**Big data Exercise #7 Ch10 – 의사결정나무**

|  |
| --- |
| Source Code( Data 추출) |
| Graphical user interface, text, application, chat or text message  Description automatically generated  Graphical user interface, text, application, email  Description automatically generated |
| Console( Data 추출) |
| Text  Description automatically generated  Text  Description automatically generatedText  Description automatically generated with medium confidenceTable  Description automatically generated |
| Source Code(모델링) |
| Text  Description automatically generated |
| Console(모델링) |
| 8:2 으로 나눴을 때 |
| Diagram  Description automatically generated  Diagram  Description automatically generatedDiagram  Description automatically generated |
| 7:3으로 나눴을 때 |
| Diagram  Description automatically generatedDiagram  Description automatically generatedDiagram  Description automatically generated |
| Source Code(결과비교)- |
| Graphical user interface, text, application  Description automatically generated |
| Console(결과비교) |
| 8:2 |
| Table  Description automatically generated |
| 7:3 |
| Table  Description automatically generated |

\* 8: 2로 나눈 데이터 의사 결정 나무

\* 7: 3으로 나눈 데이터 의사 결정 나무

1. Ch8 과제에서 사용한 carsale.csv 파일을 다시 보세요. 그 파일에는 세 가지 종류의 차를 산 고객들의 나이(age), 사는 지역(zip), 수입(salary), 신용(Credit), 주택(house) 정보가 들어있습니다.

다섯 가지 정보를 사용하여 의사결정나무를 만들어보세요. 데이터는 예제에서와 같이 8:2 혹은 7:3 등으로 자유롭게 나누어 검증해 보세요. Global pruning, CF의 설정 또한 자유롭게 해 보세요.

설정을 바꾸어 여러가지로 해 보고, 가장 정확도가 높게 나오는 결과를 그 설정과 함께 보여주세요. (정확도는 정답수/예측수의 비율에 의해 결정됩니다.)

Ch8의 과제에서 사용한 Carsale.csv파일을 이용해서 의사결정나무를 만들어봤다.

read.csv()를 통해서 파일을 읽어왔고, 파일에 있는 나이(age), 사는 지역 (zip), 수입(salary), 신용(Credit), 주택(house) 정보를 Factor 요인으로 넣어두었고, 이를 이용해서 세 가지 차 종류를 추론하는 모델을 만들어보았다. 데이터를 8:2 혹은 7:3으로 나눈 데이터 셋을 이용했고, 각각의 데이터 셋을 가지치기가 없는 경우, 지역적 가지치기, 전역적 가지치기 모델로 나눠서 분석을 해봤다.

놀랍게도 7:3으로 나눈 경우 세가지 가지치기 방법은 같은 결과가 나왔고, 8:2로 나온 세가지 결과가 각각 같았다. 8:2로 나눈 모델은 84개중 83개(98.8%)를 맞췄고, 7:3으로 나눈 모델은 126개중 123개(97.6%)를 맞췄다.

따라서 8:2로 나누어 학습을 할 경우 정답률이 높았고, 8:2로 나눴을 대, Global pruning과 CF값은 각각의 경우( NoGlobalpruning=True, CF=1 ), ( NoGlobalpruning=True, CF=0.25 ), ( NoGlobalpruning=False, CF=1 )인 경우였다.

그리고 추가적으로 ‘가지치기’에 AI관련 자료를 읽으며 연구해본 결과, ‘가지치기’는 Tree모델이 너무 복잡해지거나, 내용이 많아졌을 경우 학습을 할 때, 과적합(Overfitting)이 일어날 수 있다고 한다. 따라서 의사결정 모델의 과적합을 막기 위해서 가지치기 방법을 활용해 이를 막는다고 한다. ( 과적합이 되는 경우 훈련이 train data에 과도하게 학습되어, test data에 대한 예측이 떨어지는 경우를 말한다) 따라서 결정 모델의 과정에서 과적합이 일어날 경우(test data에 예측률이 많이 떨어질 경우) 가지치기를 해서 모델의 정확도 향상을 위해 사용해 볼 수 있는 좋은 방법이 될 것이다.

**문제 해결 Solution**

1. 예제와 다르게 파일을 읽어오고 나서 읽어온 파일들의 항목이 stringAsFactors = TRUE를 이용할 경우, Audi, Benz, BMW등이 character변수임으로 Factor로 이용 될 것 같았다. 그래서 구글링을 통해서 이미 입력된 항목을 factor로 바꿀 수 있는 방법을 찾아보았다. 그 결과 관련된 함수를 찾을 수 있었는데, 블로그(<https://bpapa.tistory.com/31>)에서 붓꽃 관련하여 factor로 바꿀 수 있는 예시를 찾았고, 그 함수를 일반화하여<dataframe$factor로 바꿀항목 <- as.factor(dataframe$factor로 바꿀항목) 과 같은 방식으로 바꿔져서 문제를 해결 하였다.

실제 이용한 코드:

Logo

Description automatically generated with low confidence

2.

이번 학습 시에는 csv데이터에 id가 평가항목으로 이용될 수 있었다. csv의 자료를 살펴보니, id에 관한 항목은 구매항목과 다르게 독립적으로 증가하고 있었다. 또한 문제에서 5가지 따라서 나이(age), 사는 지역(zip), 수입(salary), 신용(Credit), 주택(house) 등을 통해서 학습하라고 명시 되어있었다. 따라서 Id는 label을 예측과 관련 없는 데이터라는 점과 문제에서 고객들의 학습정보로 제시하지 않았다는 점에서 제외하고 모델링을 해야한다고 파악 했다. 따라서 모델링을 할 경우에는 이 id항목이 이용되지 않을 수 있도록 c객체를 이용해서 필요한 열을 추출해서 학습해주었다.