EURON 3주차 예습과제_정윤지

CHAPTER 04: 분류

01. 분류(classification)의 개요

- 지도학습은 명시적인 정답이 있는 데이터가 주어진 상태에서 학습하는 머신러닝 방식(레이블)
- 지도학습의 대표적인 유형인 분류(Classification)는 학습 데이터로 주어진 데이터의 피처와 레이블 값을 머신러닝 알고리즘으로 학습해 모델을 생성, 생성된 모델에 새로운 데이터 값이 주어졌을 때 미지의 레이블 값을 예측하는 것
- 기존 데이터가 어떤 레이블에 속하는지 패턴을 알고리즘으로 인지한 뒤, 새롭게 관측된 데이터에 대한 레이블을 판별하는 것

분류 구현이 가능한 알고리즘

- 베이즈 통계와 생성 모델에 기반한 나이브 베이즈(naive bayes)
- 독립 변수와 종속 변수의 선형 관계성에 기반한 로지스틱 회귀(logistic regression)
- 데이터 균일도에 따른 규칙 기반의 결정 트리(dscision tree)
- 개별 클래스 간의 최대 분류 마진을 효과적으로 찾아주는 서포트 벡터 머신(support vector machine)
- 근접 거리를 기준으로 하는 최소 근접(nearest neighbor) 알고리즘
- 심층 연결 기반의 신경망(neural network)
- 서로 다른(또는 같은) 머신러닝 알고리즘을 결합한 앙상블(ensemble)

앙상블 방법(ensemble method)의 개요

- 분류에서 가장 각광을 받는 방법 중 하나
- 신경망에 기반한 딥러닝을 제외한 정형 데이터의 예측 분석 영역에서 앙상블이 매우 높은 예측 성능을 발휘
- 서로 다른/같은 알고리즘을 단순히 결합한 형태도 있으나, 일반적으로 배깅(bagging)과 부스팅(boosting)으로 구분
- 랜덤 포레스트(random forest): 대표적인 배깅 방식, 뛰어난 예측 성능, 상대적으로 빠른 수행 시간, 유연성
- 그래디언트 부스팅(gradient boosting): 부스팅의 효시, 뛰어난 예측 성능, 수행 시간이 오래 요소, 최적화 모델 튜닝의 어려움
- xgboost와 lightGBM 등 기존 그래디언트 부스팅의 예측 성능을 한 단계 발전시키면서, 수행 시간을 단축시킨 알고리즘이 등장, 가장 활용도 높은 알고리즘으로 자리 잡음

앙상블은 대부분 동일한 알고리즘을 결합
앙상블의 기본 알고리즘으로 일반적으로 사용하는 것은 결정트리

결정트리

- 매우 쉽고, 유연하게 적용될 수 있는 알고리즘
- 데이터의 스케일링이나 정규화 등의 사전 가공 영향이 매우 적음
- 예측 성능 향상을 위해 복잡한 규칙 구조를 가져야 하며 이로 인한 과적합이 발생해 예측 성능이 저하될 수 있다는 단점이 존재
- 단점은 앙상블 기업에서 장점으로 작용하기도 하는데, 앙상블은 매우 많은 여러 개의 약한 학습기(예측 성능이 상대적으로 떨어지는 학습 알고리즘)를 결합해 확률적으로 보완과 오류가 발생한 부분에 대한 가중치를 업데이트하면서 예측 성능을 향상 시키는데, 결정 트리가 좋은 약한 학습기가 되기 때문임

02. 결정 트리

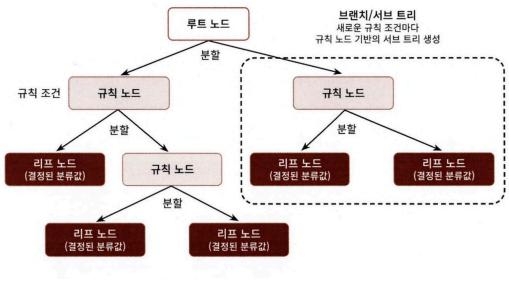
ML 알고리즘 중 직관적으로 이해하기 쉬운 알고리즘

데이터에 있는 규칙을 학습을 통해 자동으로 찾아내는 트리(tree) 기반의 분류 규칙을 만드는 것

규칙을 가장 쉽게 표현하는 방법은 if/else 기반으로 나타내는 것

데이터의 어떤 기준을 바탕으로 규칙을 만들어야 가장 효율적인 분류가 될 것인가가 알고리즘의 성능을 좌우

결정 트리의 구조



- 규칙 노드(dicision node)로 표시된 노드는 규칙 조건
- 리프 노드(leaf node)로 표시된 노드는 결정된 클래스 값
- 새로운 규칙 조건 마다 서브 트리(sub tree)가 생성
- 데이터 세트에 피처가 있고, 피처가 결합해 규칙 조건을 만들 때마다 규칙 노드가 생성
- 규칙이 많다는 것은 분류를 결정하는 방식이 복잡하다는 의미히고, 과적합으로 이어지기 쉬움
- 트리의 깊이(depth)가 깊어질수록 결정 트리의 예측 성능이 저하될 가능성이 높음
- 가장 한 적은 결정 노드로 높은 예측 정확도를 가지려면 데이터를 분류할 때 최대한 많은 데이터 세트가 해당 분류에 속할 수 있도록 결정 노드의 규칙이 정해져야 함
- 어떻게 트리를 분할(split) 할 것인가가 중요, 최대한 균일한 데이터 세트를 구성하도록 분할 하는 것이 필요

균일한 데이터 세트

- 모든 데이터의 구성이 동일해야 균일도가 높은 데이터임
- 데이터 세트의 균일도는 데이터를 구분하는 데 필요한 정보의 양에 영향을 미침

- 결정 노드는 정보 균일도가 높은 데이터 세트를 먼저 선택할 수 있도록 규칙 조건을 만듦

- 정보 균일도가 데이터 세트로 쪼개질 수 있도록 조건을 찾아 서브 데이터 세트를 만들고, 이 서브 데이터 세트에서 균일도가 높은 자식 데이터 세트를 쪼개는 방식으로 자식 트리를 내려가면서 반복, 데이터 값을 예측
- 정보의 균일도를 측정하는 대표적인 방법은 엔트로피를 이용한 정보 이득(information gain)지수와 지니 계수

정보 이득
- 엔트로피 개념을 기반으로 함
- 엔트로피는 주어진 데이터 집합의 혼잡도를 의미
- 서로 다른 값이 섞여 있으면 엔트로피가 높고, 같은 값이 섞여 있으면 엔트로피가 낮음
- 정보 이득 지수는 1에서 엔트로피 지수를 뺀 값(1-엔트로피 지수)
- 결정 트리는 정보 이득 지수로 분할 기준을 결정
- 정보 이득이 높은 속성을 기준으로 분할

지니 계수
- 경제학에서 불평등 지수를 나타낼 때 사용하는 계수
- 지니 계수가 낮을 수록 데이터 균일도가 높은 것으로 해석, 지니 계수가 낮은 속성을 기준으로 분할

- 결정 트리 알고리즘을 사이킷런에서 구현한 DecisionTreeClassifier는 지니 계수를 이용해 데이터 세트를 분할
- 결정 트리의 일반적인 알고리즘은 데이터 세트를 분할하는 데 가장 좋은 조건, 즉 정보 이득이 높거나 지니 계수가 낮은 조건을 찾아, 자식 트리 노드에 걸쳐 반복적으로 분할한 뒤, 데이터가 모두 특정 분류에 속하게 되면 분할을 멈추고 분류를 결정

결정 트리 모델의 특징
- 결정 트리의 가장 큰 장점은 정보의 '균일도'를 기반으로 하여 알고리즘이 쉽고 직관적임
- 결정 트리는 룰이 명확하고 어떻게 규칙 노드와 리프 노드가 만들어지는 지 알 수 있으며, 시각화가 가능
- 특별한 경우를 제외하고 각 피처의 스케일링과 정규화 등의 전처리 작업이 필요하지 않음
- 과적합으로 정확도가 떨어질 수 있음(피처 정보의 균일도에 따른 룰 규칙으로 서브 트리를 만들면 피처가 많고 균일도가 다양하게 존재할 수록 트리의 깊이가 커지고 복잡해짐)
- 복잡한 학습 모델은 실제 상황에 유연하게 대처할 수 없어, 예측 성능이 떨어짐
- 트리의 크기를 사전에 제한하는 것이 성능 튜닝에 도움이 됨

결정 트리 파라미터
- 사이킷런은 결정 트리 알고리즘을 구현한 DecisionTreeClassifier와 DecisionTreeRegressor 클래스를 제공
- DecisionTreeClassifier는 분류를 위한 클래스, DecisionTreeRegressor는 회귀를 위한 클래스
- 사이킷런의 결정 트리 구현은 CART 알고리즘 기반

파라미터

- min_samples_split: 노드를 분할하기 위한 최소한의 샘플 데이터 수, 과적합 제어에 사용, 디폴트 값은 2이며 작게 설정할수록 과적합 가능성이 증가
- min_sample_leaf: 분할이 될 경우 왼쪽과 오른쪽의 브랜치 노드에서 가져야 할 최소한의 샘플 데이터 수, 과적합 제어 용도, 비대칭적 데이터의 경우 특정 클래스의 데이터가 극도로 작을 수 있으므로 이 경우 작게 설정 필요
- max_features: 최적의 분할을 위해 고려해야 할 최대 피처 개수, 디폴트 값은 none으로 데이터 세트의 모든 피처를 사용해 분할 수행
- max_depth: 트리의 최대 깊이를 규정, 디폴트 값은 none으로 완벽하게 클래스 결정 값이 될 때까지 깊이를 계속 키우며 분할하거나 노드가 가지는 데이터 개수가 min_sample_split 보다 작아질 때까지 깊이를 증가시킴
- max_leaf_nodes: 말단 노드의 최대 개수

결정 트리 모델의 시각화
- Graphviz 패키지를 사용
- 사이킷런은 graphviz 패키지와 쉽게 인터페이스할 수 있도록 export_graphivz() API를 제공
- 함수 인자로 학습이 완료된 Estimator, 피처의 이름 리스트, 레이블 이름 리스트를 입력하면 학습된 결정 트리 규칙을 실제 트리 형태로 시각화 하여 보여줌

결정 트리 규칙의 구성
리프 노드
- 더 이상 자식 노드가 없는 노드
- 최종 클래스(레이블) 값이 결정되는 노드
- 오직 하나의 클래스 값으로 최종 데이터가 구성되거나 리프 노드가 될 수 있는 하이퍼 파라미터 조건을 충족해야 함

브랜치 노드
- 자식 노드가 있는 노드
- 자식 노드를 만들기 위한 분할 규칙 조건을 가짐

노드 내 기술된 지표의 의미
- Petal length(cm) <= 2.45 와 같이 피처의 조건이 있는 자식 노드를 만들기 위한 규칙 조건(이 조건이 없으면 리프 노드)
- gini는 다음의 value=[]로 주어진 데이터 분포에서의 지니 계수
- samples는 현 규칙에 해당하는 데이터 건수
- value=[]는 클래스 값 기반의 데이터 건수

결정 트리는 규칙 생성 로직을 미리 제어하지 않으면 완벽하게 클래스 값을 구별하기 위해 트리 노드를 계속 생성
이로 인해 매우 복잡한 규칙 트리가 만들어져 모델이 쉽게 과적합되는 문제를 가지게 됨

결정 트리는 이러한 이유로 과적합이 상당히 높은 ML 알고리즘임
결정 트리 알고리즘을 제어하는 대부분 하이퍼 파라미터는 복잡한 트리가 생성되는 것을 막기 위함

결정 트리 과적합
- 결정 트리가 학습 데이터를 분할해 예측을 수행하는 방법과 이로 인한 과적합 문제
- 일부 이상치 데이터까지 분류하기 위해 분할이 자주 일어나, 결정 기준 경계가 많아지고, 리프 노드 안에서 데이터가 모두 균일하거나 하나만 존재해야 하는 엄격한 분할 기준으로 인해 결정 기준 경계가 많아지고 복잡해짐
- 이렇게 복잡한 모델은 학습 데이터 세트의 특성과 약간 다른 형태의 데이터 세트를 예측하면 예측 정확도가 떨어짐
- 이상치에 크게 반응하지 않으면 더 일반화된 분류 규칙에 따라 분류됐음을 알 수 있음
- 학습 데이터에만 지나치게 최적화된 분류 기준은 오히려 테스트 데이터 세트에서 정확도를 떨어뜨릴 수 있음

결정 트리 실습 - 사용자 행동 인식 데이터 세트
- 실습

03. 앙상블 학습
앙상블 학습 개요
- 앙상블 학습을 통한 분류는 여러 개의 분류기를 생성하고 그 에측을 결합함으로써 보다 정확한 최종 예측을 도출하는 기법
- 앙상블 학습의 목표는 다양한 분류기의 예측 결과를 결합함으로써 단일 분류기보다 신뢰성 높은 예측 값을 얻는 것
- 쉽고 편하면서도 강력한 성능을 보유

앙상블 학습의 유형
보팅(voting)
- 여러 개의 분류기가 투표를 통해 최종 예측 결과를 결정
- 일반적으로 서로 다른 알고리즘을 가진 분류기를 결합하는 것

배깅(bagging)
- 여러 개의 분류기가 투표를 통해 최종 예측 결과를 결정
- 각각의 분류기가 모두 같은 유형의 알고리즘 기반이지만, 데이터 샘플링을 서로 다르게 가져가면서 학습을 수행해 보팅을 수행하는 것
- 대표적인 배깅 방식이 랜덤 포레스트 알고리즘

부스팅(boosting)
- 여러 개의 분류기가 순차적으로 학습을 수행하되, 앞에서 학습한 분류기가 예측이 틀린 데이터에 대해서는 올바르게 예측할 수 있도록 다음 분류기에게는 가중치를 부여하면서 학습과 예측을 진행하는 것

- 계속해서 분류기에게 가중치를 부스팅하면서 학습을 진행하기 때문에 부스팅 방식으로 불림
- 예측 성능이 뛰어나 앙상블 학습을 주도하고 있으며 대표적인 부스팅 모듈로 그래디언트 부스트, XGBoost, LightGBM이 있음

스태킹
- 여러 가지 다른 모델의 예측 결괏값을 다시 학습 데이터로 만들어서 다른 모델로 재학습시켜 결과를 예측하는 방법

보팅 유형 - 하드 보팅(Hard Voting)과 소프트 보팅(Soft Voting)

하드 보팅
- 하드 보팅을 이용한 분류(classification)는 다수결 원칙과 비슷
- 예측한 결괏값들 중 다수의 분류기가 결정한 예측값을 최종 보팅 결괏값으로 선정하는 것

소프트 보팅
- 분류기들의 레이블 값 결정 확률을 모두 더하고 이를 평균해서 이들 중 확률이 가장 높은 레이블 값을 최종 보팅 결괏값으로 선정
- 일반적으로 소프트 보팅이 보팅 방법으로 적용

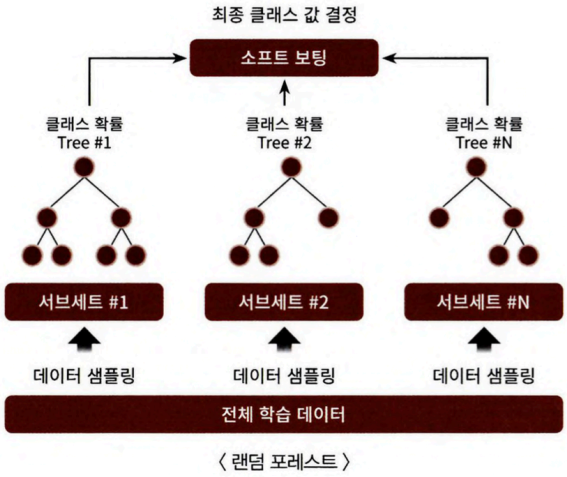
보팅 분류기(Voting Classifier)
사이킷런은 보팅 방식의 앙상블ㅇ르 구현한 VotingClassifier 클래스를 제공
보팅 방식의 앙상블을 이용해 위스콘신 유방암 데이터 세트를 분석 - 실습

- 보팅, 배깅, 부스팅 등의 앙상블 방법은 전반적으로 다른 단일 ML 알고리즘보다 뛰어난 예측 성능을 가지는 경우가 많음
- ML 모델의 성능은 다양한 테스트 데이터에 의해 검증되므로 어떻게 높은 유연성을 가지고 현실에 대처할 수 있는가가 중요한 ML 모델의 평가 요소
- 결정 트리 알고리즘은 쉽고 직관적인 분류 기준을 가지고 있지만 정확한 예측을 위해 학습 데이터의 예외 상황에 집착한 나머지 오히려 과적합이 발생해 실제 테스트 데이터에서 예측 성능이 떨어지는 현상이 발생하기 쉬운데, 앙상블 학습에서는 이 같은 결정 트리 알고리즘의 단점을 많은 분류기를 결합해 다양한 상황을 학습하게 함으로써 극복하고 있음
- 결정 트리 알고리즘의 장점은 그대로 취하고, 단점은 보완하면서 편향-분산 트레이드 오프의 효과를 극대화할 수 있음

04. 랜덤 포레스트
배깅(bagging)
- 보팅과 다르게, 같은 알고리즘으로 여러 개의 분류기를 만들어서 보팅으로 최종 결정하는 알고리즘

- 배깅의 대표적인 알고리즘은 랜덤 포레스트

랜덤 포레스트
- 앙상블 알고리즘 중 비교적 빠른 수행 속도를 가지고 있으며, 다양한 영역에서 높은 예측 성능을 보임
- 기반 알고리즘은 결정 트리로, 쉽고 직관적인 장점을 그대로 가지고 있음
- 여러 개의 결정 트리 분류기가 전체 데이터에서 배깅 방식으로 각자의 데이터를 샘플링해 개별적으로 학습을 수행한 뒤 최종적으로 모든 분류기가 보팅을 통해 예측 결정을 하게 됨



〈 랜덤 포레스트 〉

- 랜덤 포레스트는 개별적인 분류기의 기반 알고리즘은 결정 트리지만 개별 트리가 학습하는 데이터 세트는 전체 데이터에서 일부가 중첩되게 샘플링된 데이터 세트
- 여러 개의 데이터 세트를 중첩되게 분리하는 것을 부트스트래핑 분할 방식이라고 함
- 부트스트랩은 통계학에서 여러 개의 작은 데이터 세트를 임의로 만들어 개별 평균의 분포도를 측정하는 등의 목적을 위한 샘플링 방식을 지칭, 랜덤 포레스트의 서브세트 데이터는 이런 부트스트래핑으로 데이터가 임의로 만들어짐
- 서브 세트 데이터 건수는 전체 데이터 건수와 동일하지만, 개별 데이터가 중첩되어 만들어짐
- 데이터가 중첩된 개별 데이터 세트에 결정 트리 분류기를 각각 적용하는 것

랜덤 포레스트 하이퍼 파라미터 및 튜닝
- 트리 기반의 앙상블 알고리즘의 단점은 하이퍼 파라미터가 너무 많고, 그로 인해 튜닝을 위한 시간이 많이 소모된다는 것, 시간을 소모했음에도 튜닝 후 예측 성능이 크게 향상되는 경우가 많지 않음

- n_estimators: 랜덤 포레스트에서 결정 트리의 개수를 지정, 많이 설정할수록 좋은 성능을 기대할 수 있지만 계속 증가시킨다고 성능이 무조건 향상되는 것은 아님
- max_features: 결정 트리에 사용된 max_features 파라미터와 같은
- max_depth, min_samples_split: 결정 트리에서 과적합을 개선하기 위해 사용되는 파라미터가 랜덤 포레스트에도 또같이 적용될 수 있음