15: General Inference

Jerry Cain
April 29, 2022
General Inference: Introduction
Inference
Inference

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:

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

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

Brute-force computation of a full joint probability mass function is often intractable.
N can be large...
Conditionally Independent RVs

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 $X_1, X_2, ..., X_n$ are called conditionally independent given $Y$ if:

for all $x_1, x_2, ..., x_n, y$:

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

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

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

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?

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

- Flu
- Undergrad
- Fever
- Tired

$P(F_{ev} = 1|F_{lu} = 1) = 0.9$
$P(F_{ev} = 1|F_{lu} = 0) = 0.05$
Constructing a Bayesian Network

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?

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

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

- $P(U = 1) = 0.8$

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

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

Using a Bayes Net

\[
P(F_{ev} = 1|F_{lu} = 1) = 0.9 \\
P(F_{ev} = 1|F_{lu} = 0) = 0.05
\]

\[
P(T = 1|F_{lu} = 0, U = 0) = 0.1 \\
P(T = 1|F_{lu} = 0, U = 1) = 0.8 \\
P(T = 1|F_{lu} = 1, U = 0) = 0.9 \\
P(T = 1|F_{lu} = 1, U = 1) = 1.0
\]
Inference: Math
Inference via math

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

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

Compute joint probabilities using chain rule.

\[ P(F_{ev} = 1|F_{lu} = 1) = 0.9 \quad P(T = 1|F_{lu} = 0, U = 0) = 0.1 \]
\[ P(F_{ev} = 1|F_{lu} = 0) = 0.05 \quad P(T = 1|F_{lu} = 0, U = 1) = 0.8 \]
\[ P(T = 1|F_{lu} = 1, U = 0) = 0.9 \quad P(T = 1|F_{lu} = 1, U = 1) = 1.0 \]
Inference via math

1. Compute joint probabilities
   \[ P(F_{lu} = 1, F_{ev} = 0, U = 0, T = 1) \]
   \[ P(F_{lu} = 0, F_{ev} = 0, U = 0, T = 1) \]

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

\[ P(F_{lu} = 1) = 0.1 \quad \text{and} \quad P(U = 1) = 0.8 \]

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

\[ P(F_{ev} = 1|F_{lu} = 1) = 0.9 \quad P(F_{ev} = 1|F_{lu} = 0) = 0.05 \]

\[ P(T = 1|F_{lu} = 0, U = 0) = 0.1 \quad P(T = 1|F_{lu} = 0, U = 1) = 0.8 \]

\[ P(T = 1|F_{lu} = 1, U = 0) = 0.9 \quad P(T = 1|F_{lu} = 1, U = 1) = 1.0 \]
Inference via math

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

1. Compute joint probabilities

\[
P(F_{lu} = 1, U = 1, F_{ev} = 1, T = 1) \\
... \]

\[
P(F_{lu} = 0, U = 1, F_{ev} = 0, T = 1) ?
\]

2. Definition of conditional probability

\[
\frac{\sum_{y} P(F_{lu} = 1, U = 1, F_{ev} = y, T = 1)}{\sum_{x} \sum_{y} P(F_{lu} = x, U = 1, F_{ev} = y, T = 1) } = 0.122
\]

\[
P(F_{lu} = 1) = 0.1 \\
P(U = 1) = 0.8
\]

\[
P(F_{ev} = 1|F_{lu} = 1) = 0.9 \\
P(F_{ev} = 1|F_{lu} = 0) = 0.05
\]

\[
P(T = 1|F_{lu} = 0, U = 0) = 0.1 \\
P(T = 1|F_{lu} = 0, U = 1) = 0.8 \\
P(T = 1|F_{lu} = 1, U = 0) = 0.9 \\
P(T = 1|F_{lu} = 1, U = 1) = 1.0
\]
Inference via math

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

Flu \quad Under-grad

Fever \quad Tired

\[ P(F_{ev} = 1|F_{lu} = 1) = 0.9 \]
\[ P(F_{ev} = 1|F_{lu} = 0) = 0.05 \]
\[ P(T = 1|F_{lu} = 0, U = 0) = 0.1 \]
\[ P(T = 1|F_{lu} = 0, U = 1) = 0.8 \]
\[ P(T = 1|F_{lu} = 1, U = 0) = 0.9 \]
\[ P(T = 1|F_{lu} = 1, U = 1) = 1.0 \]

Solving inference questions precisely is possible, but sometimes tedious.

Can we use sampling to do approximate inference?
Rejection Sampling
Rejection sampling algorithm

Step 0:
Require a fully specified Bayesian Network

\[ P(F_{tu} = 1) = 0.1 \]
\[ P(U = 1) = 0.8 \]

\[ P(F_{ev} = 1 | F_{tu} = 1) = 0.9 \]
\[ P(F_{ev} = 1 | F_{tu} = 0) = 0.05 \]
\[ P(T = 1 | F_{tu} = 0, U = 0) = 0.1 \]
\[ P(T = 1 | F_{tu} = 0, U = 1) = 0.8 \]
\[ P(T = 1 | F_{tu} = 1, U = 0) = 0.9 \]
\[ P(T = 1 | F_{tu} = 1, U = 1) = 1.0 \]
Rejection sampling algorithm

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

```python
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 \( (F_{lu}=1,U=1,T=1) \)
return len(samples_event)/len(samples_observation)
```

[flu, und, fev, tir]
Rejection sampling algorithm

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

```python
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 $(F_{lu} = 1, U = 1, T = 1)$
    return len(samples_event)/len(samples_observation)
```

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

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?

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: $P(E) = \lim_{n\to\infty} \frac{n(E)}{n}$, $n = \# \text{ of total trials}$, $n(E) = \# \text{ trials where } E \text{ occurs}$
Rejection sampling algorithm

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

```python
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 $(F_{lu} = 1, U = 1, T = 1)$
    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]
...
[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_{iu} = a, U = b, F_{ev} = c, T = d)
according to the joint probability?

Create a sample using the Bayesian Network!

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

```python
# 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]
```

![Bayesian Network Diagram]

- \( P(F_{lu} = 1) = 0.1 \)
- \( P(U = 1) = 0.8 \)
- \( P(F_{ev} = 1|F_{lu} = 1) = 0.9 \)
- \( P(F_{ev} = 1|F_{lu} = 0) = 0.05 \)
- \( P(T = 1|F_{lu} = 0, U = 0) = 0.1 \)
- \( P(T = 1|F_{lu} = 0, U = 1) = 0.8 \)
- \( P(T = 1|F_{lu} = 1, U = 0) = 0.9 \)
- \( P(T = 1|F_{lu} = 1, U = 1) = 1.0 \)
Rejection sampling algorithm

# 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)
    #
    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
# 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)
    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]
# 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 $P(F_{lu} = 1|U = 1, T = 1)$?

```python
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 $(F_{lu} = 1, U = 1, T = 1)$
return len(samples_event)/len(samples_observation)
```

[flu, und, fev, tir]
Rejection sampling algorithm

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

```python
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 $(F_{lu} = 1, U = 1, T = 1)$
    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)
Rejection sampling algorithm

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

```python
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_event = reject_inconsistent(samples_observation, event)
    return len(samples_event) / len(samples_observation)

# Method: Reject Inconsistent
# -------------------
# Rejects all samples that do not align with the outcome.
# Returns a list of consistent samples.
def reject_inconsistent(samples, outcome):
    consistent_samples = []
    for sample in samples:
        if check_consistent(sample, outcome):
            consistent_samples.append(sample)
    return consistent_samples

(\begin{align*}
U &= 1, \\
T &= 1
\end{align*})
Rejection sampling algorithm

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

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

```python
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

# Conditional
(F_{lu} = x, U = 1, F_{ev} = y, T = 1)  \quad (F_{lu} = 1)
return consistent_samples
```

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

Inference question:

```python
def reject_inconsistent(samples, outcome):
    consistent_samples
    return consistent_samples
```
Rejection sampling algorithm

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

```python
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)
```

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:

1. Probability estimates
2. Conditional probability estimates
3. Expectation estimates

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

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