머신러닝 : 지도학습(분류, 회귀,...), 비지도학습(클러스터링, 차원 축소, 강화학습)

데이터를 기반으로 통계적인 신뢰도 강화하고 예측 오류를 최소화하기 위한 다양한 수학적 기법 적용해 패턴 스스로 인지하고 신뢰도 있는 예측 결과 도출

데이터에 매우 의존적

지도학습: 학습을 위한 다양한 피처와 분류 결정값인 레이블 데이터로 학습한 뒤(학습 데이터 세트), 별도의 테스트 데이터 세트에서 미지의 레이블 예측

사이킷런 패키지 내의 모듈명은 sklearn으로 시작하는 명명규칙

sklearn.datasets : 사이킷런에서 자체 제공하는 데이터 세트 생성 모듈 모임

sklearn.tree : 트리 기반 머신러닝 알고리즘 구현한 클래스 모임 -> ex) 의사결정트리

sklearn.model\_selection : 학습데이터와 검증데이터, 예측데이터로 분리하거나 최적의 하이퍼 파라미터(알고리즘 별로 최적의 학습을 위해 직접 입력하는 파라미터, 성능 튜닝 가능)로 평가하기 위한 모듈 모임

지도학습의 분류 알고리즘을 구현한 클래스를 Classifier로, 회귀 알고리즘을 구현한 클래스를 Regressor로 지칭, 합쳐서 Estimator

즉, 지도학습의 모든 알고리즘을 구현한 클래스 통칭하여 Estimator라고 부름, fit()과 predict()를 내부에서 구현

cross\_val\_score() 같은 evaluation함수, GridSearchCV 같은 하이퍼 파라미터 튜닝 지원 클래스는 이 Estimator를 인자로 받음. cross\_val\_score(), GridSearchCV.fit() 함수 내에서 인자로 받은 Estimator의 fit()과 predict() 호출해서 평가하거나 하이퍼 파라미터 튜닝 수행

비지도학습인 차원축소, 클러스터링, 피처 추출 등을 구현한 클래스도 fit(), transform() 적용

fit(): 입력 데이터의 형태에 맞춰 데이터를 변환하기 위한 사전 구조 맞추는 작업

transform(): fit() 이후 입력 데이터의 차원 변환, 클러스터링, 피처 추출 등의 작업 수행

fit\_transform(): fit()과 transform()을 별도로 호출할 필요를 줄여주지만, 사용에 주의가 필요

머신러닝 모델 구축하는 주요 프로세스

: 피처 처리(피처 가공, 변경, 추출) -> ML 알고리즘 학습/예측 -> 평가

내장된 데이터 셋(분류나 회귀 용도)

datasets.load\_boston(), datasets.load\_breast\_cancer(), datasets.load\_iris(), datasets.load\_digits(), datasets.load\_diabetes()

cf) fetch 계열 명령어는 데이터 크기가 커서 인터넷에서 내려받는 데이터

표본데이터 생성기

Datasets.make\_classifications() --> 분류를 위한 데이터 세트 만듦

.make\_blobs() --> 클러스터링을 위한 데이터 세트 만듦

분류나 회귀를 위한 내장된 예제 데이터의 구성

1. 일반적으로 딕셔너리 형태
2. 키의 구성

* Data : 피처의 데이터 세트, ndarray
* Target: 분류 시 레이블 값, 회귀 시 숫자 결과값 데이터 세트, ndarray
* target\_names: 개별 레이블 이름, ndarray/list
* feature\_names: 피처의 이름, ndarray/list
* DESCR: 데이터세트에 대한 설명과 각 피처의 설명, string

정리

머신러닝은 데이터의 가공 및 변환 과정의 전처리, 데이터를 학습 데이터와 테스트 데이터로 분리 후에 학습 데이터를 기반으로 머신러닝 알고리즘을 적용해 모델 학습. 학습된 모델 기반으로 테스트 데이터에 대한 예측 수행하고 예측 결과값과 실제 값을 비교해 머신러닝 모델에 대한 평가 수행

* 전처리: 오류 데이터의 보정이나 결손값 처리 등 다양한 데이터 클렌징 작업, 레이블 인코딩/원-핫 인코딩 같은 인코딩 작업, 데이터의 스케일링/정규화 작업 등으로 데이터를 사전 처리
* 학습 데이터 세트로 학습한 뒤 반드시 별도의 테스트 데이터 세트로 평가되어야 함.
* 교차 검증 지원하기 위해 KFold, StratifiedKFold, cross\_val\_score() 등의 다양한 클래스와 함수 제공
* 최적 하이퍼 파라미터를 교차 검증을 통해 추출하기 위해 GridSearchCV 제공