Machine Learning

Programming the logic (= define pattern)

vs.

Machine learning through data (= hidden pattern recognition)

---

scikit-learn modules

Data processing

> `sklearn.preprocessing`

1. `LabelEncoder`
2. `OneHotEncoder` (`pd.get_dummies`)

> `sklearn.feature_selection`

> `sklearn.decomposition`

Validation & Tuning

> `sklearn.model_selection`

최적의 모델을 선택하기 위한 과정 ;

validation을 통해 모델 성능을 평가하고, hyperparameter tuning

- train-test set 분리 `train_test_split`
- cross validation `KFold` `StratifiedKFold` `cross_val_score` `cross_validate`
- hyperparameter tuning `GridSearchCV` 등

Evaluation

> `sklearn.metrics`

1. accuracy (정확도)

   전체 예측 중 제대로 예측한 비율

   (TP + TN) / (TP + TN + FP + FN)

   보통 테스트셋에 대한 accuracy를 측정해서 모델을 평가하지만,

   만약 데이터가 불균형하다면 이 수치는 의미가 없다.

   (fraud detection / disease detection / ad click 등)

   이렇게 불균형한 데이터에 대해서는 다른 지표를 사용해야 함 (F1 score 등)

2. precision & recall

   `precision` (정밀도)

   1이라고 예측한 것 중 실제로 1인 비율

   - TP / (TP + FP)
   - False Positive 낮추는 데 초점

   `recall = sensitivity` (재현율)

   실제 1을 1이라고 제대로 예측한 비율

   - TP / (TP + FN)
   - = TPR (True Positive Rate)
   - 실제 양성을 음성으로 예측하면 큰일나는 경우 (질병 판단, 사기 감지)
   - False Negative 낮추는 데 초점

3. F1 score

   `F1-score`

   = `precision` 과 `recall` 의 조화평균 (불균형 완화)

   불균형한 데이터 분류 문제의 평가 지표

4. AUC

   `True Positive Rate` = `recall` = TP / (TP + FN)

   `False Positive Rate` = `1 - sensitivity`

   를 두 축으로 그린 그래프인 `ROC curve` 의 아래 면적이 `AUC` 이다.

Models

> `sklearn.linear_model, tree, svm, neighbors, cluster, ensemble` 등

estimator class method

- 지도학습

  fit( ) -> 학습

  predict( ) -> 예측

- 비지도학습 / feature extraction

  fit( ) -> 학습이 아닌, 데이터 구조 맞추는 것

  transform( )

  fit_transform( )