**1. SVM**

서포트 벡터 머신(SVM)은 두 개의 서로 다른 클래스에 속하지만 복잡한 다차원 경계로 분리된 데이터에 대한 최적의 분류기를 찾는 방법으로 제안되었다. 단일 클래스 분류, 즉 단일 클래스의 훈련 데이터 가용성을 기반으로 하는 이상 감지로 사용이 확장되었다.

원래 two-class 방법에는 데이터를 원래 특징 공간에서 복잡한 초평면이 아닌 초평면에 의해 효과적으로 분리할 수 있는 다차원(종종 무한 차원) 공간으로 변환하는 것이 포함된다. 이 변환된 특징 공간에 들어가면 두 클래스(과 ) 클래스의 경계를 나타내는 두 개의 평행 초평면은 다음 식 (1)에 의해 정의될 수 있다.

(1)

여기서 는 특징의 다차원 공간에서 일반적인 점이고, 는 두 초평면에 수직인 벡터이며, **/**는 법선 벡터를 따라 평면까지의 거리를 나타낸다. SVM 최적 경계 문제는 다음 식 (2)에 표시된 정확한 분류와 마진 최대화 사이의 균형으로 공식화한다.

* Training set 의 정확한 분류

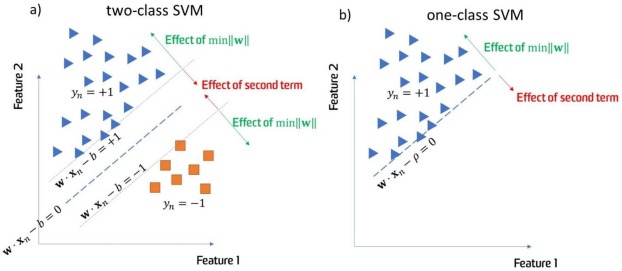
(2)

* 마진 최대화, 즉 두 평면 사이의 거리

따라서 와 의 최적 경계(두 평면 사이의 절반에 위치함, 즉 )의 값은 식 (3)에서 볼 수 있다.

(3)

문제는 원래(변환되지 않은) 특징 공간에서 직접 재구성하고 해결할 수 있지만 이것은 작업의 범위를 벗어난다. 더 흥미롭게도 식 (3)에서 나타내는 two-class SVM은 두 가지 항과 반대 효과를 가지고 있다는 것이다. 첫 번째, 마진의 최대화는 경계를 ± 무한대(거리 최대화)로 밀어내는 반면, 두 번째 항인 ‘softly’의 존재는 경계를 두 데이터셋 사이의 공간의 실제 부분으로 밀어냄으로써 문제를 제한한다. 이 개념을 2차원 그래픽으로 나타내면 [그림 1. (a)]로 나타난다.



[그림 1.] two-class SVM(a) 분류와 one-class SVM(b) 분류 비교

식 (3)과 (4)의 두 항의 효과는 도표에 개념적으로 표현된다.

one-class SVM에서, 클래스 하나가 없으면 두 번째 항이 분명히 손상될 것이고, 두 평면 중 하나만 제한할 수 있으므로 경계가 무한대, 즉 고정된 분류 출력으로 이동하게 될 것이다.

이러한 이유로 단일 평면 을 기반으로 하는 one-class SVM은 식 (4)에서 다음과 같이 재구성된다.

(4)

여기서 첫 번째 항은 마진을 원점에서 최대한 멀리 밀어내는 반면 두 번째 항은 [그림 1. (b)]과 같이 경계를 가로지르는 점의 수를 최소화하여 ‘soft’ 제약을 적용한다. 하이퍼파라미터 는 이 경계가 inlier 주변에 얼마나 밀접한지 제어하고 outlier 비율에 대한 상한을 설정한다.

**2. Decision tree**

의사 결정 트리는 어떤 현상에 대한 관찰을 해당 현상의 목표 값에 관한 추론에 통합하는 예측 모델이다. 데이터에서 의사 결정 트리를 추론하기 위해 머신 러닝을 사용하는 것을 의사 결정 트리 학습이라고 하며, 이는 공간 데이터 마이닝에서 일반적으로 사용된다. 모든 내부 노드는 변수에 해당하고 모든 에지는 해당 변수의 가능한 값을 나타내는 자식에 해당한다. 각 리프는 루트에서 리프까지의 경로로 표시되는 입력 변수 값이 주어지면 대상 변수의 값을 나타낸다. 의사 결정 트리는 속성 값 테스트를 기반으로 소스 세트를 하위 집합으로 분할하여 “학습”할 수 있다. 이 프로세스는 파생된 각 하위 집합에 대해 재귀적으로 반복된다. 분할이 더 이상 유용하지 않거나 파생된 하위 집합의 모든 샘플에 하나의 분류를 적용할 수 있을 때 재귀가 완료된다. 의사 결정 트리는 분류 및 예측에 사용할 수 있는 데이터 세트의 관계에 대해 사람이 이해할 수 있는 규칙을 생성한다. 이 기법은 질병 진단 및 사고 예측과 같은 여러 분야에서 광범위하게 사용되었다.

여기서 가장 중요한 문제는 의사 결정 트리 구조의 결과가 가능한 단순해야 한다는 것이다. 의사 결정 트리는 입력 데이터를 정확하게 분류하고 훈련 데이터에서 배운 지식을 일반화하여 가능한 한 정확하게 새로운 데이터를 분류할 수 있어야 한다. 대부분의 의사 결정 트리는 증분 될 수 있다. 즉, 새로운 훈련 데이터를 추가하여 의사 결정 트리를 업데이트할 수 있다.

의사 결정 트리는 여러 유형이 있다. 일부 의사 결정 트리의 출력은 possible value의 이산 집합이다. 이러한 트리를 분류 트리라고 한다. 출력이 실수이면 트리를 회귀 트리라고 한다. 이 트리는 리프 노드의 숫자를 예측한다. 두 출력이 모두 있는 트리를 분류 및 회귀 트리(CART)라고 한다.

CART 모델에서는 먼저 모든 데이터가 첫 번째(루트) 노드에 배치된다. 그런 다음 각 분기에서 가장 높은 균질성과 순도를 가진 splitter 변수를 기준으로 루트에 분할이 생성된다. 트리 브랜치에서 변수를 분류하는 프로세스는 각 노드의 데이터가 특정 클래스에 속할 만큼 충분히 균질해질 때까지 계속된다. 트리 끝에 있는 노드를 터미널 또는 리프 노드라고 하며 결과 클래스(부적합하고 적합 그리고 매우 적합함)이다. 루트 노드와 터미널 노드 사이의 노드를 내부 노드라고 한다. splitter 변수를 선택하고 각 노드를 두 개의 하위 노드로 분할하기 위한 다양한 인덱스가 있다. 그 중 하나인 지니 지수를 예를 들면, 노드의 모든 데이터가 하나의 범주에 속할 경우 노드에서 가장 순도가 높은 노드가 생성되고 지니 지수는 0이 된다. 그러나 노드의 데이터가 모든 범주에 속할 경우 가장 높은 지니 값을 얻는다. 각 노드의 모든 변수에 대해 지니지수를 계산하여 지니지수 값이 가장 낮은 변수를 splitter로 선정한다. 지니지수는 식 (1)과 같이 정의된다.

(1)

여기서 및 는 특정 등급으로 분류되는 개체의 확률이다.