PS8

Graded

Student

Chetan Hiremath

Total Points

88 / 100 pts

Question 1

8.1 Probability Warmup

10 / 10 pts

✔ **+ 10 pts** Correct

Question 2

8.2 Bayes Net for College Admissions

15 / 15 pts

✔ **+ 15 pts** Correct

Question 3

8.3 Bayes Net for Earthquakes

20 / 20 pts

✔ **+ 20 pts** Correct

Question 4

8.5 Fuzzy Logic **3** / 15 pts

4.1 **(a)** **3** / 3 pts

✔ **+ 3 pts** Correct

4.2 **(b)** **Resolved** **0** / 3 pts

✔ **+ 0 pts** Missing/ incorrect

Regrade Request **Submitted on: Apr 15**

My fuzzy logic is correct. How have I lost points? Will you regrade it and let me know?

check for x=30, 65

Reviewed on: Apr 18

4.3 **(c)** **Resolved** **0** / 3 pts

✔ **+ 0 pts** Missing/incorrect

Regrade Request **Submitted on: Apr 15**

My fuzzy logic is correct. How have I lost points? Will you regrade it and let me know?

if you put x=20 in your function you get 0. also for 45 you get 1

Reviewed on: Apr 18

4.4 **(d)** **Resolved** **0** / 3 pts

✔ **+ 3 pts** Correct

✔ **– 1.5 pts** A few can be infinitely large or 1 is always considered "a few"

✔ **– 1 pt** Not fuzzy

✔ **– 1 pt** Domain is positive real numbers

Regrade Request **Submitted on: Apr 15**

How have I lost points even though my fuzzy logic is right? Also, my domain has positive real numbers. Will you regrade it and let me know?

check x = 1,5. Also your domain should be pos integers

Reviewed on: Apr 18

4.5 (e) Resolved 0 / 3 pts

✔ + 0 pts Missing/incorrect

Regrade Request Submitted on: Apr 15

How have I lost points when I have to compare these functions and provide reasonable explanations only? Will you regrade it and let me know?

your answer is not correct.
"Correct answer: Note “old” and “not old” are complements, but “not old” and “young” need not be the same."
they're not "exactly opposite".

Reviewed on: Apr 16

Question 5

8.6 Approximate Inference 1

20 / 20 pts

✔ + 20 pts Correct

Question 6

8.7 Approximate Inference 2

20 / 20 pts

✔ + 20 pts Correct

Question assigned to the following page: 6

6.

Sources- This code is modified, used, and borrowed from ApproximateInference.ipynb's Python code and R&N Textbook's Likelihood Weighting Pseudocode.

```
from random import random

TRIALS = 10**7
probs = {'B': [0.001], 'E': [0.002], 'A': [[0.001, 0.29], [0.95, 0.98]], 'J':
[0.01, 0.95], 'M': [0.01, 0.70]}

def likelihood_weighting(B, e, Q, n):
    counts = [0,0]
    for trial in range(1,n+1):
        sample = {}
        weight = 1
        for x in B:
            if x in e:
                v = e[x]
                sample[x] = v
                if x == 'B' or x == 'E':
                    if sample[x]:
                        weight *= (B[x][0])
                    else:
                        weight *= (1 - B[x][0])
                elif x == 'A':
                    if sample[x]:
                        weight *= (B[x][sample['B']][sample['E']])
                    else:
                        weight *= (1 - B[x][sample['B']][sample['E']])
                elif x == 'J' or x == 'M':
                    if sample[x]:
                        weight *= (B[x][sample['A']])
                    else:
                        weight *= (1 - B[x][sample['A']])
```

Question assigned to the following page: 6

```
            else:
                if x == 'A':
                    sample[x] = (random() < B[x][sample['B']][sample['E']])
                elif x == 'J' or x == 'M':
                    sample[x] = (random() < B[x][sample['A']])
                elif x == 'B' or x == 'E':
                    sample[x] = (random() < B[x][0])
        v = sample[Q]
        counts[v] += weight
    return [round(x / sum(counts), 3) for x in counts]

p_b = probs['B'][0]
p_e = probs['E'][0]
e = {'M': False, 'B': True, 'E': True}
P_j = likelihood_weighting(probs, e, 'J', TRIALS)
p_j = P_j[0]
e = {'B': True, 'E': True}
P_m = likelihood_weighting(probs, e, 'M', TRIALS)
p_m = P_m[0]

print('P(-J, -M, B, E) =', p_j*p_m*p_b*p_e)
```

Question assigned to the following page: 6

Result-

```
P(-J, -M, B, E) = 6.8452e-08
```

I have used the program from ApproximateInference.ipynb in a separate file to modify it and included the probabilities of the third question since I need to verify the answers and make sure that the answers of these questions will provide the same results. I have used Rejection Sampling since the evidence is true. Then, I have compiled the modified program that is giving me the same results. If I run the Python code to generate results, then the results will be random due to the random library. But the results will find the approximate joint probabilities. Therefore, the equal results are shown successfully.

Question assigned to the following page: 1

1. $P(R|B_1) = \frac{2}{2+3+6} = \frac{2}{11}$

$P(R|B_2) = \frac{4}{4+4+3} = \frac{4}{9}$

$P(R|B_3) = \frac{3}{3+6+3} = \frac{3}{10}$

$P(B_1) = P(B_2) = P(B_3) = \frac{1}{3}$

Use the Bayes' Rule to calculate the probabilities of boxes that are randomly selected when the selected ball is red.

$$P(B_2|R) = \frac{\frac{1}{3}(\frac{2}{11})}{\frac{1}{3}(\frac{2}{11} + \frac{4}{9} + \frac{3}{10})} = \frac{180}{917} = 0.196$$

$$P(B_2|R) = \frac{\frac{1}{3}(\frac{4}{9})}{\frac{1}{3}(\frac{2}{11} + \frac{4}{9} + \frac{3}{10})} = \frac{440}{917} = 0.480$$

$$P(B_3|R) = \frac{\frac{1}{3}(\frac{3}{10})}{\frac{1}{3}(\frac{2}{11} + \frac{4}{9} + \frac{3}{10})} = \frac{297}{917} = 0.324$$

Question assigned to the following page: 4.1

4a. $M(x) = 0$ if $x < 30$.

$M(x) = \frac{100 - x}{35}$ if $30 \leq x \leq 65$.

$M(x) = 1$ if $x > 65$.

Question assigned to the following page: 4.2

b. $M(x) = 0$ if $x > 65$.

$M(x) = \frac{x - 30}{35}$ if $30 \leq x \leq 65$.

$M(x) = 2$ if $x < 30$.

Question assigned to the following page: 2

2. $P(A|D) = \frac{P(A)\sum_B \sum_C P(B|A)P(C|A)P(D|B,C)}{P(D)}$

$P(A,B,C) = P(A)(P(B|A)P(C|A)P(D|B,C) + P(\neg B|A)P(C|A)P(D|\neg B,C) + P(B|A)P(\neg C|A)P(D|B,\neg C) + P(\neg B|A)P(\neg C|A)P(D|\neg B,\neg C)) / P(D)$

$= \frac{\frac{1}{2}(1(1)(1) + 0(1)(\frac{1}{2}) + 1(0)(\frac{1}{2}) + 0(0)(0))}{P(D)}$

$= \alpha(\frac{1}{2})$

$P(\neg A|D) = P(\neg A)(P(B|\neg A)P(C|\neg A)P(D|B,C) + P(\neg B|\neg A)P(C|\neg A)P(D|\neg B,C) + P(B|\neg A)P(\neg C|\neg A)P(D|B,\neg C) + P(\neg B|\neg A)P(\neg C|\neg A)P(D|\neg B,\neg C)) / P(D)$

$= \frac{\frac{1}{2}(\frac{1}{2}(\frac{1}{2})(1) + \frac{1}{2}(\frac{1}{2})(\frac{1}{2}) + \frac{1}{2}(\frac{1}{2})(\frac{1}{2}) + \frac{1}{2}(\frac{1}{2})(0))}{P(D)}$

$= \alpha(\frac{1}{4})$

$\underline{P}(A|D) = \langle \frac{\frac{1}{2}}{\frac{1}{2}+\frac{1}{4}}, \frac{\frac{1}{4}}{\frac{1}{2}+\frac{1}{4}} \rangle = \langle \frac{2}{3}, \frac{1}{3} \rangle = \langle 0.667, 0.333 \rangle$

Question assigned to the following page: 4.3

c. $M(x) = 0$ if $x > 45$.

$M(x) = \frac{x-20}{25}$ if $20 \leq x \leq 45$.

$M(x) = 1$ if $x < 20$.

Question assigned to the following page: 3

3. $P(\neg J, \neg M, B, E) = P(B)P(E)\sum_A P(A|B,E)P(\neg J|A)P(\neg M|A)$

$= P(B)P(E)(P(A|B,E)P(\neg J|A)P(\neg M|A) +$
$P(\neg A|B,E)P(\neg J|\neg A)P(\neg M|\neg A))$

$= 0.001(0.002)(0.98(0.05)(0.30) + 0.02(0.99)$
$(0.99))$

$= 0.000000068604$

Question assigned to the following page: 4.4

d. $M(x) = 0$ if $x \geq 5$.

$M(x) = \frac{x-1}{4}$ if $1 \leq x \leq 5$.

$M(x) = 1$ if $x < 1$.

Question assigned to the following page: 4.5

e. "Not Old" and "Young" don't have to be same because the person, who is not old, overlaps with person's age range. But they are not same since they are exactly opposite and have different age ranges. Their functions are different, so they are distinct fuzzy sets.

Question assigned to the following page: 5

5.

Source- This code is modified, used, and borrowed from ApproximateInference.ipynb's Python code.

```
from random import random
import matplotlib.pyplot as plt

TRIALS = 10**6
evidence = 0
query = 0
pall = []
pdenomall = []

for trial in range(1,TRIALS+1):
    randA, randB, randC, randD = ( random() for i in range(4) )
    A = (randA < 0.5)
    if A:
      B = (randB < 1.0)
      C = (randC < 1.0)
    else:
      B = (randB < 0.5)
      C = (randC < 0.5)
    if B and C:
        D = (randD < 1.0)
    elif B and not C:
        D = (randD < 0.5)
    elif not B and C:
        D = (randD < 0.5)
    else:
        D = (randD < 0.0)
```

Question assigned to the following page: 5

```
    # REJECTION SAMPLING: ONLY PROCESS IF EVIDENCE True
    if D:
        evidence += 1
        if A:
            query += 1
        p = query/evidence
        pall.append(p)
    pdenom = evidence/trial
    pdenomall.append(pdenom)

print("P(A|D) ~=", pall[-1])
print("P(A) ~=", pdenomall[-1])
print("P(A,D) ~=", pall[-1]*pdenomall[-1])
plt.plot(pall, label = 'Pall')
plt.plot(pdenomall, label = 'Pdenomall')
plt.legend()
plt.show()
```

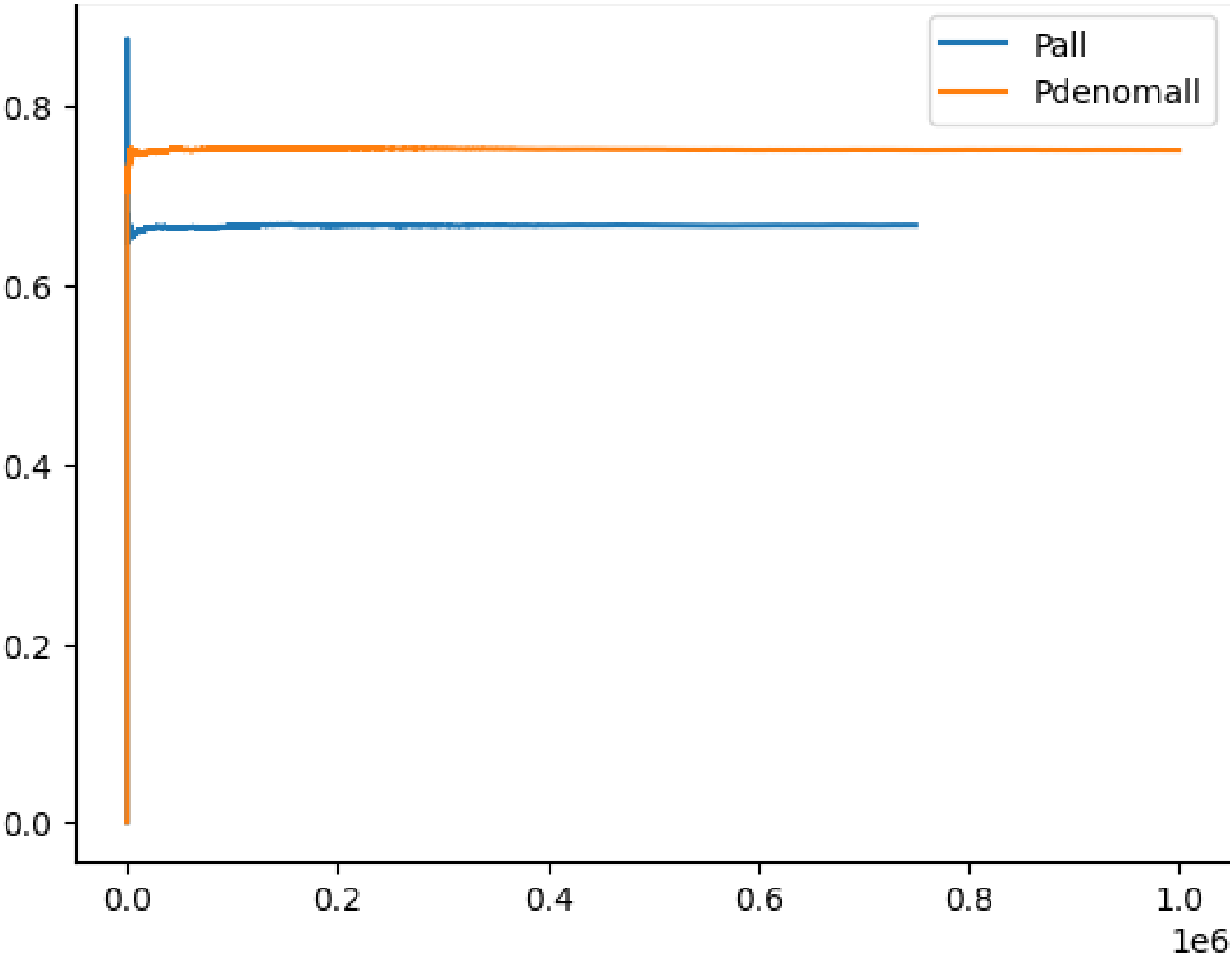
Question assigned to the following page: 5

Result-

```
P(A|D) ~= 0.6670436712520016
P(D) ~= 0.750654
P(A,D) ~= 0.500719
```

I have used the program from ApproximateInference.ipynb in a separate file to modify it and included the probabilities of the second question since I need to make sure that the answers of these questions will provide the same results. I have used Rejection Sampling since the evidence is true. Then, I have compiled the modified program that is giving me the same results. Also, I have generated a graph with labels to make sure that the results are same. I am getting the same results on the graph too. If I run the Python code to generate results, then the results will be random due to the random library. But the results will find the approximate probabilities. Therefore, the equal results are shown successfully.

Question assigned to the following page: 5


Pall
Pdenomall
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1e6