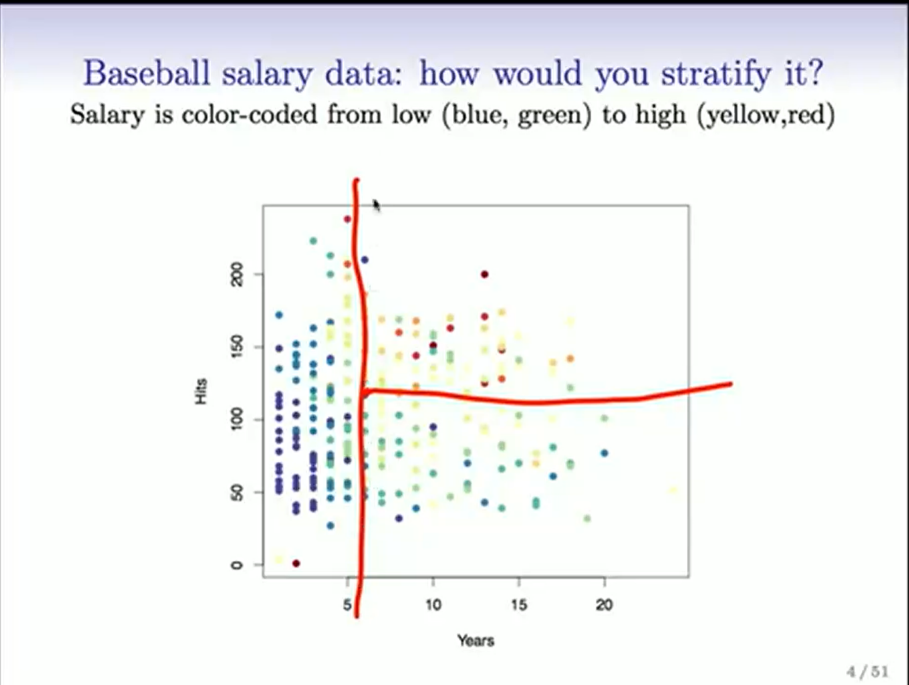
머신러닝 프로그래밍 과제

20235127김민주



먼저 이 트리를 보고 레이아웃이 뭔지 해석해보면, 이것은 일련의 분열입니다. 상단에는 모든 데이터가 있다. 그리고 이 “연도는 4.5 미만”은 분할이다. 왼쪽에는 리그에서 4.5년 미만의 선수들이 있는 칸막이다. 오른쪽에는 리그에서 4.5년 이상 활동한 선수들이 있다.이 분할은 다섯 개이지만 대략적으로—4.5는 4에서 5 사이이다. 그래서 맨 뒤에 있는 트리는 왼쪽과 오른쪽 영역으로 나누는 분할이다. 그래서 이 트리는 우선, 플레이어들의 다년간의 경험을 나누어 볼 것이라고 말한다. 4.5년 미만인 사람들은 왼쪽 노드에 배정되고, 4.5년 이상인 사람들은 오른쪽 노드에 배정된다. 맨 아래에 있는 숫자는 평균 로그 응답이다. 오른쪽에서는 히트곡에 대한 추가 분할을 진행한다. 경력이 4.5년 이상인 선수 중 117.5 안타 미만인 선수는 이 지점에 배정되며, 그렇지 않은 경우 이 지점에 배정된다. 이건 알고리즘에 의해 발견되었다. 사실, 강의에서 설명할 알고리즘은 훨씬 더 큰 트리를 만든 다음, 이 트리를 아래에서 가지치기하여 이 세 개의 노드 트리를 만드는 것이다. 맨 아래에 있는 노드들은 터미널 노드이기 때문에 터미널 노드라고 불린다. 더 이상 분리되지 않는다. 비터미널 노드를 내부 노드라고하며, 이 경우 트리에 두 개의 내부 노드가 있습니다. 터미널 노드는 예측 공간의 분할을 설명하는 노드이기 때문에 관심 있는 노드이다. 그럼 이 나무를 어떻게 해석해야할까? 경험이 적은 선수는 연봉이 낮을 수 있다. 왼쪽 지점에서는 더 이상 분할하지 않았기 때문에 경험이 적은 선수의 연봉을 결정하는데 비트 수가 그다지 중요하지 않은 것 같다. 오른쪽에는 더 많은 경험을 가진 선수들이 있는 곳에서 타격이 중요하다. 따라서 4.5년 이상의 경력을 가진 사람들에게는 안타 수가 중요하지만, 4.5 미만의 경력을 가진 사람들에게는 중요하지 않은 것 같다. 트리는 실제로 특징 공간을 연속적으로 분할해서 얻은 나무와 매우 비슷하다고 말했다. 트리가 성장하면 응답 값의 평균에서 볼 수 있듯이 예측은 각 터미널 노드에 속하게 된다. 상자들이 균질하고 각 상자에서 매우 유사한 관측값을 가지기를 원하기 때문이다. 그리고 상자들 사이에서는 매우 다를 것이다. 그래서 이것이 상자 찾기 문제를 제기하는 합리적인 방법처럼 보인다. 하지만 실제로는 계산적으로 해결하기가 너무 어렵다는 것이 밝혀졌다. 이 기준의 값이 가장 작은 10개의 상자를 찾고싶지만, 사실 그것은 계산적으로 불가능하다. 10은 풀 수 있을지 모르지만, 확실히 20이나 50을 넘어서면 매우 어렵다. 특히 상자를 만들 수 있는 방법을 몇가지나 생각해보면 숫자가 엉망이 될 뿐이다. 트리는 일종의 명백한 방법인 근사치를 사용한다. 위에서 아래로 욕심을 부리는 접근 방식. 그리고 위에서 아래로 내려오는 이유는 위에서 전체 관측값 세트로 시작한 다음 각 레벨에서 하나씩 두 조각으로 나누기 때문이다. 전체 데이터 세트와 모든 예측 변수를 맨 위에서 시작하여, 가장 작은 기준을 생성하는 예측 변수와 분할을 찾는 것이다. 이는 노드의 평균을 중심으로 각 응답의 제곱의 합을 의미합니다.