w를 추론할 수도 있겠지만, 그런 정보가 가능한 상황은 실전에서 쉽게 마주하기 쉽지않을 것이다. 또한 더 가장 작은 KSC의 목적함수값을 산출하는 초깃값이 반드시 최선의 초깃값이 아니었던 문제가 있기 때문에, 최적의 초깃값을 찾기가 어려운 문제가 있다.

[참고문헌]

[1] Wagstaff K. (2004) Clustering with Missing Values: No Imputation Required. In: Banks D., McMorris F.R., Arabie P., Gaul W. (eds) Classification, Clustering, and Data Mining Applications. Studies in Classification, Data Analysis, and Knowledge Organisation. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-17103-1_61

[2] Chi, Jocelyn & Chi, Eric & Baraniuk, Richard. (2015). k -POD A Method for k -Means Clustering of Missing Data. The American Statistician. 70. 1-29. 10.1080/00031305.2015.1086685.

[3] Pfaffel, Oliver. (2020). CLUSTIMPUTE: AN R PACKAGE FOR K-MEANS CLUSTERING WITH BUILD-IN MISSING DATA IMPUTATION. 10.13140/RG.2.2.20143.36007.