Consider the following information about houses lining a street. Bob lives to the left of Alice. Carol lives to the left of Bob. David lives to the left of Bob. (Assume each person lives in a distinct house, and they all live on the same side of the street.) Furthermore, assume that the 'leftOf' relationship has the properties of a total order (antireflexive, symmetric, transitive). Write out axioms for 'leftOf' in FOL. Let the set of facts and axioms be S.

(To keep things simple, you can use terms like ‘A’ to refer to Alice’s house, ‘B’ for Bob’s house, etc. That way, you don’t have to have separate objects for people and houses.)

1a. Draw a model depicting the scene described. Call it M1, such that sat(M1,S) or

M1
models(S).

1b. Write M1 out formally in terms of <U,D,R>

1c. Draw a
different
model of S, M2.

1d. Write out M2 as <U,D.R>

1e. Draw another model M3 in which there are at least 5 people living on the street.

1f. Write M3 as <U,D,R>

1g. Suppose we added another axiom defining ‘rightOf’ as the
opposite
of ‘leftOf’, to

create the theory (set of sentences) S’. Write the new axiom for ‘rightOf’.

- Does S’ |= rightOf(A,D)?

-

Does S’ |= rightOf(C,D)?

•

Does S’ |=
x rightOf(x,A)?

Use model theory arguments (reasoning about models) to explain your answers.

1a.

C — D — B — A

1b. M1 written formally

M1 = ⟨U, D, R⟩

- U = {A, B, C, D}
- Denotation D: D(A) = A, D(B) = B, D(C) = C, D(D) = D
- Relation R(leftOf):

  C leftOf D

  C leftOf B

  C leftOf A

  D leftOf B

  D leftOf A

  B leftOf A

  So R(leftOf) = {(C,D), (C,B), (C,A), (D,B), (D,A), (B,A)}

1c.

D — C — B — A

1d.

U = {A, B, C, D}

Denotation D: D(A) = A, D(B) = B, D(C) = C, D(D) = D

leftOf = {(D, C), (D, B), (D, A), (C, B), (C, A), (B, A)}

1e.

Add person E

E — D — C — B — A

1f. M3 formally

U = {A, B, C, D, E}

Denotation D: D(A) = A, D(B) = B, D(C) = C, D(D) = D, D(E) = E

leftOf = {(E,D), (E, C), (E, B), (E, A), (D, C), (D, B), (D, A), (C, B), (C, A), (B, A)}

1g.

New axiom:

$\forall x \forall y$ [ rightOf(x,y) $\leftrightarrow$ leftOf(y,x) ]

Let S′ = S $\cup$ {this axiom}.

Does S′ $\vDash$ rightOf(A, D)?
This asks for every model from M1, M2, M3 is leftOf(D,A) true and it is true

rightOf(A,D) $\leftrightarrow$ leftOf(D,A).

Could there be a model this is False the answer is no as we know D is left of B and B is left of A. so leftOf(D,B) and leftOf(B,A) imply leftOf(D,A)

Answer: Yes

Does S′ $\vDash$ rightOf(C, D)?

rightOf(C,D) ↔ leftOf(D,C).

Asking is C always right of D meaning is D always left of C

M1: C left of D → leftOf(D,C) false

M2: D left of C → leftOf(D,C) true

Since the truth is different across models it is not entailed.

Answer: No.

Does $S' \models \exists x$ rightOf(x, A)?

rightOf(x,A) ↔ leftOf(A,x).

A is the rightmost in all valid models

Thus leftOf(A,x) is false for all x.

Therefore $\exists x$ rightOf(x,A) is false in every model means its negation is entailed.

Answer: $S' \models \neg\exists x$ rightOf(x,A).

Consider two factors that influence whether a student passes a given test: a) being smart, and b) studying. Suppose 30% of students believe they are intrinsically smart. But since students do not know a priori whether they are smart enough to pass a test, suppose 40% of will study for it anyway. (assume Smart and Study are independent). The causal relationship of these variables on the probability of actually passing the test can be expressed in a conditional probability table (CPT) as follows:

| P(pass\|Smart,Study) | ¬smart | smart |
|---|---|---|
| ¬study | 0.2 | 0.7 |
| study | 0.6 | 0.95 |

prior probabilities: P(smart)=0.3, P(study)=0.4

2. Bayesian Inference

P(Smart)=0.3

P(Study)=0.4

a) Write out the equation for calculating joint probabilities, P(Smart,Study,Pass).

P(S, St, Pass) = P(S) × P(St) × P(Pass | S, St)

Similarly for negated combinations.

P(S, St, ¬Pass) = P(S) × P(St) × (1 - P(Pass | S, St))

b) Calculate all the entries in the full joint probability table (JPT) [a 4x2 matrix, like Fig 12.3 in the textbook; [Note: names of variables are capitalized, lower-case indicates truth value, e.g. ‘pass’ means Pass=T, and ‘-pass’ means Pass=F.]

2b.

Compute P(S)*P(St)*CPT.

Pass rows

- smart, study, pass n not pass

  P = 0.3 × 0.4 × 0.95 = 0.114

P(¬pass) = 0.3 x 0.4 x 0.05 = 0.006

- ¬smart, study, pass n not pass

  P = 0.7 × 0.4 × 0.60 = 0.168

  P(¬pass) = 0.7 × 0.4 × 0.4 = 0.112

- ¬smart, ¬study, pass n not pass

  P = 0.7 × 0.6 × 0.2 = 0.084

  P(¬pass) = 0.7 x 0.6 x 0.8 = 0.336

- smart, ¬study, pass n not pass

  P = 0.3 × 0.6 × 0.70 = 0.126

  P(¬pass) = 0.3 × 0.6 × 0.3 = 0.054

| Smart | Study | Pass | Probability |
|---|---|---|---|
| F | F | T | 0.084 |
| F | F | F | 0.336 |
| T | F | T | 0.126 |
| T | F | F | 0.054 |
| F | T | T | 0.168 |
| F | T | F | 0.112 |
| T | T | T | 0.114 |
| T | T | F | 0.006 |

c) From the JPT, compute the probability that a student is smart, given that they pass the test but did not study.

2c.

P(smart | pass ^ ¬study) = P(smart,¬study, pass) / P(pass ^ ¬study) = 0.126/(0.084+0.126) = 0.60

d) From the JPT, compute the probability that a student did not study, given that they are smart but did not pass the test.

P( ¬study | smart ^ ¬pass) = P(smart,¬study,¬pass) / P(smart ^ ¬pass) = 0.054/(0.054 + 0.006) = 0.90

e) Compute the marginal probability that a student will pass the test given that they are smart.

P(pass | smart) = P(smart, pass) / P(smart) = (0.126+0.114) / 0.3 = 0.8

f) Compute the marginal probability that a student will pass the test given that they study

P(pass | study) = P(study, pass) / P(study) = (0.168 + 0.114) / 0.4 = 0.705

## 3. Bayesian Networks.

Here is a probabilistic model that describes what it might mean when a person sneezes, e.g. depending on whether they have a cold, or whether a cat is present and they are allergic. Scratches on the furniture would be evidence that a cat had been present.

a) Using Equation 13.2 in the textbook (p. 415), write out the expression for the joint probability for any state (i.e. combination of truth values for the 5 variables in this

problem). [Note: Use capital letters for names of variables, and lower-case to indicate truth value, e.g. 'cold' means Cold=T, and '-cold' means Cold=F.]
P(Cold,Sneeze,Allergic,Scratches,Cat) = ?

P(Cold,Cat,Allergic,Sneeze,Scratches) = P(Cold) P(Cat) P(Allergic | Cat) P(Sneeze | Cold, Allergic) P(Scratches | Cat)

b) Use the equation above to calculate the
joint probability
that the person sneezes, but
does not have a cold, has a cat, is allergic, and there are scratches on the furniture:

P(-cold,sneeze,allergic,scratches,cat) = P(-cold) P(cat) P(allergic | cat) P(sneeze | -cold, allergic) P(scratches | cat) = 0.95 * 0.02 * 0.75 * 0.7 * 0.5 = 0.0049875

c) Use
normalization
to calculate the conditional probability that a person has cat, given
that they sneeze and are allergic to cats, but do not have a cold, and there are scratches on the furniture.

Using bayes normalization

P(cat|-cold,sneeze,allergic,scratches) = P(cat, -cold,sneeze,allergic,scratches) / (P(cat, -cold,sneeze,allergic,scratches) + P(-cat, -cold,sneeze,allergic,scratches))
= 0.0049875/ (0.0049875 + (0.95 * 0.98 * 0.05 * 0.7 * 0.05)) = 0.7538

d) Use Bayes' Rule to re-write the expression for P(cat|scratches). Look up the values for the numerator in the table above.

Bayes rule
P(cat|scratches) = P(scratches|cat) P(cat) / P(scratches)
= 0.5 * 0.02 / P(scratches)

= 0.01 / P(scratches)

e) The denominator in the answer for (d) would require
marginalization
over how many

joint probabilities? Write out the expressions for these (i.e. expand the denominator, but you don't have to calculate the actual values)

Cold: 2 values
Allergy: 2 values
Sneeze: 2 values
Cat: 2 values

s^4 = 16 joint terms

P(cold, allergy, sneeze, scratches, cat)
P(cold, allergy, ¬sneeze, scratches, cat)
P(cold, ¬allergy, sneeze, scratches, cat)
P(cold, ¬allergy, ¬sneeze, scratches, cat)
P(¬cold, allergy, sneeze, scratches, cat)
P(¬cold, allergy, ¬sneeze, scratches, cat)
P(¬cold, ¬allergy, sneeze, scratches, cat)
P(¬cold, ¬allergy, ¬sneeze, scratches, cat)
P(cold, allergy, sneeze, scratches, ¬cat)
P(cold, allergy, ¬sneeze, scratches, ¬cat)
P(cold, ¬allergy, sneeze, scratches, ¬cat)
P(cold, ¬allergy, ¬sneeze, scratches, ¬cat)
P(¬cold, allergy, sneeze, scratches, ¬cat)
P(¬cold, allergy, ¬sneeze, scratches, ¬cat)
P(¬cold, ¬allergy, sneeze, scratches, ¬cat)
P(¬cold, ¬allergy, ¬sneeze, scratches, ¬cat)

3. PDDL and Situation Calculus

To start a car, you have to be at the car and have the key, and the car has to have a charged battery and the tank has to have gas. Afterwards, the car will be running, and you will still be at the car and have the key after starting the engine.

a. Write a PDDL operator to describe this action.

(note: you can express this ego-centrically – you don't have to refer explicitly to the person starting the car; but the operator should take the car being started as an argument)

StartCar(c)
Preconditions: AtCar(c) ^ HaveKey ^ BatteryCharged(c) ^ HasGas(c)
Effect: Running(c) ^ AtCar(c) ^ HaveKey

b. Describe the same operator using Situation Calculus (remember to add a situation argument to your predicates).

Preconditions: Poss(StartCar(c),s) ⇔ AtCar(c,s) ^ HaveKey(s) ^ BatteryCharged(c, s) ^ HasGas(c, s)
Successor state axiom:
Running(c, do(StartCar(c),s))
AtCar(c, do(StartCar(c),s)) <-> AtCar(c,s)
HaveKey(do(StartCar(c),s)) <-> HaveKey(s)

c. Add a Frame Axiom that says that starting this car will not changes whether any other car is out of gas (tank empty).

Given fluent: HasGas(d,s)
Action: StartCar(c)
Constraint: d ≠ c

(d ≠ c) -> (HasGas(d, do(StartCar(c),s)) <-> HasGas(d,s))