EURON 2주차 과제

분류

01. 분류(Classification)의 개요

지도학습 : 레이블, 즉 명시적인 정답이 있는 데이터가 주어진 상태에서 학습하는 머신러닝 방식

지도학습의 대표적인 유형 : 분류

▼ 분류

- 학습데이터로 주어진 데이터의 피처와 레이블 값을 머신러닝 알고리즘으로 학습하여 모델 생성
- 생성된 모델에 새로운 데이터 값이 주어졌을 때 미지의 레이블 값 예측
- 다양한 머신러닝 알고리즘으로 구현 가능 → 주로 앙상블 방법을 다룰 예정

  ▼ 앙상블 방법

  - 정형 데이터의 예측 분석 영역에서 높은 예측 성능으로 각광받음
  - 배깅과 부스팅 방식으로 나뉨 (ex. 랜덤 포레스트 - 배깅, 그래디언트 부스팅 - 부스팅)
  - 기본 알고리즘 : 결정 트리

02. 결정 트리

- ML 알고리즘 중 직관적으로 이해하기 쉬움
- 데이터에 있는 규칙을 학습을 통해 자동으로 찾아내 트리 기반의 분류 규칙 생성
- 일반적으로 규칙은 if/else 기반으로 나타냄

▼ 구조

- 규칙 노드 : 규칙 조건
- 리프 노드 : 결정된 클래스 값
- 서브 트리 : 새로운 교칙 조건마다 생성

- 트리의 깊이가 깊어질수록 결정 트리의 예측 성능이 저하될 가능성 상승
- 트리를 어떻게 분할할 것인가가 중요 → 최대한 균일한 데이터 세트를 구성하도록

- 결정 노드는 정보 균일도가 높은 데이터 세트를 먼저 선택할 수 있도록 규칙 조건 생성
- 정보의 균일도 측정하는 방법 : 엔트로피를 이용한 정보 이득 지수, 지니 계수

결정 트리 모델의 특징

- 장점 : 정보의 '균일도', 알고리즘이 쉽고 직관적임
- 단점 : 과적합으로 정확도가 떨어짐 → 트리의 크기를 사전에 제한하는 것이 성능 튜닝에 도움

결정 트리 파라미터

- 사이킷런은 결정 트리 알고리즘을 구현한 DecisionTreeClassifier와 DecisionTreeRegressor 클래스 제공
- DecisionTreeClassifier : 분류를 위한 클래스
- DecisionTreeRegressor : 회귀를 위한 클래스

결정 트리 모델의 시각화

- Graphviz 패키지 사용
- 사이킷런에서 인터페이스 가능한 export_graphvis() API 제공
- Graphviz를 이용해 결정 트리 알고리즘의 규칙 생성 트리를 시각적으로 이해할 수 있음

결정 트리 과적합(Overfitting)

- 복잡한 모델은 학습 데이터 세트의 특성과 약간만 다른 형태의 데이터 세트를 예측하면 예측 정확도가 떨어짐
- 학습 데이터에만 지나치게 최적화된 분로 기준은 오히려 테스트 데이터 세트에서 정확도를 떨어뜨릴 수 있음

결정 트리 실습 - 사용자 행동 데이터 세트

ipynb 파일 참고

03. 앙상블 학습

앙상블 학습 개요

- 여러 개의 분류기 생성 → 예측 결합 → 단일 분류기보다 정확한 최종 예측 도출

▼ 학습 유형 : 보팅, 배깅, 부스팅

- 보팅 & 배깅 공통점 : 여러 개의 분류기가 투표를 통해 최종 예측 결과를 결정

▼ 보팅

- 서로 다른 알고리즘을 가진 분류기 결합

▼ 배깅

- 같은 알고리즘 / 데이터 샘플링 다르게 (ex. 랜덤 포레스트 알고리즘)
- 부트스트래핑 분할 방식 : 개별 분류기에 할당된 학습 데이터는 원본 학습 데이터를 샘플링해 추출
- 중첩을 허용

▼ 부스팅

- 여러 개의 분류기가 순차적으로 학습을 수행, 다음 분류기에 가중치 부여
- 그래디언트 부스트, XGBoost, LightGBM 등

보팅 유형 - 하드보팅과 소프트보팅

- 하드보팅 : 다수결 원칙과 비슷, 다수의 분류기가 결정한 예측값을 최종 보팅 결괏값으로 선정
- 소프트보팅 : 분류기들의 레이블 값 결정 확률을 모두 더하여 평균, 그 중 확률이 가장 높은 레이블 값을 최종 보팅 결괏값으로 선정
- 일반적으로 소프트 보팅 적용 : 예측 성능 우수

보팅 분류기

- 사이킷런에서 보팅 방식의 앙상블을 구현한 VotingClassifier 클래스 제공

- 보팅으로 여러 개의 기반 분류기를 결합한다고 해서 무조건 기반 분류기보다 예측 성능이 향상되지는 않음 → 전반적으로 뛰어난 예측 성능

04. 랜덤 포레스트

랜덤 포레스트의 개요 및 실습

- 앙상블 알고리즘 중 비교적 빠른 수행 속도
- 다양한 영역에서 높은 예측 성능
- 사이킷런에서 RandomForestClassifeir 클래스를 통해 랜덤 포레스트 기반의 분류 지원

랜덤 포레스트 하이퍼 파라미터 및 튜닝

- 트리 기반의 앙상블 알고리즘은 하이퍼 파라미터가 너무 많고, 그로 인한 시간 소모가 큼
- GridSearchCV를 이용해 랜덤 포레스트의 하이퍼 파라미터 튜닝 → 튜닝 시간 절약