

과제 2: 추론 [5팀]



과목명: 인공지능(목 7-9)

교수명: 한경수 교수님

팀원: 20190274 정택원

20201092 정유진

20210833 남민지

목차

1. 문제 도메인 선정	3
2. 문제 선택 배경	3
3. 지식베이스(KB) 구축	3
4. 질문에 대한 답변 추출 및 코드.....	5
5. 현실과의 괴리.....	6
6. 프로젝트 실행 결과.....	11
7. 개인 보고서.....	13

1. 문제 도메인 선정

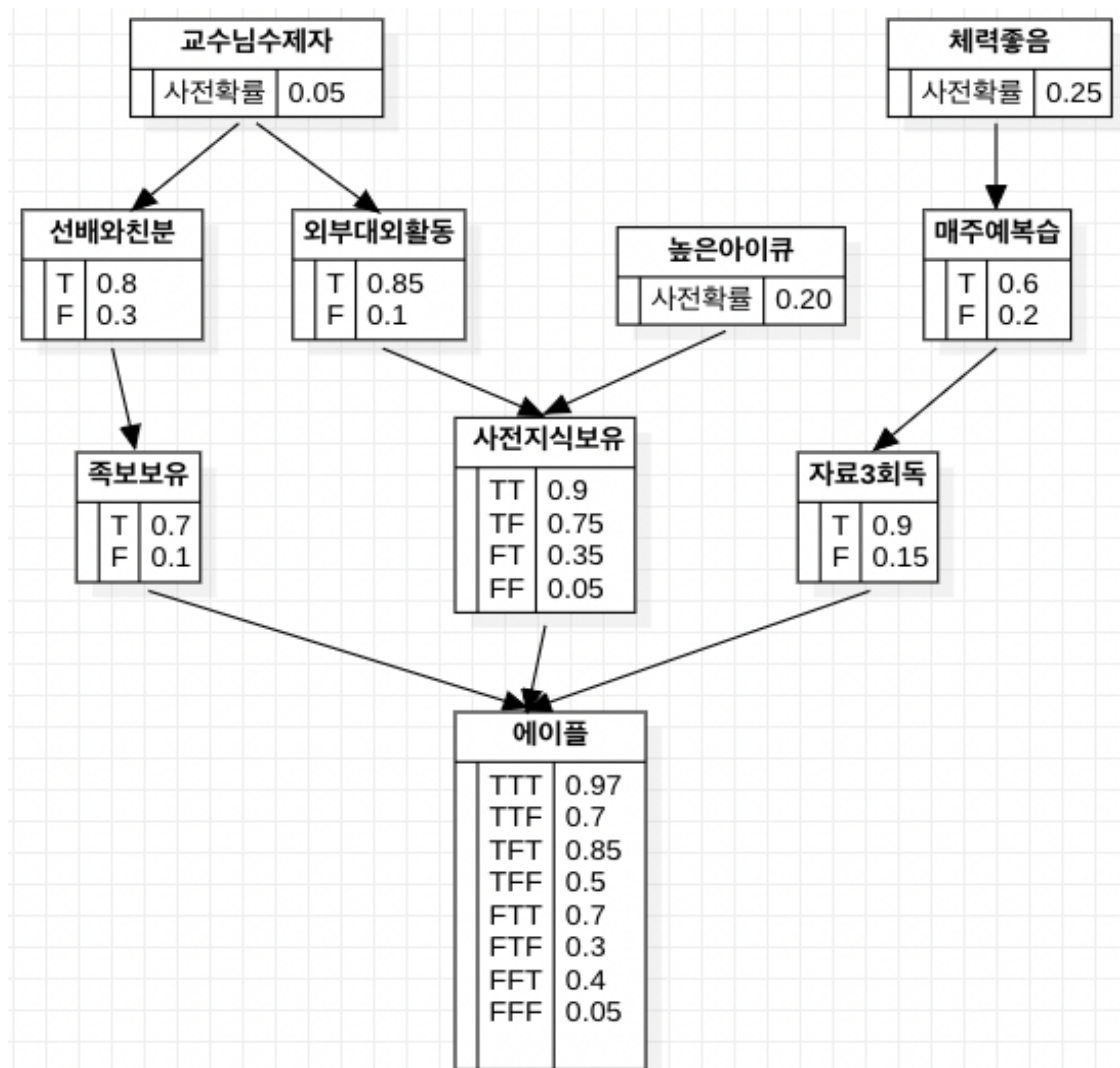
본 팀은 사용자의 상황을 입력 받아, A+의 성적을 받을 확률을 추론한다. 추론 알고리즘으로는 Gibbs 샘플링 알고리즘을 사용하며, 변수 소거를 통한 추론의 확률과 비교하여 Gibbs 샘플링이 얼마나 정확하게 측정하는지 확인한다.

2. 문제 선택 배경

수강하는 과목에서 받을 수 있는 최고 학점은 A+이다. 학기가 지남에 따라 A+성적을 부여할 수 있는 기준 또한 달라진다. 무엇보다 중요한 것은 학생으로서 얼마나 성실히 수업에 참여했는가이다. 본 팀은 성실함을 포함한 여러가지 변수들을 정의하고 사용자로부터 변수(이하 노드)의 참 거짓 값을 입력 받아, 학생들이 최고 학점을 받을 확률을 추론해 주는 것이 흥미로울 것 같아 위와 같은 주제를 선정하게 되었다.

3. 지식베이스(KB) 구축

본 팀의 추론 방식으로 Gibbs샘플링을 이용하기 위해 베이지 네트워크를 직접 제작하였다.



[그림 1] 베이지 네트워크 정의

- '교수님의 수제자': 평소 행실이 우수하여 교수님의 마음 한 켠에 자리잡은 학생이다.
- '높은 아이큐': 학습을 하는 데에 있어 높은 아이큐는 큰 영향을 미친다.
- '체력 좋음: 건강한 육체는 건강한 마음을 갖게 한다.
- '선배와 친분': 교수님의 마음 한 켠을 이미 자리잡은 선배들을 소개받을 수 있다.
- '외부 대외활동': 진로 탐색 학점제, 다양한 학술제 등을 교수님과 함께 진행할 수 있다.
- '죽보 보유': 선배와 친분이 있다면 현재 수강하고 있는 교수님의 문제 제출 경향을 선배들을 통해 미리 파악할 수 있다.
- '사전 지식 보유': 높은 아이큐로 청소년기에 많은 독서와 경험을 할 수 있고, 외부 대외활동을 통해 수업의 내용을 뛰어넘는 고급 과정을 알 수 있다.
- '매주 예복습': 체력이 좋지 않으면 3시간의 수업을 수강하고 20시간을 취침해야 피로가 풀리는 학생이 존재할 수 있다. 체력이 좋으면 수업 직후 예복습이 가능하다.
- '자료 3회독': 매주 예복습을 통해 수업 내용에 대해 탐구욕이 솟구쳐 시험을 치루기 전까지 제공된 자료를 3회독 이상 할 수 있다.
- '에이플': 가장 많은 조상을 갖는 노드로, 본 팀이 추론하고자 하는 노드이다.

4. 질문에 대한 답변 추출 및 코드

사용자로부터 받을 수 있는 입력은 2가지가 존재한다.

→ 첫번째는 아무런 입력을 받지 않는다. 아무런 입력을 받지 않으면, A학점의 실제 비율인 35%와 근접하게 추론을 할 수 있도록 사전/사후 확률을 조정하였다. 결과적으로 약 36%의 확률을 추론한다.

→ 두번째는 베이지 네트워크에 존재하는 노드들의 값을 사용자로부터 입력 받는다. 모든 노드에 대해 True, False값 중 하나를 입력 받고 해당 상황에 대해 A+학점을 받을 확률을 추론한다.

‘에이플’의 확률이 아닌 다른 변수의 확률 추론 또한 가능하다. 추론을 진행할 때에는 추론하고자 하는 변수와 증거변수는 달라야 한다.

```
In [7]: # 결정론적인 노드 x
Aplus1 = BayesNet([
    ("교수님의수제자", "", 0.05),
    ("선배와친분", "교수님의수제자", {T: 0.8, F: 0.3}),
    ("족보보유", "선배와친분", {T: 0.7, F: 0.1}),
    ("외부대외활동", "교수님의수제자", {T: 0.85, F: 0.1}),
    ("높은아이큐", "", 0.2),
    ("사전지식보유", "외부대외활동 높은아이큐", {(T, T): 0.9, (T, F): 0.75, (F, T): 0.35, (F, F): 0.05}),
    ("체력 좋음", "", 0.25),
    ("매주예복습", "체력 좋음", {T: 0.6, F: 0.2}),
    ("자료3회독", "매주예복습", {T: 0.9, F: 0.15}),
    ("에이플", "족보보유 사전지식보유 자료3회독",
     {(T, T, T): 0.97, (T, T, F): 0.7, (T, F, T): 0.85, (F, T, T): 0.7,
      (T, F, F): 0.5, (F, T, F): 0.3, (F, F, T): 0.4, (F, F, F): 0.05})
])

In [25]: ans_dist = elimination_ask('에이플', dict(), Aplus1)
print('진 확률: ', end='')
ans_dist.show_approx()
print(round(ans_dist[True]*100, 3), end='%\n')

print('Gibbs sampling: ', end='')
dist = gibbs_ask('에이플', dict(), Aplus1, 10000)
print(round(dist[True]*100, 3), end='%\n')

진 확률: 36.173%
Gibbs sampling: 36.139%
```

[그림 2] 증거변수가 주어지지 않은 상황에서의 샘플링

→ 위 셀에서 주어진 베이지 네트워크를 정의하였다.

→ 아래 셀에서 기각 샘플링을 통해 추론한 진 확률과 10,000개의 샘플을 이용한 Gibbs샘플링 결과를 출력한다.

```
In [30]: ans_dist = elimination_ask('에이플', dict(교수님의수제자=True, 높은아이큐=True, 체력 좋음=False), Aplus1)
print('진 확률: ', end='')
ans_dist.show_approx()
print(round(ans_dist[True]*100, 3), end='%\n')

print('Gibbs sampling: ', end='')
dist = gibbs_ask('에이플', dict(교수님의수제자=True, 높은아이큐=True, 체력 좋음=False), Aplus1, 10000)
print(round(dist[True]*100, 3), end='%\n')

진 확률: 59.044%
Gibbs sampling: 58.61%
```

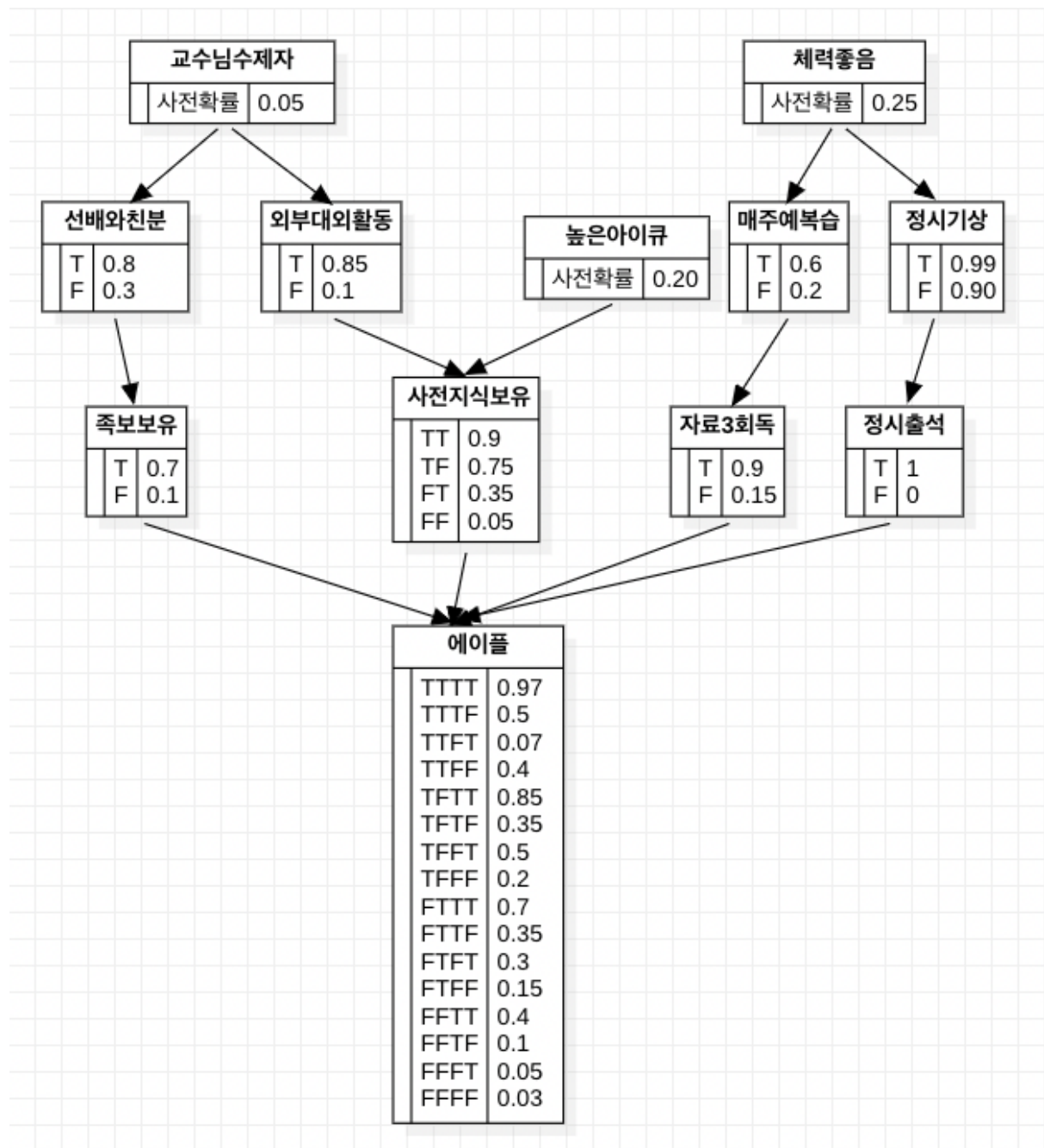
[그림 3] 증거변수가 주어진 상황에서의 샘플링

→ 교수님의 수제자, 높은 아이큐, 체력 좋음 변수의 값을 입력 받아 10,000개의 샘플을 이용한 Gibbs샘플링 결과를 출력한다.

5. 현실과의 괴리

Gibbs샘플링의 수렴은 에르고드성을 충족해야 하며, 베이지 네트워크에 결정론적 조건부 분포가 없어야 한다. 하지만, 수업에 일정 횟수 이상 결석을 하게 된다면 성적은 결정론적으로 F를 받게 된다. 본 팀이 추론하고자 하는 A+ 성적은 당연히 받을 수 없다.

결정론적인 조건부 분포를 갖는 노드를 정의하고, 실제 추론과 다른 점을 찾아 수정하는 과정을 소개한다.



[그림 4] 결정론적 조건부 분포를 포함한 베이지 네트워크

```

In [9]: # 결정론적 노드 o
Aplus2 = BayesNet([
    ('교수님의수제자', '', 0.05),
    ('선배와친분', '교수님의수제자', {T: 0.8, F: 0.3}),
    ('족보보유', '선배와친분', {T: 0.7, F: 0.1}),
    ('외부대외활동', '교수님의수제자', {T: 0.85, F: 0.1}),
    ('높은아이큐', '', 0.2),
    ('사전지식보유', '외부대외활동 높은아이큐', {(T, T): 0.9, (T, F): 0.75, (F, T): 0.35, (F, F): 0.05}),
    ('체력좋은', '', 0.25),
    ('매주예복습', '체력좋은', {T: 0.6, F: 0.2}),
    ('자료3회독', '매주예복습', {T: 0.9, F: 0.15}),
    ('정시기상', '체력좋은', {T: 0.99, F: 0.90}),
    ('정시출석', '정시기상', {T: 1, F: 0}),
    ('에이플', '족보보유 사전지식보유 자료3회독 정시출석',
    {(T, T, T, T): 0.97, (T, T, F, T): 0.7, (T, F, T, T): 0.85, (F, T, T, T): 0.7,
    (T, F, F, T): 0.5, (F, T, F, T): 0.3, (F, F, T, T): 0.4, (F, F, F, T): 0.05,
    (T, T, T, F): 0.5, (T, T, F, F): 0.4, (T, F, T, F): 0.35, (F, T, T, F): 0.35,
    (T, F, F, F): 0.2, (F, T, F, F): 0.15, (F, F, T, F): 0.1, (F, F, F, F): 0.03})
])

In [35]: ans_dist = elimination_ask('에이플', dict(), Aplus2)
print('진 확률: ', end='')
ans_dist.show_approx()
print(round(ans_dist[True]*100, 3), end='%\n')

for _ in range(2):
    print('Gibbs sampling: ', end='')
    dist = gibbs_ask('에이플', dict(), Aplus2, 10000)
    print(round(dist[True]*100, 3), end='%\n')

진 확률: 34.657%
Gibbs sampling: 14.519%
Gibbs sampling: 36.83%

```

[그림 5] 결정론적 조건부 분포를 포함한 추론 결과

- 결정론적 조건부 분포를 포함한 결과는 정확하지 않음을 알 수 있다.
- 해당 베이즈 네트워크에서는 정시기상과 정시출석이 반드시 같은 값을 가지게 된다. 변수를 랜덤하게 하나만 변경하는 Gibbs 샘플링에서는 두 값의 동시 변경이 불가능하므로 정시기상과 정시출석은 절대 변할 수 없는 값이 된다. 하나만 바꾸면 모순이 발생하기 때문이다. 따라서 최초의 샘플 생성시 만들어진 값이 마치 증거변수처럼 변하지 않고 고정된다.

```

In [11]: ans_dist = elimination_ask('에이플', dict(정시기상=True), Aplus2)
print('정시기상이 True일 때 A+일 확률: ', end='')
print(round(ans_dist[True]*100,3), end="%\n")

ans_dist = elimination_ask('에이플', dict(정시기상=False), Aplus2)
print('정시기상이 False일 때 A+일 확률: ', end='')
print(round(ans_dist[True]*100,3), end="%\n")

정시기상이 True일 때 A+일 확률: 36.365%
정시기상이 False일 때 A+일 확률: 14.315%

```

[그림 6] 결정론적 노드를 증거로 한 변수 소거 확률

- 정시기상을 증거변수로 주었을 때의 확률과 Gibbs 샘플링의 확률이 유사함을 확인할 수 있다.

```

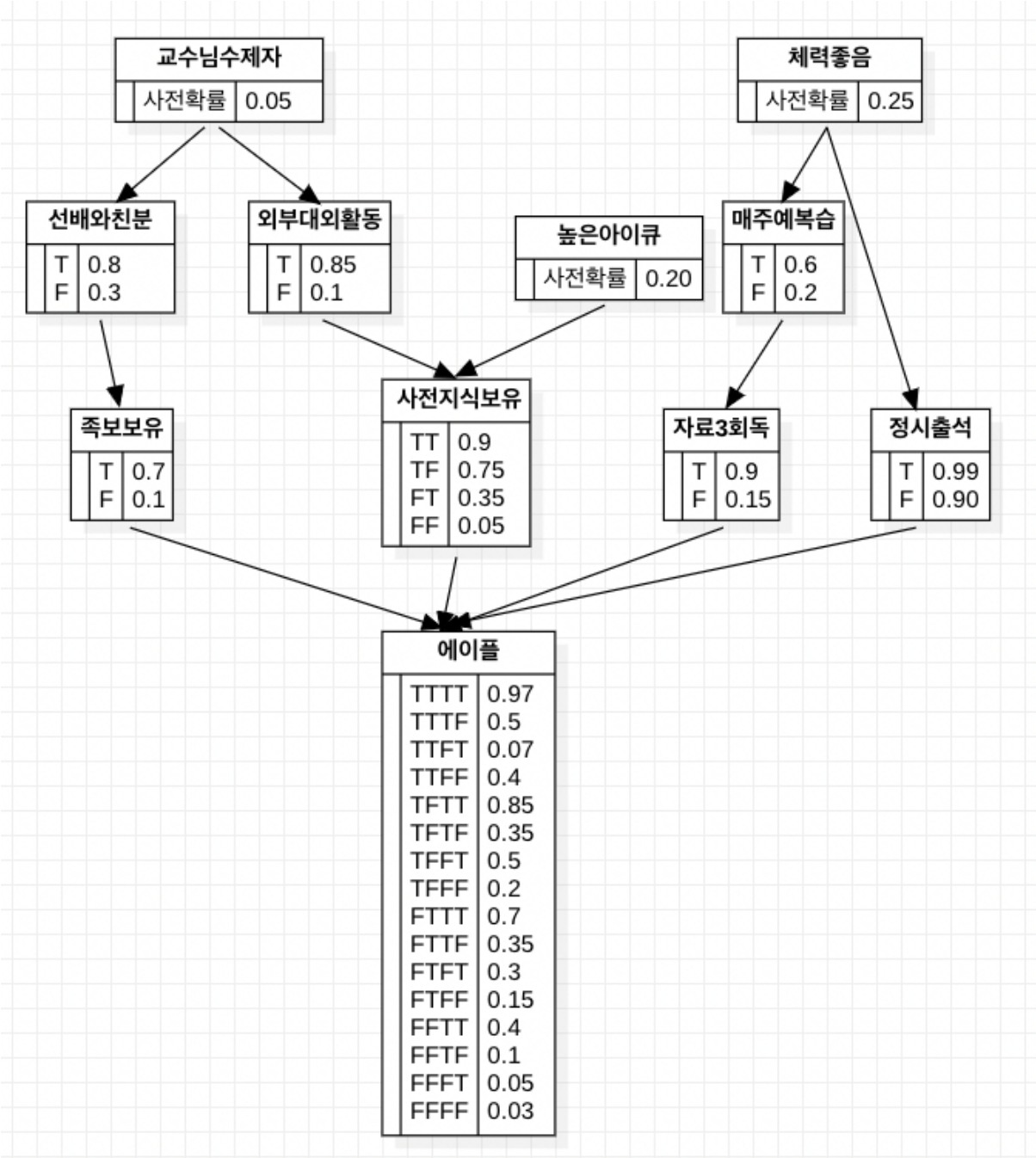
def normalize(self):
    """모든 확률값의 합이 1이 되도록 정규화하여 정규화된 분포를 리턴함.
    확률값 합이 0이면, ZeroDivisionError가 발생됨."""
    total = sum(self.prob.values())
    if not np.isclose(total, 1.0):
        for val in self.prob:
            if total == 0:
                self.prob[val] = 1
            else:
                self.prob[val] /= total
    return self

```

[그림 7] probability.py 코드수정

→ 또한 정규화 과정에서 ZeroDivisionError가 발생하여 코드를 수정해야한다.

절대적인 결정론적 조건부 분포를 제거하고 변수를 수정하여 샘플링 중 다양한 샘플을 생성하지 못하는 문제를 해결한 새로운 베이지 네트워크를 정의한다. 정시기상과 정시출석은 항상 같은 값을 갖기 때문에 하나를 생략하여 베이지 네트워크를 간소화 하였다.



[그림 8] 결정론적 조건부 분포를 제거한 베이지 네트워크

```
In [11]: # 결정론적 노드 수정
Aplus3 = BayesNet([
    ('교수님의수제자', '', 0.05),
    ('선배와친분', '교수님의수제자', {T: 0.8, F: 0.3}),
    ('족보보유', '선배와친분', {T: 0.7, F: 0.1}),
    ('외부대외활동', '교수님의수제자', {T: 0.85, F: 0.1}),
    ('높은아이큐', '', 0.2),
    ('사전지식보유', '외부대외활동 높은아이큐', {(T, T): 0.9, (T, F): 0.75, (F, T): 0.35, (F, F): 0.05}),
    ('체력 좋음', '', 0.25),
    ('매주예복습', '체력 좋음', {T: 0.6, F: 0.2}),
    ('자료3회독', '매주예복습', {T: 0.9, F: 0.15}),
    ('정시출석', '체력 좋음', {T: 0.99, F: 0.90}),
    ('에이플', '족보보유 사전지식보유 자료3회독 정시출석',
    {(T, T, T, T): 0.97, (T, T, F, T): 0.7, (T, F, T, T): 0.85, (F, T, T, T): 0.7,
    (T, F, F, T): 0.5, (F, T, F, T): 0.3, (F, F, T, T): 0.4, (F, F, F, T): 0.05,
    (T, T, T, F): 0.5, (T, T, F, F): 0.4, (T, F, T, F): 0.35, (F, T, T, F): 0.35,
    (T, F, F, F): 0.2, (F, T, F, F): 0.15, (F, F, T, F): 0.1, (F, F, F, F): 0.03})
])
```

```
In [37]: # 10,000 => 2초, 100,000 => 15초
ans_dist = elimination_ask('에이플', dict(), Aplus3)
print('진 확률: ', end='')
ans_dist.show_approx()
print(round(ans_dist[True]*100, 3), end='%\n')

for _ in range(10):
    print('Gibbs sampling: ', end='')
    dist = gibbs_ask('에이플', dict(), Aplus3, 10000)
    print(round(dist[True]*100, 3), end='%\n')
```

```
진 확률: 34.657%
Gibbs sampling: 34.391%
Gibbs sampling: 34.309%
Gibbs sampling: 34.811%
Gibbs sampling: 34.28%
Gibbs sampling: 35.06%
Gibbs sampling: 34.89%
Gibbs sampling: 34.661%
Gibbs sampling: 34.609%
Gibbs sampling: 34.799%
Gibbs sampling: 33.75%
```

[그림 9] 결정론적 조건부 분포를 제거한 추론 결과

→ 결정론적 조건부 분포를 포함한 추론 결과보다 더 나은 확률을 갖는다. '정시출석' 변수 값이 True, False여도 결정론적 조건부 분포를 제거하였기 때문에, Gibbs 샘플링 과정에서 충분히 값이 변화할 수 있다. 그 결과로 본 팀의 의도에 맞게 에르고드성을 띄어 어느 곳에 치우치지 않는 샘플들이 생성될 수 있다.

6. 프로젝트 실행 결과

반갑습니다! 아래에 출력되는 변수들 중 추론하고싶은 변수를 선택해주세요.
프로젝트의 목적에 맞게 11을 선택하시는것을 권장합니다.

- 1: 교수님의수제자
- 2: 선배와친분
- 3: 족보보유
- 4: 외부대외활동
- 5: 높은아이큐
- 6: 사전지식보유
- 7: 체력좋은
- 8: 매주예복습
- 9: 자료3회독
- 10: 정시출석
- 11: 에이플

추론하고자 하는 변수 선택(ex: 11): 11
'에이플'의 확률을 추론합니다.
증거변수를 선택하시겠습니까? (아니오: 0, 예: 1) 0

증거변수 없이 '에이플'의 확률을 추론합니다.

진 확률(정확추론; 변수 소거 알고리즘): 34.657%
rejection_sampling: 34.72%
likelihood_weighting: 34.34%
Gibbs sampling: 34.59%

[그림 10] 프로젝트 실행: 증거변수 없이 추론 결과

제작한 프로젝트를 실행한 결과이다. 추론하고자 하는 변수인 '에이플'을 선택하였고, 증거변수는 선택하지 않았다. 그에 따른 진 확률, 기각 샘플링, 중요도 샘플링, Gibbs 샘플링 결과를 출력한다.

반갑습니다! 아래에 출력되는 변수들 중 추론하고싶은 변수를 선택해주세요.
프로젝트의 목적에 맞게 11을 선택하시는것을 권장합니다.

- 1: 교수님의수제자
- 2: 선배와친분
- 3: 족보보유
- 4: 외부대외활동
- 5: 높은아이큐
- 6: 사전지식보유
- 7: 체력 좋음
- 8: 매주예복습
- 9: 자료3회독
- 10: 정시출석
- 11: 에이플

추론하고자 하는 변수 선택(ex: 11): 11
'에이플'의 확률을 추론합니다.
증거변수를 선택하시겠습니까? (아니오: 0, 예: 1) 1

증거변수를 선택해주세요(ex: 10, 선택 중단: 0): 1
'교수님의수제자'의 값을 입력해주세요(T or F)T

선택된 증거변수: 교수님의수제자: True

증거변수를 선택해주세요(ex: 10, 선택 중단: 0): 10
'정시출석'의 값을 입력해주세요(T or F)T

선택된 증거변수: 교수님의수제자: True 정시출석: True

증거변수를 선택해주세요(ex: 10, 선택 중단: 0): 0

증거변수 교수님의수제자: True 정시출석: True 를 기반으로 '에이플'의 확률을 추론합니다.

진 확률(정확추론; 변수 소거 알고리즘): 58.777%
rejection_sampling: 54.978%
likelihood_weighting: 58.163%
Gibbs sampling: 56.54%

[그림 11] 프로젝트 실행: 증거변수 제공 추론 결과

추론하고자 하는 변수는 '에이플'이고 증거변수를 선택하였다. '교수님의수제자', '정시출석' 변수의 값을 True로 입력하였고, 그에 따른 결과를 추론한다.

컴퓨터공학과 20190274 정택원

15주차 수업을 듣기 전, 팀원들과 회의에서 집 주변 아르바이트 구인 공고를 지식베이스로 넣음으로써 사용자가 원하는 급여를 받을 수 있는지 없는지에 대한 추론 주제를 제안하였습니다. 변수로는 시간대, 시급, 각종 수당 등이 있었습니다. 아쉽게도 선정된 주제는 단순 베이즈 모델을 이용한 스팸 메일 분석으로 정해졌습니다. 입력을 분석하여 키워드를 추출하고, 그에 따라 어떤 메일의 스팸 여부를 추론하는 주제였지만, 15주차 수업을 듣고 근사 추론과 정확 추론을 비교하면 더 멋진 추론 결과를 도출할 수 있을 것이라 생각하여 Gibbs샘플링을 통한 근사 추론을 진행하기로 하였습니다.

학생들이 흥미로워 할 수 있는 주제로 바꾸었습니다. 주제는 A+학점을 받을 수 있는 확률을 추론하는 것으로 정했습니다. 추론 결과를 A, B, C, F 학점으로 출력 가능하도록 설계하고 싶었지만, 코드상 변수의 값을 Boolean(T, F)으로만 설정 가능하다는 제한이 존재하였습니다. 여러 제약을 모두 고려하여 나온 추론 방식은, 사용자의 상황을 입력 받아, 그에 따른 A+학점을 받을 확률을 추론하는 것입니다. 성적에 영향을 미칠 변수들을 정의하고 그에 맞게 상황을 일반화하여 변수의 확률을 정의했습니다. 그 후 코드를 동작 시켜 보며 A+학점을 받을 수 있는 확률을 실제 비율과 유사하게 35%로 추론될 수 있도록 변수의 확률을 수정하였습니다.

베이즈 네트워크를 주어진 코드에 맞게 정의하였습니다. 근사추론에서는 결정론적 조건부 분포가 존재하면 안되지만, 실제 학점은 결석 4번이면 A+학점을 받을 수 없습니다. 현실과 맞추어 추론을 하기 위해 결정론적 조건부 분포를 포함하여 Gibbs샘플링을 통한 추론을 하였습니다. 정규화를 하는 과정에서 확률이 0인 변수가 존재하여 ZeroDivisionError가 발생하여 코드를 알맞게 수정하였습니다. 그 후 결정론적 조건부 분포를 갖는 변수를 제거하고 수정하여, 출석에 관련한 샘플이 만들어져도 에르고드성을 만족하도록 하였습니다.

베이즈 네트워크를 시각화하기 위해 starUML을 이용하여 제작하였습니다.

근사 추론과 정확 추론이 얼마나 차이가 있는지 확인하기 위하여, 변수 소거 알고리즘과 근사 추론 알고리즘들을 동시에 실행하면서 추론 결과의 차이를 비교하여 어디에 문제가 발생하고 어떤 경우에 에르고드성을 충족하지 않는지 디버깅을 하였습니다.

코드의 공유를 위해 **깃허브 레파지토리**¹를 생성하여 형상 관리를 통한 작업을 진행하였습니다.

팀원과 철야를 하며 실 생활에 일어나는 현상을 추론하는 과정은 매우 흥미로웠습니다. 이번 과제를 통해 지식 베이스(KB)가 주어진 모델에서의 정확 추론, 근사 추론 알고리즘들의 단계별 절차를 깊게 이해할 수 있었습니다. 지식 베이스기반에서 더 나아가 머신 러닝은 어떠한 학습 과정을 거쳐 추론을 하는지 궁금해졌습니다. 감사합니다.

¹ <https://github.com/chtw2001/aiclass2>

산업경영공학과 20201092 정유진

과제 선정에 있어 여러 의견이 나왔었다. 지식추론과 확률추론 중 어떤 추론을 사용해야 할지 다양하게 나왔다. 14주차가 진행되기전 13주차 강의가 끝난 후 진행된 회의 였기 때문에 근사추론에 대한 주제는 나오지 않고 일차논리와 명제논리, 베이지논리 등에 대한 주제들이 나왔다. 주제에 있어서는 통장 일차논리와 베이지모델에 대한 주제를 제시했었다. 팀원들도 각자 1-2개의 의견을 냈고 처음에는 남민지 학우님이 제시하신 단순 베이지모델을 사용한 스팸메일, 키워드를 분석하는 주제로 정했었다. 하지만 14주차 근사추론에 대한 강의 이후 근사추론을 이용한 A+를 맞을 확률을 추론해주는 것이 되었다.

파이썬 프로그램에 대해서는 남민지 학우님과 정택원 학우님이 진행하셨다. 정택원 학우님이 깃을 통해 프로그래밍을 공유해 주셨다.

과제를 진행하며 추론에 대해 더 이해가 갔던 것 같다. 단순 외우기보다 과제 진행을 위해 실습이 진행되어 이해가 더 잘 되었다. 여러 추론들 중 주제를 정하기가 어려웠고 내가 생각한 주제가 배운 추론과 맞는 주제인지 확신이 잘 서지 않았지만 팀 회의 진행을 통해 내가 맞게 내용을 이해하고 있다는 것을 알게 해주었다. 파이썬 프로그램들도 기존의 프로그래밍은 이해할 수 있었지만 새로운 주제에 맞게 프로그래밍하는 것은 팀원분들이 진행해주었다.

중간 전보다 어려워진 내용이었다. 점점 어려워진 추론들이 등장했고 주제에 대해서도 진도가 나갈 때마다 다른 생각을 하게 되었다. 인공지능에 대해 단순하게 생각하고 있었는데 이번 강의를 통해 답변이 단순히 아니라 여러 과정이 있다는 것을 알게 해주었다. 이번 강의를 통해 근사추론을 더 잘 이해할 수 있었다.

20210833 남민지

첫 회의는 베이지 네트워크를 배우기 전이었다. 단순 베이지 모델이나 논리적 추론을 통해 해결할 수 있는 주제를 정해야 했다. 나는 논리적 추론을 이용한 분리수거와 날씨 별 적절한 옷 추천, 단순 베이지 모델을 이용한 텍스트 감정 추측과 스팸 메일 필터링을 제안했다. 팀원들과 의견을 나눈 결과 스팸 메일 필터링으로 의견이 모였고, 실제 스팸 메일을 분석하여 모델을 만들자고 하였다. 하지만 이후 수업에서 배운 근사추론이 너무 매력적이었고, 스팸 메일 필터링은 스팸 메일을 분석하는 시간이 오래 걸릴 것이라 예상했다. 추론 과정보다 더 오래 걸린다면 과제의 목적과 다른 것에 너무 많은 시간을 투자하는 것이 될 것이라 생각하여 주제를 변경했다.

어차피 통계 같은 객관적인 자료를 수집하기 어려운 이상 직관적이고 공감할 구석이 있는, 일반적인 지식을 사용할 수 있는 주제를 선택하고자 했다. 우리는 전원 대학생이고, A+에 대한 욕심이 있었으므로 정보를 주면 학점에 대한 확률을 계산하는 추론을 주제로 삼았다. 처음에는 학점을 각각 A, B, C, F 확률로 계산하고자 하였으나 주어진 샘플 코드는 True와 False만 입력할 수 있었다. 그래서 과제의 방향을 A+를 받을 확률로 변경했으며, 각 변수의 세부사항도 조금 변경했다.

우선은 확률을 계산하기 전에 어떤 변수가 어떤 변수의 값에 영향을 미칠지 고민하여 인과관계를 정리했다. 이후 조건부 확률 표를 아직 정의하지 않은 베이지 네트워크를 그려 전체 형태를 파악했다. 이후 팀원과의 상의와 개인적인 경험을 바탕으로 확률을 추정하고 코드를 작성했다. 변수 소거를 통한 확률 계산을 참고하며, 아무런 조건이 주어지지 않았을 때 A+일 확률이 실제 A 학점의 비율인 35%와 유사하도록 신경 썼다.

추론방법으로 깃스 샘플링을 선택하였으므로, 에르고드성을 충족하지 못했을 때 깃스 샘플링이 수렴하지 못하는 것을 확인해보고 싶었다. 출력 관리에 실패했을 때, 다른 요소와는 무관하게 반드시 A+이 False가 되도록 설정했다. 하지만 깃스 샘플링 결과 변수 소거를 통한 확률 값에 수렴하는 결과가 나왔다. 지금 와서 생각해보면, 해당 네트워크는 그저 문맥 국한 독립성을 가지고 있는 네트워크였다. 출력 관리가 True일 때 A+이 반드시 True인 것이 아닌 이상 해당 그래프를 무작위로 돌아다니는 것에 문제가 생기지 않기 때문이다.

오랜 시간 고민하고 찾아본 끝에 두 변수의 값이 완전히 결정되어야 에르고드성이 미 충족될 것임을 깨달았다. A가 B를 완전히 결정한다면 반대로 B가 A를 결정한다고 볼 수 있다. 변수를 하나만 변경하는 깃스 샘플링 특성상 하나만 변경했을 때 모순이 발생하므로 두 변수 모두 절대 변하지 않는 값이 되었다. 처음 상태를 생성할 때 결정되는 두 변수의 값이 마치 증거변수처럼 변하지 못했다. 변수 소거를 통한 실제 확률과 결정론적 변수를 증거로 주었을 때의 확률을 비교하여 해당 사실을 확인했다.

탐색과제와 비교하면 코드 작업이 훨씬 적었지만, 사고과정이 더 필요한 만큼 쉽지는 않은 과제였다. 하지만 팀원과 학교에서 철야 하며 문제를 토의하는 과정은 재미있었다. 베이지 네트워크와 깃스 샘플링의 원리에 대해 더 확실하게 이해할 수 있었다.