EDA

- data에 대해 깊은 이해를 하는 과정
- domain
- understand how the data was generated
  - 적절한 validation을 위해
- Exploring anonymized data
  - 각 컬럼의 정보, data의 값을 임의의 값으로 바꿔서 제공하는 경우가 많다.
  - 파악하기 어려운 경우 baseline model을, EDA로 연구해야한다.
  - explore individual featrure
    - column의 의미와 type에 대해 파악하라
    - `df.dtypes, df.info(), value_counts()`
  - explore featrure relation
    - columns 끼리의 관계 파악하라
  - Dataset cleaning
    - duplicated feature
    - constant feature(`train.nunique(axis=1)==1`) 찾아내기
    - duplicated rows
    - check if dataset is shuffled

validation and overfitting

- validation type
  - holdout
    - ngroups = 1
    - data가 많을 때 사용가능
    - `sklearn.model_selection.ShuffleSplit`
  - k-fold
    - ngroups = k
    - 주로 data가 부족하다고 느낄 때 사용한다.
    - 여러개의 fold로 나눈뒤에 해당 score를 평균내어 사용한다.
    - 당연히 시간이 더 걸린다.
    - `sklearn.model_selection.Kflod`
- validation set을 만들때 Stratification 필요한 경우
  - small dataset
  - unbalanced dataset
  - multiclass classification
- Data splitting strategies
  - time-based split이 필요한지 random 해도 되는지 잘 살펴보자.
    - Random, rowwise
    - Timewise
    - By id
  - train/test와 비슷한 상황을 만들어주는 validation set을 만들자.
- Problems occuring during validation
  - validation stage
    - causes of different scores and optimal parameters
      - 1. Too little data
      - 2. Too diverse and incosistent data
  - submission stage
    - LB score가 validation score 보다 높거나 낮을때

        - LB와 validation score랑 corr하지 않을때
    - why?
        - Kfold score들이 비슷한지 확인하라.
        - train/test의 분포가 다른 경우 존재
            - leaderboard probing : 예를 들어, LB data가 train의 target 보다 평균적으로 1이 크다면 train prediction에 1을 더한다.
        - LB data의 수가 너무 적은 경우 존재
- Data leakage
    - Leaks in time series
        - 시간에 따라 split해야한다.
        - split 잘해도 미래에 대한 정보를 갖고 있는 경우가 많다. 그러면 leak이니까 조심하자.
    - 대회이외의 외부로부터 얻게 되는 정보
    - leaderboard probing