|  |
| --- |
| 데이터 기반 서울시 대중교통 환승센터 입지 선정  Selecting optimal position of building new bus transfer center  based on data  손현오, OOO, OOO, OOO  요약문  서울은 상시 교통체증이라고 해도 과언이 아닐 정도로 항상 교통이 혼잡합니다. 실제로 최근 국토교통부의 분석 결과에 따르면 전국 교통 혼잡구간은 최근 3년간 늘어났다고 합니다. 본 논문에서는 이러한 교통체증을 줄이기 위해 어디에 버스환승센터를 추가적으로 설치했을 때 가장 효율적일지를 보여줍니다. 직접 수집한 데이터를 통해 비용함수를 제작하였고 이를 통해 선정된 후보지 중에서 최적의 후보지를 도출합니다. 또한 이를 데이터분석 도구 Rapidminer를 통해 다시 한번 결과를 도출합니다. 최종적으로 도출된 지역들을 조사하여 최종 후보지로서 자격을 확인합니다.  Abstraction  It is no exaggeration to say that Seoul is always crowded with traffic. In fact, according to a recent analysis by the Ministry of Land, Infrastructure and Transport, the number of traffic congestion sections nationwide has increased over the past three years. In this paper, it will be most effective when additionally installed a bus transfer center to reduce such traffic jams. A cost function was created using the data directly collected, and the optimal candidate site was derived from the selected candidate sites. In addition, the results are derived once again through the data analysis tool Rapidminer, and the finally derived regions are investigated to confirm their qualifications as final candidates.  **핵심용어**: Bus Transfer Center, Data Analysis, Cost Function, Optimal Candidate |
|  |

데이터 기반 서울시 대중교통 환승센터 입지 선정

Python과 RapidMiner 기반 데이터 분석 프로세스 개발

# **서론**

# **연구 목표**

본 연구는 데이터 분석을 이용해 서울시에서 제공한 대중교통 이용 데이터를 기반으로 기존의 환승센터 입지(강변역, 개화역, 구로디지털단지역, 구파발역, 도봉산역, 서울역, 여의도, 잠실, 청량리역)를 제외한 새로운 환승센터의 입지에 적절한 지역을 선정하는 것을 목적으로 한다.

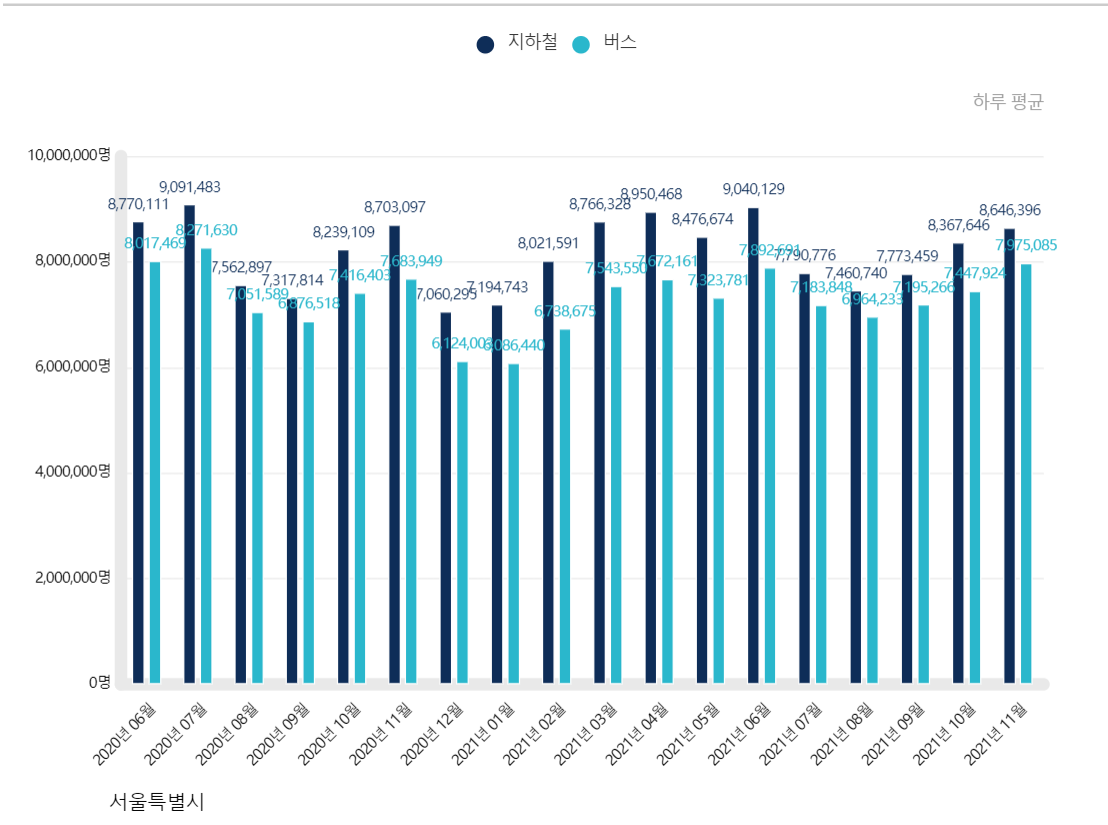
본 연구에서는 정확한 입지 선정을 위해 두 개의 분석 도구를 사용한다. 분석 도구로는 Python3와 Rapidminer를 사용했다. Python은 데이터 분석을 위한 다양한 라이브러리를 지원하고 있으며, Rapidminer는 데이터 준비, 기계학습, 예측 모델 배포를 위한 데이터분석 S/W 플랫폼이라는 점을 이유로 채택하였다. 두 분석 도구들로 도출된 결과를 비교 분석하여 서울시의 대중교통 환승센터 추진 목적에 적합한 입지를 선정하였다.

# **현황 분석**

테이블이(가) 표시된 사진

자동 생성된 설명

서울시 대중교통은 2013년부터 꾸준히 많은 이용객 수를 유지하고 있다. 2021년 9월 13일 서울시가 발표한 대중교통 이용객 현황을 보면, 2020년 대중교통 이용객 수가 감소했음을 알 수 있다. 그러나 이러한 일시적인 감소 추세는 국토교통부 생활교통복지과에서 2021년 3월에 발표한 보도자료대로 수도권의 대중교통 이용률의 감소는 코로나-19 확산의 영향이다. 이후 서울열린데이터 광장에서 제공한 ‘서울시 지하철 호선 별, 역 별 승차하 인원 정보’ 데이터를 보더라도 대중교통 이용객 수가 코로나 확산에 따라 일시적으로 감소했다가 증가하여 일정 수준 이상으로 유지되고 있음을 알 수 있다.



이용객 수의 증가는 대중교통 이용에 불편을 초래하여 자가용의 이용 증가로 이어졌으며, 자가용의 이용 증가는 교통 혼잡의 증가로 이어져 서울시민의 불편함을 증가시켰다. 이에 따라 서울시에서는 시민 편의성을 위해 대중교통 활성화를 추진하게 되고, 대중교통 활성화 추진 계획으로 시작된 것이 대중교통 환승 센터이다.

서울시에서 밝힌 서울 주요 정책을 통해 언급한 대중교통 환승 센터의 정의는 다음과 같다. 대중교통 환승 센터는 버스와 버스, 버스와 지하철 등 버스와 타 교통수단을 잇는 환승 교통시스템으로 대중교통을 이용하는 시민이 다른 교통수단을 보다 빠르고 편리하게 환승할 수 있도록 대중교통 이용 환경을 개선하고, 서울 외곽에서 도심으로 진입하는 승용차 위주의 교통량을 줄이고 대중교통 이용을 활성화하여 도심 교통환경을 개선하는 역할을 하고 있다.

# **본론**

# **입지 선정기준**

버스-버스, 버스-타 교통수단을 잇는 대중교통 환승 센터(이하 환승 센터)는 승용차 위주의 교통량을 줄이고 대중교통 이용을 활성화하여 도심 교통환경을 개선하는 역할을 하고 있다. 이러한 환승 센터 정책의 시행 전후를 비교한 결과는 다음과 같다.

|  |  |
| --- | --- |
| **시행 전** | **시행 후** |
| - 수단간 연계가 어려움  - 승용차 위주의 지속적인 교통량 증가  - 혼잡하고 긴 환승 동선으로 인한 환승 불편  - 노선 및 환승 정보 부족 | - 교통수단간 환승 편의  - 대중교통 활성화로 도심교통 환경 개선  - 환승 동선 최적화로 환승 시간 단축  - 체계적인 노선 및 환승 정보 제공 |

환승 센터 정책 시행 결과 나타난 특징을 지표화 할 수 있도록 기준을 마련하였다. ‘교통수단 간 환승 편의성’은 지하철역 주변의 버스정류장, 지하철역의 개수를 통해 측정할 수 있다. ‘대중교통 활성화로 도심교통 환경 개선’은 공간성 확보를 위한 도로의 차선 수로 간접적 평가가 가능하다. 이는 환승 동선 최적화로 환승 시간 단축 효과에도 기여할 수 있다. ‘체계적인 노선 및 환승 정보 제공’은 후보지 주변 버스 노선 개수 및 빈도를 통해 지표화 할 수 있다.

환승 센터 입지를 선정하는 기준을 위해 데이터를 기반으로 변수를 설정하여 아래와 같은 총 8개의 입지선정 기준을 마련하였다.

|  |  |  |
| --- | --- | --- |
|  | 입지선정 기준 (변수) | 역할 |
| 1 | 500m 내 정류장 개수 | 지하철역을 기준으로 반경 500m 내 존재하는 정류장의 개수 |
| 2 | 500m 내 지하철역 개수 | 지하철역을 기준으로 반경 500m 내 존재하는 지하철역의 개수 (=환승 가능한 다른 호선) |
| 3 | 통행 버스 노선 개수 | 지하철역 반경 500m를 지나는 버스 노선 개수  (광역/간선/지선/순환 등의 버스 노선 모두 포함) |
| 4 | 통행 버스 노선 빈도 | 역 반경 500m를 지나는 버스 노선의 (총 운행거리) × (운행 횟수) (일별) |
| 5 | 주변의 환승 주차장 개수 | 지하철역 기준 반경 1.5km 주변의 환승 주차장\* 개수 (\*지하철로 환승하면 주차요금 결제 시 환승 할인 지원하는 주차장) |
| 6 | 지하철역 주변 차선 수 | 지하철역 주변의 차선 수 합 |
| 7 | 지하철/버스 승 하차 인구 | 지하철/버스 승 하차 인구 합  (버스는 지하철역 반경 500m 내 정류장만 고려) |
| 8 | 기존 환승 센터와의 거리 | 기존 환승 센터 6곳과의 거리의 합 (청량리, 서울역, 여의도, 구로디지털단지, 잠실, 삼성) |

# **Python 기반 입지 도출 과정**

### **데이터 분석 프로세스**

본 연구에서는 버스, 지하철의 승 하차 인구를 파악하고자 서울특별시TOPIS 종합교통관제센터에서 제공하는 대중교통 O/D 현황 월별 데이터를 활용하였다.

텍스트이(가) 표시된 사진

자동 생성된 설명

2021년 1월부터 9월까지의 대중교통 월별 데이터를 사용하였다. 월별 평일, 토요일, 일요일 하루씩 데이터가 제공되어 Unique한 데이터의 개수가 많지 않다. 버스 노선과 승 하차, 정류장명, 승객 수 위주로 분석하고, 이들을 메인으로 정류장별 승객 수 총합을 도출했다. 노선 별 승객수가 가장 많았던 정류장, 가장 많이 이용하는 정류장과 가장 이용하지 않는 정류장을 시각화 하여 확인해본 결과 상위 정류장과 하위 정류장의 차이가 뚜렷하게 나타났다. 이러한 관점에서 지하철 승객수에 대한 탐색도 진행하였다.

* + 1. **1단계: Top 39 환승 센터 후보지 선정**

지하철역을 기준으로 1월부터 9월까지의 월별 승차 수와 총 승객 수 합, 지하철역 위치에 해당하는 좌표를 도출하여 Top 39 환승 센터 후보지를 필터링했다. 지하철역 좌표의 경우, 서울교통공사와 코레일에서 제공하는 지하철역 좌표 데이터셋을 활용하였다.

테이블이(가) 표시된 사진

자동 생성된 설명

Python3의 folium 지도 시각화 패키지를 사용하여 1~9월의 버스 승차 수는 빨간색, 1~9월의 버스 하차 수는 파란색, 1~9월의 지하철 승차 수는 초록색, 1~9월의 지하철 하차 수는 연두색으로 표시하였다. 지하철의 경우 승 하차 인구 위치가 대부분 겹쳐 하나로 표시되는 것을 확인할 수 있다.

지도이(가) 표시된 사진

자동 생성된 설명

앞서 파악했던 지하철역, 버스정류장의 위치를 중심으로 반경 500m를 표현해보았다. 빨간 반경이 버스정류장, 파란 반경이 지하철이며 승차 수 기준 Top 39개의 지하철역과 그 주변의 버스정류장을 대상으로 시각화 하였다. 결과적으로 서울 남부 지역에서 지하철역과 버스정류장의 영향력이 많이 겹치고 있는 것을 파악할 수 있었다.

지도이(가) 표시된 사진

자동 생성된 설명

이후 가시성 높은 시각화를 위해 아이콘을 활용해 지하철과 버스의 승 하차 Top20을 추려 표현하였다. 지하철은 서울 남부, 버스도 서울 남부와 일부 도봉로에 집중된 것을 확인할 수 있다.

지도이(가) 표시된 사진

자동 생성된 설명

* + 1. **변수기반 최종입지 선정**

Top39개로 필터링 된 환승 센터 후보지를 Top10으로 다시 선별하여 최종 3개의 입지를 선정하고자 하였다. Top10로 선별된 후보지는 쌍문역, 수유역, 미아사거리역, 고속터미널역, 서울대입구역, 신림역, 신도림역, 합정역, 홍대입구역, 강남역이다.

지도이(가) 표시된 사진

자동 생성된 설명

이들 후보지 주변 500m 반경을 그려 지하철역, 버스정류장, 버스노선의 개수를 구하기 위하여 python에서 Distance 함수를 만들었다.

텍스트이(가) 표시된 사진

자동 생성된 설명

이렇게 만들어진 Distance 함수를 기반으로 후보지 근처 500m 안에 존재하는 지하철역과 버스 정류장의 개수를 계산했다. 이를 위해 ‘서울시 버스정류소 좌표 데이터’와 ‘서울시 지하철역 데이터’를 활용하였다. 특히, 상/하행 버스 노선에 따라 다른 위치의 동일한 정류장명을 사용하는 경우가 많아 정류장 ARS-ID를 기반으로 정류장 개수와 노선 개수를 측정하였다. 예를 들어, 홍대입구역을 지나는 노선은 ['110A고려대', '마포08', '7016', '마포06', '마포05', '마포15', '110B국민대', '7734', '마포09', 270, '271A', '7711', '273', '마포16'] 로 도출되었다.

서울교통공사에서 제공한 (지하철역에서 300m 이내에 위치한) 환승 주차장 위치데이터를 기반으로 환승 센터 후보지의 500m 이내에 환승 주차장의 개수를 계산하였다. 이 또한 앞서 개발한 Distance 함수를 기반으로 환승 센터 후보지와 환승 주차장 간의 거리를 측정하였다.

지도이(가) 표시된 사진

자동 생성된 설명

기존 환승 센터 위치와의 거리 지표는 기존 환승 센터 6곳과의 후보지와의 거리 합으로 표현하였다.

테이블이(가) 표시된 사진

자동 생성된 설명

최종적으로 8개의 입지선정 기준에 대해 10개의 후보지별 값을 계산했다.

텍스트이(가) 표시된 사진

자동 생성된 설명

최종 3개의 후보지를 선정하기 위해 8개의 변수에 가중치를 적용하여 최종 Top3를 선정하였다.



# **RapidMiner 기반 입지 도출 과정**

* + 1. **1단계: Top 39 환승센터 후보지 선정**



서울시에 기존 존재하는 환승 센터의 공통점은 모두 지하철을 중심으로 형성되어 있다는 점이다. 그래서 먼저 지하철의 이용인구 데이터를 이용해서 Top39 데이터 셋을 뽑아냈다. 여기서 기존 환승 센터는 SetMinus Operator를 이용해서 제외했고 Top39 데이터 셋에다가 위치 좌표를 새로운 열로 추가시켰다.

* + 1. **2단계: RapidMiner Extension Operator 개발 및 변수기반 최종입지 선정**

8가지 기준을 구하기 위해 상용 RapidMiner 버전에 없는 새로운 Operator 7개를 개발하였다. 총 6개의 Operator를 새로 구현하였고, 이들과 기존 Operator를 이용하여 Top39개의 후보지 별 적합 Cost 함수 프로세스를 구현하였다. RapidMiner를 통한 서울시 환승센터 입지선정에서는 자체 개발한 Operator 7개를 중심으로 분석 프로세스를 구축했다.

**① Extraction Text Operator**

데이터셋의 특정 컬럼에서 괄호((), {}, [] 등의 소괄호, 중괄호, 대괄호) 안 텍스트를 자동으로 삭제해주는 텍스트마이닝 전처리 Operator이다. Operator 개발 필요성은 다음과 같다.

1. 현장에서 생성되는 Raw Data를 정제하고, 여러 데이터셋 간의 원활한 매핑, 조인을 위해
2. 수백만개의 Row로 구성된 데이터셋에서의 빠르고 효과적인 텍스트 작업

실제로 개발한 기능은 소괄호/중괄호/대괄호 등 사용자가 입력한 시작문자, 끝 문자 사이의 단어를 모두 제거하는 기능이다. 예를 들어 ‘미아사거리(4호선)’ 텍스트에 대해 ‘미아사거리’를 추출하는 것이다.

사용 방법은 다음과 같이 등록한 데이터셋을 input으로 하여 간단하게 attribute를 선택하고 해당하는 attribute 내의 괄호 안 텍스트를 삭제하는 원리이다.

|  |  |
| --- | --- |
|  |  |
|  |

**② How Many In Circle Operator**

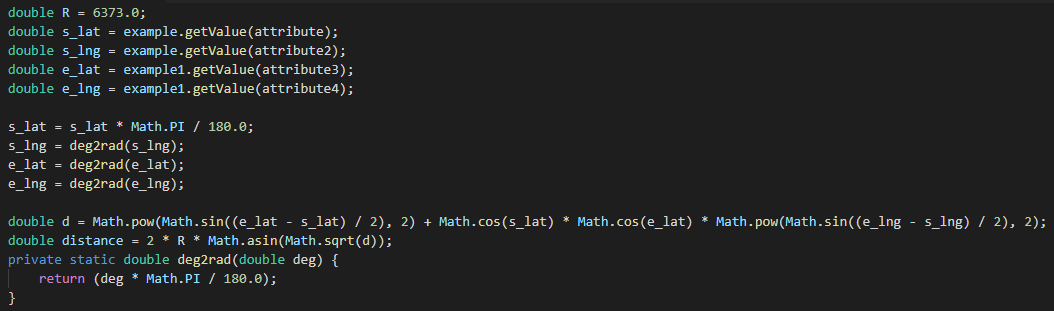
시작점의 좌표와 도착점의 좌표를 입력하고, 기준(km)과 새로운 열의 이름을 정하면 기준 원 안의 개수를 구할 수 있는 Operator이다. 개발 필요성은 다음과 같다.

1. 서울시 대중교통 데이터에서 500m 내 정류장과 지하철 역이 몇 개인지 파악하고자
2. 좌표 간 거리를 계산하기 위해

RapidMiner 자체 Operator 중 좌표를 이용하여 거리를 구하는 기능은 없고, 기준좌표 중심의 반경에 포함되는 개수를 구하는 기능 역시 존재하지 않는다. 실제로 개발한 기능은 위도와 경도를 기준으로 각각 거리가 얼마나 되는지를 구할 수 있는 역함수인 아크하버사인을 활용해 좌표 간 거리를 구하는 것이다. 해당 오퍼레이터를 통해 Top39 후보지 위치에서 500m 내의 버스 정류장 개수, 지하철 역 개수, 1.5km 내의 환승 주차장 개수 등 3가지 기준을 구할 수 있다.

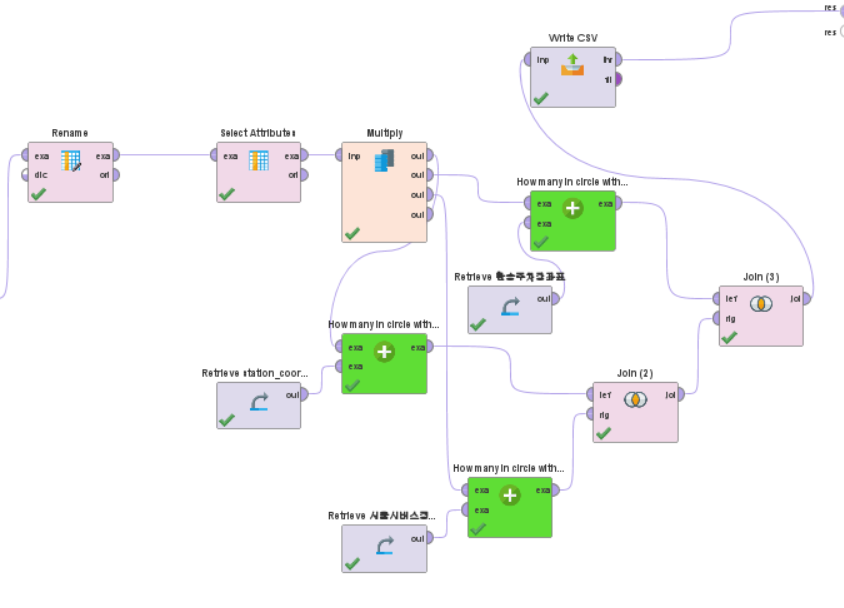
(아크하버사인 공식)

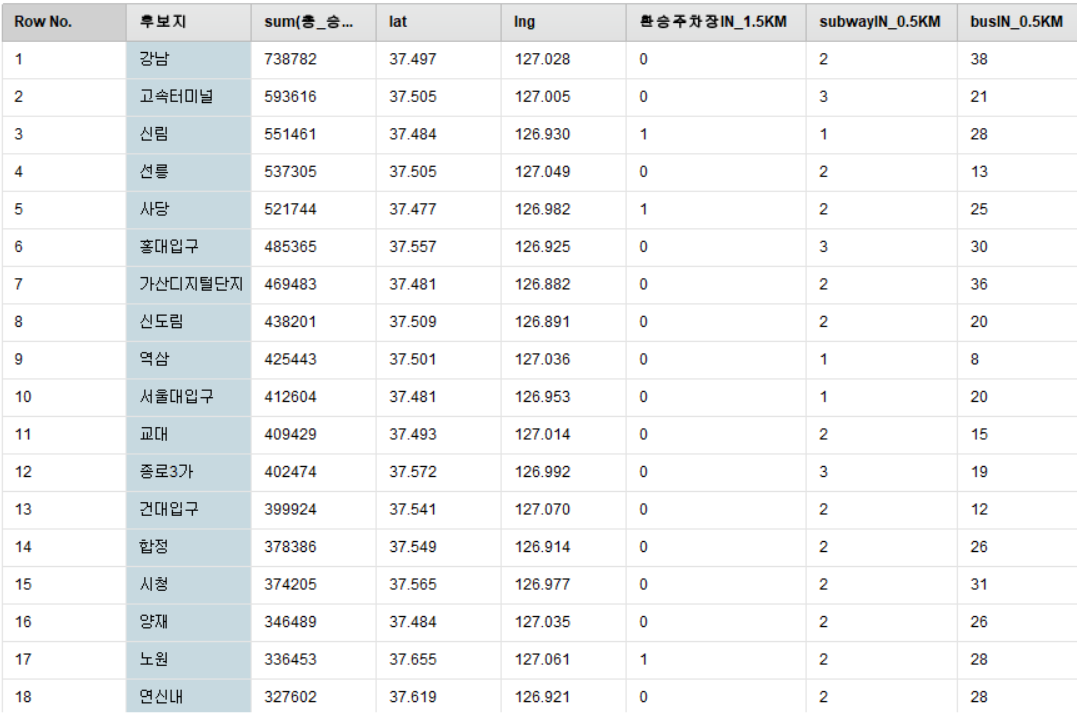
Operator 개발을 위한 아크하버사인 구현 소스코드는 다음과 같다.



|  |  |
| --- | --- |
|  |  |
|  |

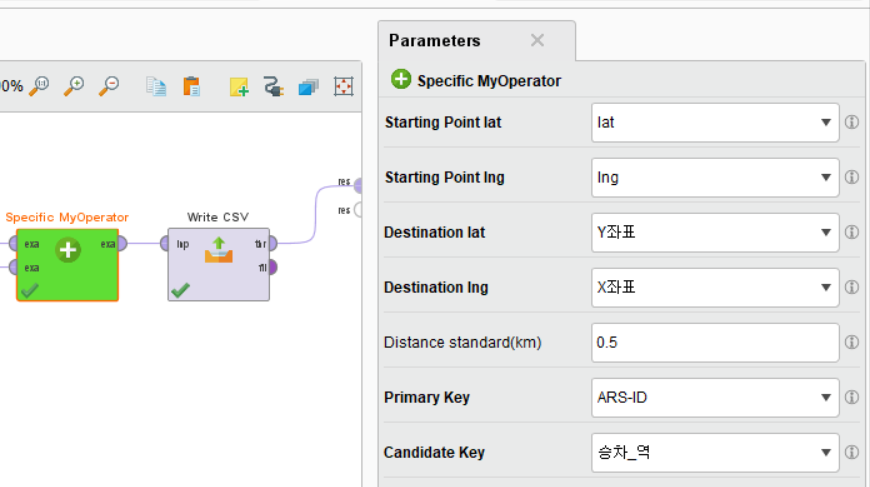
지금까지의 프로세스와 그 결과물은 다음과 같다.



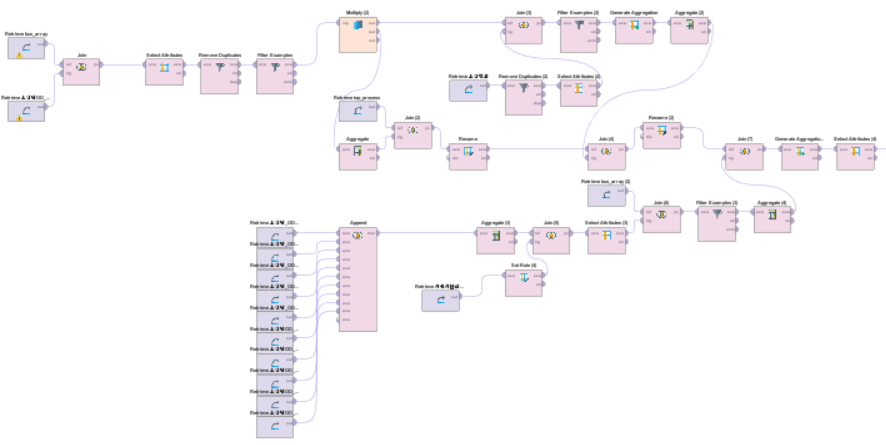


**③ Specific MyOperator**

How Many In Circle Operator를 이용하면 기준(km)안의 개수를 구할 수는 있지만, 어떤 원소가 포함되는지 그 원소 리스트를 배열로 뽑아낼 수 없으므로 새로운 Operator를 개발했다. 이 Operator의 핵심 원리는 새로운 행의 추가이다. Matrix 형식의 데이터셋에 배열을 삽입할 수 없는 한계점이 있으므로 OS Process에서 착안하여 데이터가 나올 때마다 행을 추가해서 마치 배열이 저장되는 것처럼 구현하였다.



Operator 활용 프로세스는 다음과 같다.



이러한 과정을 거쳐 환승 센터 후보지의 반경 500m 내에 지나가는 버스 노선의 개수, 후보지를 지나가는 노선의 길이 \* 총 운행횟수, 후보지 이용인구 등을 구했다.

**④ Distance Metric Operator**

환승 센터 후보지들과 기존 환승 센터들 과의 거리 합을 계산하기 위해 새로운 Operator를 개발하였다. 출발 좌표와 도착 좌표를 이용하면 거리 계산이 가능하며 거리들의 총합을 구하도록 Operator를 만들었다.

|  |  |
| --- | --- |
|  |  |
|  |

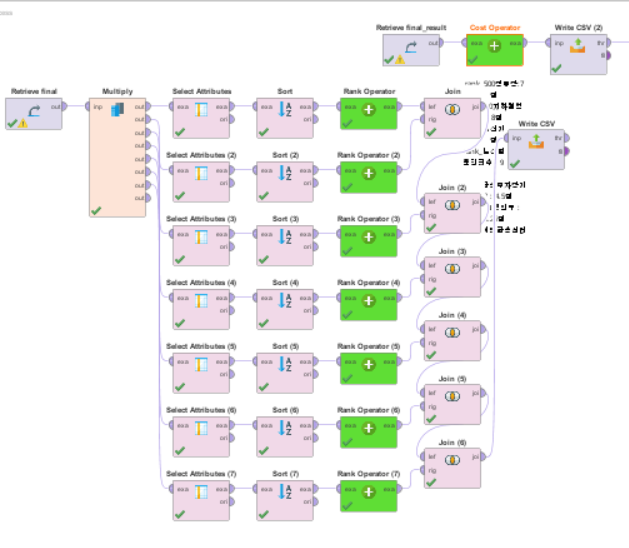
이렇게 4개의 Operator를 추가 개발하여 총 7개의 변수에 해당하는 후보지 값을 도출했다.



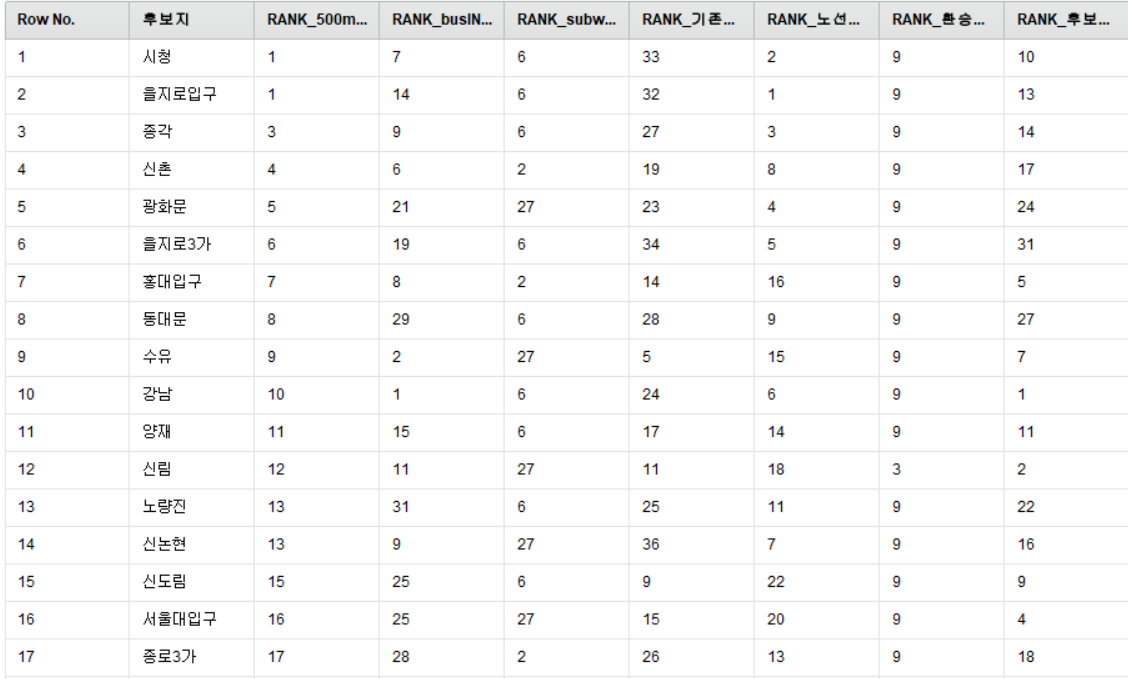
**⑤ Rank Operator**

|  |  |
| --- | --- |
|  |  |

지금까지 계산한 총 7개의 변수 별 등수(Ranking)를 구하고, 그 등수의 열을 합치는 과정이다. 사용 방법은 Attribute를 설정하여 해당 attribute의 등수를 동점까지 분류한다. 기존 RapidMiner에서 제공하는 Rank Operator는 나의 등수를 새로운 열로 나타내기 위해 generate ID 등을 이용하여 입지선정과 같은 여러 데이터를 다루는 상황에서는 적합하지 않았다. 이를 보완하여 동점자까지 자동으로 처리하여 각 열의 Rank를 새로운 열로 만들어주는 Operator를 고안하였다.



해당 오퍼레이터를 사용하여 변수별로 Ranking 한 결과이다.



**⑥ Cost Operator**

사용자가 원하는 가중치(Weight)를 곱하여 여러 항목을 통합하는 단일 결과를 낼 수 있는 Operator를 개발하였다. 여러 변수들을 입력 값으로 하는 cost 함수를 이용해 각종 변수들과 가중치를 입력 받아 변수와 가중치를 곱해 하나의 점수로 만들고 최종 후보지를 도출하는 과정이 필요하다.

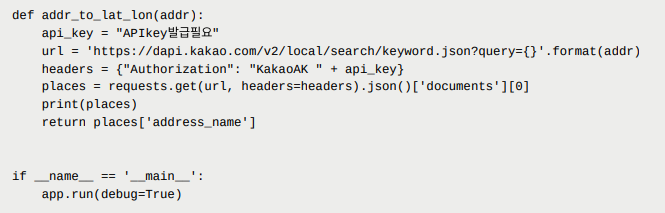
Cost 함수에는 현재 500m 내 정류장 개수, 지하철 역 개수, 버스 노선 개수, 버스 노선 빈도, 환승 주차장 개수, 주변 차선 수, 지하철/버스 승 하차 인구, 기존 환승 센터까지의 거리 총 8개의 변수와 각각의 변수마다의 가중치가 포함되었다.

|  |  |
| --- | --- |
|  |  |

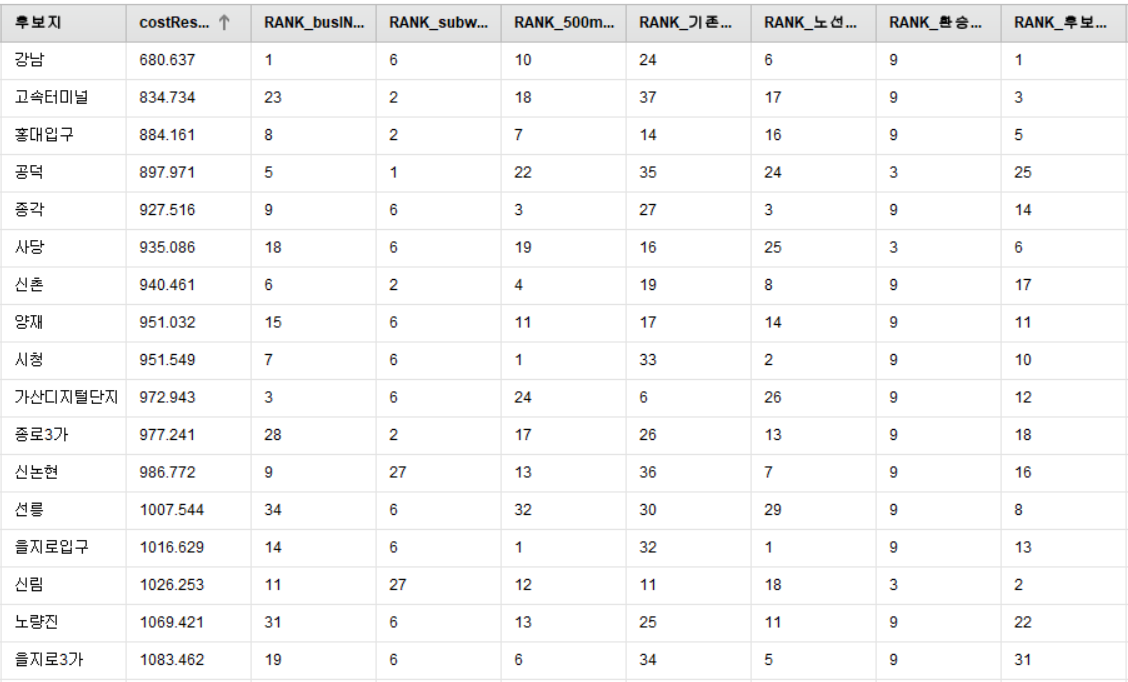
**⑦ Coords To Address Match Operator**

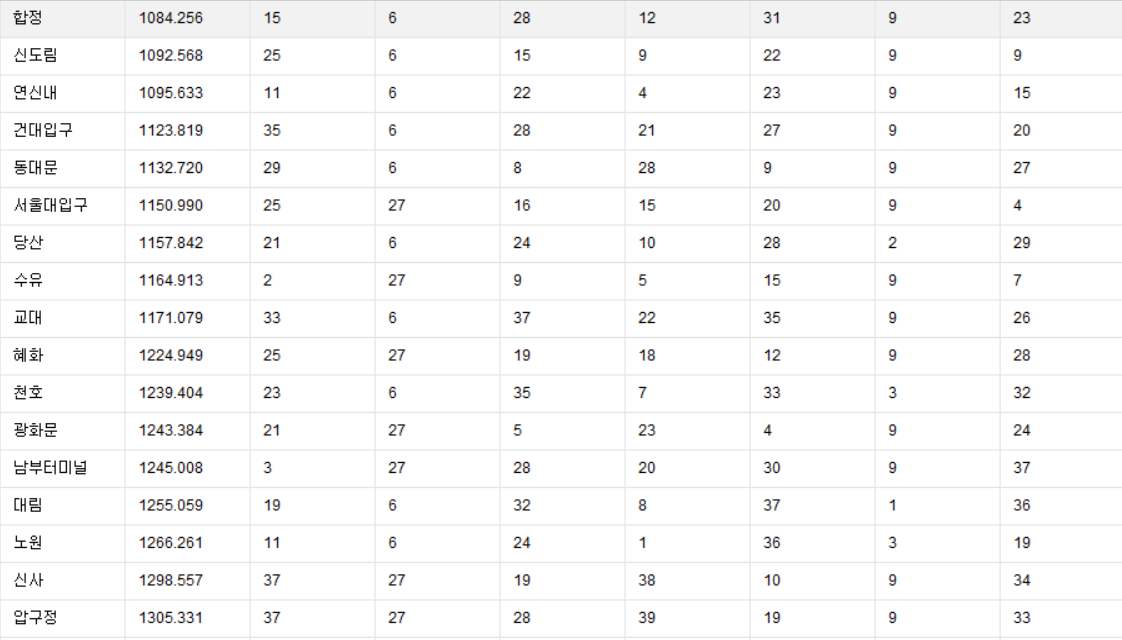
좌표 데이터를 이해하기 쉬운 지번(도로명) 주소로 변환할 수 있는 Operator이다. 사용자는 좌표 데이터로 위치를 파악하는 데 어려움을 겪으며 공간 데이터에 대한 접근 용이성이 필요하므로 Operator를 자체 개발하였다.

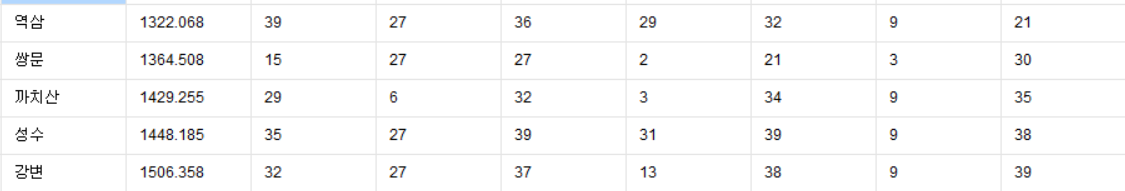
MAPS API에서 제공하는 좌표 데이터를 지번(도로명) 주소로 매칭하여 반환한다. 로컬 Operator에서 Gradle로 등록한 외부 라이브러리가 RapidMiner의 Extension에 직접 추가되지 않는 이슈가 발생했다. 이를 위해 Python Flask를 통해 API 형태의 서버를 띄우고 원하는 값만 입력하여 사용 가능하도록 하였다.



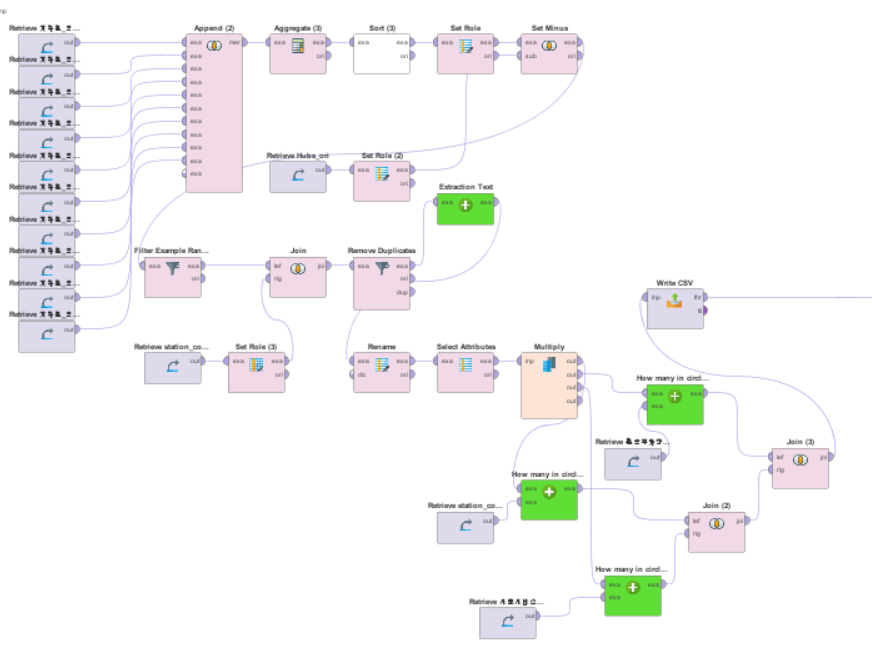
7가지 Operator를 개발하여 다음과 같은 최종 결과물이 도출되었다.

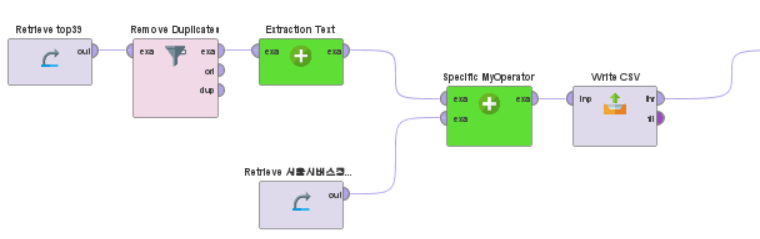


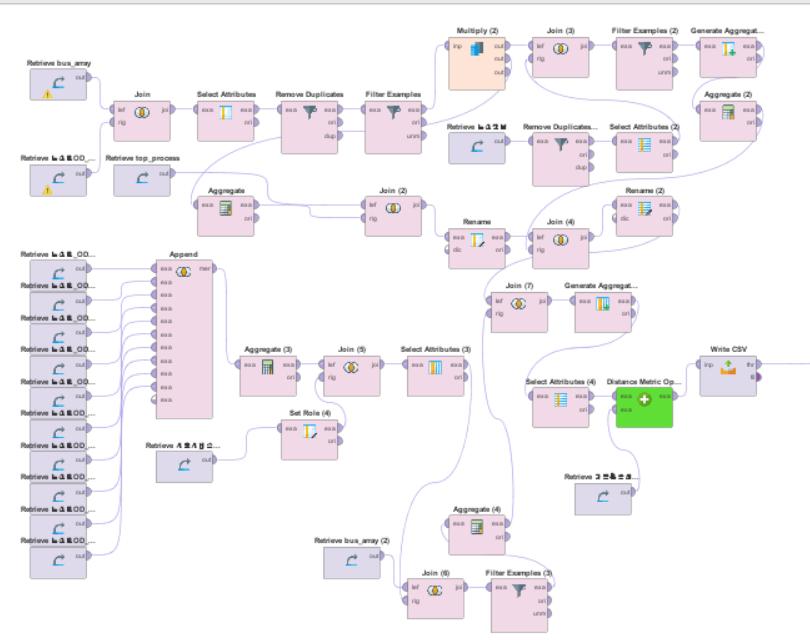


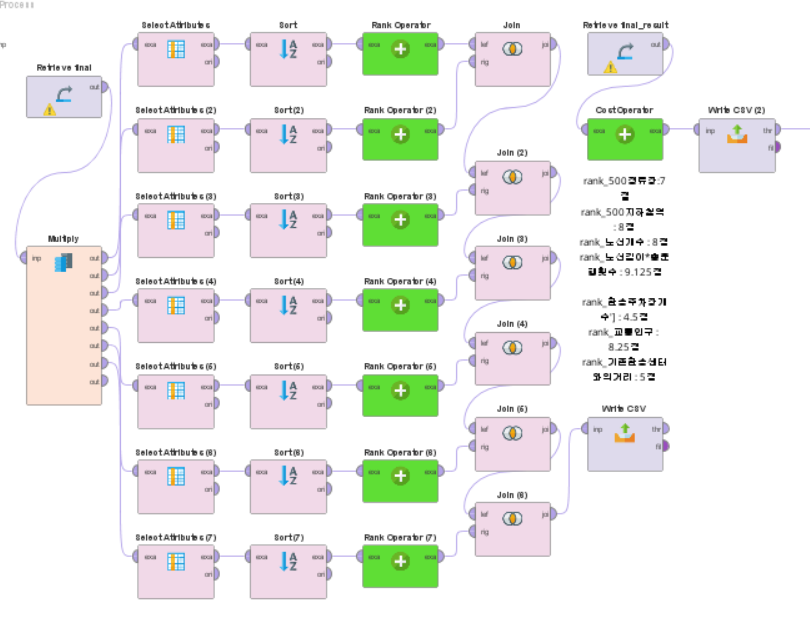


RapidMiner 전체 Process









# **결론**

# **Python과 RapidMiner 기반 입지 도출과정 비교 및 분석**

서울시의 대중교통 환승 센터에 적합한 지점을 찾고자 두 가지 기술을 사용하여 분석하고 결론을 도출하였다.

RapidMiner의 경우 기존에 존재하는 Operator만 이용하게 되면 빠르게 가시적인 결과를 확인할 수 있어 시각화와 편리함에 높은 점수를 부여할 수 있다. 그러나 Operator 들의 제한으로 인해 RapidMiner로 세밀한 데이터 전처리에는 다소 어려움을 겪었다. 또한 필요한 Operator가 없을 경우 직접 개발해야 한다는 점이 이슈가 될 수 있다.

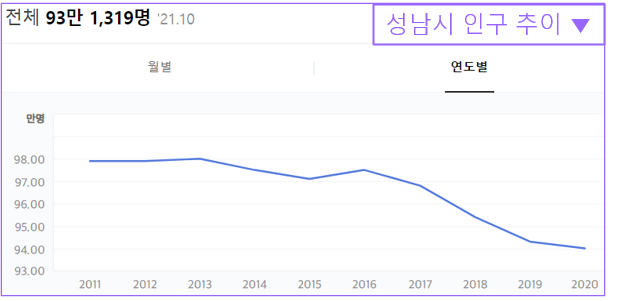
반면, Python3로 모델을 만드는 과정에서는 정교하고 세밀하게 코드를 작성하며, 여러 유용한 패키지를 손쉽게 가져올 수 있었다. 결과적으로 데이터와 분석과정에 대한 이해도가 높다면 Python을 활용하고, Operator의 다양성만 확보된다면 초보자에겐 RapidMiner가 더 활용하기 좋다고 판단된다.

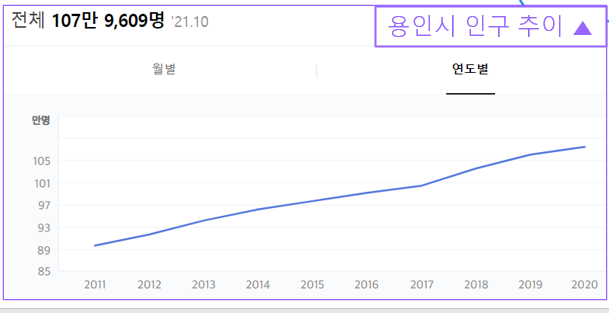
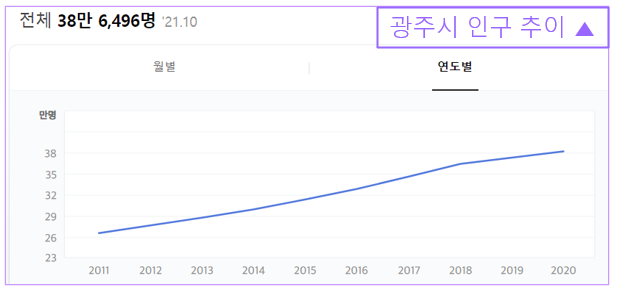
# **최종 환승 센터 입지 선정에 대한 결론 도출 및 활용방안 제시**

결과적으로 Cost 함수 결과값이 제일 작은 역인 강남, 홍대입구, 수유 3개의 최종 후보지에 대한 조사를 추가적으로 진행하여 환승 센터 입지로서의 데이터 분석 결과를 뒷받침하였다.

### **강남역**

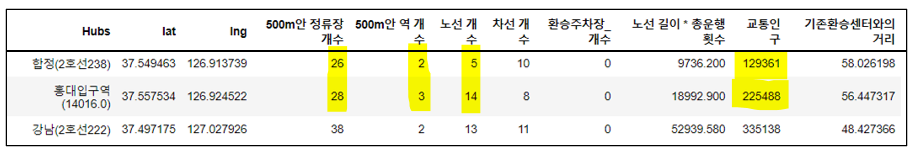
경기도에서 직접적으로 강남을 거쳐 서울에 유입되는 경기도 광주, 용인의 인구추이를 확인해본 결과 인구 추이가 늘고 있었고 성남의 인구는 줄어드는 추세이나, 여전히 많은 인구가 있었다.

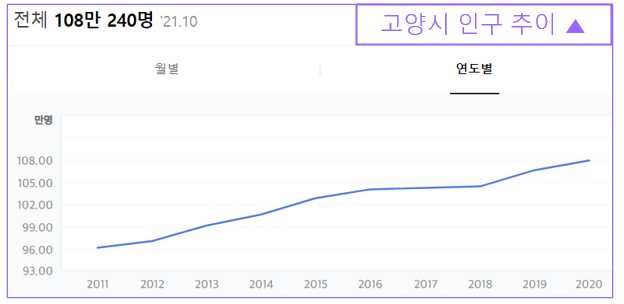




* + 1. **홍대입구역**

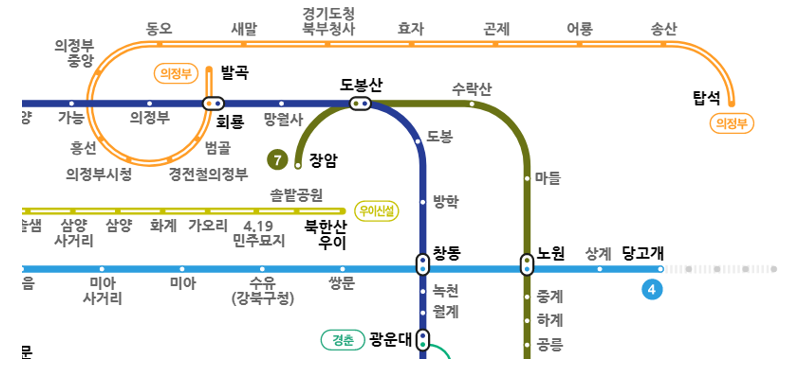
홍대입구는 홍대 인근 역인 합정도 후보지 내에 있어 이 둘의 수요가 합쳐지면 더욱 영향력 있는 후보지가 될 수 있다. 추가로 신도시 계획 지역인 고양은 경의중앙선과 공항철도를 이용하므로 두 지하철이 만나는 홍대입구가 중요한 요충지가 될 것으로 보이며 고양시 인구추이 또한 계속해서 증가하고 있다.

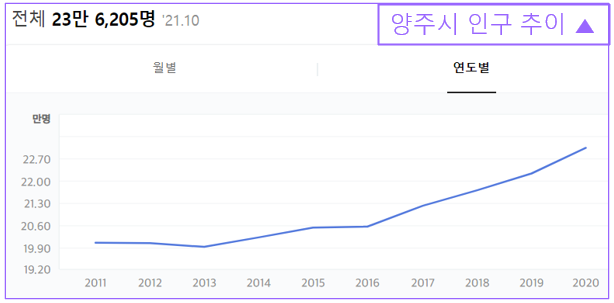
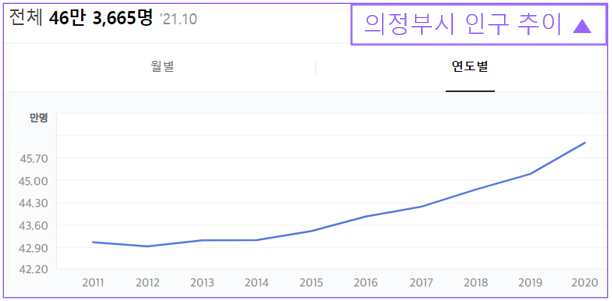




* + 1. **수유역**

인근에 후보지인 쌍문역과 미아사거리가 있어 이를 한 지점으로 고려한다면 매우 경쟁력 있는 후보가 될 수 있다. 또한 보다시피 경기 북부의 의정부와 양주에서 서울을 진입하는 거점지로 해당 시의 인구 추이 증가와 더불어 노원, 도봉, 강북 인구를 통한 유동인구 증가가능성이 있다.



# **부족했던 점 및 앞으로의 과제**

프로젝트를 진행하면서 여러 문제점이 있었다.

RapidMiner Tool 사용 시에는 RapidMiner의 Join Operator를 사용했으나, Open Source 버전에서의 Join Operator 구현이 되어있지 않아 분석에 어려움이 있었다.

또한 변수 선정에 보다 많은 시간을 들였다면 실제로 본 연구에서 정한 8개의 변수들이 입지 선정에 얼마나 중요한 영향을 미치는지는 정확하게 파악할 수 있었을 것이다.

그러나 서울시 환승 센터의 입지선정 과정 뿐만 아니라 데이터 분석이 가능한 여러 Tool을 사용해보며 분석 용이성과 접근성, 분석 프로세스에 대해 깊이 있게 이해할 수 있었다. 직접 데이터를 기반으로 선정한 변수들을 고려하고 변수들의 값과 가중치를 생각하여 하나의 함수로 개발하고, 이를 통해 결과값을 도출해보았다는 것에 의의를 가지고 프로젝트에 임할 수 있었다.

이러한 변수들을 생각해 봄으로써 추가적인 변수들을 구상하고 확장해 나가는 것은 이전에 변수들을 만들지 않았을 때보다 유리하게 작용하여, 앞으로 본 연구와 프로젝트에 살을 붙여 확장해 나가는 것이 수월할 것으로 예상된다.