

(6주차) 머신러닝 모델링

✓ 2023.05.29 ~ 06.04 (6/2 금 1차발표) 🔗

✓ 5/29, 5/31 (홍봉희)

- 특허식으로 위험여부 판별하고 위험운전계수 라벨링
 - 5(매우위험), 4(위험), 3(보통), 2(안전), 1(매우안전)

♥ Iteration 2 : 개발 계획 🔗

💬 데이터 분석 🔗

- 급회전 경우 구해야됨!!!!
 - 조합방법 :
 - 가속(급회전이 같이 발생하면 *, 단독 발생발생하면 +)
 - (+) 감속(급회전이 같이 발생하면 *, 단독 발생발생하면 +)
 - 시공간에 대한 위험운전 점수 등 표시 => 파라미터로 시간(선택) + 공간은 내가 좌표 찍은 곳
- 머신러닝(xgboost 어찌구)

💬 백엔드

- 로그인 기능 구현
 - 회원가입, 회원탈퇴, 회원정보관리, 로그인
- 파일 전송 관련 (안)
 - 프론트에서 csv 파일 업로드 하면 파일 내용 풀어서 json 형태로 스프링부트로 전송
 - 스프링부트 데이터베이스에 입력하고 (플라스크로 부터 받을 데이터는 null값 입력) 플라스크로 전송
 - 플라스크에서 칼럼 만들어서 스프링부트로 응답
 - 스프링부트에서 데이터베이스 입력하고 리액트로 응답 (기존 구현했던거 참고)
- 예측모델링 api
 - 리액트에서 시간과 좌표정보로 요청하면 스프링에서 DB 로그 기록 후 플라스크로 전송
 - 플라스크에서 예측값 응답
 - 스프링에서 DB 로그 기록 후 리액트로 응답

💬 프론트엔드

- 로그인 기능 관련
 - 로그인 폼, 회원가입 폼
 - 회원정보관리 페이지
 - 로그인을 하면 페이지 이동할 수 있도록 하기!
- 파일 전송 관련 화면 개발
 - 업로드할 수 있는 폼 만들기
 - 결과 화면은 기존 지도랑 마커 화면 가져오기 (차트는 필요없쥬)

- 예측모델링
 - 시간 선택할 수 있는 셀렉트 박스 만들기
 - 지도에서 좌표 찍을 수 있게 하고
 - 예측결과 응답이 오면 커스텀 오버레이 생성
 - 예상 위험운전지수 xxx 구간입니다! 조심하셈!!

가속 기준 xgboost 분류(5/30)

- 정상주행의 경우가 대부분, 정상주행 데이터 빼고 돌림

```
y_pred = model.predict(X_test)
```

Executed at 2023.05.30 14:21:51 in 51ms

```
accuracy_score(y_pred, y_test)
```

Executed at 2023.05.30 14:21:51 in 15ms

0.9658553466374438

```
from sklearn.metrics import f1_score
```

```
f1_score(y_test, y_pred, average='weighted')
```

Executed at 2023.05.30 14:21:51 in 46ms

0.9501743099140395

- 결과
 - accuracy score: 0.96..
 - f1 score: 0.95..
- 말도 안되는 결과지만 일단 돌리라고 하니까 돌려봄...