7장 앙상블 학습과 랜덤 포레스트

투표 기반 분류기

직접 투표(hard voting) 분류기: 다수결 투표로 정해지는 분류기, 약한 학습기여도 충분히 많고 다양하다면 강한 학습기가 될 수 있다. 이는 큰 수의 법칙 때문이다.

간접 투표(soft voting) 분류기: 개별 분류기의 예측을 평균 내어 확률이 가장 높은 클래스를 예측, 확률이 높은 투표에 비중을 더 두기 때문에 직접투표 방식보다 성능이 좋다.

배깅과 페이스팅

배깅: 같은 알고리즘을 사용하지만 훈련 세트의 서브셋을 무작위로 구성하여 분류기를 각기 다르게 학습시키는 것. 샘플링을 할 때 중복을 허용한다. 페이스팅에 비해 일반적으로 선호된다.

페이스팅: 배깅과 유사하지만 중복을 허용하지 않는다.

랜덤 패치와 랜덤 서브스페이스

랜덤 패치: 훈련 특성과 샘플을 모두 샘플링하는 방식

랜덤 서브스페이스: 훈련 샘플을 모두 사용하고 특성은 샘플링하는 방식

\*특성 샘플링은 더 다양한 예측기를 만들며 편향을 늘리는 대신 분산을 낮춥니다.

랜덤 포레스트

랜덤 포레스트: 배깅을 적용한 결정 트리 앙상블, 무작위로 선택한 특성후보 중에서 최적의 특성을 찾는 식으로 무작위성을 주입

엑스트라 트리: 랜덤 포레스트는 트리를 만들 때 특성마다 최적의 임계값을 찾지만 익스트림 랜덤 트리 앙상블은 최적의 임계값을 찾는 대신에 후보특성을 사용해 무작위로 분류한 다음 그 중에서 최상의 분할을 선택한다. 편향은 늘어나지만 분산을 줄어든다.

부스팅

약한 학습기를 여러 개 연결하여 강한 학습기를 만드는 앙상블 방법.

아다 부스트(AdaBoost): 이전 모델이 과소적합했던 훈련 샘플의 가중치를 더욱 높이는 것. 이렇게 하면 새로운 예측기는 학습하기 어려운 샘플에 점점 더 맞춰지게 된다.

그래디언트 부스트(GradientBoost): 이전 예측기가 만든 잔여 오차에 새로운 예측기를 학습시킨다. 트리의 수가 적을 경우 학습이 충분히 이루어지지 않지만 너무 많을 경우 과대적합이 발생할 수 있다. 이를 대비하여 더 이상 검증오차가 향상되지 않으면 조기 종료를 구현한다.

스태킹

앙상블에 속한 모든 예측기의 예측을 취합하는 간단한 함수를 사용하는 대신 모델을 사용.

8장 차원 축소

차원을 축소시키면 일부 정보가 유실됩니다. 그래서 훈련속도가 빨라지지만 성능은 조금 나빠질 수 있습니다. 하지만 어떤 경우 훈련데이터의 차원을 축소시키면 잡음이나 불필요한 세부사항을 걸러내어 성능을 높일 수 있습니다.

차원 축소를 위한 접근 방법

투영: 모든 훈련샘플은 사실 고차원 공간안의 저차원 부분공간에 놓여 있습니다. 따라서 정사영을 시키는 방법으로 투영이 가능합니다. 하지만 많은 경우 부분 공간이 뒤틀리거나 휘어 있기도 합니다.

매니폴드: 훈련 샘플이 놓여있는 매니폴드를 모델링하는 식으로 작동. 이는 대부분의 실제 고차원 데이터셋이 더 낮은 저차원 매니폴드에 가깝게 놓여있다는 매니폴드 가정에 근거한다. 예를 들어 MNIST의 경우 선으로 연결되어 있고 어느정도 중앙에 있는 등의 제약이 있다. 이러한 제약이 데이터셋을 저차원의 매니폴드로 압축할 수 있도록 도와준다.

PCA

투영을 이용한 차원축소 기술이다. 훈련세트의 분산이 최대인 축을 찾는다. 이어서 첫번째 축과 직교하며 남은 분산을 최대한 보존하는 두 번째 축을 찾는다. 이를 반복한다. 일반적으로 설명된 분산의 비율을 기준으로 주성분의 개수를 정한다.

점진적 PCA(IPCA): 훈련 세트를 미니배치로 나눈 뒤 IPCA 알고리즘에 한번에 하나씩 주입한다. 이를 통해 전체 훈련 세트를 메모리에 올려야 하는 PCA의 문제점을 보완

커널 PCA(kPCA): 고차원 공간(특성 공간)에서 선형 결정 경계는 원본 공간에서는 복잡한 비선형 결정 경계에 해당함. 이를 PCA에 적용해 차원 축소를 위한 비선형 투형을 수행.

LLE(지역 선형 임베딩)

비선형 차원축소 기법에 해당하며 매니폴드 학습이다. LLE는 먼저 각 훈련샘플이 가장 가까운 이웃에 얼마나 선형적으로 연관 되어있는지 측정한다(선형적인 지역관계 모델링). 이후 국부적인 관계가 가장 잘 보존되는 훈련세트의 저차원 표현을 찾습니다(관계를 보존하는 차원축소).

9장 비지도 학습

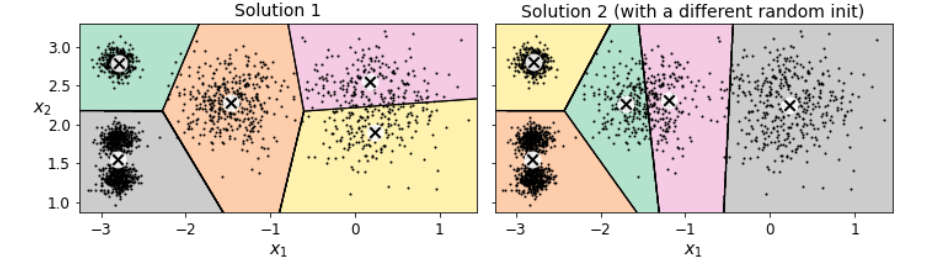
하드 클러스터링 vs 소프트 클러스터링

하드 클러스터링: 어떤 점이 오직 하나의 군집에만 속하는 군집 Ex) K-means

소프트 클러스터링: 어떤 데이터 점이 둘 이상의 군집에 속할 수 있는 군집, 각 군집에 속할 확률을 제공 ex) 가우스 혼합 모델

K-means

가장 빠르고 단순한 군집기법이다. 기본적인 K-means에서 센트로이드는 무작위로 초기화되며 알고리즘은 그저 점진적으로 센트로이드의 위치가 향상되도록 반복된다. 하지만 이럴 경우 같은 입력에 따라 다른 결과가 나올 수 있다.



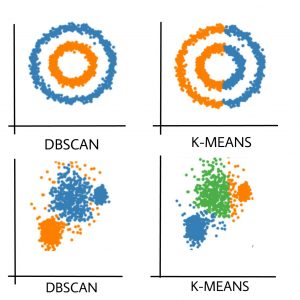
해결방법 중 하나는 이러한 다양한 결과를 계속 구한 후 그 중에서 센트로이드의 결과가 가장 좋은 것을 선택한다.

Accelerated k-means: triangle inequality(삼각 부등식)을 이용하여 불필요한 계산을 회피하여 군집을 빠르게 진행하는 방법

Mini batch k-means: 군집을 진행할 때 데이터 셋이 너무 크다면 속도가 느려지는 등의 문제가 발생할 수 있다. 미니 배치 k-means는 먼저 데이터 셋에서 랜덤으로 샘플을 뽑은 후 이들을 기준으로 군집을 진행하고 이어서 새로운 샘플을 뽑아 군집을 발전시킨다. 이 방법은 일반적인 k-means에 비해 속도가 빠르다.

DBSCAN 클러스터링: 밀도기반의 공간적 군집화

K-Means 군집의 한계인 오목한 형태의 데이터를 군집하기 위해 데이터의 밀도를 기준으로 인스턴스들을 공간적으로 군집화하는 기법



K-Means는 동심원 모양으로 데이터가 모여있거나 반달모양으로 형성되어있을 경우 클러스터의 중심으로부터 거리기준으로 군집

DBSCAN은 코어 데이터들을 계속 밀도있게 연결해나가 동일한 클러스터로 판단하여 군집