

15: General Inference

Jerry Cain

April 29, 2022

Table of Contents

2	General Inference: Intro
12	Bayesian Networks
19	Inference: Math
25	Rejection Sampling



General Inference: Introduction

Inference

*Web*MD[®]

Inference

WebMD Symptom Checker WITH BODY MAP

INFO SYMPTOMS QUESTIONS CONDITIONS DETAILS TREATMENT

What are your symptoms?

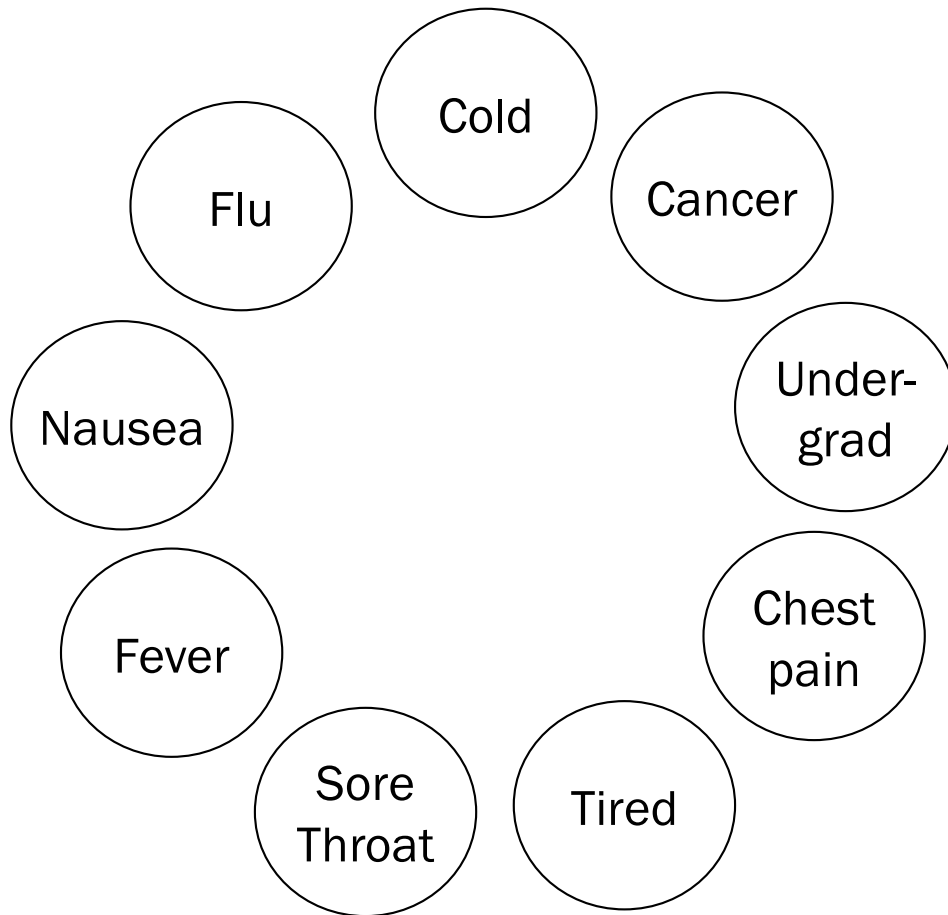
Type your main symptom here

My Symptoms

- nausea
- fever 100.5f to 102f
- severe headache
- shaking chills

Results Strength: MODERATE

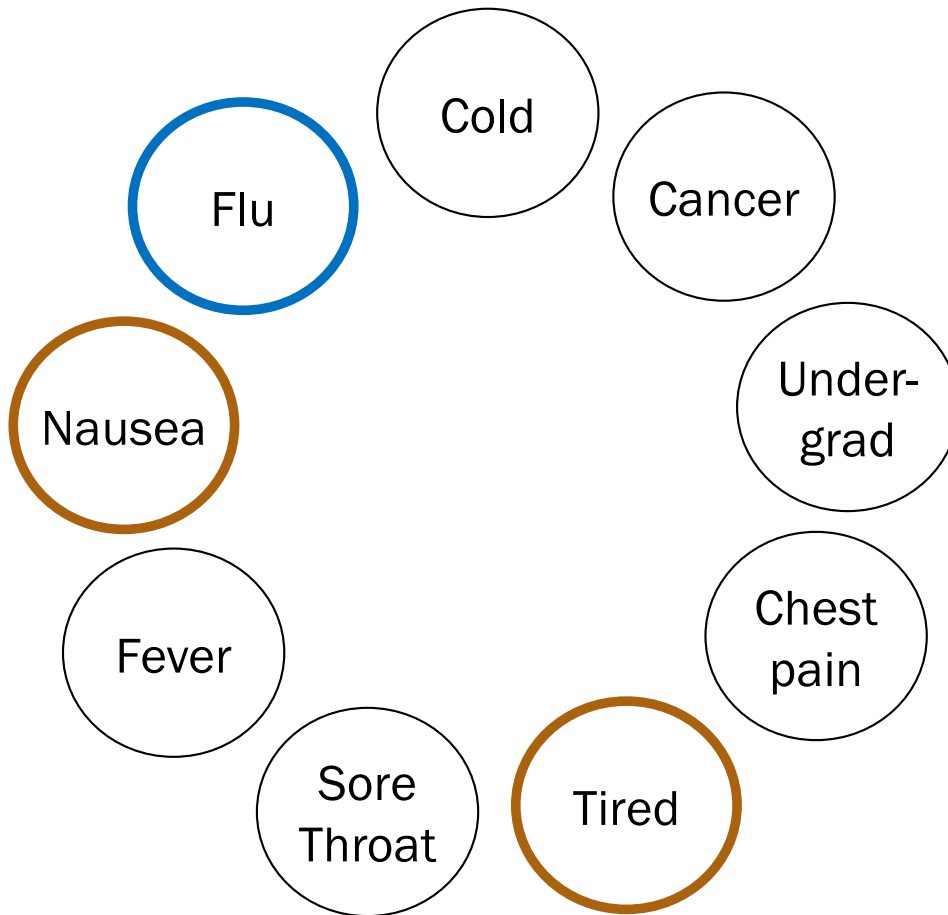
Inference



General inference question:

Given the values of some random variables, what is the conditional distribution of some other random variables?

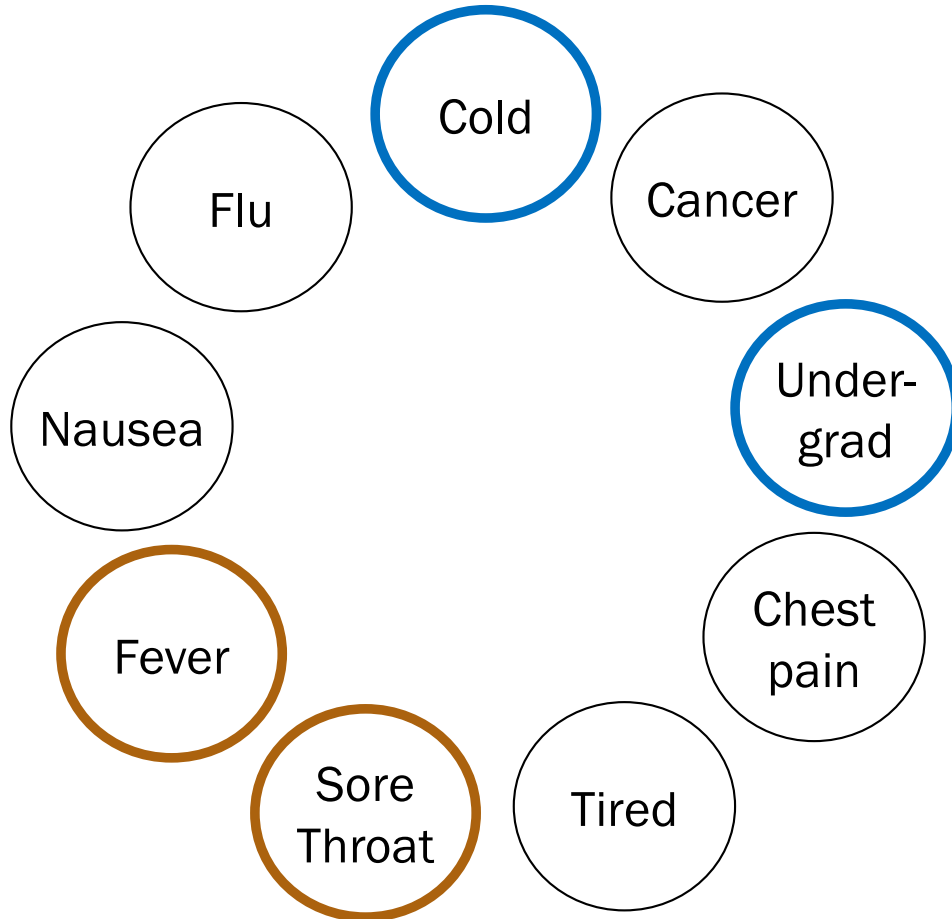
Inference



One inference question:

$$P(F = 1 | N = 1, T = 1) \\ = \frac{P(F = 1, N = 1, T = 1)}{P(N = 1, T = 1)}$$

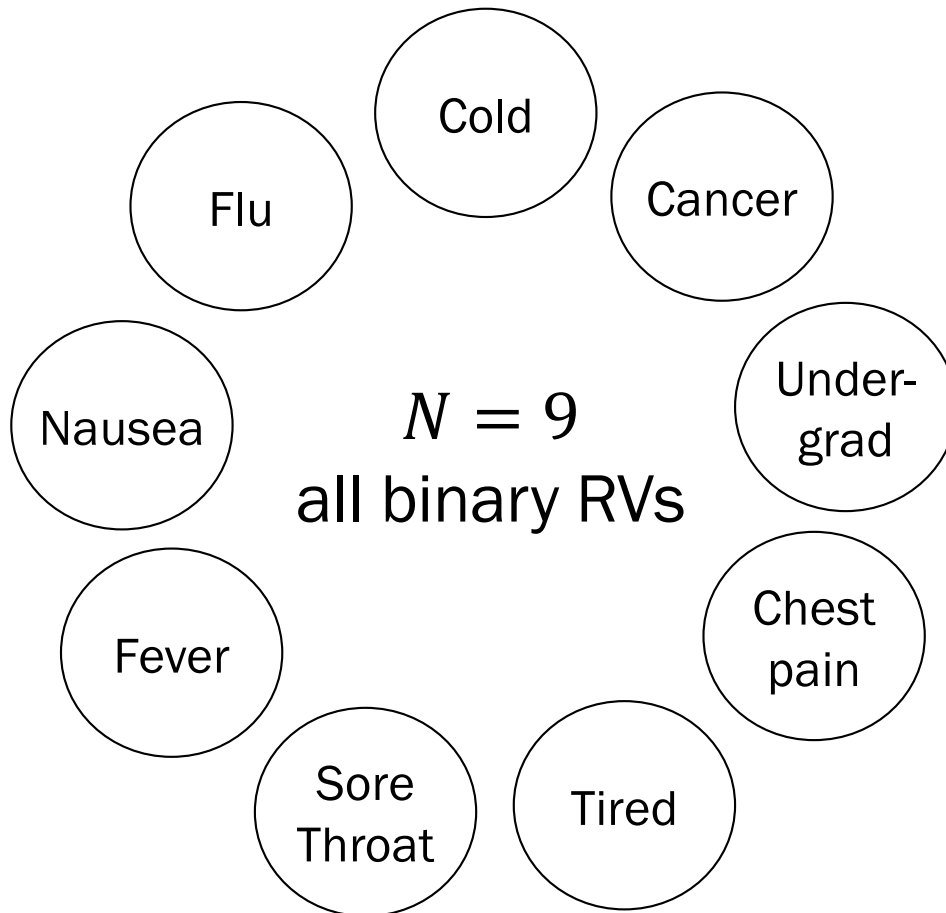
Inference



Another inference question:

$$P(C_o = 1, U = 1 | S = 0, F_e = 0) \\ = \frac{P(C_o = 1, U = 1, S = 0, F_e = 0)}{P(S = 0, F_e = 0)}$$

Inference



If we know the full **joint distribution**, we can answer all probabilistic inference questions.

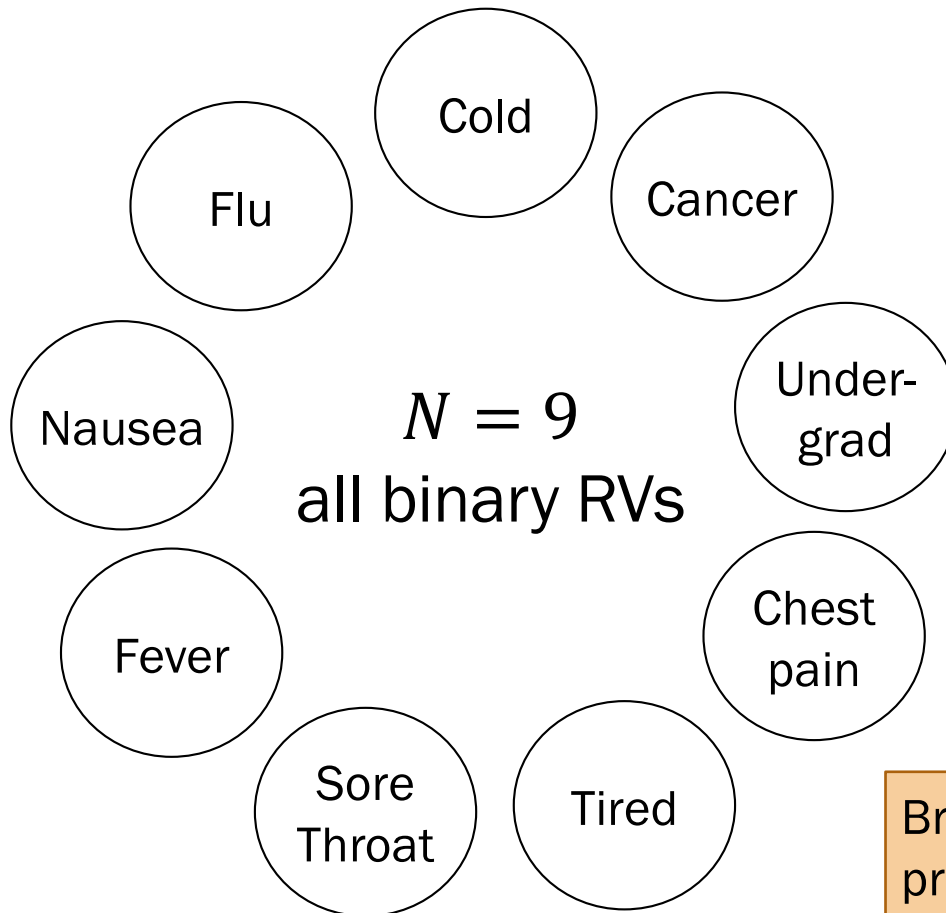
What is the size of the joint probability table?

- A. 2^{N-1} entries
- B. N^2 entries
- C. 2^N entries
- D. None/other/don't know



Inference

$$P(x_1 = -, x_2 = -, x_3 = -, \dots, x_n = -)$$



If we know the full **joint distribution**, we can answer all probabilistic inference questions.

What is the size of the joint probability table?

- A. 2^{N-1} entries
- B. N^2 entries
- C. 2^N entries
- D. None/other/don't know

Brute-force computation of a full joint probability mass function is often intractable.

Conditionally Independent RVs

E is the new sample space.

Recall that two events A and B are conditionally independent given E if:

$$P(AB|E) = P(A|E)P(B|E)$$

n discrete random variables X_1, X_2, \dots, X_n are called **conditionally independent given Y** if:

for all x_1, x_2, \dots, x_n, y :

$$P(X_1 = x_1, X_2 = x_2 | Y = y) = P(X_1 = x_1 | Y = y) \cdot P(X_2 = x_2 | Y = y)$$

$$P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n | Y = y) = \prod_{i=1}^n P(X_i = x_i | Y = y)$$

This implies the following (cool to remember for later):

$$\log P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n | Y = y) = \sum_{i=1}^n \log P(X_i = x_i | Y = y)$$



Bayesian Networks

*Baye
Nets*

A simpler WebMD

Flu

Under-grad

Fever

Tired

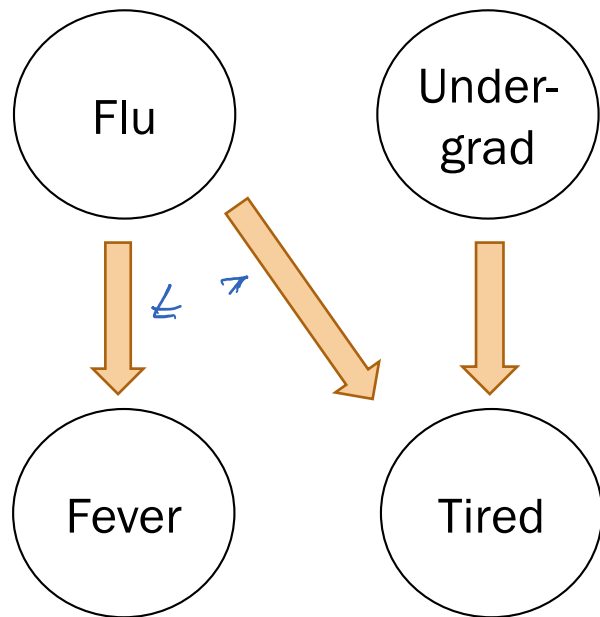
Great! Just specify $2^4 = 16$ joint probabilities?

$$P(F_{lu} = a, F_{ev} = b, U = c, T = d)$$

What would an infectious diseases (ID) expert do?

Describe the joint distribution using causality!

Constructing a Bayesian Network

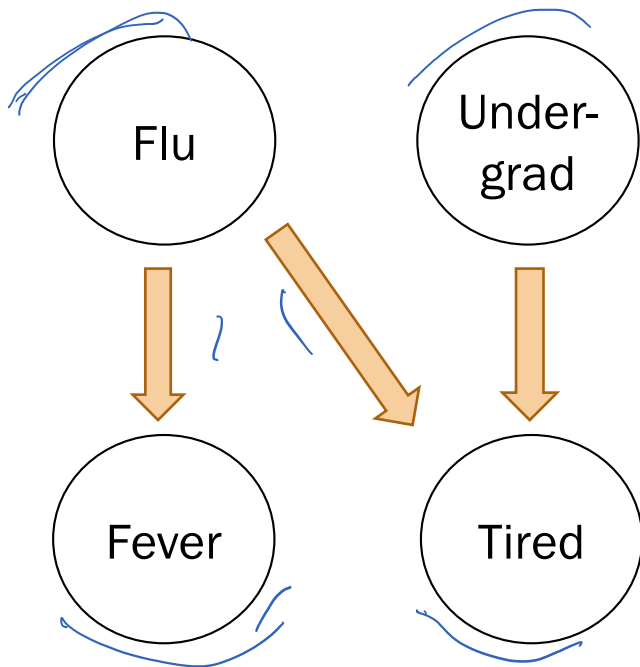


What would an ID expert do?

1. Describe the joint distribution using causality.

2. Assume conditional independence.

Constructing a Bayesian Network



In a Bayesian Network,
Each random variable is **conditionally independent** of its non-descendants, **given its parents**.

- Node: random variable
- Directed edge: conditional dependency

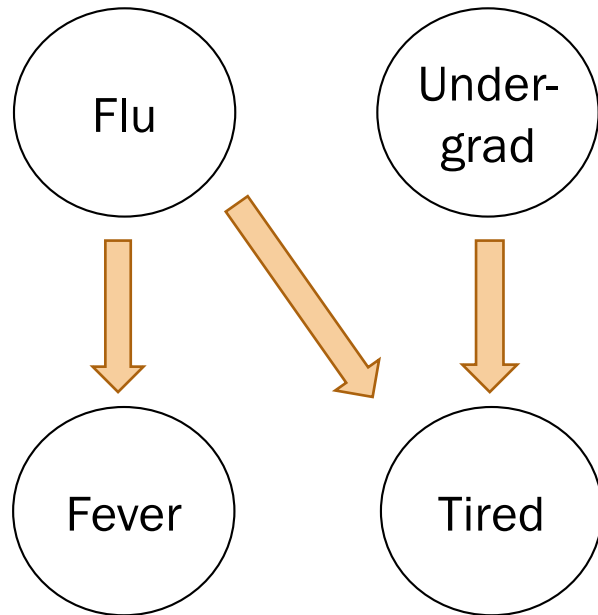
Examples:

- $P(F_{ev} = 1 | T = 0, F_{lu} = 1) = P(F_{ev} = 1 | F_{lu} = 1)$
- $P(F_{lu} = 1, U = 0) = P(F_{lu} = 1)P(U = 0)$

Constructing a Bayesian Network

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



$$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$$

$$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$$

What would an ID expert do?

1. Describe the joint distribution using causality.
- ✓ 2. Assume conditional independence.
3. Provide $P(\text{values}|\text{parents})$ for each random variable

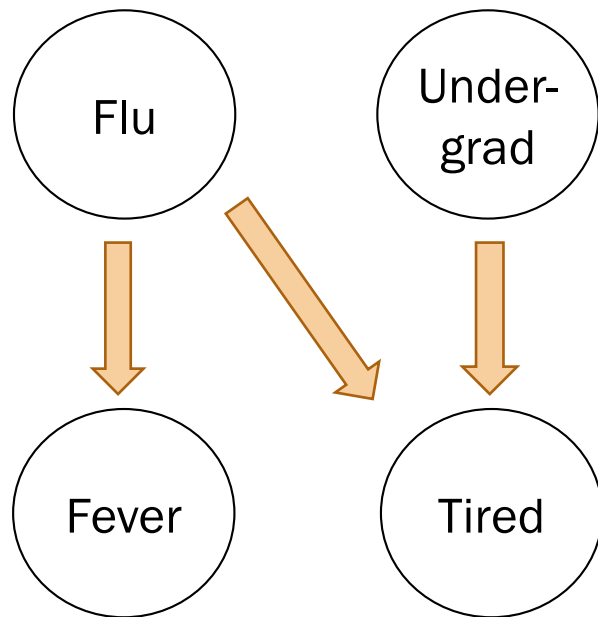
← What conditional probabilities should our expert specify?



Constructing a Bayesian Network

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



$$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$$
$$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$$

What would an ID expert do?

1. Describe the joint distribution using causality.
2. Assume conditional independence.
3. Provide $P(\text{values}|\text{parents})$ for each random variable

What conditional probabilities should our expert specify?

$$P(T = 1 | F_{lu} = 0, U = 0)$$

$$P(T = 1 | F_{lu} = 0, U = 1)$$

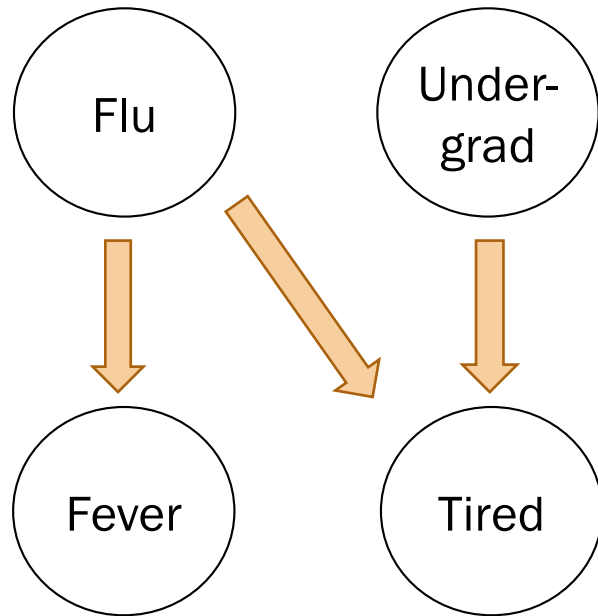
$$P(T = 1 | F_{lu} = 1, U = 0)$$

$$P(T = 1 | F_{lu} = 1, U = 1)$$

Using a Bayes Net

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



$$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$$

$$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$$

$$P(T = 1 | F_{lu} = 0, U = 0) = 0.1$$

$$P(T = 1 | F_{lu} = 0, U = 1) = 0.8$$

$$P(T = 1 | F_{lu} = 1, U = 0) = 0.9$$

$$P(T = 1 | F_{lu} = 1, U = 1) = 1.0$$

Lisa Yan, Chris Piech, Mehran Sahami, and Jerry Cain, CS109, Spring 2022

What would a CS109 student do?

1. Populate a Bayesian network by asking an infectious diseases expert or by using reasonable assumptions

2. Answer inference questions

Our focus today

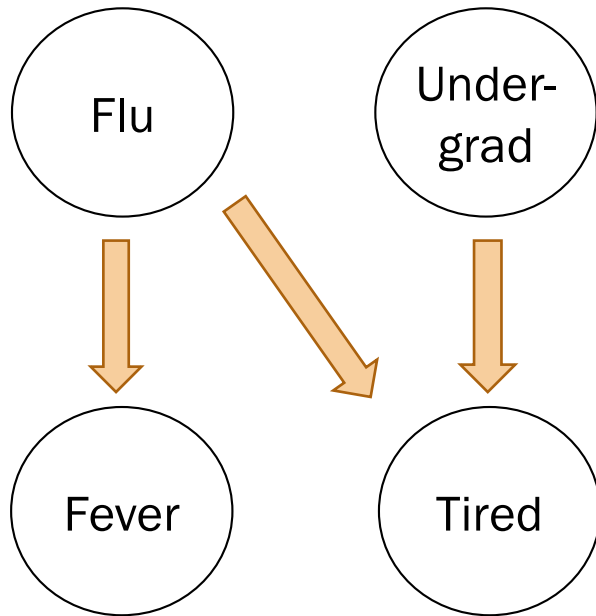


Inference: Math

Inference via math

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



$$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$$

$$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$$

$$P(T = 1 | F_{lu} = 0, U = 0) = 0.1$$

$$P(T = 1 | F_{lu} = 0, U = 1) = 0.8$$

$$P(T = 1 | F_{lu} = 1, U = 0) = 0.9$$

$$P(T = 1 | F_{lu} = 1, U = 1) = 1.0$$

Lisa Yan, Chris Piech, Mehran Sahami, and Jerry Cain, CS109, Spring 2022

1. $P(F_{lu} = 0, U = 1, F_{ev} = 0, T = 1)$?

Compute joint probabilities using chain rule.

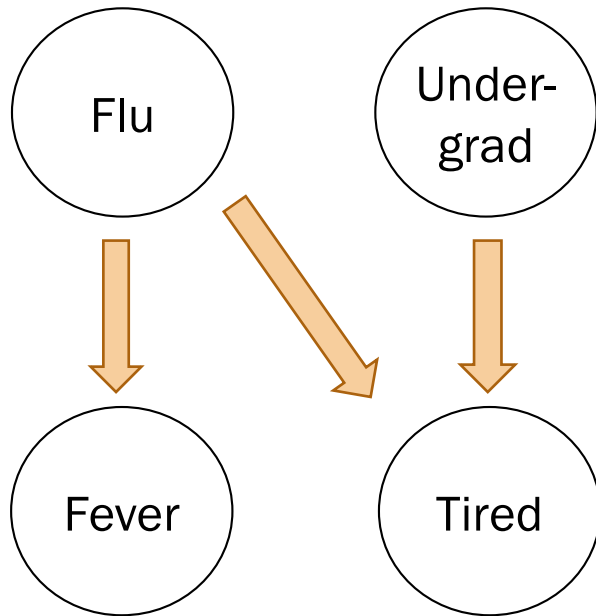
$$P(F_{lu}=0) P(U=1 | F_{ev}=0) P(F_{ev}=0 | F_{lu}=0, U=1) P(T=1 | F_{lu}=0, U=1, F_{ev}=0)$$

$$(0.9) \quad (0.8) \quad (0.95) \quad (0.8)$$

Inference via math

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



$$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$$

$$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$$

$$P(T = 1 | F_{lu} = 0, U = 0) = 0.1$$

$$P(T = 1 | F_{lu} = 0, U = 1) = 0.8$$

$$P(T = 1 | F_{lu} = 1, U = 0) = 0.9$$

$$P(T = 1 | F_{lu} = 1, U = 1) = 1.0$$

2. $P(F_{lu} = 1 | F_{ev} = 0, U = 0, T = 1)$?

1. Compute joint probabilities

$$\left[\begin{array}{l} \rightarrow P(F_{lu} = 1, F_{ev} = 0, U = 0, T = 1) \\ P(F_{lu} = 0, F_{ev} = 0, U = 0, T = 1) \end{array} \right]$$

2. Definition of conditional probability

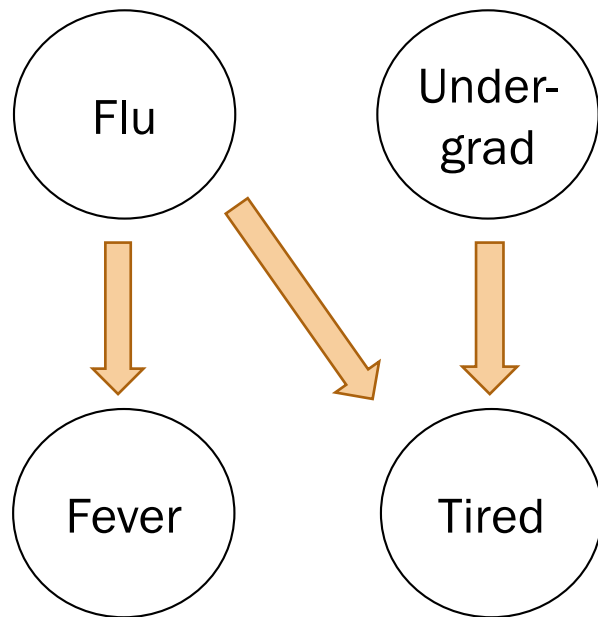
$$\frac{P(F_{lu} = 1, F_{ev} = 0, U = 0, T = 1)}{\sum_x P(F_{lu} = x, F_{ev} = 0, U = 0, T = 1)}$$

$$= 0.095$$

Inference via math

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



3. $P(F_{lu} = 1 | U = 1, T = 1)$?

$$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$$

$$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$$

$$P(T = 1 | F_{lu} = 0, U = 0) = 0.1$$

$$P(T = 1 | F_{lu} = 0, U = 1) = 0.8$$

$$P(T = 1 | F_{lu} = 1, U = 0) = 0.9$$

$$P(T = 1 | F_{lu} = 1, U = 1) = 1.0$$

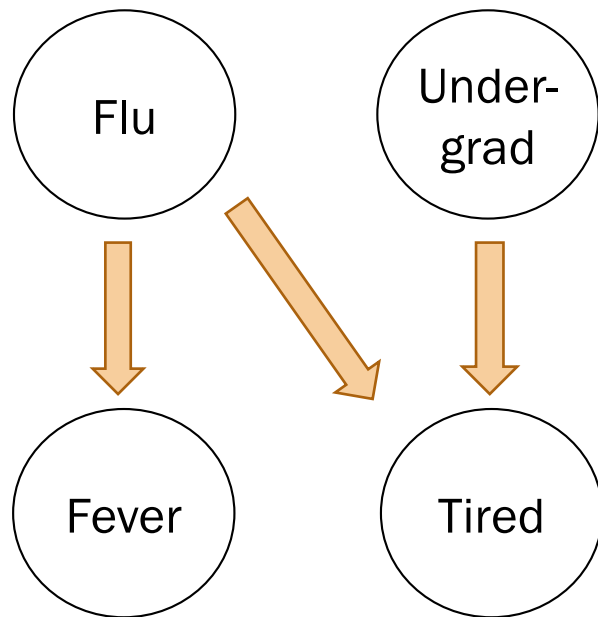
Lisa Yan, Chris Piech, Mehran Sahami, and Jerry Cain, CS109, Spring 2022



Inference via math

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



$$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$$

$$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$$

$$P(T = 1 | F_{lu} = 0, U = 0) = 0.1$$

$$P(T = 1 | F_{lu} = 0, U = 1) = 0.8$$

$$P(T = 1 | F_{lu} = 1, U = 0) = 0.9$$

$$P(T = 1 | F_{lu} = 1, U = 1) = 1.0$$

Lisa Yan, Chris Piech, Mehran Sahami, and Jerry Cain, CS109, Spring 2022

3. $P(F_{lu} = 1 | U = 1, T = 1)$?

1. Compute joint probabilities

4
$$\left[\begin{array}{l} P(F_{lu} = 1, \underline{U = 1}, \overbrace{F_{ev} = 1, T = 1}^{\text{...}}) \\ P(F_{lu} = 0, \underline{U = 1}, \overbrace{F_{ev} = 0, T = 1}^{\text{...}}) \end{array} \right]$$

2. Definition of conditional probability

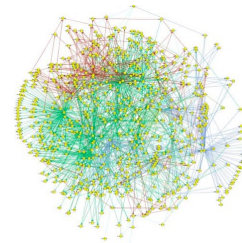
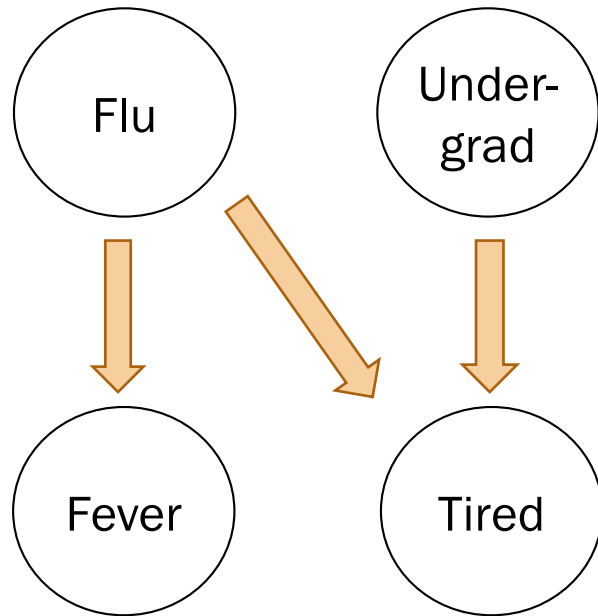
$$\frac{\sum_y P(F_{lu} = 1, U = 1, F_{ev} = y, T = 1)}{\sum_x \sum_y P(F_{lu} = x, U = 1, F_{ev} = y, T = 1)}$$

$$= 0.122$$

Inference via math

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



Solving inference questions precisely is possible, but sometimes tedious.

Can we use sampling to do approximate inference?

$$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$$
$$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$$

$$P(T = 1 | F_{lu} = 0, U = 0) = 0.1$$
$$P(T = 1 | F_{lu} = 0, U = 1) = 0.8$$
$$P(T = 1 | F_{lu} = 1, U = 0) = 0.9$$
$$P(T = 1 | F_{lu} = 1, U = 1) = 1.0$$





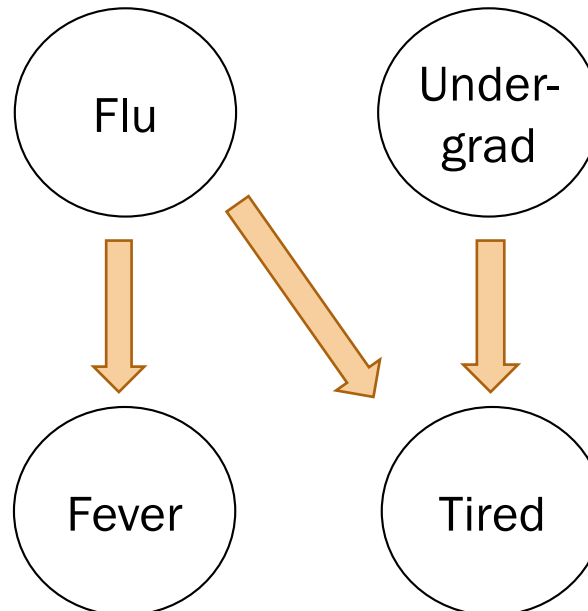
Rejection Sampling

Rejection sampling algorithm

Step 0:
Require a fully specified
Bayesian Network

$$P(F_{lu} = 1) = 0.1$$

$$P(U = 1) = 0.8$$



$$P(F_{ev} = 1|F_{lu} = 1) = 0.9$$

$$P(F_{ev} = 1|F_{lu} = 0) = 0.05$$

$$P(T = 1|F_{lu} = 0, U = 0) = 0.1$$

$$P(T = 1|F_{lu} = 0, U = 1) = 0.8$$

$$P(T = 1|F_{lu} = 1, U = 0) = 0.9$$

$$P(T = 1|F_{lu} = 1, U = 1) = 1.0$$

Rejection sampling algorithm

Inference question:

What is $P(F_{lu} = 1 | U = 1, T = 1)$?

$[0, 1, 0, 1]$

[flu, und, fev, tir]

```
def rejection_sampling(event, observation):  
    samples = sample_a_ton()  
    samples_observation = ...  
    # number of samples with (U = 1, T = 1)  
    samples_event = ...  
    # number of samples with (Flu = 1, U = 1, T = 1)  
    return len(samples_event) / len(samples_observation)
```

```
Sampling...  
[0, 1, 0, 1]  
[0, 1, 0, 1]  
[0, 1, 0, 1]  
[0, 0, 0, 0]  
[0, 1, 0, 1]  
[0, 1, 1, 1]  
[0, 1, 0, 0]  
[1, 1, 1, 1]  
[0, 0, 1, 1]  
...  
[0, 1, 0, 1]  
Finished sampling
```

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):  
    ( samples = sample_a_ton()  
      samples_observation = ...  
        # number of samples with (U = 1, T = 1)  
      samples_event =  
        # number of samples with (Flu = 1, U = 1, T = 1)  
    return len(samples_event)/len(samples_observation)
```

$$\text{Probability} \approx \frac{\# \text{ samples with } (F_{lu} = 1, U = 1, T = 1)}{\# \text{ samples with } (U = 1, T = 1)}$$

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

$$\text{Probability} \approx \frac{\# \text{ samples with } (F_{lu} = 1, U = 1, T = 1)}{\# \text{ samples with } (U = 1, T = 1)}$$

Why would this definition of approximate probability make sense?

Why would this approximate probability make sense?

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

Why would this definition of approximate probability make sense?

$$\text{Probability} \approx \frac{\# \text{ samples with } (F_{lu} = 1, U = 1, T = 1)}{\# \text{ samples with } (U = 1, T = 1)}$$

Recall our definition of probability as a frequency: $P(E) = \lim_{n \rightarrow \infty} \frac{n(E)}{n}$ $n = \#$ of total trials
 $n(E) = \#$ trials where E occurs

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):  
    samples = sample_a_ton()  
    samples_observation = ...  
    # number of samples with (U = 1, T = 1)  
    samples_event = ...  
    # number of samples with (Flu = 1, U = 1, T = 1)  
    return len(samples_event) / len(samples_observation)
```

[flu, und, fev, tir]

```
Sampling...  
[0, 1, 0, 1]  
[0, 1, 0, 1]  
[0, 1, 0, 1]  
[0, 0, 0, 0]  
[0, 1, 0, 1]  
[0, 1, 1, 1]  
[0, 1, 0, 0]  
[1, 1, 1, 1]  
[0, 0, 1, 1]  
...  
[0, 1, 0, 1]  
Finished sampling
```

Rejection sampling algorithm

```
N_SAMPLES = 100000
# Method: Sample a ton
# -----
# create N_SAMPLES with likelihood proportional
# to the joint distribution
def sample_a_ton():
    samples = []
    for i in range(N_SAMPLES):
        sample = make_sample() # a particle
        samples.append(sample)
    return samples
```

How do we make a sample
 $(F_{lu} = a, U = b, F_{ev} = c, T = d)$
according to the
joint probability?

Create a sample using the Bayesian Network!

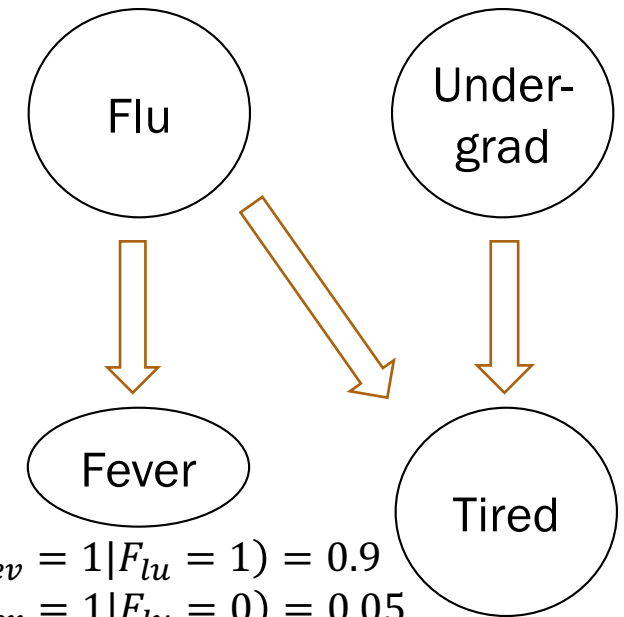
Rejection sampling algorithm

```
# Method: Make Sample
# -----
# create a single sample from the joint distribution
# based on the medical "WebMD" Bayesian Network
def make_sample():
    # prior on causal factors
    flu = bernoulli(0.1)
    und = bernoulli(0.8)

    # choose fever based on flu
    if flu == 1: fev = bernoulli(0.9)
    else: fev = bernoulli(0.05)

    # choose tired based on (undergrad and flu)
    #
    # TODO: fill in
    #
    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
```

$$P(F_{lu} = 1) = 0.1 \quad P(U = 1) = 0.8$$



$$P(F_{ev} = 1|F_{lu} = 1) = 0.9$$
$$P(F_{ev} = 1|F_{lu} = 0) = 0.05$$

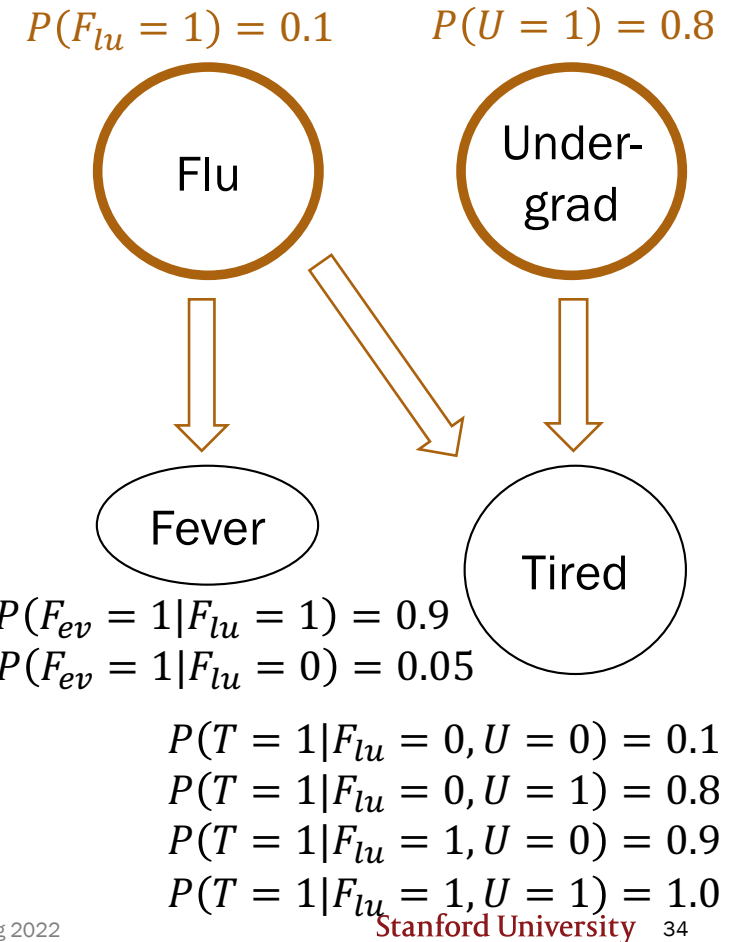
$$P(T = 1|F_{lu} = 0, U = 0) = 0.1$$
$$P(T = 1|F_{lu} = 0, U = 1) = 0.8$$
$$P(T = 1|F_{lu} = 1, U = 0) = 0.9$$
$$P(T = 1|F_{lu} = 1, U = 1) = 1.0$$

Rejection sampling algorithm

```
# Method: Make Sample
# -----
# create a single sample from the joint distribution
# based on the medical "WebMD" Bayesian Network
def make_sample():
    # prior on causal factors
    flu = bernoulli(0.1)
    und = bernoulli(0.8)

    # choose fever based on flu
    if flu == 1: fev = bernoulli(0.9)
    else: fev = bernoulli(0.05)

    # choose tired based on (undergrad and flu)
    #
    # TODO: fill in
    #
    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
```

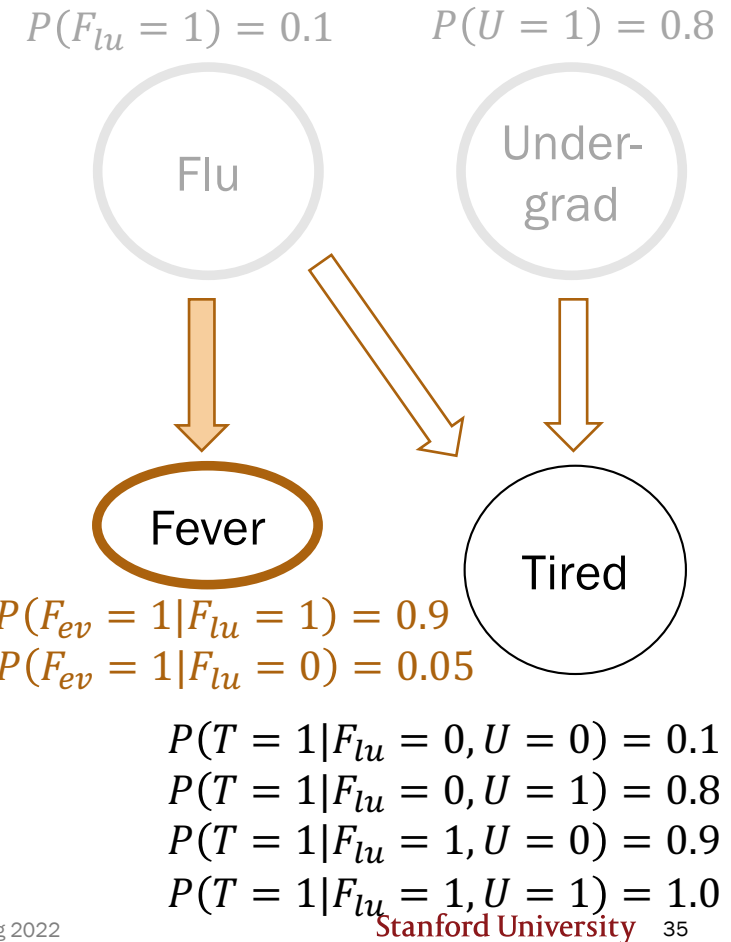


Rejection sampling algorithm

```
# Method: Make Sample
# -----
# create a single sample from the joint distribution
# based on the medical "WebMD" Bayesian Network
def make_sample():
    # prior on causal factors
    flu = bernoulli(0.1)
    und = bernoulli(0.8)

    # choose fever based on flu
    if flu == 1: fev = bernoulli(0.9)
    else: fev = bernoulli(0.05)

    # choose tired based on (undergrad and flu)
    #
    # TODO: fill in
    #
    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
```



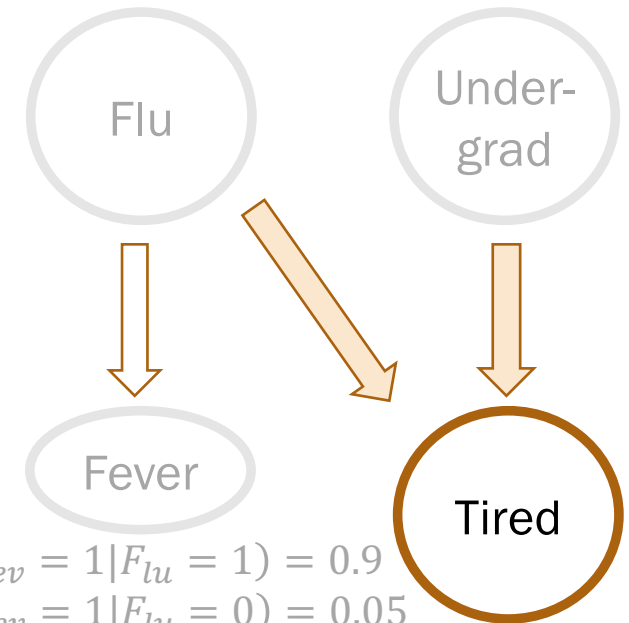
Rejection sampling algorithm

```
# Method: Make Sample
# -----
# create a single sample from the joint distribution
# based on the medical "WebMD" Bayesian Network
def make_sample():
    # prior on causal factors
    flu = bernoulli(0.1)
    und = bernoulli(0.8)

    # choose fever based on flu
    if flu == 1: fev = bernoulli(0.9)
    else: fev = bernoulli(0.05)

    # choose tired based on (undergrad and flu)
    #
    # TODO: fill in
    #
    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
```

$$P(F_{lu} = 1) = 0.1 \quad P(U = 1) = 0.8$$



$$P(F_{ev} = 1|F_{lu} = 1) = 0.9$$
$$P(F_{ev} = 1|F_{lu} = 0) = 0.05$$

$$P(T = 1|F_{lu} = 0, U = 0) = 0.1$$
$$P(T = 1|F_{lu} = 0, U = 1) = 0.8$$
$$P(T = 1|F_{lu} = 1, U = 0) = 0.9$$
$$P(T = 1|F_{lu} = 1, U = 1) = 1.0$$



Rejection sampling algorithm

```

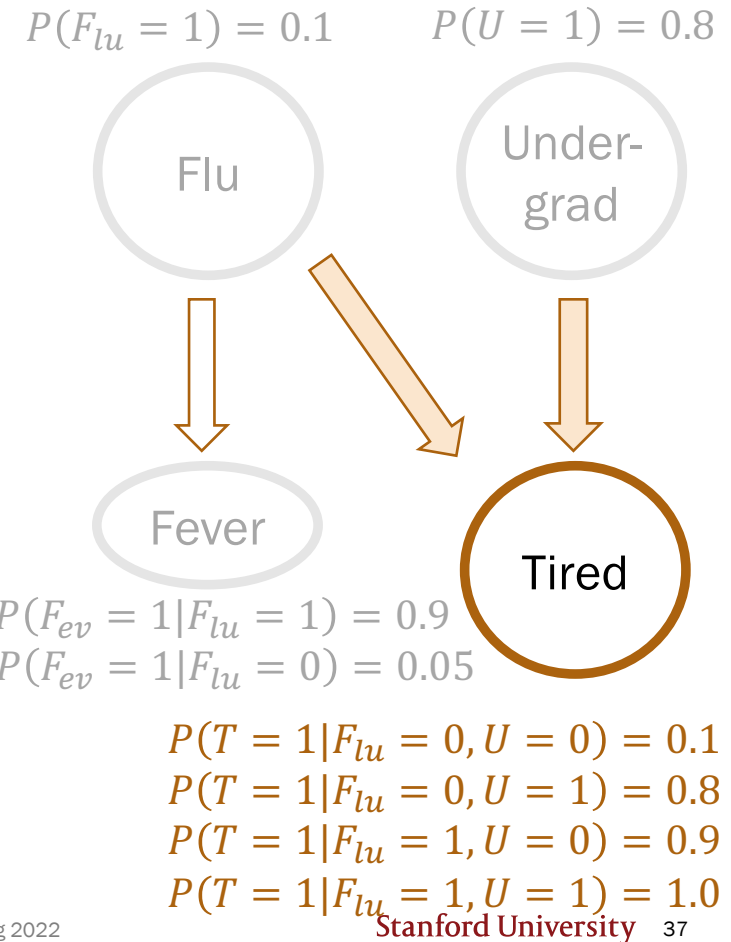
# Method: Make Sample
# -----
# create a single sample from the joint distribution
# based on the medical "WebMD" Bayesian Network
def make_sample():
    # prior on causal factors
    flu = bernoulli(0.1)
    und = bernoulli(0.8)

    # choose fever based on flu
    if flu == 1: fev = bernoulli(0.9)
    else:       fev = bernoulli(0.05)

    # choose tired based on (undergrad and flu)
    if flu == 0 and und == 0: tir = bernoulli(0.1)
    elif flu == 0 and und == 1: tir = bernoulli(0.8)
    elif flu == 1 and und == 0: tir = bernoulli(0.9)
    else:                       tir = bernoulli(1.0)

    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]

```



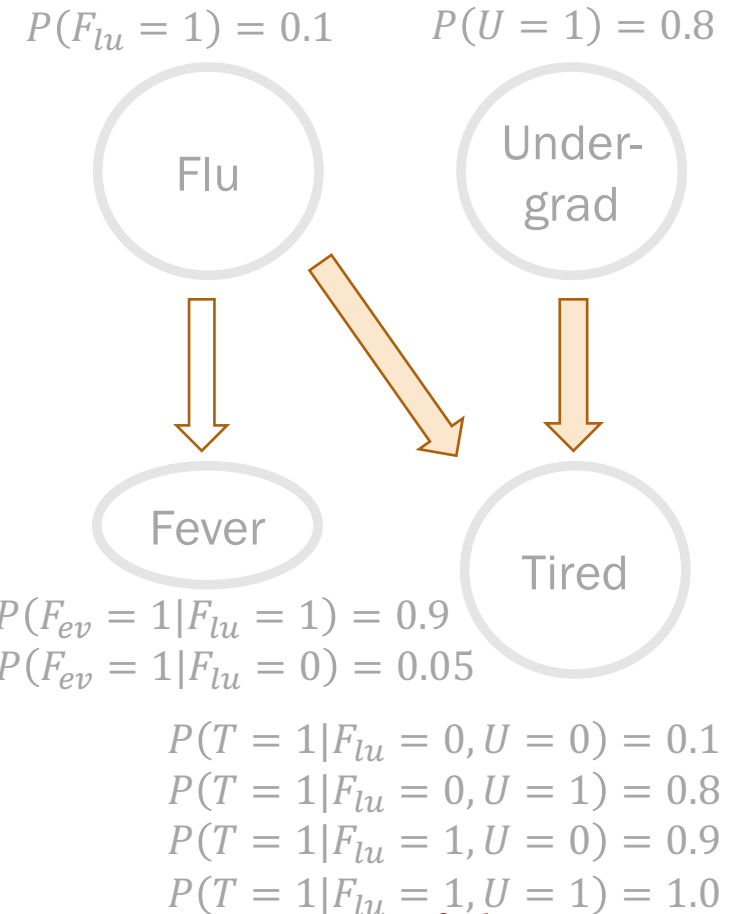
Rejection sampling algorithm

```
# Method: Make Sample
# -----
# create a single sample from the joint distribution
# based on the medical "WebMD" Bayesian Network
def make_sample():
    # prior on causal factors
    flu = bernoulli(0.1)
    und = bernoulli(0.8)

    # choose fever based on flu
    if flu == 1: fev = bernoulli(0.9)
    else:       fev = bernoulli(0.05)

    # choose tired based on (undergrad and flu)
    if flu == 0 and und == 0: tir = bernoulli(0.1)
    elif flu == 0 and und == 1: tir = bernoulli(0.8)
    elif flu == 1 and und == 0: tir = bernoulli(0.9)
    else:                       tir = bernoulli(1.0)

    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
```



Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):  
    samples = sample_a_ton() # 100,000 samples  
    samples_observation = ...  
        # number of samples with (U = 1, T = 1)  
    samples_event = ...  
        # number of samples with (Flu = 1, U = 1, T = 1)  
    return len(samples_event) / len(samples_observation)
```

[flu, und, fev, tir]

```
Sampling...  
[0, 1, 0, 1]  
[0, 1, 0, 1]  
[0, 1, 0, 1]  
[0, 0, 0, 0]  
[0, 1, 0, 1]  
[0, 1, 1, 1]  
[0, 1, 0, 0]  
[1, 1, 1, 1]  
[0, 0, 1, 1]  
...  
[0, 1, 0, 1]  
Finished sampling
```

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):  
    samples = sample_a_ton()  
    samples_observation = ...  
        # number of samples with (U = 1, T = 1)  
    samples_event =  
        # number of samples with (Flu = 1, U = 1, T = 1)  
    return len(samples_event) / len(samples_observation)
```

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):
    samples = sample_a_ton()
    samples_observation =
        reject_inconsistent(samples, observation)
    samples_event =
        # number of samples with  $(F_{lu} = 1, U = 1, T = 1)$ 
    return len(samples_event) / len(samples_observation)
```

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):  
    samples = sample_a_ton()  
    samples_observation =  
        reject_inconsistent(samples, observation)  
    samples_event =  
        # number of samples with  $(F_{lu} = 1, U = 1, T = 1)$   
    return len(samples_event) / len(samples_observation)
```

Keep only samples that are consistent
with the observation $(U = 1, T = 1)$.

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):
```

```
    samples = sample_a_ton()
```

```
    samples_observation =  
        reject_inconsistent(samples, observation)
```

```
    samples_# Method: Reject Inconsistent  
            # -----  
            # Rejects all samples that do not align with the outcome.  
            # Returns a list of consistent samples.
```

```
    return
```

```
        def reject_inconsistent(samples, outcome):
```

```
            consistent_samples = []
```

```
            for sample in samples:
```

```
                if check_consistent(sample, outcome):
```

```
                    consistent_samples.append(sample)
```

```
            return consistent_samples
```

$(None, (None, 1))$

$(U = 1, T = 1)$

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):  
    samples = sample_a_ton()  
    samples_observation =  
        reject_inconsistent(samples, observation)  
    samples_event =  
        reject_inconsistent(samples_observation, event)  
    return len(samples_event) / len(samples_observation)
```

Conditional event = samples with $(F_{lu} = 1, U = 1, T = 1)$.

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):
    samples = sample_a_ton()
    samples_observation =
        reject_inconsistent(samples, observation)
    samples_event =
        reject_inconsistent(samples_observation, event)
    return samples_event

def reject_inconsistent(samples, outcome):
    consistent_samples =
        (F_{lu} = x, U = 1, F_{ev} = y, T = 1)
    consistent_samples =
        (F_{lu} = 1)
    return consistent_samples
```

Condition

$(F_{lu} = x, U = 1, F_{ev} = y, T = 1) \mid (F_{lu} = 1) = 1$.

Rejection sampling algorithm

Inference question: What is $P(F_{lu} = 1 | U = 1, T = 1)$?

```
def rejection_sampling(event, observation):  
    samples = sample_a_ton()  
    samples_observation =  
        reject_inconsistent(samples, observation)  
    samples_event =  
        reject_inconsistent(samples_observation, event)  
    return len(samples_event)/len(samples_observation)
```

$$\text{Probability} \approx \frac{\# \text{ samples with } (F_{lu} = 1, U = 1, T = 1)}{\# \text{ samples with } (U = 1, T = 1)}$$

To the code!



Rejection sampling

If you can sample enough from the joint distribution, you can answer any probability inference question.

With enough samples, you can correctly compute:

- Probability estimates
- Conditional probability estimates
- Expectation estimates

Why? Because your samples represent the joint distribution incredibly well!

[flu, und, fev, tir]

Sampling...

[0, 1, 0, 1]

[0, 1, 0, 1]

[0, 1, 0, 1]

[0, 0, 0, 0]

[0, 1, 0, 1]

[0, 1, 1, 1]

[0, 1, 0, 0]

[1, 1, 1, 1]

[0, 0, 1, 1]

...

[0, 1, 0, 1]

Finished sampling

$$P(\text{has flu} \mid \text{undergrad and is tired}) = 0.122$$