[어서와~ 머신러닝은 처음이지?]

2장. 의사결정나무

- 장형석
- 국민대학교 빅데이터경영MBA과정 교수
- 숙명여자대학교 빅데이터센터 연구소장
- chjang1204@nate.com
- 010-3302-5543

1. 생활에서 만나는 문제

사례

- 아파트상가내의 적당한 규모의 중저가브랜드의 화장품 가게를 운영하는 새미씨는 오랫동안 영업을 해서인지 대략 동네의 누가 주로 주기적으로 스킨과 로션을 사러 오고 어떤 사람이 좀 비싼 에센스나 영양크림을 사러 오는지 누가 어떤 브랜드를 선호하는지 아는 정도까지 되었다. 그러다보면 가끔씩 할인을 해주는데 어떤 사람은 할인이벤트가 고마워서 물건을 더 자주 사러 오거나 더 많은 양을 사는 경우도 있다. 새미씨는 대형백화점의 화장품 코너를 이용하지 않고 자신의 가게에 오는 손님이 고맙기만 하다. 그런데 이왕이면 자신의 가게를 선호하는 사람에게는 더 좋은 혜택을 주고 싶다. 그래서 할인쿠폰제도를 운영하기로 했다. 그런데 발행한 할인쿠폰을 어떤 사람에게 주어야 할까? 이왕이면 할인쿠폰을 좋아하고 반응도가 높은 사람에게 주어야 하지 않을까? 그런 사람을 어떻게 구분할 수 있을까? 일일히 그 많은 손님에게 물어봐야 할까?

새미는 한가지 아이디어를 내었다. 우선은 자신의 가게를 찾아오는 손님마다 설문지를 돌려서 간단한 인적사항이나 개인정보를 조사하면서 할인쿠폰을 선물로 주었다. 그리고 6개월간 할인쿠폰에 대한 반응도를 YES/NO로 체크하기 시작했다. 그리고는 어떤 패턴을 가진 사람이 할인쿠폰을 좋아하는지 관찰해보기로 했다. 즉 문제는 이것이다.

1. 생활에서 만나는 문제

설문 데이터셋

	A	B	C	D	E	F	G
1	고객번호	성별	나이	직장여부	결혼여부	차량보유여부	쿠폰반응여부
2	1	남	30대	NO	YES	NO	NO
3	2	여	20대	YES	YES	YES	NO
4	3	여	20대	YES	YES	NO	NO
5	4	여	40대	NO	NO	NO	NO
6	5	여	30대	NO	YES	NO	NO
7	6	여	30대	NO	NO	YES	NO
8	7	여	20대	NO	YES	NO	NO
9	8	여	20대	NO	YES	YES	YES
10	9	여	30대	YES	YES	NO	YES

속성

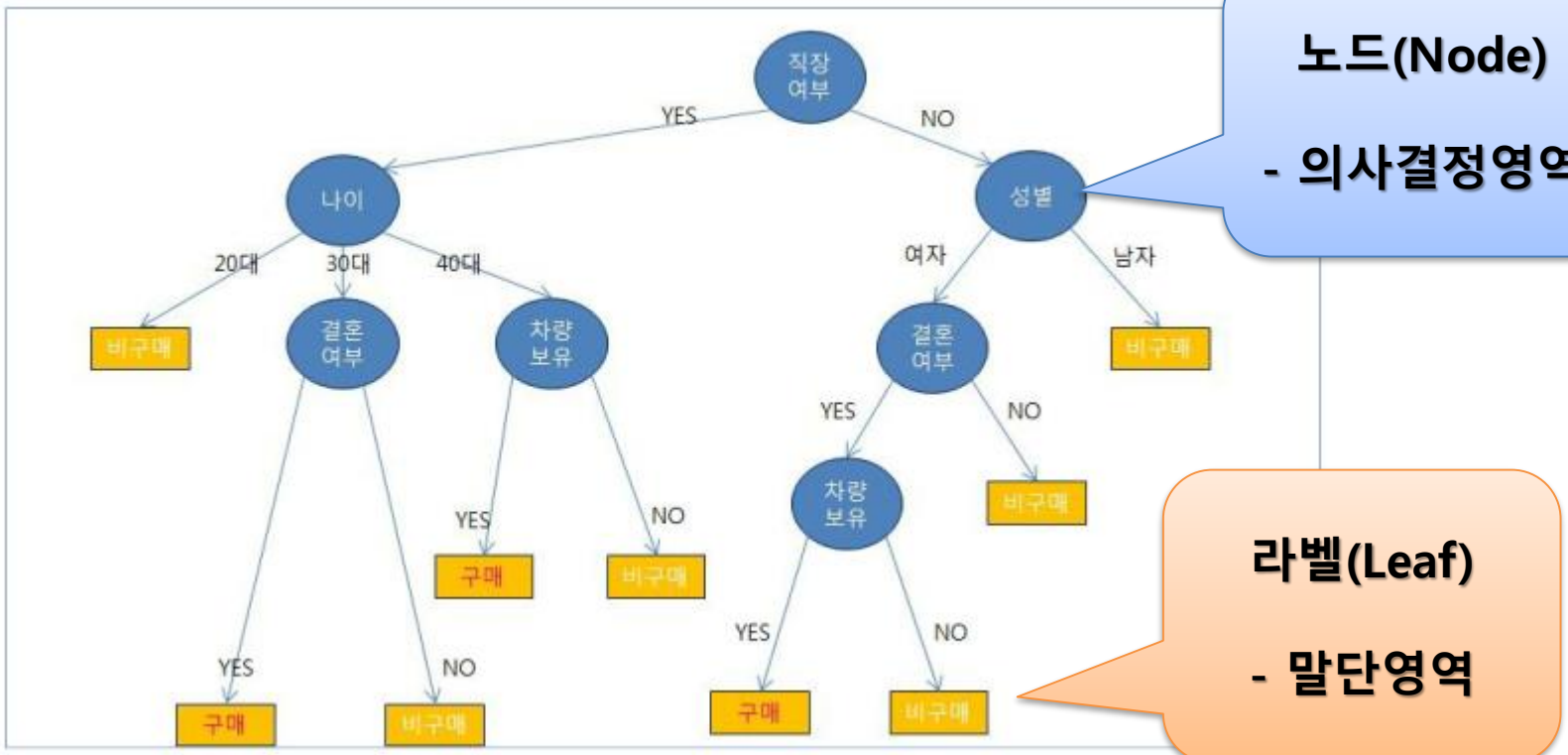
레이블

1.분류 문제

2.지도 학습

2. 해결책을 위한 사고실험

의사결정나무(Decision Tree)



3. 알고리즘과 수학적 정리

클로드 새넌의 정보이론

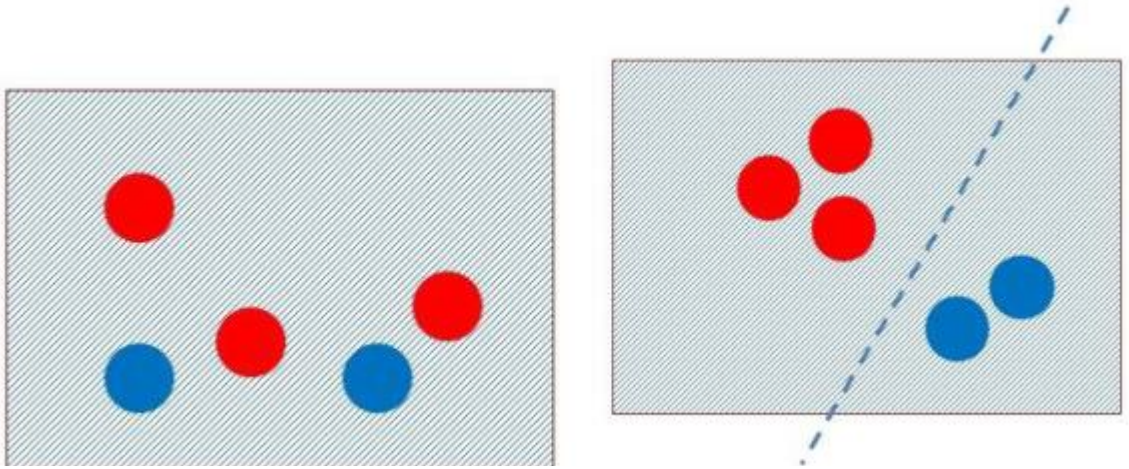
- 정보 이득(Information Gain) : 데이터를 분할하기 전과 후의 변화

  어떤 속성으로 데이터를 분할할 때 가장 높은 정보 이득을 취할 수 있는가?

- 섀넌 엔트로피(Shannon Entropy) : 무질서도, 혼잡도

3. 알고리즘과 수학적 정리

Entropy



빨강의 정보 : $-\log_2(\textit{빨강이 선택될 확률}) = -\log_2(\frac{3}{5})$

파랑의 정보 : $-\log_2(\textit{파랑이 선택될 확률}) = -\log_2(\frac{2}{5})$

이것을 가중치 평균을 하게 되면 정보의 기대값과 같은 값이 나오게 되는데 다음과 같이 된다. -부호는 음수를 방지하기 위하여 붙인다.

$-\frac{3}{5}\log_2(\frac{3}{5}) - \frac{2}{5}\log_2(\frac{2}{5})$ = 바구니의 Entropy = 0.9709506

이것을 좀 더 일반화시키면 다음과 같다. 여기서 n은 분류항목의 개수이다.

정보엔트로피 H = 정보의 기대값 = $-\sum_{i=1}^{n} p(x_i)\log_2 p(x_i)$

3. 알고리즘과 수학적 정리

Entropy

정보이득(information gain) Δ = Entropy(T) − Entropy(A)

정보이득비율 = Δ / Entropy(T)

```
> x <- c("red","blue","blue","red","red")
>
> #정보엔트로피를 구하는 함수
> info_entropy <- function(x) {
+
+   factor_x <- factor(x)
+   entropy <- 0
+   for(str in levels(factor_x)) {
+       pro <- sum(x == str) / length(x)
+       entropy <- entropy - pro * log2(pro)
+   }
+   return (entropy)
+ }
>
> info_entropy(x)
[1] 0.9709506
```

3. 알고리즘과 수학적 정리

어떤 변수를 기준으로 먼저 분할하는 것이 좋은가?

```
> #새미의 데이터셋에 대하여 적용해 본 엔트로피 계산
> #맨처음의 '쿠폰반응여부'의 엔트로피
> first_entropy <- info_entropy(skin[,"쿠폰반응여부"])
>
> for(str in colnames(skin)[1:5]) {
+
+     #str=조건변수 , factors=조건값집합
+     factors <- levels(skin[[str]])
+
+     #조건변수를 각각의 가능한 속성값으로 분류하였을때 '쿠폰반응여부'에 대한 엔트로피의 합계
+     sum_entropy <- 0
+     for(str2 in factors) {
+       test_x <- skin[skin[[str]] == str2,][6]
+       sum_entropy <- sum_entropy + info_entropy(test_x[,1])
+     }
+     cat(str , '----->' , sum_entropy,'\n')
+
+ }
성별 -----> 1.641098
나이 -----> 2.796506
직장여부 -----> 1.887994
결혼여부 -----> 0.9709506
차량보유여부 -----> 1.932395
```

엔트로피가 가장 작은 순서대로 나열한다면 결혼여부 > 성별 > 직장여부 > 차량보유여부 > 나이 순이 된다. 이것은 LABEL을 가장 깔끔하게 나누는 순서라고 생각해도 무방하다. 따라서 root node는 '결혼여부' 로 선택하는 것이 맞다. 그리고 해당변수로 선택하여 데이터를 나눈 후에는 위와 똑같은 작업을 또 다시 하여 엔트로피가 가장 작게 나오는 변수를 찾으면 된다. 하지만 이런 귀찮은 작업을 단 한 줄의 코드로 작업해주는 R의 함수들이 아주 많이 존재하므로 연구목적이 아니라면 이런 일련의 알고리즘은 대략적으로나마 이해하고 잊어버려도 상관없다.

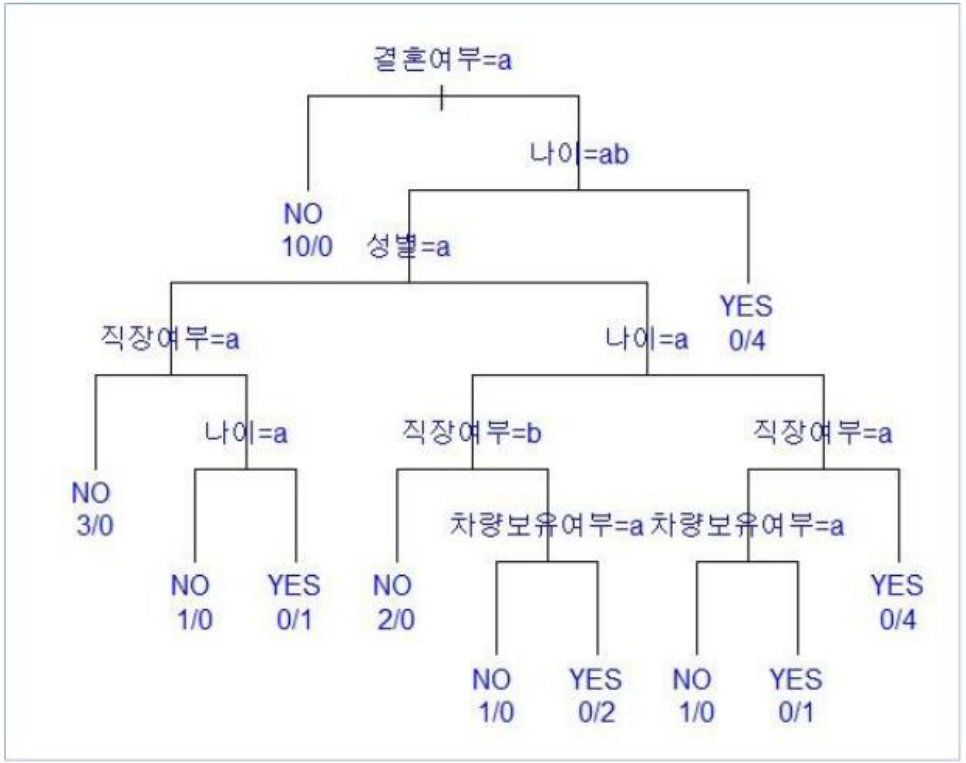
4. 코딩구현과 구체적인 해결방안

의사결정나무 : ID3 알고리즘 - rpart( )

- 트리 만들기

```
>
> library(rpart)
> tree1 <- rpart(쿠폰반응여부 ~ . , data = skin , control=rpart.control(minsplit = 2))
> plot(tree1 , compress = T , uniform = T , margin=0.1)
> text(tree1 , use.n = T , col = "blue")
```

- 그래프 그리기



4. 코딩구현과 구체적인 해결방안

의사결정나무

- 트리 출력하기

```
> tree1
n= 30

node), split, n, loss, yval, (yprob)
      * denotes terminal node

  1) root 30 12 NO (0.6000000 0.4000000)
    2) 결혼여부=NO 10  0 NO (1.0000000 0.0000000) *
    3) 결혼여부=YES 20  8 YES (0.4000000 0.6000000)
      6) 나이=20대,30대 16  8 NO (0.5000000 0.5000000)
       12) 성별=남 5  1 NO (0.8000000 0.2000000)
         24) 직장여부=NO 3  0 NO (1.0000000 0.0000000) *
         25) 직장여부=YES 2  1 NO (0.5000000 0.5000000)
           50) 나이=20대 1  0 NO (1.0000000 0.0000000) *
           51) 나이=30대 1  0 YES (0.0000000 1.0000000) *
       13) 성별=여 11  4 YES (0.3636364 0.6363636)
         26) 나이=20대 5  2 NO (0.6000000 0.4000000)
           52) 직장여부=YES 2  0 NO (1.0000000 0.0000000) *
           53) 직장여부=NO 3  1 YES (0.3333333 0.6666667)
            106) 차량보유여부=NO 1  0 NO (1.0000000 0.0000000) *
            107) 차량보유여부=YES 2  0 YES (0.0000000 1.0000000) *
         27) 나이=30대 6  1 YES (0.1666667 0.8333333)
           54) 직장여부=NO 2  1 NO (0.5000000 0.5000000)
            108) 차량보유여부=NO 1  0 NO (1.0000000 0.0000000) *
            109) 차량보유여부=YES 1  0 YES (0.0000000 1.0000000) *
           55) 직장여부=YES 4  0 YES (0.0000000 1.0000000) *
      7) 나이=40대 4  0 YES (0.0000000 1.0000000) *
```

4. 코딩구현과 구체적인 해결방안

고려사항 : 가지치기(Pruning)

```
>>> treepredict.prune(tree, 0.1)
>>> treepredict.printtree(tree)
0:google?
T-> 3:21?
      T-> {'Premium': 3}
      F-> 2:yes?
            T-> {'Basic': 1}
            F-> {'None': 1}
F-> 0:slashdot?
      T-> {'None': 3}
      F-> 2:yes?
            T-> {'Basic': 4}
            F-> 3:21?
                  T-> {'Basic': 1}
                  F-> {'None': 3}
```

과잉적합 문제
(Overfitting)

지정된 엔트로피 이하면
분할을 중단

=> 단순화가 가능

```
>>> treepredict.prune(tree, 1.0)
>>> treepredict.printtree(tree)
0:google?
T-> 3:21?
      T-> {'Premium': 3}
      F-> 2:yes?
            T-> {'Basic': 1}
            F-> {'None': 1}
F-> {'None': 6, 'Basic': 5}
```

5. 다시 한번 정리해보자

엔트로피의 대안 : 카이제곱스퀘어

카이제곱스퀘어는 각 셀마다 계산한 값을 모두 더한 값이다. 이것을 새미씨의 데이터셋에서 '결혼여부' 와 '쿠폰반응여부' 에 적용해보자.

```
> xtabs(~ 결혼여부 + 쿠폰반응여부 , data = skin)
        쿠폰반응여부
결혼여부 NO YES
     NO  10   0
     YES  8  12
> chisq.test(xtabs(~ 결혼여부 + 쿠폰반응여부 , data = skin))

        Pearson's Chi-squared test with Yates' continuity correction

data:  xtabs(~결혼여부 + 쿠폰반응여부, data = skin)
X-squared = 7.6562, df = 1, p-value = 0.005658
```

p-value가 0.05보다 훨씬 작으니 독립이다. 다른 변수로 시도를 해보면 알겠지만 '결혼여부' 가 가장 좋은 값을 보인다.

5. 다시 한번 정리해보자

엔트로피의 대안 : 지니계수

```
> #새미의 데이터셋에 대하여 적용해 본 지니계수 계산
> #맨처음의 '쿠폰반응여부'의 지니계수
> first_gini <- info_gini(skin[,"쿠폰반응여부"])
>
> for(str in colnames(skin)[1:5]) {
+
+   #str=조건변수 , factors=조건값집합
+   factors <- levels(skin[[str]])
+
+   #조건변수를 각각의 가능한 속성값으로 분류하였을때 '쿠폰반응여부'에 대한 엔트로피의 합계
+   sum_gini <- 0
+   for(str2 in factors) {
+     test_x <- skin[skin[[str]] == str2,][6]
+     sum_gini <- sum_gini + info_gini(test_x[,1])
+   }
+   cat(str , '----->' , sum_gini,'\n')
+
+ }
성별 -----> 0.7716049
나이 -----> 1.364796
직장여부 -----> 0.9243054
결혼여부 -----> 0.48
차량보유여부 -----> 0.9537888
```

```
> #지니계수를 구하는 함수
> info_gini <- function(x) {
+
+   factor_x <- factor(x)
+   gini_sum <- 0
+   for(str in levels(factor_x)) {
+     pro <- sum(x == str) / length(x)
+     gini_sum <- gini_sum + pro^2
+   }
+   return (1 - gini_sum)
+ }
```

기타

의사결정나무

- 분류기 : Label이 범주형
- 예측기 : Label이 수치형

Random Forest

- Tree를 여러 개 만듬

=> 확률적으로 결과를 반환

감사합니다.