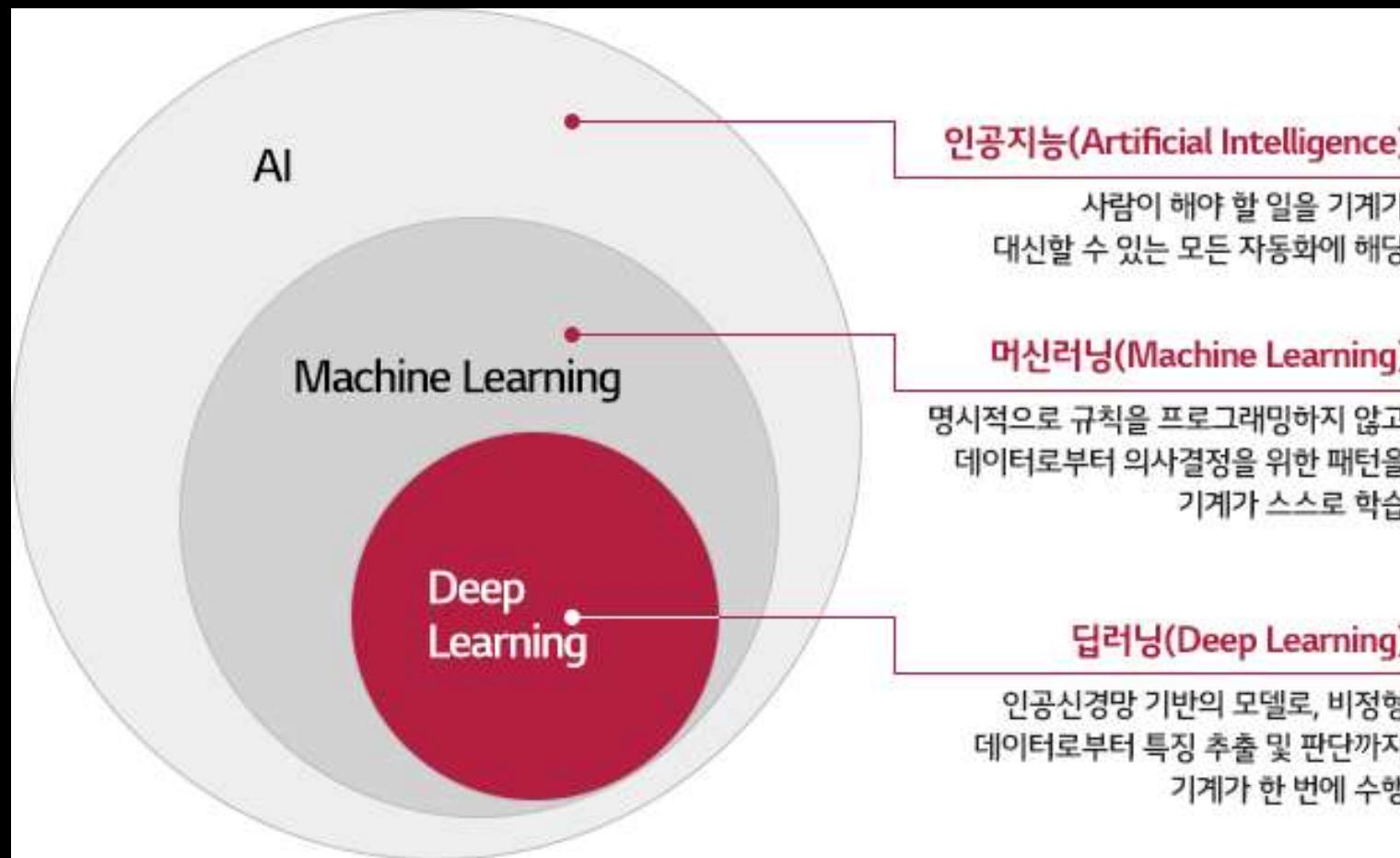


머신러닝 기법을 활용한 글로벌 가치사슬(GVC) 분석 : 재화 중심으로

20210118
김승준
전산학부
HSS310(C)

AI Basic



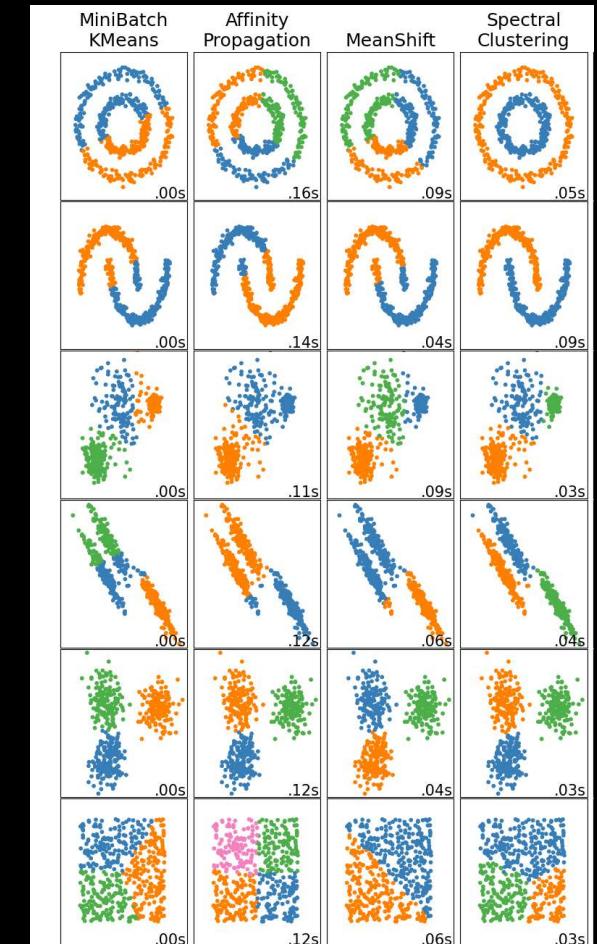
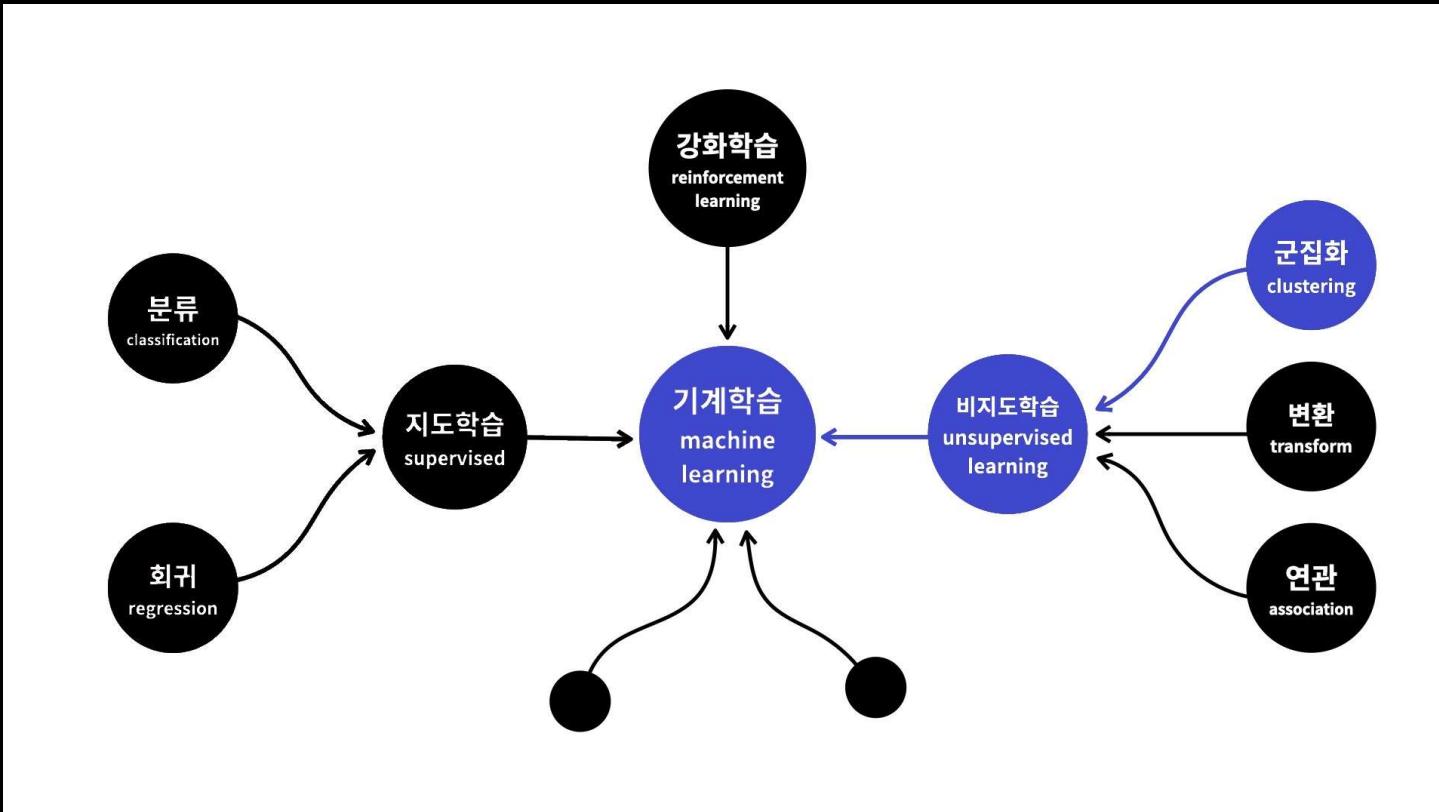
AI Basic

d차원

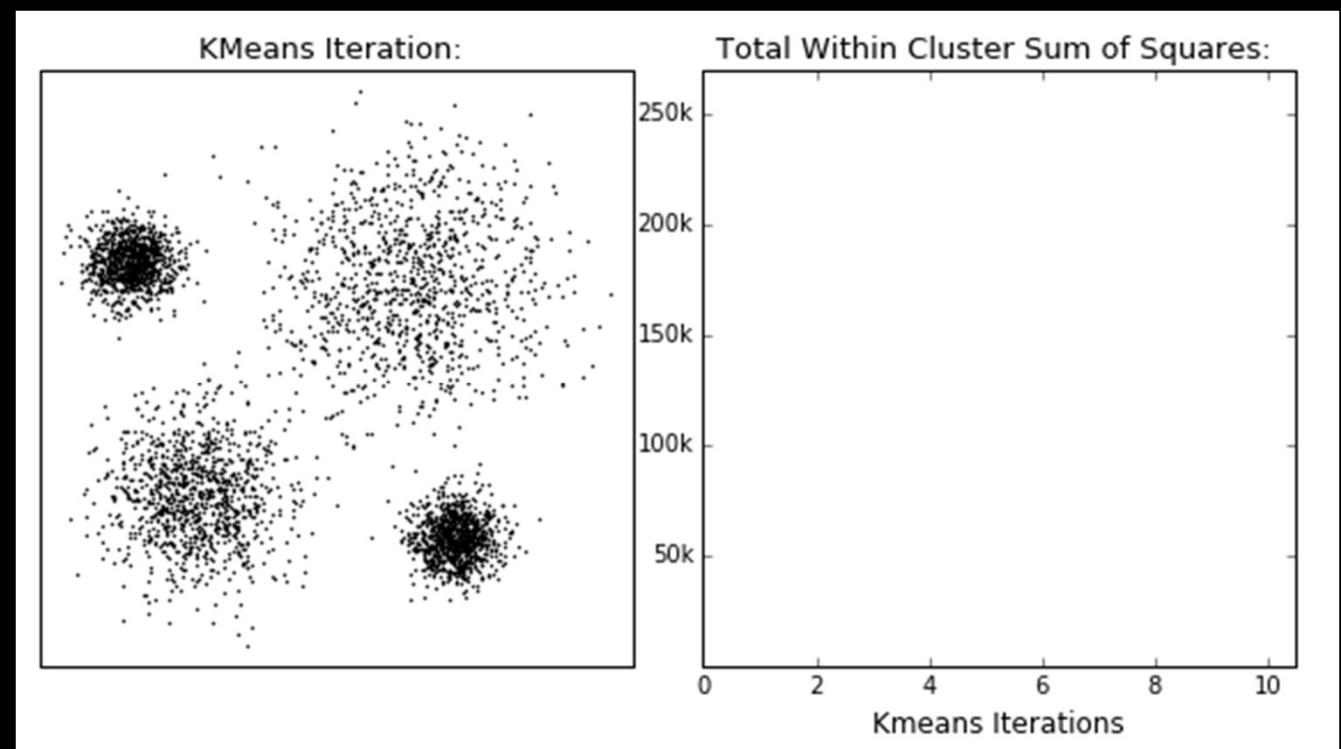
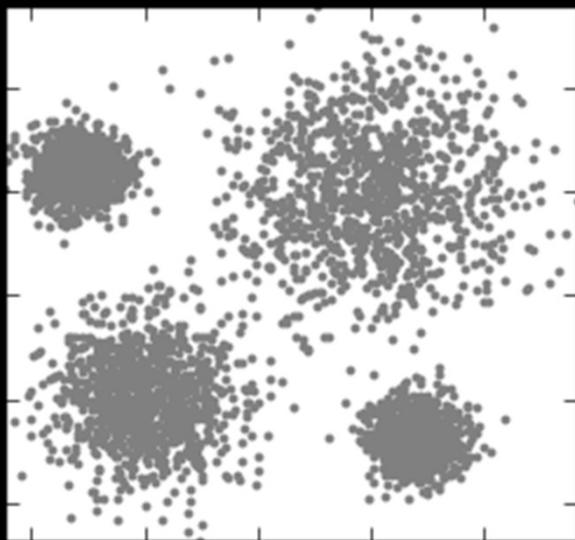
$x_1(1)$	$x_1(2)$...	$x_0(d-1)$	$x_0(d)$
$x_2(1)$	$x_2(2)$...	$x_2(d-1)$	$x_2(d)$
...
$x_{n-1}(1)$	$x_{n-1}(2)$...	$x_{n-1}(d-1)$	$x_{n-1}(d)$
$x_n(1)$	$x_n(2)$...	$x_n(d-1)$	$x_n(d)$

n개의 데이터

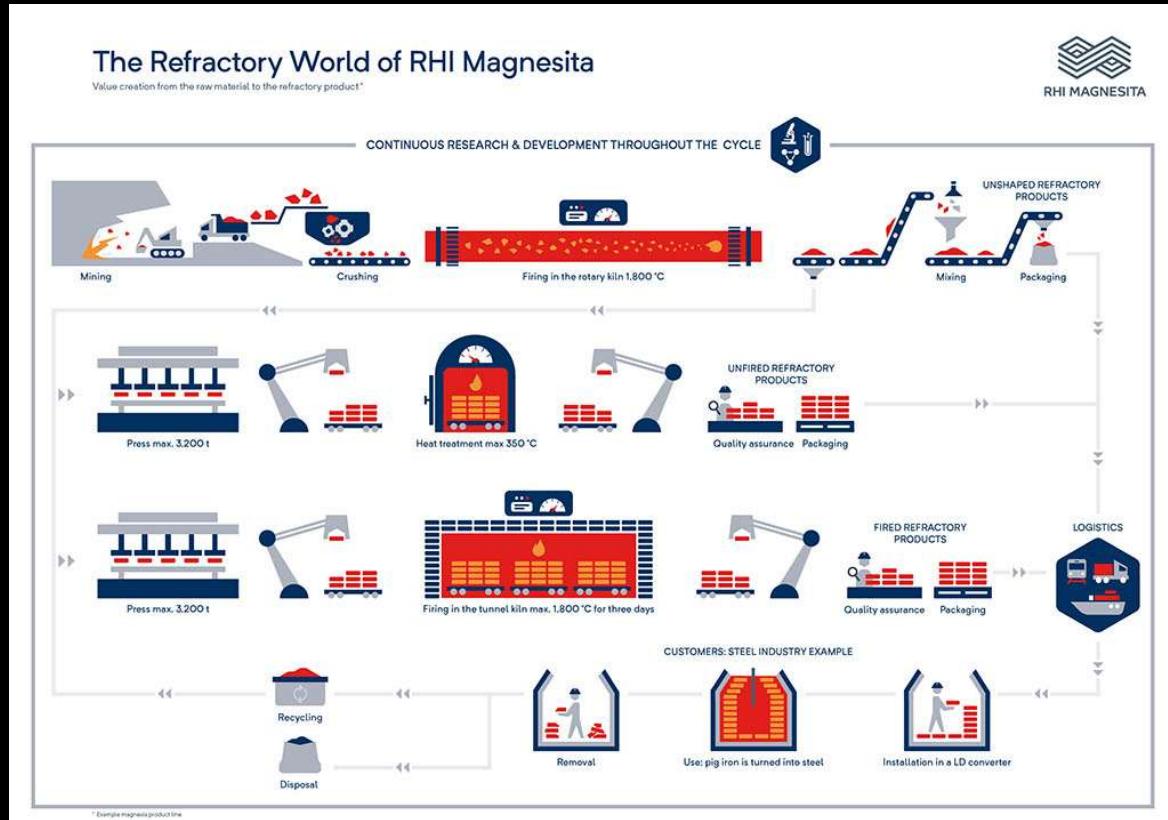
AI Basic



AI Basic



Value chain



기업 : 부가가치 창출, 경쟁 우위를 위해

Value chain

우리 :

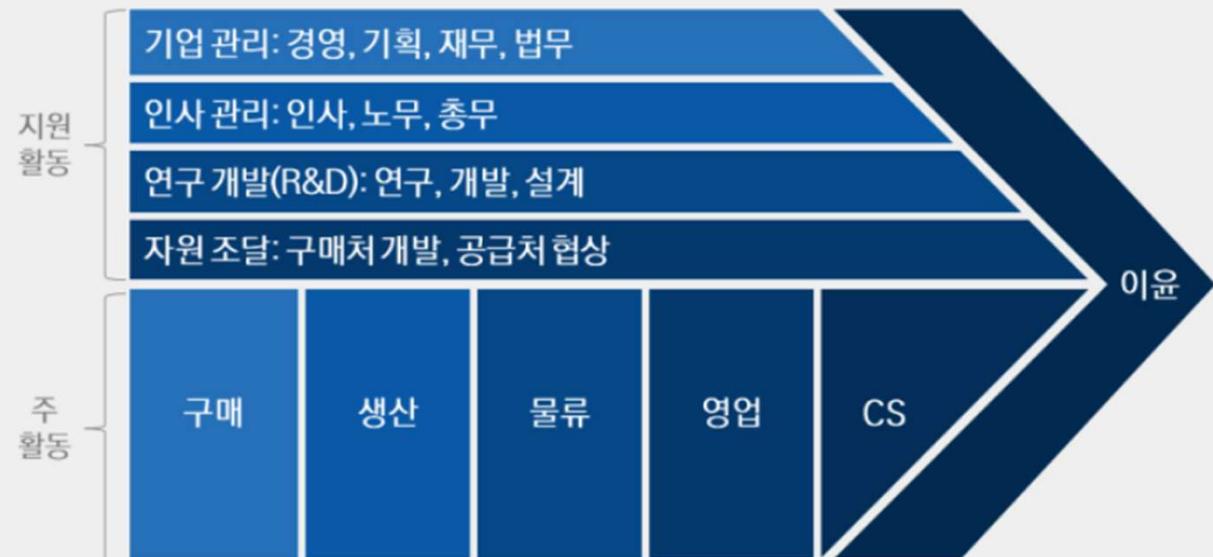
전기차 산업의 테슬라



Value chain analysis

가치사슬 분석(Value Chain Analysis)

가치사슬 분석 프레임워크는 기업의 활동을 '사슬'처럼 연결하여 시각화 함으로써, 자사가 어떻게 가치를 창출하고 어떤 프로세스에 초점을 두는지를 파악함으로써 기업이 각 활동이 어떻게 사업에 공헌하는지를 파악할 수 있는 도구다.



CAREERNERS

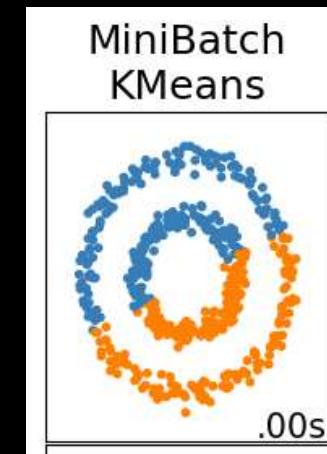
Basis

가정1. 인과관계가 강한 한 사슬 내부의 재화는 무역 금액 변화율에 대해 가까운 distance 가진다.

가정2. 모든 features가 방향이 비슷하므로 hyperplane만으로 클러스터링이 가능하다.

가정3. 원자재, 중간재, 완성재, 그리고 그 중간 서비스나 보완재는 가치를 더하는 순서대로 distance가 점점 짧아진다
(완성재 -> 중간재 -> 원자재 순 인과관계 크기, 맞는 가정인가?).

-> 간단하고, 계산 속도가 빠르고, 클러스터 개수를 직접 지정 가능한 Kmeans Algorithm 채택



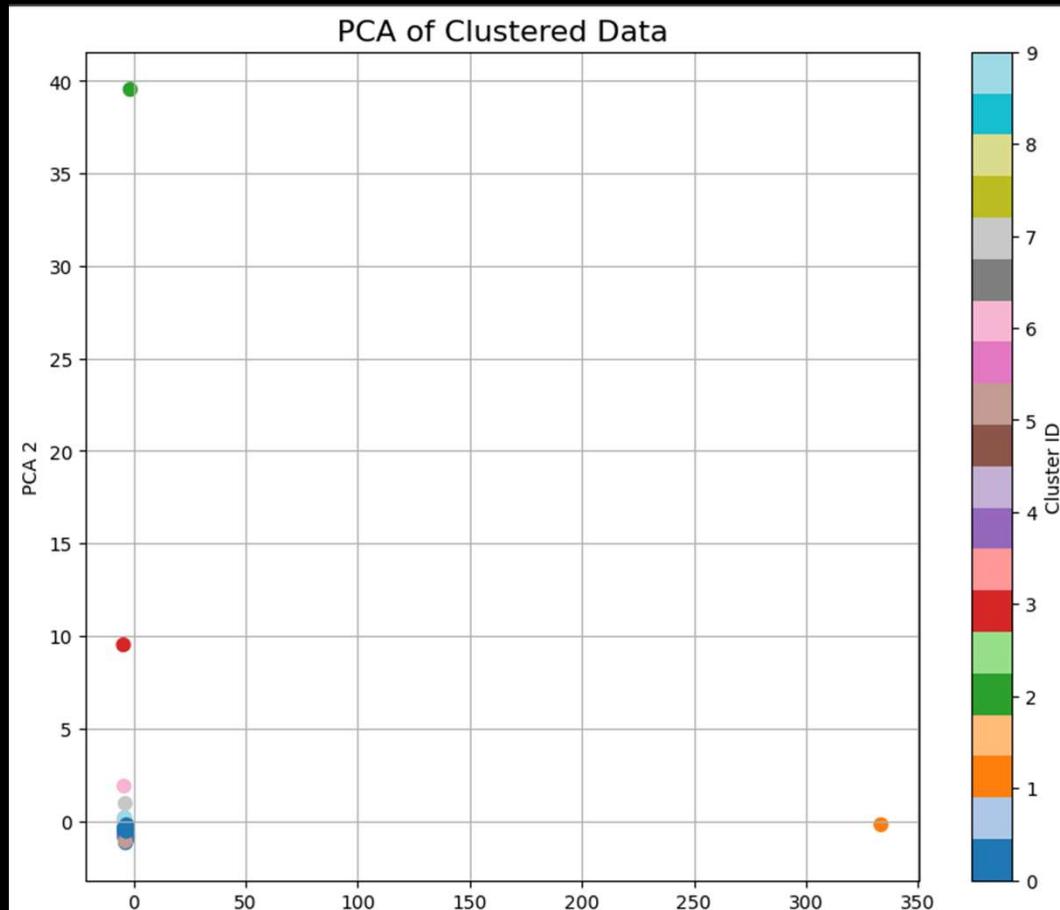
Data Source

1. 관세청 수출입무역통계, 원자재, 중간재, 완성재 균형 맞춰 100개 선정

The screenshot shows the homepage of the Korea Customs Service's Trade Statistics website. The top navigation bar includes links for '한눈에 보는 무역통계' (Trade Statistics at a Glance), 'ENG', and a search bar. Below the navigation is a horizontal menu with tabs: '수출입통계' (Export-Import Statistics), '테마별 무역통계' (Thematic Trade Statistics), '수출입물류통계' (Trade Logistics Statistics), '관세행정통계' (Customs Administration Statistics), '사용자지원' (User Support), and '무역통계소개' (Introduction to Trade Statistics). The main content area is titled '수출입 실적' (Export-Import Performance) and displays a table of trade data. The table has columns for 품목명 (Item Name), HS코드 (HS Code), 기간 (Period), 품목 (Item), 수출 (Exports), 수입 (Imports), and 무역수지 (Trade Balance). The data is presented in a grid format with various filters and sorting options available.

품목명	HS코드	기간	품목		수출		수입		무역수지
			HS코드	품목명	중량	금액	중량	금액	
석유와 역청유(瀝青油)	2709	총계	4.8	17	409,448,696.2	277,456,354	-277,456,337		
석유와 역청유(瀝青油)	2709	2022.01	0.0	0	12,619,193.5	7,791,770	-7,791,770		
석유와 역청유(瀝青油)	2709	2022.02	0.0	0	10,563,730.3	7,259,431	-7,259,431		
석유와 역청유(瀝青油)	2709	2022.03	1.6	6	11,469,770.9	8,466,710	-8,466,704		
석유와 역청유(瀝青油)	2709	2022.04	0.0	0	11,492,668.5	9,594,652	-9,594,652		
석유와 역청유(瀝青油)	2709	2022.05	0.0	0	10,839,659.1	8,945,185	-8,945,185		
석유와 역청유(瀝青油)	2709	2022.06	2.7	2	9,024,140.5	9,552,251	-9,552,240		

Preprocessing(0)



Preprocessing(1) : Data load, feature extract

Feature data : 물품코드, 분기별 수입+수출 금액

The screenshot shows a Google Colab notebook titled "dataprocess.ipynb". The left panel displays the code cells and their outputs, while the right panel shows the resulting data frames.

Code Cells:

- [1] import pandas as pd
- [2] from google.colab import drive
drive.mount('/content/drive')
- [3] data1=pd.read_csv('/content/drive/MyDrive/valuechainproject/csv1.csv')
- [4] data1

Data Frame (Left):

	기간	HS코드	품목명	수출 중량	수출 금액	수입 중량	수입 금액	무역수지	
0	총계	Nan		NaN	4745154.0	18700760	1.213145e+09	569136316	-550435555
1	2022-01	1001.0	밀과 메슬린(meslin)	0.0	0	3.290137e+05	118122	-118122	
2	2022-01	1005.0	옥수수	0.3	3	1.019851e+06	337687	-337685	
3	2022-01	1201.0	대두(부수었는지에 상관없다)	0.2	2	1.277796e+05	81692	-81690	
4	2022-01	1507.0	대두유와 그 분획물(정제했는지에 상관없이며 화학적으로 변성 가공한 것은 제외한다)	218.1	444	3.523420e+04	48360	-47916	
...	
716	2024-12	7106.0	은(금이나 백금을 도금한 은을 포함하며, 가공하지 않은 것·반가공한 모양이나 가루 ...)	236.1	165111	6.480000e+01	15519	149591	
717	2024-12	7108.0	금(백금을 도금한 금을 포함하며, 가공하지 않은 것·반가공한 모	1.8	99514	1.800000e+00	137691	-38178	

Data Frame (Right):

	기간	계
0	2022Q1	1327.000000
1	2022Q2	1345.000000
2	2022Q3	1059.000000
3	2022Q4	614.000000
4	2023Q1	691.000000
...
1173	2023Q4	4156.333333
1174	2024Q1	3308.666667
1175	2024Q2	3470.666667
1176	2024Q3	3671.000000
1177	2024Q4	4160.333333

1178 rows × 3 columns

Preprocessing(2) : Delete outlier

데이터 빠져있는 물품, |변화율|>5 물품 제거

```
[29] filtered_df['분기별 상승률'] = (
    filtered_df.groupby('품목명')[['계']]
    .transform(lambda x: x.pct_change())
)

>>> <ipython-input-29-dacd9a649870>:1: SettingWithCopyWarning:
      A value is trying to be set on a copy of a slice from a DataFrame.
      Try using .loc[row_indexer,col_indexer] = value instead
      See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-
      filtered_df['분기별 상승률'] = (
[30] filtered_df
```

	품목명	기간	계	분기별 상승률
0	겹붙인 종이와 판지[접착제로 겹붙인 것으로서 를 모양이나 시트(sheet) 모양으로... 로...	2022Q1	1327.000000	NaN
1	겹붙인 종이와 판지[접착제로 겹붙인 것으로서 를 모양이나 시트(sheet) 모양으로... 로...	2022Q2	1345.000000	0.013564
2	겹붙인 종이와 판지[접착제로 겹붙인 것으로서 를 모양이나 시트(sheet) 모양으로... 로...	2022Q3	1059.000000	-0.212639
3	겹붙인 종이와 판지[접착제로 겹붙인 것으로서 를 모양이나 시트(sheet) 모양으로... 로...	2022Q4	614.000000	-0.420208
4	겹붙인 종이와 판지[접착제로 겹붙인 것으로서 를 모양이나 시트(sheet) 모양으로... 로...	2023Q1	691.000000	0.125407
...
1173	휴대용 전등(긴전지·축전지·자석발전기와 같은 자체 전원기능을 갖춘 것으로 한정하며,...	2023Q4	4156.333333	0.118597

```
import numpy as np

# '2022Q1'을 제외한 분기 컬럼만 선택
cols_to_check = pivot_df.iloc[:, 2:].columns

# inf를 NaN으로 먼저 바꾸고, '2022Q1' 제외한 나머지에 NaN이 있는 행 제거
clean_pivot_df = pivot_df.replace([np.inf, -np.inf], np.nan)
clean_pivot_df = clean_pivot_df.dropna(subset=cols_to_check)

# 결과 확인
print(clean_pivot_df.shape)
clean_pivot_df.head()
pivot_df=clean_pivot_df

mask = (pivot_df[cols_to_check].abs() > 5).any(axis=1)

pivot_df=pivot_df[~mask]
```

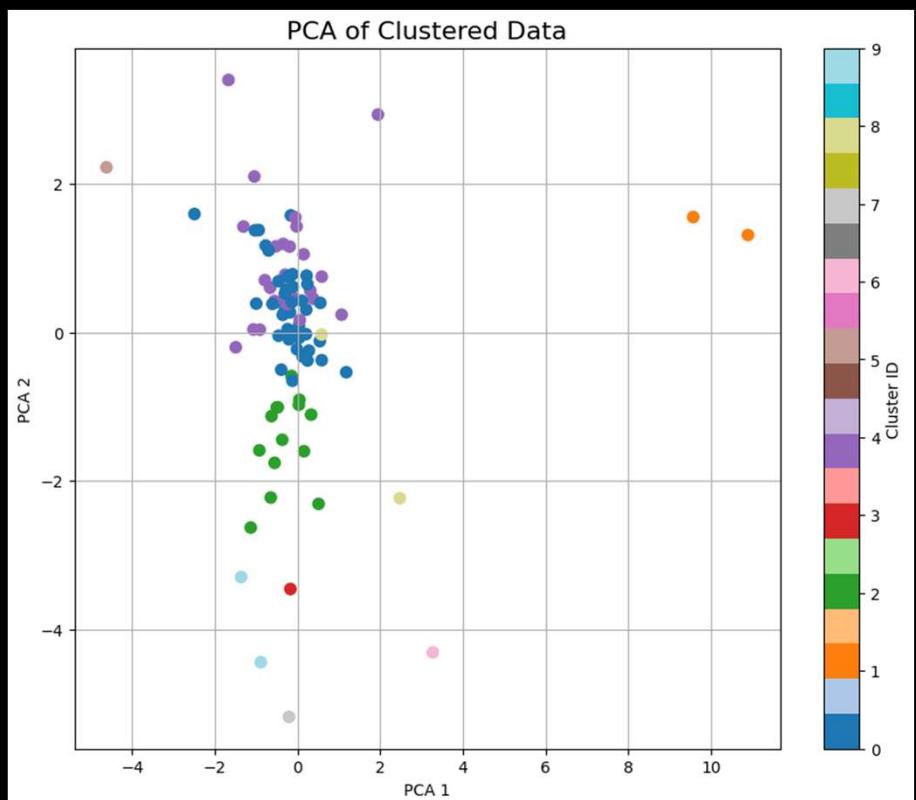
Preprocessing(2) : Normalize

Z-score

```
[46] from sklearn.preprocessing import StandardScaler  
  
# 수치형 변화를 열만 선택 (첫 번째 열은 품목명이라 제외)  
value_columns = pivot_df.columns[2:]  
  
# 스케일러로 z-score 정규화  
scaler = StandardScaler()  
pivot_df[value_columns] = scaler.fit_transform(pivot_df[value_columns])  
  
# 결과 확인  
pivot_df.head()
```

Vanila clustering

```
y_pred1 = KMeans(n_clusters=10,  
**common_params).fit_predict(feature_tensor)
```



```
# 'pivot_df'의 첫 번째 열이 '품목명'이라고 가정  
품목명_series = pivot_df.iloc[:, 1]  
  
# 클러스터 결과를 불이기  
result_df = pd.DataFrame({  
    '품목명': 품목명_series,  
    '클러스터': y_pred1  
})  
  
# 클러스터별 품목 리스트 출력  
for cluster_id, group in result_df.groupby('클러스터'): # 총 {len(group)}개 품목)  
    print(f"◆ 클러스터 {cluster_id} (총 {len(group)}개 품목)")  
    for name in group['품목명']: # 품목명)  
        print(f" - {name}")
```

◆ 클러스터 0 (총 43개 품목)
- 겹불인 종이와 판지[접착제로 겹불인 것으로서 틀 모양이나 시트(sheet) 모양으로 한정하며, 표면에 광택유와 광택유 다발, 제8544호의 것 외의 광택유 케이블, 편광재료(polarizing material)로 만든 것
- 그 밖의 도포하지 않은 종이와 판지[틀 모양이나 시트(sheet) 모양으로 한정하며, 이 류의 주제는 그 밖의 무기산과 무기 비(非)금속 산화물
- 그 밖의 차량(추진용 전동기만을 갖춘 것)
- 그 밖의 훈용 기기와 가스마스크(기계적인 부분풀과 교환용 필터를 모두 갖추지 않은 보호용 대형 기기)
- 기계요법용 기기, 마사지용 기기, 심리학적 적성검사용 기기, 오존 흡입기·산소 흡입기·에어로내과용·외과용·치과용·수의과용 기기[신파이그라피(scintigraphic)식 진단기기·그 밖의 전기식 디나멜의 괴(塊)
- 따로 분류되지 않은 조제 식료품
- 망간과 그 제품[웨이스트(waste)와 스크랩(scrap)을 포함한다]
- 면[카드(card)하지도 코움(comb)하지도 않은 것으로 한정한다]
- 물결 모양으로 하거나(corrugated)(평면지가 불은 것인지에 상관없다) 주름지거나(creped) 구겨지거나
- 반도체 디바이스(예: 다이오드·트랜지스터·반도체 기반 트랜스듀서), 갈광성 반도체 디바이스
- 불꽃점화식이나 압축점화식 내연기관의 점화용·시동용 전기기기(예: 점화용 자석발전기·자석발전기·석유가스와 그 밖의 가스 상태의 탄화수소
- 실험실용·화학용이나 그 밖의 공업용 도자제품, 농업용 도자제 통과 이와 유사한 용기, 물품의 알루미늄의 괴(塊)
- 엔스선이나 양파선·비단선·갓마선·그 밖의 저전선을 사용하는 기기(내과용·외과용·치과용·

Defining chain order

최근접 탐색(Greedy Nearest Neighbor Traversal)

```
from scipy.spatial.distance import cdist
import numpy as np

# 품목명과 feature_tensor를 담은 전체 result_df
result_df = pd.DataFrame({
    '품목명': pivot_df.iloc[:, 1].values,
    '클러스터': y_pred
})

# feature_tensor는 품목명 순서와 같은 순서라고 가정
result_df['feature'] = list(feature_tensor)

# 체인 순서 할수 정의
def build_chain_order(cluster_features, cluster_names):
    n = len(cluster_features)
    distances = cdist(cluster_features, cluster_features, metric='euclidean')
    np.fill_diagonal(distances, np.inf) # 자기 자신은 제외

    visited = [False] * n
    chain = []

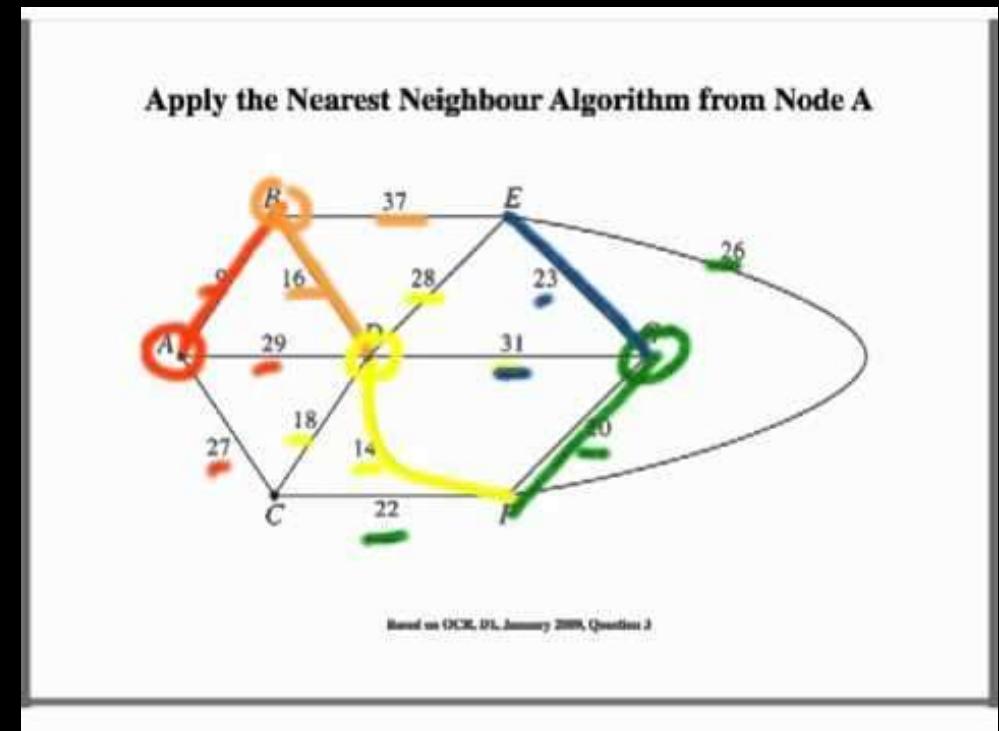
    # 시작은 가장 가까운 쌍 중 하나
    i, j = np.unravel_index(np.argmin(distances), distances.shape)
    chain.append([i, j])
    visited[i] = visited[j] = True

    while len(chain) < n:
        last = chain[-1]
        dists = distances[last]
        for idx in np.argsort(dists):
            if not visited[idx]:
                chain.append(idx)
                visited[idx] = True
                break
    return [cluster_names[i] for i in chain]

# 클러스터별 체인 출력
for cluster_id, group in result_df.groupby('클러스터'):
    cluster_names = group['품목명'].tolist()
    cluster_features = np.stack(group['feature'].values)

    chain_order = build_chain_order(cluster_features, cluster_names)

    print(f"\n◆ 클러스터 {cluster_id} 체인 순서 (총 {len(chain_order)}개)")
    for i, name in enumerate(chain_order):
        print(f"\t{i+1}: {name}")
```



Cluster Analysis(1) : Same produce process

클러스터0

5. 종이, 판지, 셀룰로오스워딩, 셀룰로오스섬유의 웹(web)
12. 한 면이나 양면을 도포한 종이와 판지
22. 원목
26. 화장지 · 안면용 티슈용 원지, 타월 · 낱킨용 원지와 이와 유사한 가정용이나 위생용 종이,
셀룰로오스워딩 · 셀룰로오스섬유의 웹(web)
30. 그 밖의 도포하지 않은 종이와 판지
31. 겹붙인 종이와 판지
33. 카본지, 셀프복사지, 그 밖의 복사지나 전사지
39. 물결 모양으로 하거나(corrugated)(평면지가 붙은 것인지에 상관없다) 주름지거나(creped) 구겨지거나(crinkled) 올록볼록하거나(embossed) 구멍을 뚫은(perforated) 종이와 판지
42. 면

클러스터0

4. 철이나 비합금강의 평판압연제품[폭이 600밀리미터 미만인 것]
7. 철이나 비합금강의 그 밖의 봉
14. 철이나 비합금강의 평판압연제품[폭이 600밀리미터 이상인 것]
24. 은(금이나 백금을 도금한 은을 포함하며, 가공하지 않은 것 · 반가공한 모양이나 가루 모양인 것으로 한정한다)
29. 철이나 비합금강의 형강(形鋼)
34. 철이나 비합금강의 그 밖의 봉[단조(鍛造) · 열간(熱間)압연 · 열간인발(熱間引拔) · 열간(熱間)압출보다 더 가공하지 않은 것으로 한정하고, 압연 후 꼬임가공된 것을 포함한다]
43. 니켈의 괴(塊)

Cluster Analysis(2) : Supply Chain + 보완재

1. 자주식(自走式) 철도용이나 궤도용 객차와 화차(제8604호의 것은 제외한다)

2. 철강으로 만든 구조물(제9406호의 조립식 건축물은 제외한다)과 구조물의 부분품

3. 실린더용량이 3,000시시를 초과하는 것
4. 이 류에 따로 분류되지 않은 기계류(고유의 기능을 가진 것으로 한정한다)
5. 전기식(전기발열에 따른 가스식을 포함한다) · 레이저나 그 밖의 광선식 · 광자빔식 · 초음파식 · 전자빔식 · 자기펄스(magnetic pulse)식 · 플라즈마 아크(plasma arc)식 납땜용 · 땜질용이나 용접용 기기금속이나 서멧(cermet)의 가열분사용 전기식 기기
6. 실린더용량이 1,500시시 초과 3,000시시 이하인 것
7. 전기기계식의 가정용 기기(전동기를 갖춘 것으로 한정하며, 제8508호의 진공청소기는 제외한다)
9. 그 밖의 차량[압축점화식 피스톤 내연기관(디젤이나 세미디젤)과 추진용 모터로서의 전동기를 둘 다 갖춘 것으로서]
11. 휴대용 전등(건전지 · 축전지 · 자석발전기와 같은 자체 전원기능을 갖춘 것으로 한정)
12. 공업용이나 실험실용 전기식 노(爐)와 오븐과 그 밖의 공업용이나 실험실용의 전자유도식이나 유전손실(dielectric loss)식 가열기
13. 알루미늄광과 그 정광(精礶)
14. 발전세트와 회전변환기

Cluster Analysis(3) : Unbalanced Clusters

- 클러스터 0 체인 순서 (총 43개)
- 클러스터 1 체인 순서 (총 2개)
- 클러스터 2 체인 순서 (총 14개)
- 클러스터 3 체인 순서 (총 2개)
- 클러스터 4 체인 순서 (총 25개)
- 클러스터 5 체인 순서 (총 2개)
- 클러스터 6 체인 순서 (총 2개)
- 클러스터 7 체인 순서 (총 2개)
- 클러스터 8 체인 순서 (총 2개)
- 클러스터 9 체인 순서 (총 2개)

표준편차 : 14.049

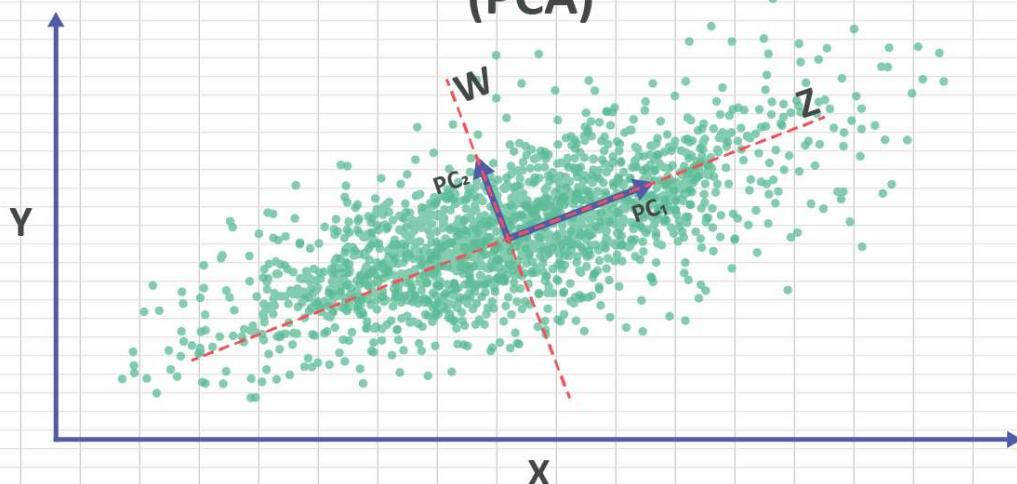
원인 : Outlier data

- 클러스터 0 체인 순서 (총 43개)
 1. 따로 분류되지 않은 조제 식료품
 2. 전기식 조명용이나 신호용 기구(제8539호의 물품은 제외한다) · 원드스크
 3. 합성필라멘트사(재봉사와 소매용은 제외하며, 67데시텍스 미만인 합성모
 4. 철이나 비합금강의 평판압연제품[폭이 600밀리미터 미만인 것으로 한정하
 5. 종이, 판지, 셀룰로오스워딩, 셀룰로오스섬유의 웹(web)[크기와는 관계
 6. 기계요법용 기기, 마사지용 기기, 심리학적 적성검사용 기기, 오존 흡입
 7. 철이나 비합금강의 그 밖의 봉
 8. 알루미늄의 괴(塊)
 9. 팔섬유와 팔섬유 다발, 제8544호의 것 외의 팔섬유 케이블, 편광재료(po
 10. 텔레비전 수신기기를 갖추지 않은 모니터와 프로젝터, 텔레비전 수신용
 11. 실험실용 · 화학용이나 그 밖의 공업용 도자제품, 농업용 도자제 통과 이
 12. 한 면이나 양면을 도포한 종이와 판지[결합재가 있는지에 상관없으며 고
 13. 황산지 · 내지지(耐脂紙) · 트레이싱지(tracing paper) · 글라신지(glassin
 14. 철이나 비합금강의 평판압연제품[폭이 600밀리미터 이상인 것으로서 열간
 15. 합성모노필라멘트(67데시텍스 이상인 것으로서 횡단면의 치수가 1밀리미
 16. 내과용 · 외과용 · 치과용 · 수의과용 기기[신티그래픽(scintigraphic)식 진
 17. 전기식의 즉시식 · 저장식 물가열기와 투입식 가열기, 난방기기와 토양가
 18. 재생 · 반(半)합성 필라멘트사[재봉사와 소매용은 제외하며, 67데시텍스]

Refined Kmeans : PCA+Silhouette score

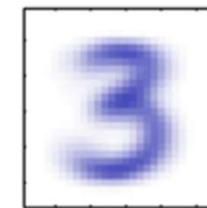
1.PCA(Principal Component Analysis)

Principal Component Analysis (PCA)

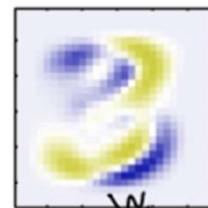


- The mean vector \bar{x} and the first 4 principal components:

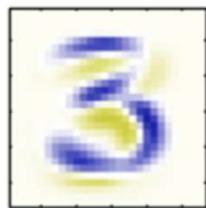
Mean



$\lambda_1 = 3.4 \cdot 10^5$



$\lambda_2 = 2.8 \cdot 10^5$



$\lambda_3 = 2.4 \cdot 10^5$



$\lambda_3 = 2.4 \cdot 10^5$

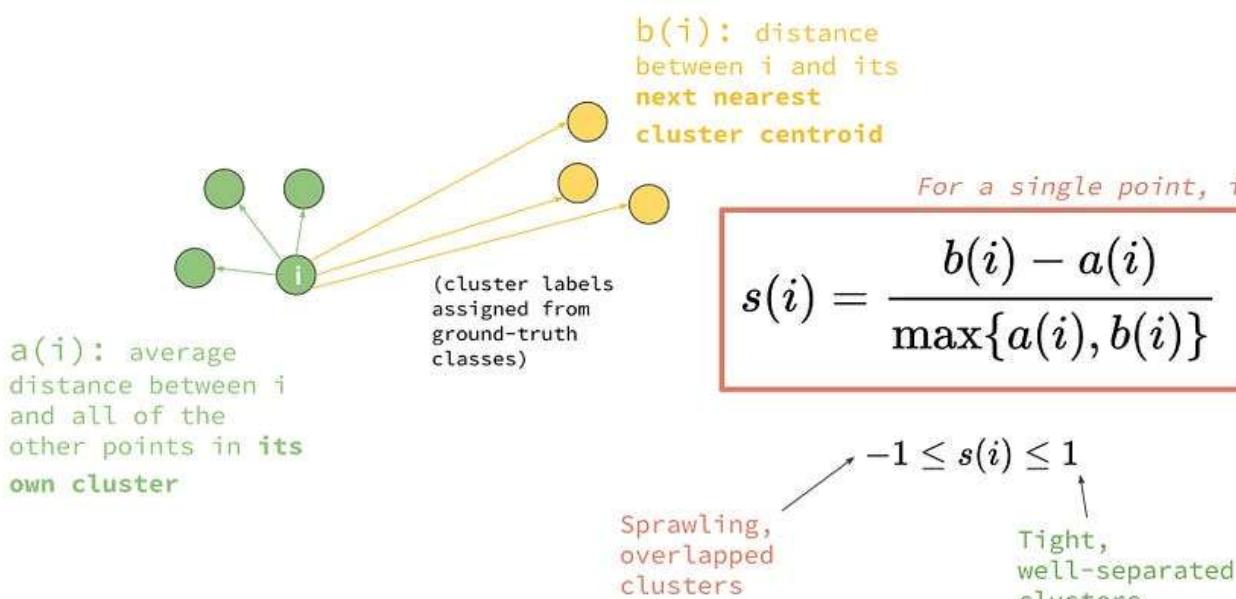


$\lambda_4 = 1.6 \cdot 10^5$



Refined Kmeans : PCA+Silhouette score

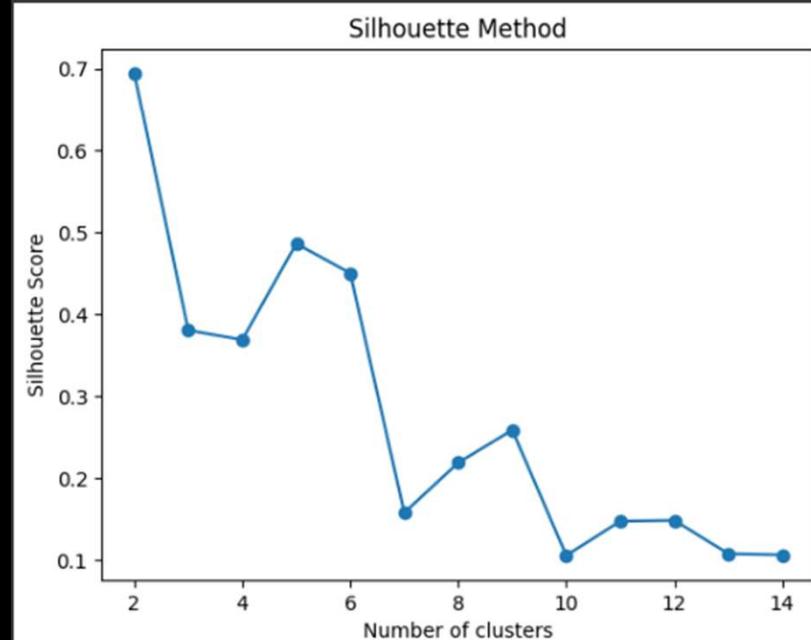
2. Silhouette score



```
X_pca2 = pca2.fit_transform(feature_tensor)
from sklearn.metrics import silhouette_score

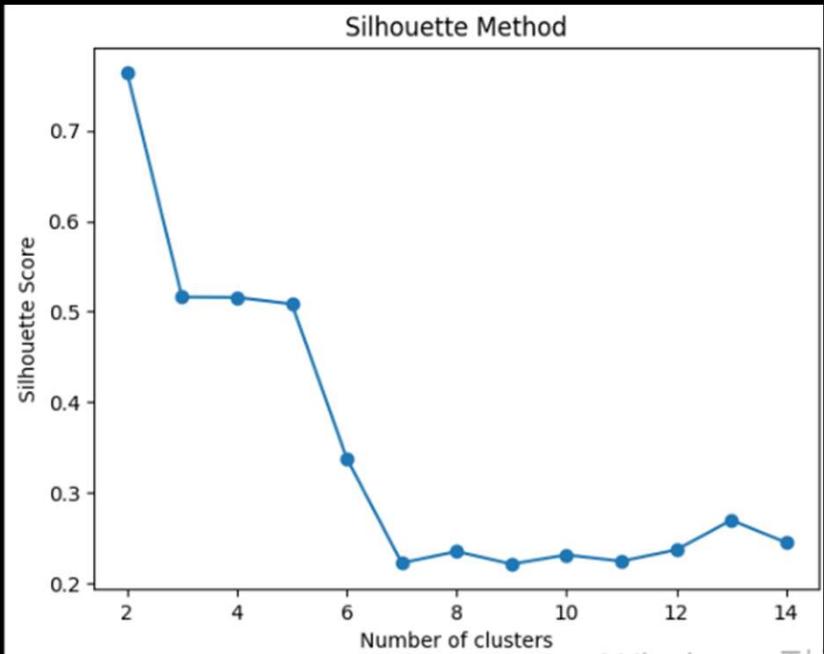
scores = []
K_range = range(2, 15) # silhouette는 k>=2에서 정의됨
for k in K_range:
    labels = KMeans(n_clusters=k, **common_params).fit_predict(X_pca2)
    score = silhouette_score(X_pca2, labels)
    scores.append(score)

plt.plot(K_range, scores, marker='o')
plt.xlabel('Number of clusters')
plt.ylabel('Silhouette Score')
plt.title('Silhouette Method')
plt.show()
```

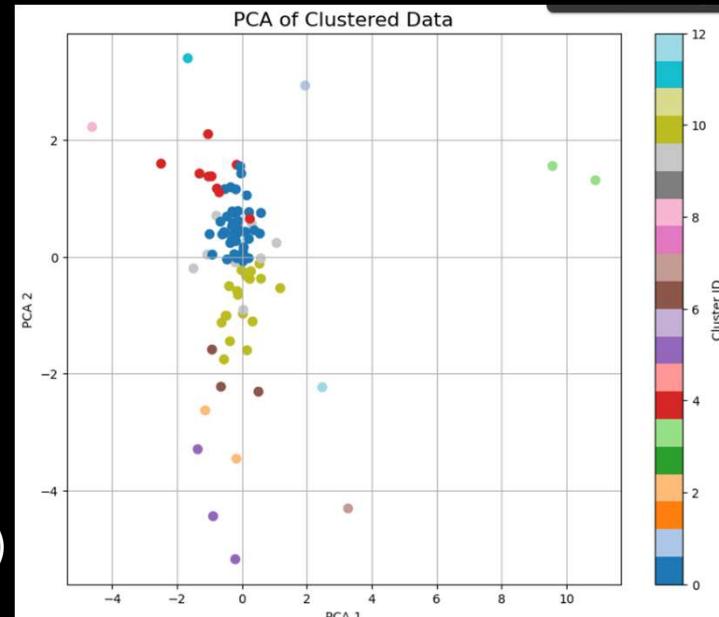


Refined Kmeans

PCA compo 4, 13 clusters



- ◇ 클러스터 0 체인 순서 (총 42개)
 - ◇ 클러스터 1 체인 순서 (총 2개)
 - ◇ 클러스터 2 체인 순서 (총 2개)
 - ◇ 클러스터 3 체인 순서 (총 2개)
 - ◇ 클러스터 4 체인 순서 (총 9개)
 - ◇ 클러스터 5 체인 순서 (총 3개)
 - ◇ 클러스터 6 체인 순서 (총 3개)
 - ◇ 클러스터 7 체인 순서 (총 2개)
 - ◇ 클러스터 8 체인 순서 (총 2개)
 - ◇ 클러스터 9 체인 순서 (총 8개)
 - ◇ 클러스터 10 체인 순서 (총 18개)
 - ◇ 클러스터 11 체인 순서 (총 2개)
 - ◇ 클러스터 12 체인 순서 (총 2개)
- 표준편차 : 11.384



한국 산업 현황

2022년 세계 수출/수입시장 점유율 상위 10개국



*자료: 국제통화기금(IMF), 한국무역협회
그래픽: 윤선정 디자인기자

세계의 가치 부가 비중 따라가지 않을 가능성 있음

GDP 대비 서비스산업 비중

단위: %, 2021년 기준

OECD 평균 71▼



자료: 세계은행

The JoongAng

Data Source

2. UN Comtrade Trade Data, 완성재, 중간재, 원자재 균형 맞춰 300개 재화 선정

The screenshot shows the UN Comtrade Database search interface. At the top, there are navigation links: Data, Data Availability, Metadata, Data Tools, Help Center, My Comtrade Free, and Logout. Below the header, a message says: "Information: Please select variables from each of the following sections to view the report. You can select from previously saved searches from 'My Saved Searches' dropdown".

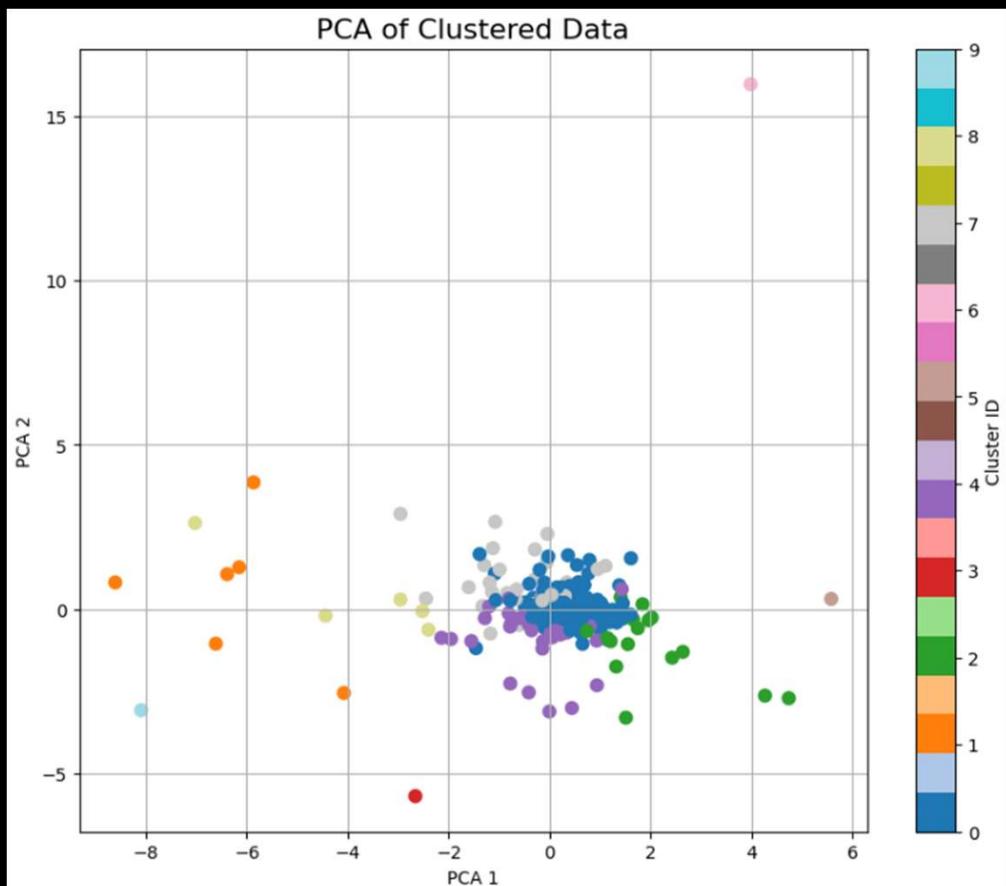
The main search area is titled "What are you looking for?". It includes sections for "Type of Product" (Goods, Services), "Frequency" (Annual, Monthly), "Classifications" (HS, SITC, BEC, As Reported), and "Refine your search".

The "Refine your search" section contains several dropdown menus:

- HS (as reported) Commodity Codes:** A list of commodity codes with "x" icons, including 270900, 271019, 260111, 260300, 260400, 290121, 290121, 290121, 290220, 290511, 290919, 291521, 720720, 740311, 750210, 760110, 810411, 847130, 847150, 851712, 852872, 854231, 870323, 870840, 870850, 870899, 871120, 610910, 620342, 620462, 611030, 630221, 100590, 100630, 120190, 120590, 070310, 020130, 020322, 030319, 030613, 030721, 190110, 190230, 200911, 210390, 220421, 840721, 840810, 841191, 842952, 843149, 850110, 850440, 850710, 853650, 854442, 280429, 281121, 290110, 290230, 290512, 300410, 300420, 300490, 300660, 300670, 490110, 490210, 490510, 491110, 852349, 851981, 852580, 852871, 853110, 854370, 020110, 020220, 0203, 020712, 020714, 150710, 150910, 151110, 151211, 151321, 271011, 271012, 271600, 271320, 271500, 380810, 380893, 381121, 381400, 382200, 390110, 390210, 390319, 390410, 390760, 392321, 392329, 392690, 401110.
- Reporters:** All
- Trade Flows:** Exports, Imports
- Partners:** World
- Modes of Transport:** TOTAL modes of transport
- 2nd Partner:** World
- Customs Codes:** TOTAL customs procedure codes

At the bottom right, there is a message in Korean: "Windows 정품 인증 [설정]으로 이동하여 Windows를 정품 인증합니다".

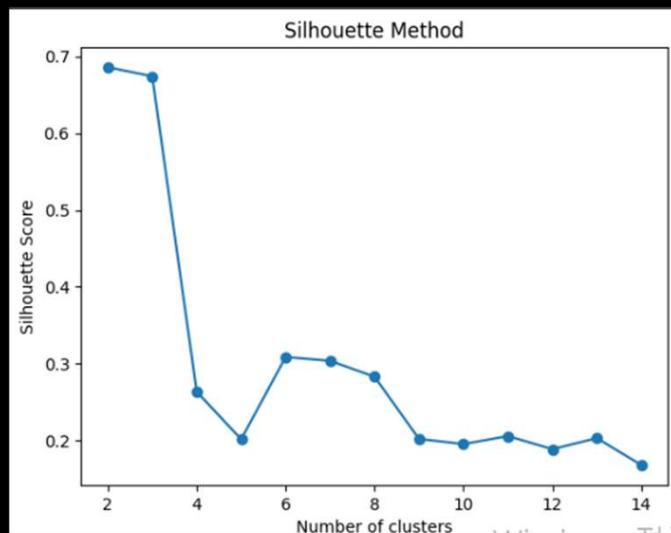
Vanila on US data



- ◇ 클러스터 0 체인 순서 (총 172개)
 - ◇ 클러스터 1 체인 순서 (총 6개)
 - ◇ 클러스터 2 체인 순서 (총 17개)
 - ◇ 클러스터 3 체인 순서 (총 1개)
 - ◇ 클러스터 4 체인 순서 (총 38개)
 - ◇ 클러스터 5 체인 순서 (총 1개)
 - ◇ 클러스터 6 체인 순서 (총 1개)
 - ◇ 클러스터 7 체인 순서 (총 31개)
 - ◇ 클러스터 8 체인 순서 (총 5개)
 - ◇ 클러스터 9 체인 순서 (총 1개)
- 표준편차 : 52.589

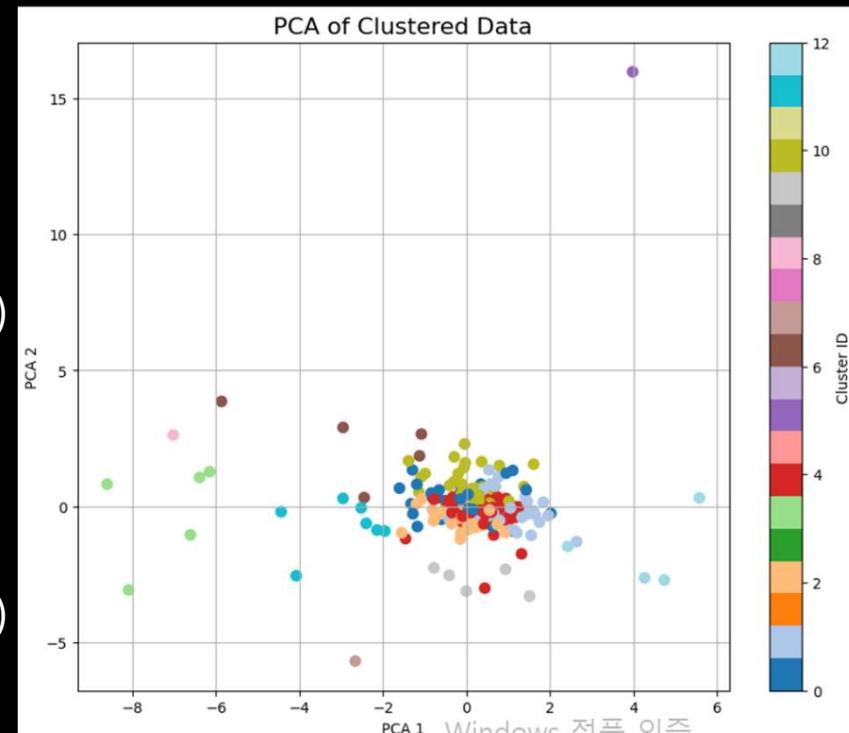
Refined on US data

PCA compo 4, 13 clusters



- ◇ 클러스터 0 체인 순서 (총 26개)
- ◇ 클러스터 1 체인 순서 (총 41개)
- ◇ 클러스터 2 체인 순서 (총 33개)
- ◇ 클러스터 3 체인 순서 (총 5개)
- ◇ 클러스터 4 체인 순서 (총 110개)
- ◇ 클러스터 5 체인 순서 (총 1개)
- ◇ 클러스터 6 체인 순서 (총 6개)
- ◇ 클러스터 7 체인 순서 (총 1개)
- ◇ 클러스터 8 체인 순서 (총 1개)
- ◇ 클러스터 9 체인 순서 (총 5개)
- ◇ 클러스터 10 체인 순서 (총 33개)
- ◇ 클러스터 11 체인 순서 (총 7개)
- ◇ 클러스터 12 체인 순서 (총 4개)

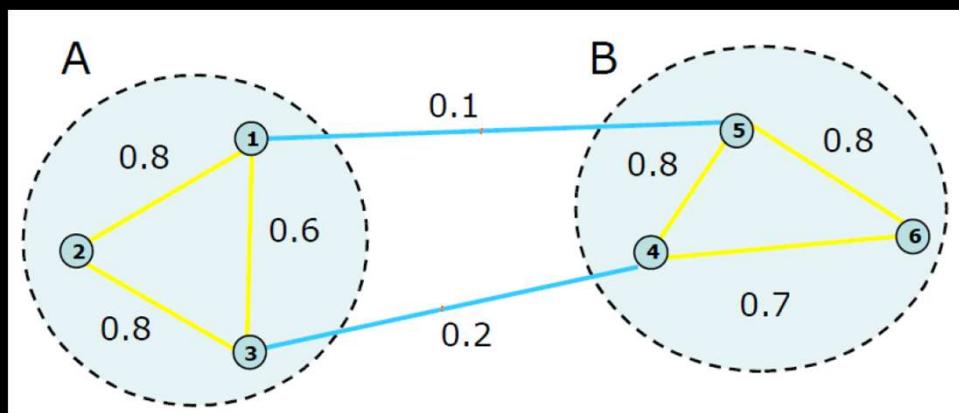
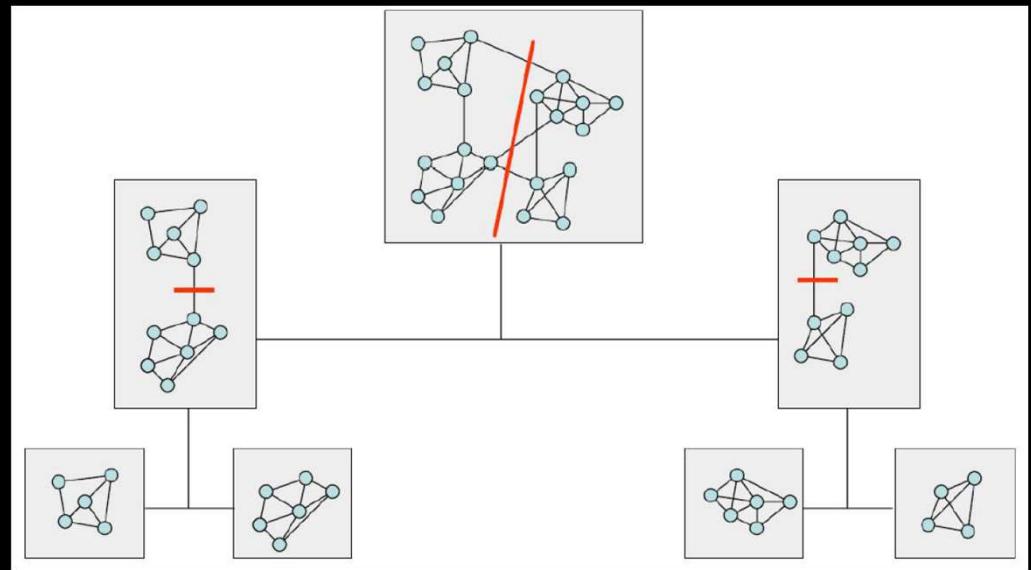
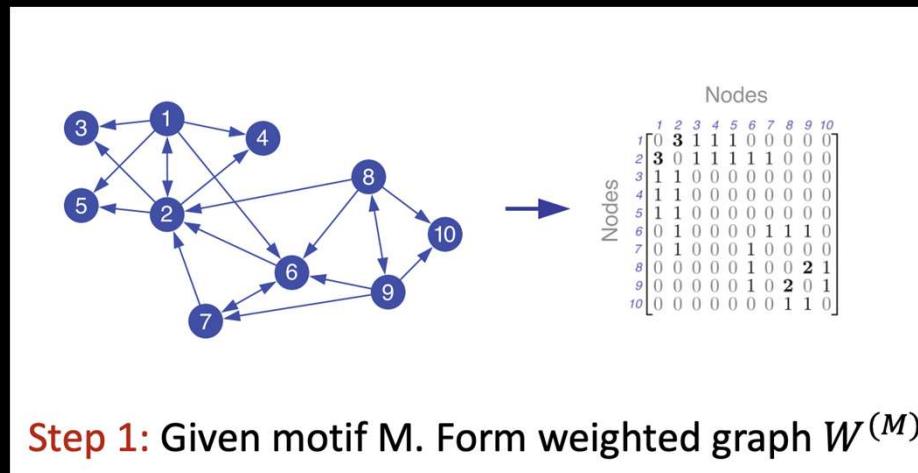
표준편차 : 30.353



Kmeans 문제점

1. 단순 distance -> 밀도 고려 안됨, 하나의 클러스터에 몰림
2. Outlier에 취약하다.(최대한 제거 했지만) Centroid 엇나감
3. 복잡한 데이터에 성능 떨어짐

Spectral Clustering

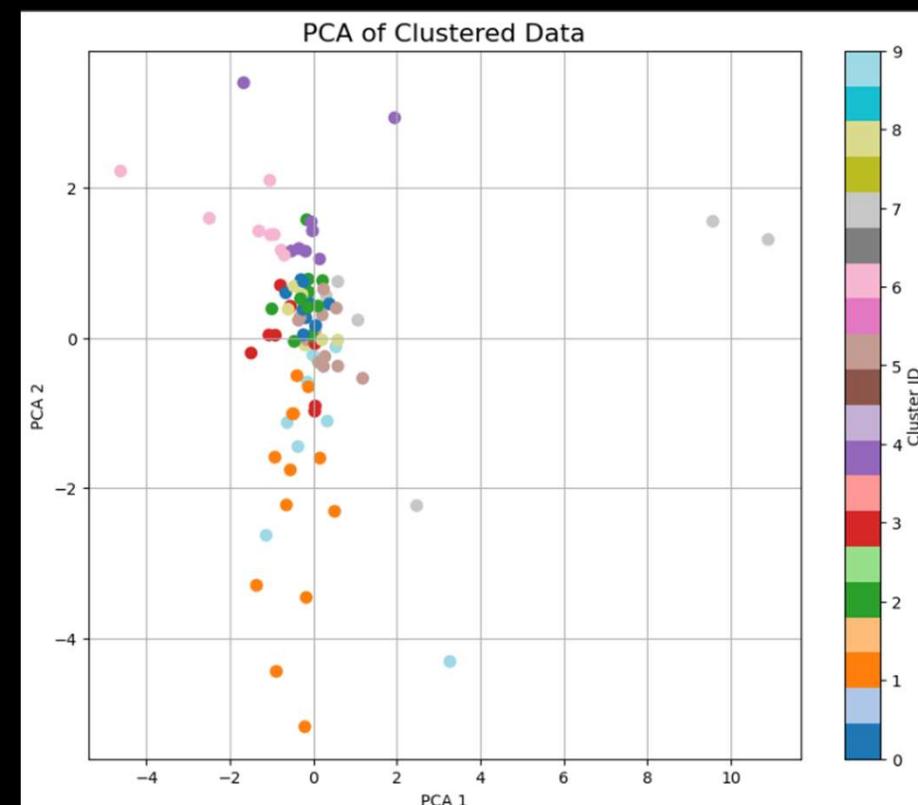


Spectral Clustering

- ◇ 클러스터 0 체인 순서 (총 8개) ◇ 클러스터 1 체인 순서 (총 6개)
- ◇ 클러스터 2 체인 순서 (총 14개) ◇ 클러스터 3 체인 순서 (총 8개)
- ◇ 클러스터 4 체인 순서 (총 7개) ◇ 클러스터 5 체인 순서 (총 15개)
- ◇ 클러스터 6 체인 순서 (총 12개) ◇ 클러스터 7 체인 순서 (총 14개)
- ◇ 클러스터 8 체인 순서 (총 8개) 표준편차 : 3.492

◆ 클러스터 6 체인 순서 (총 12개)

1. 내과용 · 외과용 · 치과용 · 수의과용 기기[신티그래픽(surgical graphic)]
2. 재생 · 반(半)합성 필라멘트사[재봉사와 소매용은 제외한다]
3. 반도체 디바이스(예: 다이오드 · 트랜지스터 · 반도체 기반의 텔레비전)
4. 엑스선이나 알파선 · 베타선 · 감마선 · 그 밖의 전리선을 발생하는 장치
5. 옥수수
6. 전자공업에 사용하기 위하여 도프처리된(doped) 화학원소
7. 전동기와 발전기(발전세트는 제외한다)
8. 전화기(셀룰러 통신망용이나 그 밖의 무선통신망용의 스마트폰)
9. 불꽃점화식이나 압축점화식 내연기관의 점화용 · 시동용
10. 그 밖의 차량(추진용 전동기만을 갖춘 것)
11. 석유가스와 그 밖의 가스 상태의 탄화수소
12. 천연고무 · 발라타(balata) · 구타페르카(gutta-percha) ·



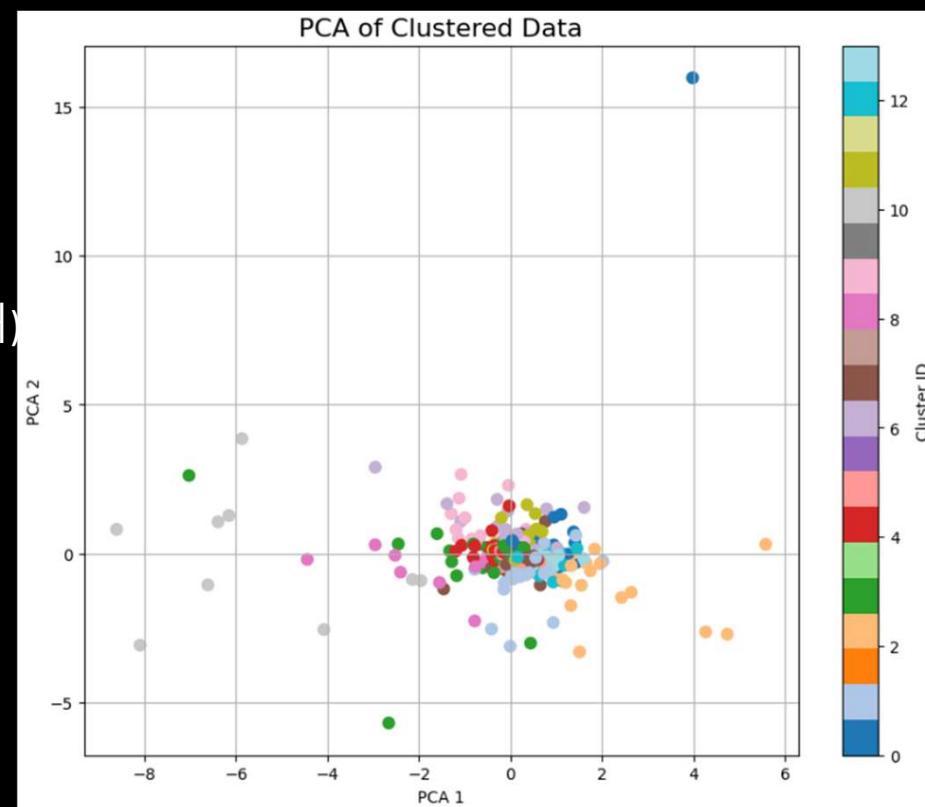
Spectral Clustering

- ◊ 클러스터 0 체인 순서 (총 18개) ◊ 클러스터 1 체인 순서 (총 28개)
- ◊ 클러스터 2 체인 순서 (총 21개) ◊ 클러스터 3 체인 순서 (총 23개)
- ◊ 클러스터 4 체인 순서 (총 32개) ◊ 클러스터 5 체인 순서 (총 10개)
- ◊ 클러스터 6 체인 순서 (총 21개) ◊ 클러스터 7 체인 순서 (총 38개)
- ◊ 클러스터 8 체인 순서 (총 9개) ◊ 클러스터 9 체인 순서 (총 13개)
- ◊ 클러스터 10 체인 순서 (총 9개) ◊ 클러스터 11 체인 순서 (총 20개)
- ◊ 클러스터 12 체인 순서 (총 12개) ◊ 클러스터 13 체인 순서 (총 19개)

표준편차 : 8.768

◆ 클러스터 2 체인 순서 (총 21개)

1. Medicaments; consisting of mixed or unmixed substances
2. Zinc; oxide and peroxide
3. Alcohols; saturated monohydric, propan-1-
4. Organic compounds; n.e.c. in chapter 29
5. Dairy produce; natural milk constituents
6. Calcium carbonate, chalk, talc, gypsum



한계

여러 인과관계 합쳐져 명확한 chain 분리 안됨

- ◆ 클러스터 6 체인 순서 (총 21개)
 1. Carbon; activated
 2. Vegetable preparations; potatoes,
 3. Ceramic tiles; flags and paving, h
 4. Cement; portland, other than white
 5. Beer; made from malt

데이터 소스가 무역 데이터밖에 없음

→ 내수 흐름 배제, 서비스 가치 창출 배제(오직 재화만 고려)

→ 모든 산업 모든 회사의 거래명세데이터 있었다면
정확했을것(불가능)

무역 데이터도 단일 국가에 국한

→ 전세계적인 가치 사슬 분석 안됨

사슬 순서 분석 부족

→ 하나의 클러스터에 들어도 어떤 과정에서 가치를 추가했는지 알기
힘듬