I-01. 파이썬 기반의 머신러닝과 생태계 이해

- 머신러닝이란?
  - 애플리케이션을 수정하지 않고 데이터를 기반으로 패턴을 학습하고 결과를 예측하는 알고리즘 기법
- 머신러닝의 분류
  - 지도학습 (Supervised Learning) : 분류, 회귀, 추천 시스템, 시각/음성 감지/인지, NLP
  - 비지도학습 (Un-supervised Learning): 클러스터링, 차원 축소, 강화학습 ——→ ? 비지도 학습의 예제에도 강화학습?
  - 강화 학습 (Reinforcement Learning)

I-03. 넘파이

- 넘파이 (NumPy) → 머신러닝의 알고리즘은 선형대수/통계 기반!
  - 파이썬에서 선형대수 기반의 프로그램을 쉽게 만들수 있도록 지원하는 패키지
  - 기반 데이터 타입 : ndarray (다차원 배열)
  - ndarray 내 데이터 값은 숫자/문자열/불값 모두 가능
  - 같은 데이터 타입만 가능 ——→ 데이터 크기가 다른 데이터 타입으로 형 변환 일괄 적용
  - 인덱싱 (Indexing) : ndarray의 데이터 세트 선택
    1. 특정 데이터만 추출 : 인덱스 값 지정
    2. 슬라이싱 (Slicing) : 연속된 인덱스상의 ndarray 추출 ex) 1:5 → 1부터 4까지에 해당하는 Narray 변환
    3. 팬시 인덱싱 (Fancy Indexing) : 일정한 인덱스 집합을 리스트 또는 ndarray 형태로 지정, 해당 위치 반환
    4. 불린 인덱싱 (Boolean Indexing) : T/F 값 인덱싱 집합을 기반으로 True에 해당하는 ndarray 위치 반환
- 선형대수 연산
  1. 행렬 내적 (행렬 곱)
     - np.dot()을 이용해 계산
     - 왼쪽 행렬의 열 개수와 오른쪽 행렬의 행 개수가 동일해야 함.
  2. 전치 행렬
     - 원 행렬에서 행과 열 위치를 교환한 원소로 구성한 행렬 (A^T)
     - np.transpose()를 이용해 계산

I-04. 데이터 핸들링 - 판다스

- 판다스 (Pandas)
  - 핵심 객체: Dataframe, 2차원 데이터를 담는 데이터 구조체
  - Index: 개별 데이터를 고유하게 식별하는 key 값
  - Series: 칼럼이 하나뿐인 데이터 구조체 → 칼럼명 x, Dataframe이 생성되며 생김
  - [] 연산자: 칼럼만 지정 가능 (칼럼 지정 연산자)
  - Indexing 방식 → vs 넘파이: 행의 위치, 열의 위치, 슬라이싱 범위 등

    1. iloc[]
       - 위치 기반 인덱싱 방식
       - 0을 시작점으로 하는 행과 열 위치를 좌표 정숫값으로 지정
       - 불린 인덱싱 지원 x

    2. loc[]
       - 명칭 (label) 기반 인덱싱 방식
       - 인덱스 값으로 행 위치, 칼럼 명칭으로 열 위치 지정
       - 슬라이싱 기호를 적용하면 종료 값까지 포함

Ⅱ-02. 붓꽃 품종 예측하기

- 분류(Classification) - 지도학습 사용
- ML 알고리즘 : 의사 결정 트리 (Decision Tree)
- 학습용 / 테스트용 데이터 분리 필요 : 예측은 학습용과 별개 테스트용 데이터 세트 사용
- 사이킷런의 Estimator 클래스 = 지도 학습의 모든 알고리즘 구현 = Classifier (분류) + Regressor (회귀)

Ⅱ-04. Model Selection 소개

- 교차 검증 : 데이터의 편중을 막기 위해 여러 세트로 구성된 학습 데이터 세트와 검증 데이터 세트에서 학습과 평가 수행
- K 폴드 교차 검증 : K개의 데이터 폴드 세트를 만들어서 K번만큼 각 폴드 세트에 학습과 검증 평가를 반복적으로 수행

Ⅱ-05. 데이터 전처리

- 사이킷런의 머신러닝 알고리즘은 문자열 값 허용 x ⟶ 피처 벡터화 or 삭제

• 데이터 인코딩

1. 레이블 인코딩 (Label encoding)
   - 카테고리 피처를 코드형 숫자 값으로 변환
   - 선형 회귀와 같은 ML 알고리즘 적용x ⟶ 숫자 값이 크고 작다는 개념도 x
   - 숫자의 중요도를 적용하지 않는 트리 계열 ML 알고리즘에서 사용
2. 원-핫 인코딩 (One-Hot Encoding)
   - 피처 값의 유형에 따라 새로운 피처를 추가해 고유 값에 해당하는 칼럼에만 1 표시, 나머지는 0 표시
   - 입력 값으로 2차원 데이터 필요
   - OneHotEncoder를 이용해 변환한 값이 희소 행렬 형태이므로 toarray()를 이용해 밀집 행렬로 변환

• 피처 스케일링 (feature scaling)

1. 표준화 (Standardization)
   - 데이터의 피처 각각이 평균이 0 이고 분산이 1인 가우시안 정규 분포를 가진 값으로 변환
2. 정규화 (Normalization)
   - 서로 다른 피처의 크기를 통일하기 위해 크기를 변환
   - 개별 데이터의 크기를 최소 0 ~ 최대 1의 값으로, 모두 똑같은 단위로 변경

- 테스트 데이터는 학습 데이터의 스케일링 기준에 따라야 한다.

Ⅲ-01. 정확도(Accuracy)

- 정확도
  - 모델 예측 성능을 나타내는 평가지표
  - 불균형한 레이블 값 분포에서 ML 모델 성능 평가에 부적합

Ⅲ-02. 오차 행렬

- 오차 행렬 (Confusion matrix)
  - 예측 클래스와 실제 클래스에 따라 TN/FP/FN/TP 형태로 분류

Ⅲ-03. 정밀도와 재현율

- 정밀도
  - 예측을 Positive로 한 대상 중에 예측과 실제 값이 Positive로 일치한 데이터의 비율
  - TP / (FP + TP)
  - 실제 Negative 양성인 데이터 예측을 positive로 잘못 판단하면 큰 영향이 발생하는 경우 중요함
- 재현율 (TPR)
  - 실제 값이 Positive인 대상 중에 예측과 실제 값이 Positive로 일치한 데이터의 비율
  - TP/(FN+TP)
  - 실제 Positive 양성인 데이터 예측을 Negative로 잘못 판단하면 큰 영향이 발생하는 경우 중요함
- 분류 결정 임계값: Positive 예측값을 결정하는 확률의 기준 (TPR ∝ 임계값)

Ⅲ-04. F1 스코어

- 정밀도와 재현율을 결합한 지표
- 정밀도와 재현율이 치우쳐 있을 때 상대적으로 낮은 값을 가진다.

Ⅲ-05. ROC 곡선과 AUC

- 이진 분류 예측 성능 측정에서 중요
- ROC 곡선
  - FPR이 변할 때 TPR이 어떻게 변하는지를 나타내는 곡선
  - ROC 곡선이 가운데 직선에서 멀어질수록 뛰어난 성능
  - ROC 곡선 면적에 기반한 AUC 값으로 성능 지표 확인 (보통 0.5 이상이 AUC)