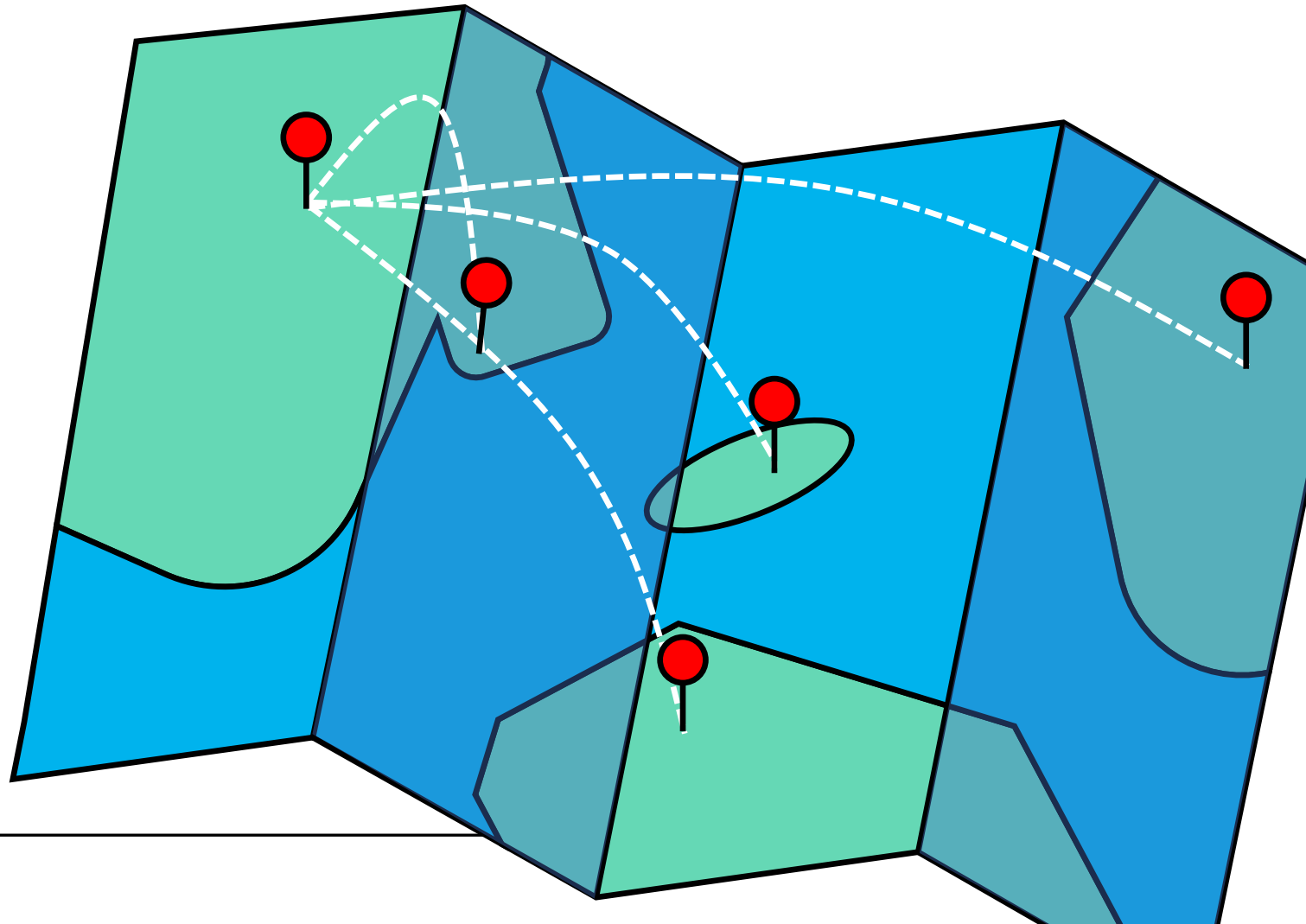
김진혁, 김보근, 노정아, 심수현, 이상혁

해외여행지 기반
국내여행지 추천 시스템



기저선대

1. 분석배경

분석 목적

국내 여행 활성화에 초점을 둔 정부의 내수활성화 대책이 발표됨

해외로 나가는 내국인 관광객에 대해서,
해외로 나가는 대신 국내 여행을 할 유인의 필요성이 대두됨

해외 여행지 선택 이유를 분석해 국내 여행지를 추천해주자!

1. 분석배경

분석 흐름

1. 여행지 데이터 수집 및 EDA
2. 국내 여행지 클러스터링을 통한 인사이트 도출
3. 해외, 국내 여행지 유사도 계산을 위한 토픽 모델링 진행
4. 컨텐츠 기반 국내 여행지 추천 시스템 구축
5. 추천 정보를 활용한 관광 활성화 도모 목적의 관광지 정책 제언

2. 데이터

토픽 추출 데이터 크롤링 및 토큰화

특정 여행지의 특징을 보여줄 수 있는 데이터로 **네이버 블로그**를 선택해

국내 228개 시군구, 해외 136개의 도시를 크롤링 한 뒤, 필터링 및 토큰화 과정을 거침

총 **142,856건**의 데이터에 대해 **okt**와 **mecab**을 이용해 **토큰화**를 진행함

2. 데이터

데이터 수집 방법

한국관광 데이터랩

한국관광 데이터랩 Data Lab

지역별 동반 유형, 지역별 방문자수,
지역별 SNS언급량, 유형별 검색건수 등

카카오 API

각 지역의 X, Y좌표

2. 데이터

데이터 설명

여행지 유사도와 더불어, 사용자 맞춤형 국내 관광지 추천을 위해

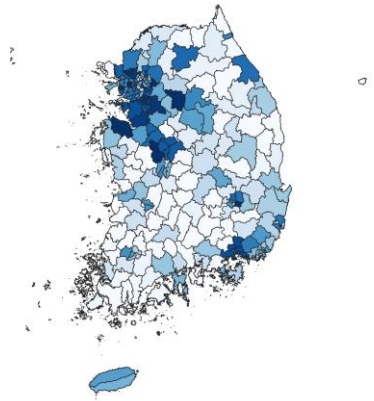
국내 여행지의 특성을 나타낼 수 있는 변수를 선정해 데이터셋을 구성함

변수	파생변수
거리	-
여행 동반자 유형	-
2022, 2023년 방문자수	방문자 증가율
22년 3월 & 23년 3월 SNS 언급량	계절별 트렌드, SNS 언급량 변화율, 최신트렌드
카테고리	-
축제	-

2. 데이터

데이터 설명

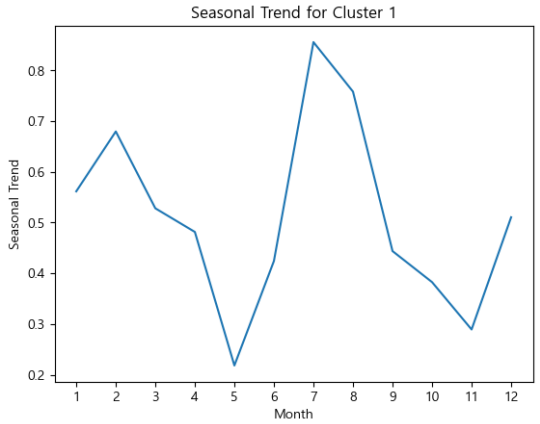
카테고리별 검색량 수 시각화



기타(대형마트)

1. 경기도 하남시
2. 경기도 고양시
3. 경기도 용인시

계절별 트렌드 변수



1. 부산 해운대구
2. 강원도 강릉시
3. 제주도 제주시

⋮

QGIS를 활용한 시각화 및 EDA를 통해 **인사이트 도출**

시계열 분해 등을 통해 **파생변수 생성**

3. 클러스터링

검색 비율이 가장 높은 유형으로 카테고리 부여시

작은 차이로 하나의 카테고리에 분류되는 문제 발생

광역	기초	기타	데이트 코스	레저 스포츠	문화 관광	쇼핑	숙박	음식	역사 관광	자연 (바다)	자연 (산,공원)	체험 관광
강원	양양	0	0	0.077	0.0151	0.0094	0.1224	0.1646	0.0302	0.2362	0.2482	0.0969

→ 강원도 양양의 경우 검색 비율이 가장 높은 유형으로 카테고리 부여시 자연(산,공원)으로 분류됨

기존의 방식으로 카테고리 부여시

바다와 산, 공원의 비율이 모두 높다는 특징을 반영한 분류 불가능

3. 클러스터링

<유형별 검색 비율>

광역	기초	기타	데이트 코스	레저 스포츠	문화 관광	쇼핑	숙박	음식	역사 관광	자연 (바다)	자연 (산,공원)	체험 관광
강원	강릉	0.0338	0.1735	0.0229	0.0439	0.0525	0.1063	0.0642	0.0668	0.2834	0.0362	0.1165
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
충북	충주	0.0335	5.00E-04	0.3333	0.0366	0.0116	0.0624	0.0235	0.0428	0.016	0.1897	0.2501

각 유형을 토픽으로 보면 각 도시의 토픽 비율로 데이터를 이해할 수 있음

11가지 군집에 속할 확률을 모두 고려해 클러스터링하면

좀 더 정확한 분류를 할 수 있을 것이라는 가정하에 클러스터링 시도

3. 클러스터링

Compositional data의 특징 만족

① 데이터의 **모든 feature값이 양수**

② feature의 **총합이 모든 데이터에 대해서 상수**

광역	기초	기타	데이트 코스	레저 스포츠	문화 관광	쇼핑	숙박	음식	역사 관광	자연 (바다)	자연 (산,공원)	체험 관광
강원	강릉	0.0338	0.1735	0.0229	0.0439	0.0525	0.1063	0.0642	0.0668	0.2834	0.0362	0.1165
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
충북	충주	0.0335	5.00E-04	0.3333	0.0366	0.0116	0.0624	0.0235	0.0428	0.016	0.1897	0.2501

Aitchison Geometry Vector Space

벡터공간의 정의

일반적인 Euclidian space R^D에서는 벡터 덧셈, 스칼라 곱셈에 대해 닫혀있고 거리를 구할 수 있음

Compositional data를 구성하는 simplex에 대해서는 이를 그대로 적용할 수 없음

- Compositional data는 덧셈, 스칼라 곱셈에 대해 닫혀있지 않음
- Compositional data의 Euclidian 거리가 비율에 따라 상대적임

Aitchison Geometry Vector Space

Compositional data의 Aitchison geometry 벡터공간은

유한한 차원을 가진 Hilbert 공간이 되므로

Simplex로 부터 Gram-Schmidt 직교법이나 SVD 등으로 직교좌표계를 구성할 수 있음

완비 내적 공간으로 유클리드 공간을 일반화한 개념

완비 내적 공간: 다음의 특징을 만족하는 내적이 가능한 벡터 공간

i) 두 벡터를 이용해 스칼라를 반환하는 함수가 존재

ii) 공간에 위치한 코시 벡터가 공간 내의 극한값으로 수렴 (= 완비성 complete)

Transformation

- Center log-ratio Transformation (CLR)

모든 구성 요소의 기하평균으로 나눈 각 구성요소에 자연로그를 취하는 방법

-> **상대적인 비율과 상호의존성을 고려하기위해 compositional data에 적용됨**

$$CLR(x) = v = \left[\log\frac{x_1}{g(x)}, \cdots, \log\frac{x_D}{g(x)}\right] \in R^D, \sum_{i=1}^{D} v_i = \log\frac{x_1 x_2 \cdots x_D}{g(x)^D} = \log 1 = 0$$

CLR의 ILR은 위 hyperplane을 spanning할 수 있는 D-1개의 orthonormal basis로 구해짐

이때 **ILR은 isometry한 변환이므로** 정확한 거리를 구할 수 있음

3. 클러스터링

Compositional data

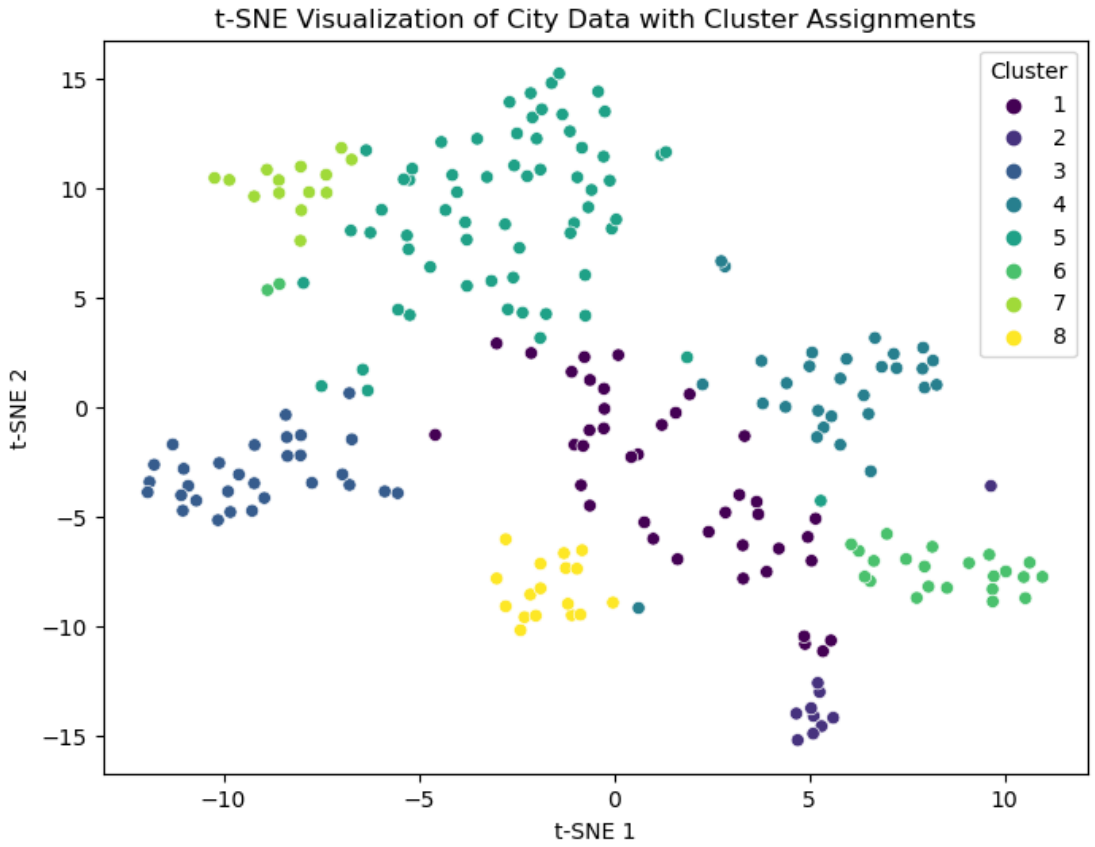
Aitchison Distance

CLR 변환 요소들 간의 유클리디안 거리

$$d(x_i, x_j) = \sqrt{\sum_{g=1}^{D} [\ln \frac{x_{gi}}{g(x_i)} - \ln \frac{x_{gj}}{g(x_j)}]^2}$$

1. Scale Invariance: 두 구성 요소 간의 거리가 **스케일에 영향을 받지 않음**
2. Perturbation Invariance: 각 구성 요소에 **상수를 곱해도 요소 사이의 거리가 변하지 않음**
3. Permutation Invariance: **거리 측정이 구성 요소의 재정렬**에 불변함
4. Sub-compositional Dominance: **한 구성의 부분 구성이 다른 구성의 해당 부분 구성보다 우위를 가지면 거리를 측정할 때 우위 관계를 반영함(큰 비율을 차지하는 구성의 차이를 더 중요하게 반영)**

3. 클러스터링



compositional data의 특징을 고려해 Aitchison 거리를 사용한 최종 군집화 결과

3. 클러스터링

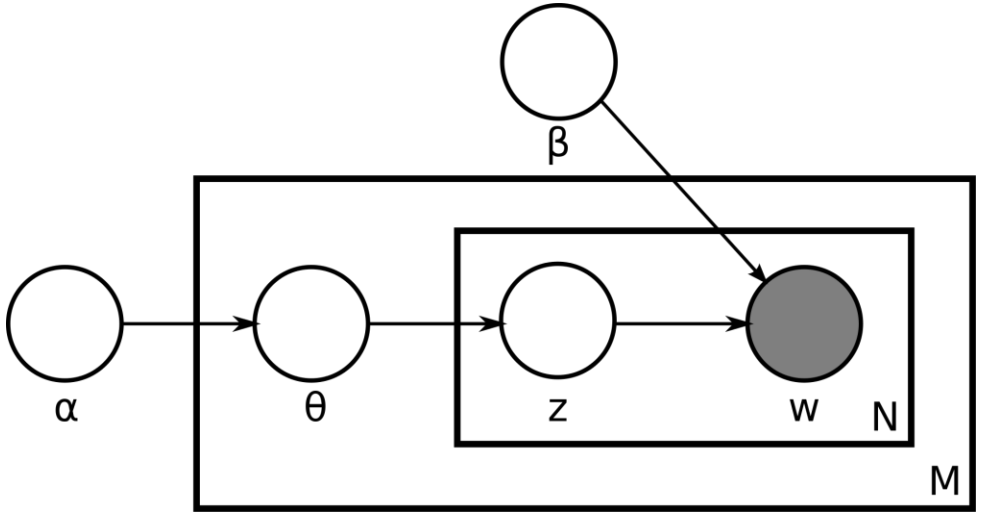
전체 데이터셋

광역	시군구	거리	가족 동반	…	회사동료 동반	1월 방문자수	…	방문자 증가율	3월 계절 트렌드	언급량 증가율	언급 트렌드	축제	클러스터
강원	강릉	166.7	0.58	…	0.009	2976083	…	1.304	0.5103	0.882	21335	4	3
경기	고양	16.2	0.74	…	0.023	12937504	…	1.143	0.6225	0.827	4515	1	5
부산	해운대	332.2	0.41	…	0.006	5386438	…	1.145	0.3913	0.995	12789	2	3
서울	종로	1.96	0.35	…	0.008	8900897	…	1.328	0.2731	1.031	41479	1	2
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
전북	임실	220.5	0.42	…	0.384	547588	…	1.346	0.8113	1.522	1956	0	1
제주	서귀포	483.4	0.75	…	0.039	4400043	…	1.121	0.6298	1.039	15810	0	1
충남	아산	88.6	0.42	…	0.072	2544841	…	1.132	0.8086	0.819	827	0	8

LDA(Latent Dirichlet Allocation)

각 문서는 토픽의 혼합으로 구성되어져 있으며,

각 토픽에서 확률 분포에 기반하여 단어를 생성한다고 가정함



M : 문서의 개수

k : 토픽의 개수

N : 단어의 개수

α : 디리클레 분포의 매개변수

θ : 문서가 주제에 속할 확률분포 $\sim Dir(\alpha)$

β : 특정 토픽이 단어를 생성할 확률

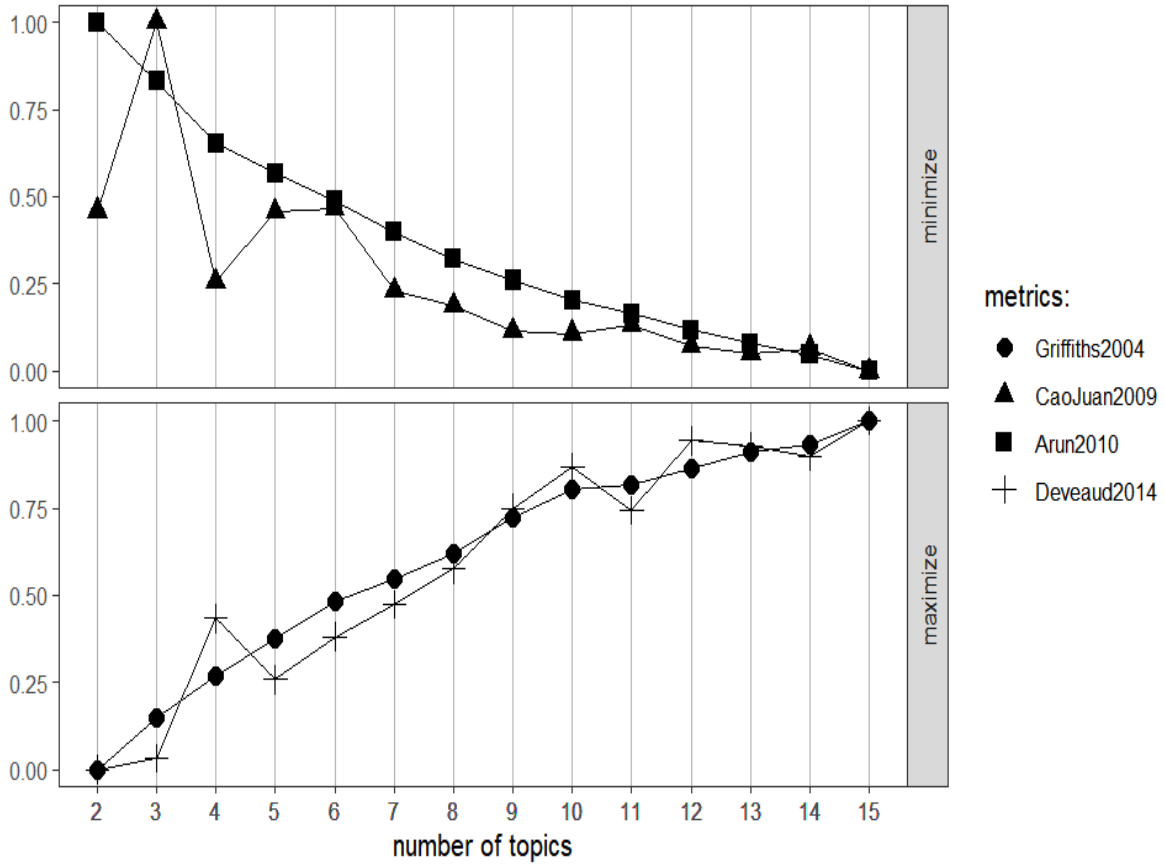
Z_n: 단어가 특정 토픽에 속할 확률 분포 $\sim Multinomial(\theta)$

w : 단어 벡터

4. LDA

최적의 토픽개수 찾기

R - ldatuning



ldatuning

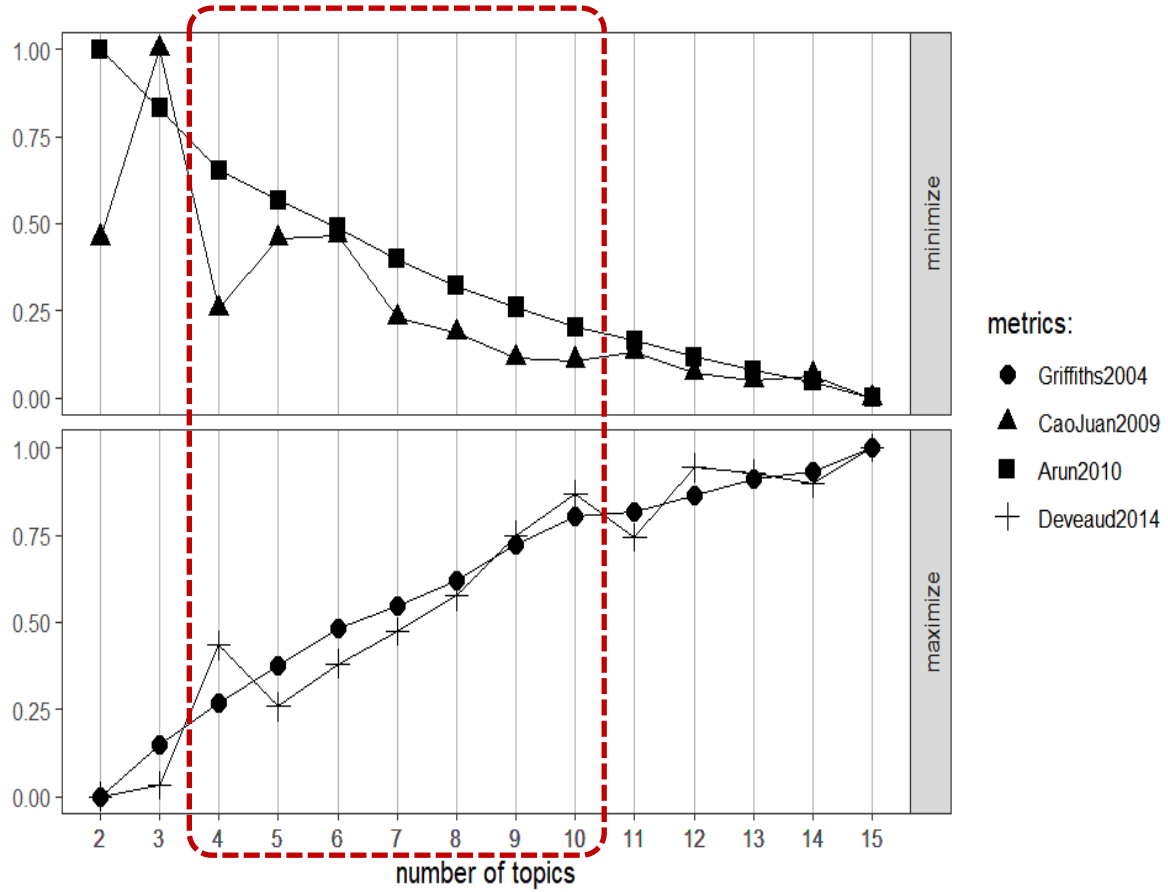
Metrics

: "Griffiths2004","CaoJuan2009"

"Arun2010", "Deveaud2014"

최적의 토픽을 구하는 Perplexity와 Coherence 이외에도 논문에서 제안한 **Metrics**로 토픽 개수를 찾을 수 있음

해외 도시만 활용한 ldatuning



토픽 4개와 10개에서 그래프가 꺾이지만

4개는 토픽 내 여러 분야가 혼재됨

10개는 토픽 간 중복 문제 발생

토픽에 대한 해석까지 고려하여 K=6 선정

- 국내 군집 재조정 결과

토픽	해외 토픽 결과 해석	국내 카테고리 재조정 결과
Topic 1	휴양, 자연(바다)	자연(바다)
Topic 2	쇼핑	쇼핑
Topic 3	음식	음식, 체험관광
Topic 4	자연(산)	자연(산), 숙박
Topic 5	역사	역사
Topic 6	문화	문화

5. 유사도 계산

유사도 계산의 종류

유사도

두 데이터가 **얼마나 비슷한 지** 나타내는 척도

데이터를 어떤 공간의 좌표(위치)처럼 벡터화했을 때, **두 데이터 사이의 거리**

유클리드 거리

코사인 유사도

JS Divergence

서로 다른 토픽 분포에서 각 문서가 나왔다는 가정 하에 LDA 진행

→ 그 결과로 나온 토픽 데이터이므로 JS Divergence 사용 가능

서로 다른 확률 분포의 차이 측정

KL-Divergence를

symmetric 하게끔 개량,

거리의 역할을 수행하도록 함

5. 유사도 계산

유사도 계산 결과

JS Divergence 기반 유사도 계산 결과

도시	0	1	2	3	4	5
로마	0.05387	0.05153	0.18288	0.06903	0.32446	0.31821

광역	기초	토픽1	토픽2	토픽3	토픽4	토픽5	토픽6	Dist
경남	함안군	0.028	0.022	0.196	0.074	0.301	0.375	0.081
울산	중구	0.001	0.069	0.225	0.111	0.408	0.184	0.180
경북	경산시	0.104	0.092	0.272	0.146	0.120	0.263	0.210
전남	곡상군	0.036	0.046	0.134	0.290	0.245	0.245	0.211
서울	종로구	0.000	0.098	0.077	0.108	0.221	0.493	0.223

Topic 5, 6의 비율이 높은 이탈리아 로마와 유사한 국내 도시들 또한

해당 토픽 비율이 로마와 유사함

6. 공간가중행렬

사용 배경

여행자의 여행 특성 고려

여행지 근처 도시나 주변 지역의 명소를

함께 방문하는 것을 고려하는 경우가 많음

추천 점수 계산 시, **주변 도시의 점수**를

함께 고려하고자 함

6. 공간가중행렬

공간가중치를 계산하기 위해서는

국내 도시 공간가중행렬을 만들어야 함

Spatial weight matrix $W = (w_{ij};\ i,j = 1,\dots,S)$:

w_{ij} : Spatial influence of unit j on unit i

w_{ii} = 1 (i.e., all diagonal elements of W are 1)

Weights based on distance	Weights based on boundaries	Combined distance-boundary weights

Combined distance-boundary weights

distance와 boundary의 관계로 표현되는 공간적 영향(Spatial influence)

*Cliff and Ord(1969)가 제안한 **공간가중행렬에 거리 가중치를 추가**하여

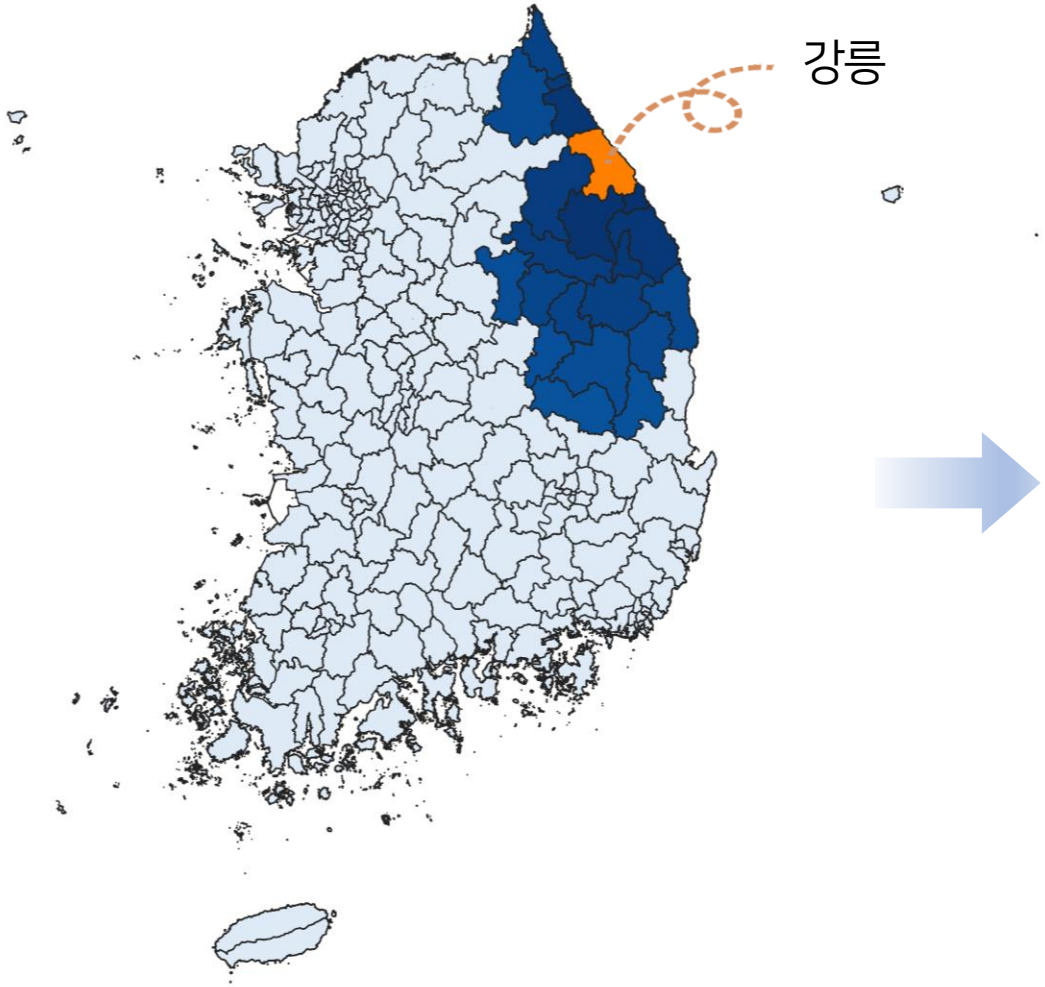
인접하진 않지만 가까운 거리에 있는 도시의 영향력 고려

$$w_{ij} = \frac{I_{ij} p_{ij}^{-\alpha}}{\sum_{k=1,\ldots,S,k\neq i} I_{ik} p_{ik}^{-\alpha}} + \frac{p_{ij}^{-\alpha}}{\beta}, where\ \alpha, \beta > 0, p_{ij} = \begin{cases} d_{ij}, & d_{ij} < 100 \\ 0, & d_{ij} \geq 100 \end{cases}$$

distance와 boundary를 모두 고려하여 가중치를 부여할 수 있기 때문에 선택

6. 공간가중행렬

- 강릉과 다른 도시들 간의 공간가중치 시각화



강릉과 거리가 **가까우면서**

인접하지 않은 도시들에도

가중치가 부여됨

7. 추천 시스템

단순히 유사한 도시를 찾는 것이 아니라

여행지를 추천해주는 것이기 때문에

도시간 유사성을 고려함과 동시에

여행지로서도 적합한 도시를 추천해주는 것이 중요했음

여행지 선택에 영향을 주는 변수를 사용해 여행지 추천

7. 추천 시스템

추천 사용 변수

< 2021 국민여행조사 中 관광여행 방문지 선택 이유 >

구분	여행지 지명도	볼거리	경비	이동거리	여행기간	숙박시설	음식	교통편	동반자유형	…
비율	16.1	21.3	4.7	12.4	12.5	3.4	8.4	2.2	8.8	…

- 여행지 지명도 -> 3월 언급 트렌드/6월 계절 트렌드/방문자수
- 볼거리 -> 해외, 국내 도시 유사도
- 이동거리 -> 거리
- 동반자 유형 -> 동반자 유형

비율이 높은 항목들에 생성했던 변수들을 각각 대응

7. 추천 시스템

추천 사용 변수

< 2021 국민여행조사 中 관광여행 방문지 선택 이유 >

구분	여행지 지명도	볼거리	경비	이동거리	여행기간	숙박시설	음식	교통편	동반자유형	…
비율	16.1	21.3	4.7	12.4	12.5	3.4	8.4	2.2	8.8	…

각 요소들이 정확히 대응하지 않고,

지명도를 언급 트렌드/계절 트렌드/방문자수로 사용하기 때문에

유저에게 직접 **가중치**를 받아 추천

7. 추천 시스템

변수 + 유저 가중치 반영 추천

추천 결과

1.서울-종로구 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족

2.서울-강남구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족

3.경기도-수원 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족

4.서울-마포구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족

5.서울-송파구 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족

6.서울-서대문구 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족

7.서울-중구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족

8.서울-은평구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족

9.서울-서초구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족

10.경기도-용인 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족

서울 강남구, 마포구 등

의도와 다른 지역이 추천

선호도 입력에 대한 모호함 존재

선호도 1과 2에 대한 입력의 차이에도 민감하게 반응

사용자의 **반응**을

고려해 **가중치 업데이트**

Incremental learning (online learning)

Online Machine Learning

데이터를 순차적으로 사용해 각 단계에서 미래 데이터에 대한
최상의 예측 변수를 점진적으로 업데이트하는 데 사용되는 머신 러닝 방법

↓

데이터가 지속적으로 흘러가는 것을 처리하기 위한 모델로서
stream 형태의 데이터들을 통해서 학습하는데 유용하게 사용

실시간으로 변화하는 유저들의 선호도에 적응하게 되는 시스템에 적합

Online Machine Learning

추천값에 대한 사용자의 평가를 받고 그에 맞춰 **모델을 업데이트**해
사용자의 **선호에 더 맞는 여행지를 추천**하고자 함

-> **매번 새로운 반응에 대한 모델 업데이트** 필요

Online learning 사용

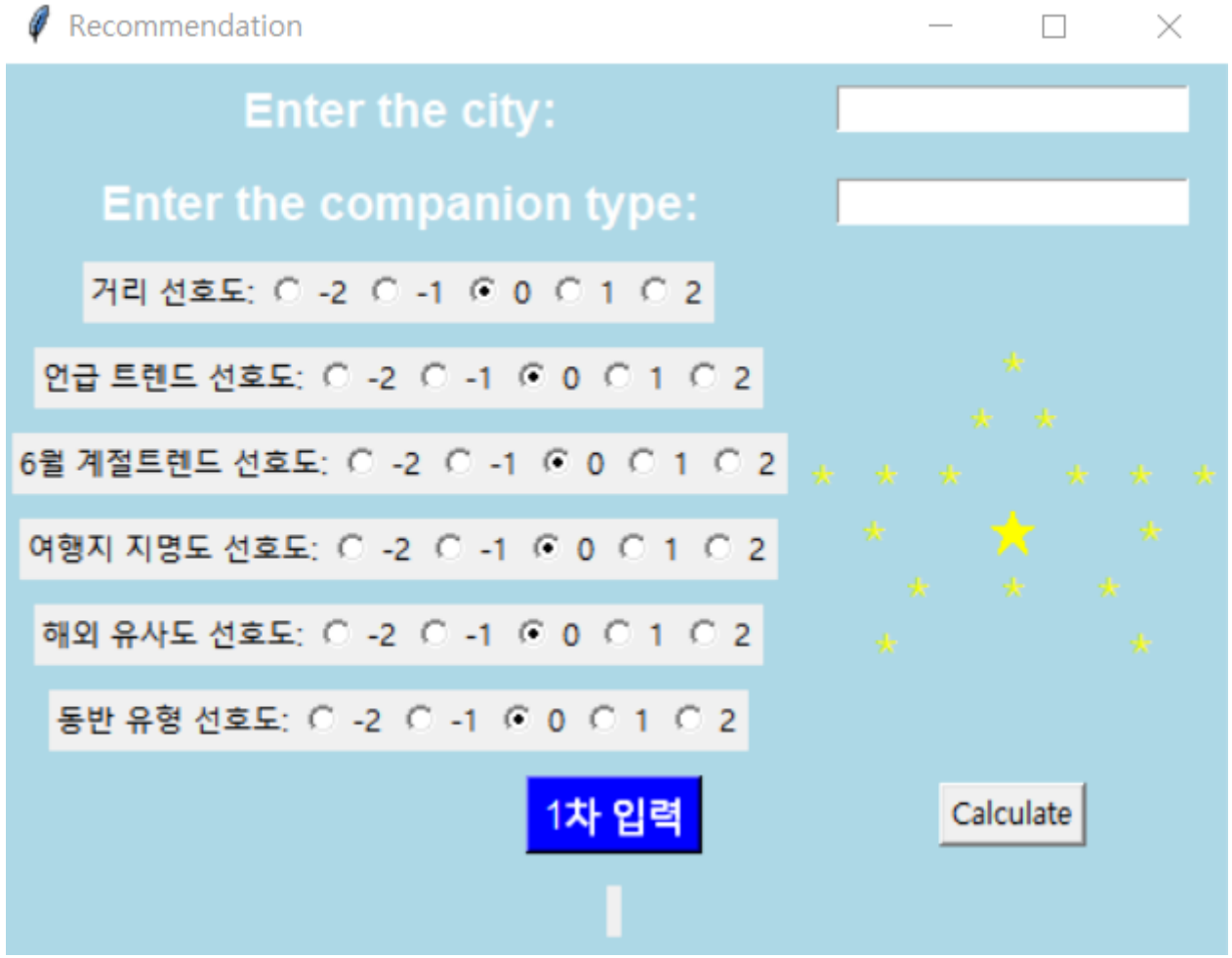
7. 추천 시스템

Online Machine Learning

1. 사용자가 가고 싶은 **해외도시** 입력
2. 사용자가 **동반자 유형** 선택
3. 각 변수에 대한 **선호도** 입력
4. 해외도시, 동반자유형, 선호도에 따른 **추천값 출력**
5. 추천에 대한 **사용자 평가 입력**
6. 평가를 반영해 **모델 업데이트**
7. 업데이트된 모델로 **새로운 여행지 추천**
8. 5 ~ 7 반복

7. 추천 시스템

Online Machine Learning



- 해외도시 : 로마
- 동반자 유형 : 친구

7. 추천 시스템

Online Machine Learning

1차 추천 결과

1.서울-종로구 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
2.서울-강남구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족
3.경기도-수원 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
4.서울-마포구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족
5.서울-송파구 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
6.서울-서대문구 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
7.서울-중구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족
8.서울-은평구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족
9.서울-서초구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족
10.경기도-용인 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족

사용자 평가를 반영해 업데이트한 추천 결과

1.경북-경주시 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
2.충북-청주시 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
3.서울-종로구 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
4.충남-공주시 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
5.전북-전주시 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
6.서울-마포구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족
7.서울-중구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족
8.경남-창원시 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족
9.경남-함안군 | 추천 만족도를 입력해주세요: ○ 불만족 ◉ 만족
10.서울-영등포구 | 추천 만족도를 입력해주세요: ◉ 불만족 ○ 만족

감사합니다