대구 교통사고 피해 예측 AI 경진대회

2023-2 KUBIG Conference

18기 신인수 이은준 정윤주 정해원

목차

데이터 소개


DACON
커뮤니티
대회
학습
랭킹
더보기
구독
De
대구 교통사고 피해 예측 AI 경진대회
알고리즘 | 정형 | 회귀 | 교통 | RMSLE | 정성평가
상금 : 1,000만원
2023.11.15 ~ 2023.12.11 09:59
+ Google Calendar
1,782명
마감
참여중
대회안내
데이터
코드 공유
토크
리더보드
제출
개요
규칙
일정
상금
동의사항
[배경]
이동수단의 발달에 따라 다양한 유형의 교통사고들이 계속 발생하고 있습니다.
한국자동차연구원과 대구디지털혁신진흥원에서는 해당 사고의 원인을 규명하고 사고율을 낮추기 위해,
시공간 정보로부터 사고위험도(ECLO)를 예측하는 AI 알고리즘 발굴을 목표로 본 대회를 개최합니다.
※ ECLO(Equivalent Casualty Loss Only) : 인명피해 심각도
• ECLO = 사망자수 * 10 + 중상자수 * 5 + 경상자수 * 3 + 부상자수 * 1
• 본 대회에서는 사고의 위험도를 인명피해 심각도로 측정

데이터 소개

주제: 대구 교통사고 피해 예측 AI 경진대회

목표: 시공간 데이터를 바탕으로
인명피해 심각도(ECLO: Equivalent Casualty Loss Only) 예측

*ECLO = 사망자수 * 10 + 중상자수 * 5 + 경상자수 * 3 + 부상자수 * 1*

회귀 문제 → **ECLO ≥ 0 예측이 관건**

Training Set - Data from 2019/1/1 ~ 2021/12/31

Variables :

- ID
- 사고일시
- 요일
- 기상상태
- 시군구
- 도로형태
- 노면상태
- 사고유형
- 사고유형 - 세부분류
- 법규위반
- 가해운전자 차종
- 가해운전자 성별
- 가해운전자 연령
- 가해운전자 상해정도
- 피해운전자 차종
- 피해운전자 성별
- 피해운전자 연령
- 피해운전자 상해정도
- 사망자수
- 중상자수
- 경상자수
- 부상자수
- ECLO

Testing Set - Data of year 2022

Variables :

- ID
- 사고일시
- 요일
- 기상상태
- 시군구
- 도로형태
- 노면상태
- 사고유형

→**Training set에 비해 변수의 개수가 감소**

제공된 외부 데이터

전국 교통사고 데이터: 대구를 제외한 2019-2021 교통사고 데이터

→ 형식은 대구시 교통사고 training set과 동일

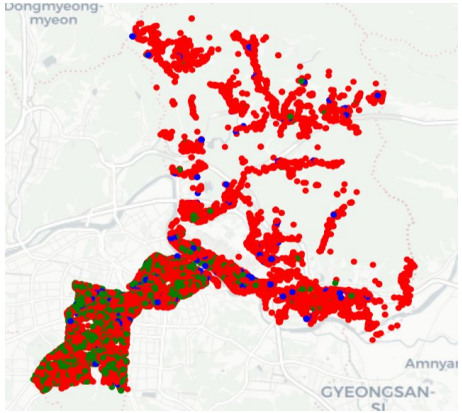
대구시 보안등 정보: 주소, 위경도, 대수, 설치연도, 설치형태(한전주, 건축물, 전용주)

대구시 어린이보호구역 정보: 주소, 위경도, 관할경찰서명, CCTV 설치 여부와 대수, 보호구역 도로폭

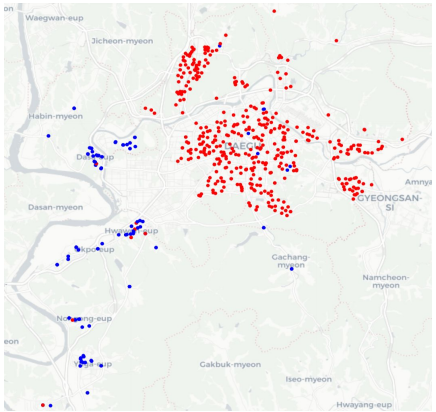
대구시 주차장 정보: 주소, 위경도, 급지 구분(1, 2, 3급지 - 가까운 역과의 거리), 요금 구분

대구시 CCTV 정보: 주소, 위경도, 설치용도, 제한속도

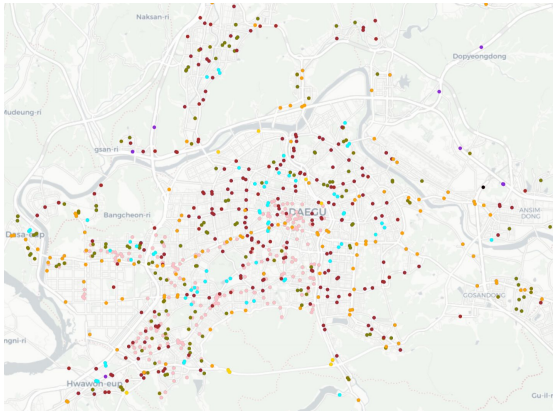
탐색적 자료분석: 외부데이터



설치 형태별 보안등 분포

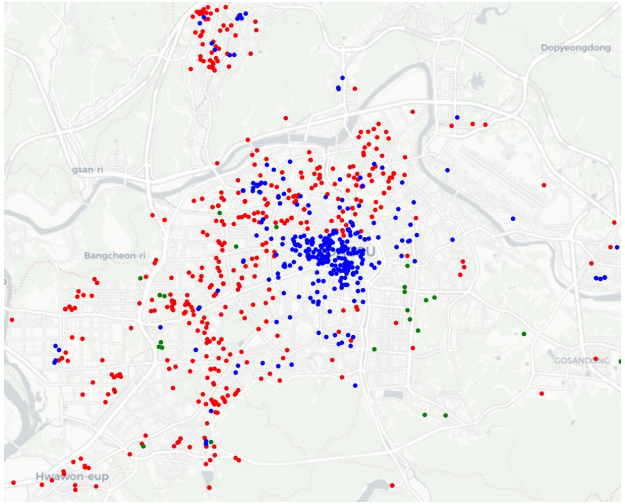


CCTV 설치여부별 어린이보호구역 분포

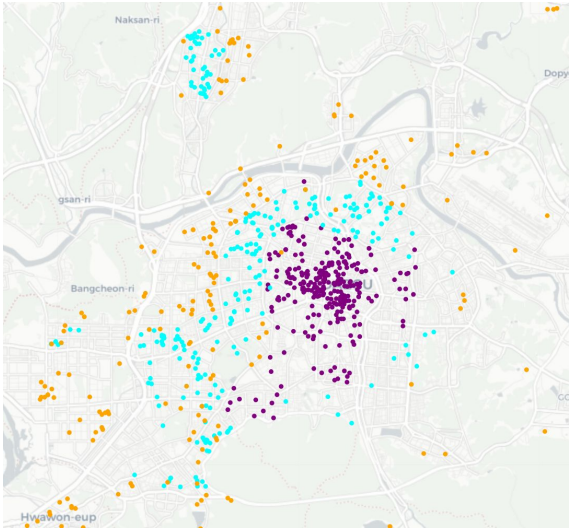


단속구분, 제한속도별 CCTV 분포

탐색적 자료분석: 외부데이터



요금구분별 주차장 분포



급지구분별 주차장 분포

→ 외부 데이터와 교통사고 ECLO간의 correlation 적음, 규칙성 부재
→ 모델 구축시 변수에서 제외

데이터 전처리 - training set(1)

시군구에 따른 데이터 간의 상관관계 분석을 위해 **각 읍,면,동으로 data split** 후
ECLO와의 상관관계 를 바탕으로 공간 분석 진행

model 정확도를 높이기 위해 **testing set에 추가적인 데이터 적용**

- 대구 보안등 정보
- 대구 어린이 보호 구역 정보
- 대구 주차장 정보
- 대구 CCTV 정보
- countrywide_accident (전국 교통사고 데이터)

시간 변수들에 대해 **시계열 분석** 진행 후 (크게 시계열 패턴이 드러나지 않았음)

Year, Month, Day, Hour로 변수 분리

범주형 데이터에 대해 **Binary encoding 진행**

데이터 전처리 - training set(2)

결측치에 대해 전처리 진행

- Median으로 대체
- 기존 데이터의 비율과 동일하도록 데이터 대체
- Drop, Fillna

ECLO와 상관관계가 낮은 데이터는 제외 (ID, 시군구 등)

데이터 및 변수의 개수가 많다고 판단, **PCA 진행**

- 6 Components → 99% Total Variance Explained

ECLO와 직접적으로 상관있는 컬럼 제외

- 사망자수, 중상자수, 경상자수, 부상자수 데이터 제외

	도로형태	노면상태	사고유형	사고유형 - 세부분류	법규위반	가해운전자 차종
도로형태	1.000000	0.030948	0.608721	0.340617	0.226331	0.039821
노면상태	0.030948	1.000000	0.027189	0.034197	0.018401	0.018099
사고유형	0.608721	0.027189	1.000000	0.856772	0.285470	0.130710
사고유형 - 세부분류	0.340617	0.034197	0.856772	1.000000	0.235829	0.087290
법규위반	0.226331	0.018401	0.285470	0.235829	1.000000	0.064231
가해운전자 차종	0.039821	0.018099	0.130710	0.087290	0.064231	1.000000
가해운전자 상해정도	0.052335	0.011330	0.199957	0.169590	0.068215	0.309903
피해운전자 차종	0.115481	0.026585	0.819010	0.322366	0.162897	0.091559
피해운전자 상해정도	0.079431	0.022052	0.556618	0.337041	0.120655	0.215538

	가해운전자 상해정도	피해운전자 차종	피해운전자 상해정도
도로형태	0.052335	0.115481	0.079431
노면상태	0.011330	0.026585	0.022052
사고유형	0.199957	0.819010	0.556618
사고유형 - 세부분류	0.169590	0.322366	0.337041
법규위반	0.068215	0.162897	0.120655
가해운전자 차종	0.309903	0.091559	0.215538
가해운전자 상해정도	1.000000	0.177645	0.241734
피해운전자 차종	0.177645	1.000000	0.459103
피해운전자 상해정도	0.241734	0.459103	1.000000

데이터 전처리 - testing set (1)

범주형 변수: train set의 카테고리 별 비율과 동일하도록 대체

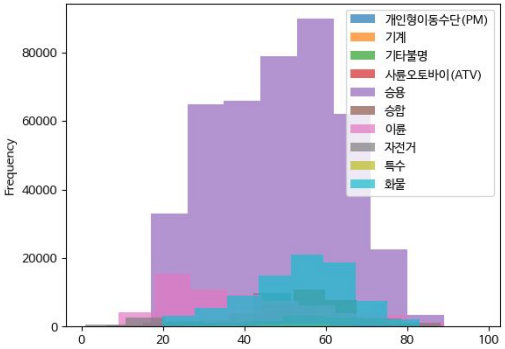
- 사고유형별 가해자 상해정도
- 사고유형별 피해자 상해정도
- 가새자 상해정도별 가해운전자 차종
- 피해운전자 상해정도별 피해운전자 차종
- 피해운전자 상해정보별 성별
- 법규위반(사고유형별)

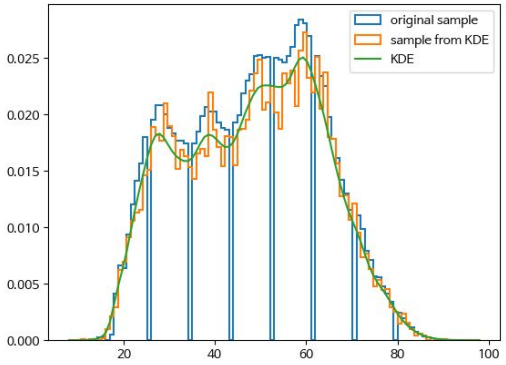
→ Model Accuracy 상승 목적

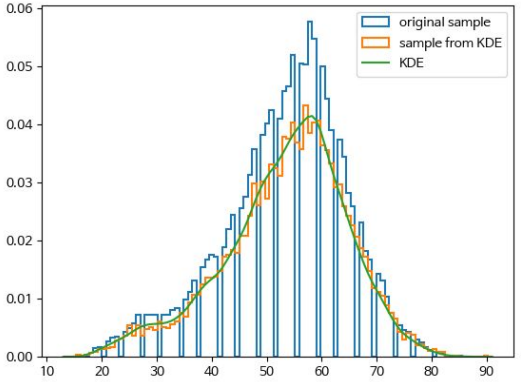
연속형 변수: 차종 별 연령 분포를 경험적 누적 분포함수(empirical CDF)를 구한 후 resampling Gaussian Kernel을 이용하여 분포함수 생성

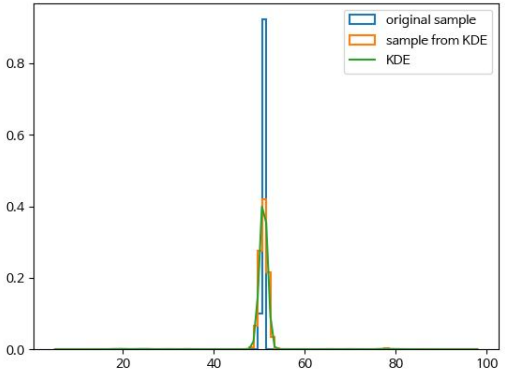
- 가해자 차종 별 연령
- 피해자 차종 별 연령

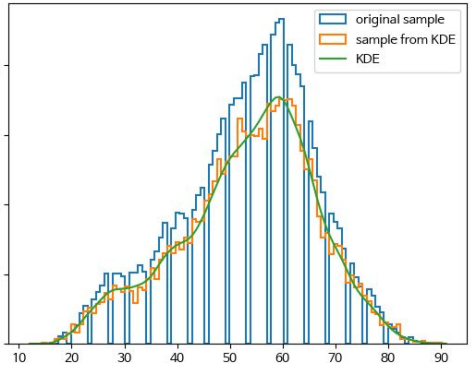
데이터 전처리 - testing set (2)

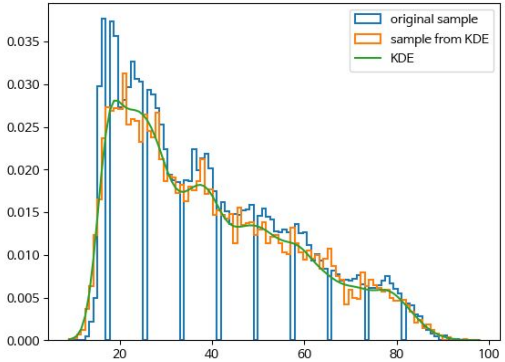




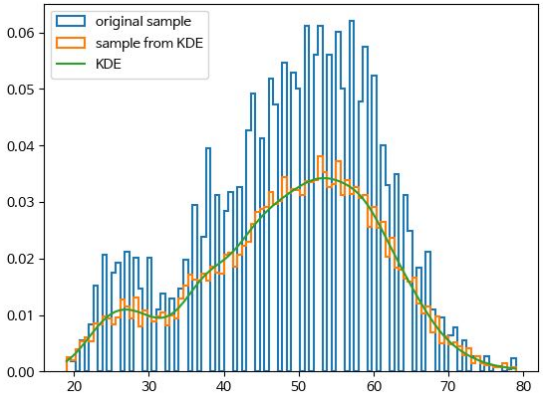


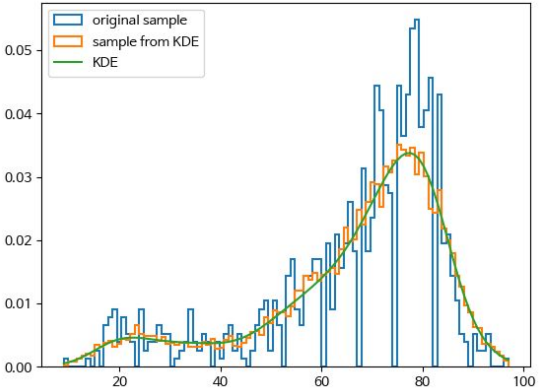


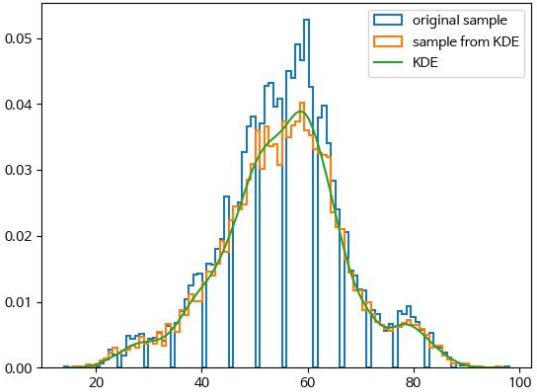




데이터 전처리 - testing set (2)







모델 선정 - Multi-output Regression

*ECLO = 사망자수 * 10 + 중상자수 * 5 + 경상자수 * 3 + 부상자수 * 1*

→ ECLO가 아니라 **사망자수, 중상자수, 경상자수, 부상자수를 예측하는 Multi-output regression으로 간주**

→ sklearn.multioutput로부터 **다변수 회귀를 한 후 예측값에 가중치를 곱하여 ECLO를 예측**

→ **Multivariate linear regression, multivariate gradient boosting regression 모두 음수 ECLO 값 예측**

→ **단변수 회귀 문제로 설정하여 모델 설정**

모델 선정 - PyCaret

PyCaret : RMSLE 기준 가장 optimal한 model인 **Huber Regressor로 모델 선정**

```
top5 = compare_models(n_select=5, sort='RMSLE')
```

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
huber	Huber Regressor	2.0577	10.1298	3.1778	-0.0391	0.4508	0.5297	0.0500
gbr	Gradient Boosting Regressor	2.1368	9.7080	3.1111	0.0039	0.4629	0.6294	2.8200
lr	Linear Regression	2.1366	9.7407	3.1164	0.0006	0.4639	0.6308	0.6190
ridge	Ridge Regression	2.1366	9.7407	3.1164	0.0006	0.4639	0.6308	0.0160
lar	Least Angle Regression	2.1366	9.7407	3.1164	0.0006	0.4639	0.6308	0.0170

모델 선정 - PyCaret

PyCaret : **PCA 처리** 후 RMSLE 기준 가장 optimal한 model인 **LGBM 로 모델 선정**

```
top5 = compare_models(n_select=5, sort='RMSLE', exclude=['lr', 'ada'])
```

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
lightgbm	Light Gradient Boosting Machine	1.4969	6.3412	2.5106	0.3244	0.3227	0.3191	
gbr	Gradient Boosting Regressor	1.5024	6.1879	2.4785	0.3420	0.3233	0.3388	
rf	Random Forest Regressor	1.4776	6.8995	2.6198	0.2638	0.3367	0.3089	
xgboost	Extreme Gradient Boosting	1.5368	6.8847	2.6194	0.2611	0.3422	0.3390	
et	Extra Trees Regressor	1.5200	7.9545	2.8152	0.1479	0.3554	0.3136	
dt	Decision Tree Regressor	1.5311	8.3630	2.8872	0.1041	0.3587	0.3163	
br	Bayesian Ridge	1.6981	7.0155	2.6420	0.2514	0.3760	0.4109	
ridge	Ridge Regression	1.6986	7.0153	2.6420	0.2514	0.3764	0.4101	

결과

PCA를 진행하지 않은 데이터보다 **PCA를 진행한 데이터의 정확도가 높음**

추가적인 데이터 사용 (보안등, 어린이 보호 구역, 주차장, CCTV 정보) 시 정확도 감소

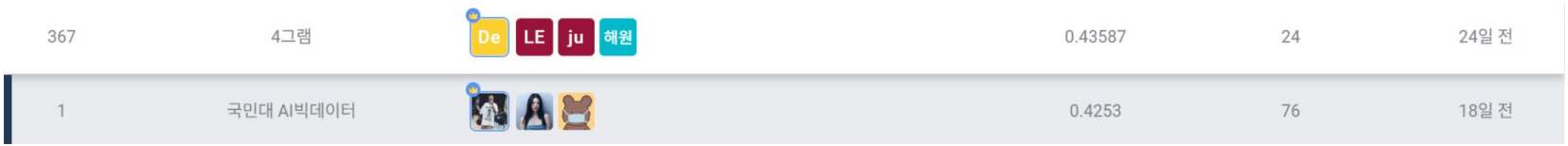
전국 교통사고 데이터 정보를 포함해 training시, 정확도 상승

Pycaret 모듈 활용 → RSMLE 기준 huber regressor의 정확도가 제일 높았음

Pycaret 모듈 활용 + PCA → RSMLE 기준 LGBM 의 정확도가 제일 높았음

사고 일시 데이터를 시계열로 data split을 했을 때 정확도가 더 높았음

예측값이 주로 **ECLO 평균으로 회귀하는 현상 발생**



결론 및 논의점

데이터 분석 과정에서 **Overfitting의 영향**성이 있었던 것으로 판단되어 구체적으로 어떤 부분에서 영향성이 컸을지에 대한 논의 필요

범주형 데이터에 대한 컬럼이 많았는데, 이 부분을 적절히 선택 및 처리해야했으나, 선택여부 판단 기준이 명확하지 않았기에 해당 부분 보완 필요

추가적인 데이터 활용을 위해서는 대구 빅데이터활용센터를 직접 방문해야 했으나, 시공간적인 한계로 방문하지 못했음.

다른 팀의 경우 Multilayer perceptron으로 많이 진행 → 후속 프로젝트에 사용하면 좋을 듯

training set보다 testing set의 변수 개수가 현저히 적어, 적절한 모델을 만들기가 어려웠음.

한 학기동안 배운 머신러닝 기법으로 최선의 모델을 선정한 의의가 있었음.