General Inference

Chris Piech
CS109, Stanford University
THE MOMENT YOU REALISE

IT'S FRIDAY
Midterm Review
(5pm today)
Pset 3 - Random Variables
For Juliette Woodrow

Due Date: Friday, Oct 28, 2:15 PM Pacific Daylight Time (this hour).
Grace Period Date: Saturday, Oct 29, 2:15 PM Pacific Daylight Time (in 24 hours).

Get Started
Extension Request Forms

New peer learn icon!
Peer Learning in CS109

To find you the best person to work with, we open the peer learn queue for 5 minutes every 30 minutes. The queue is currently open. Signups close in

101 secs

Find me a match

When you hit the button we will either match you with a peer who is at a similar conceptual point in the course, or, if a section leader is available you will get to talk to them. Once you sign up, it may take up to 5 mins to find a good pairing. Once you are matched, we will put you into a virtual session with tools for you to collaborate. You should expect to spend around 15 minutes in the session.

Past Connections

<table>
<thead>
<tr>
<th>Collaborators</th>
<th>Date</th>
<th>Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ali Raza Malik</td>
<td>10/28/2022, 1:09:00 PM</td>
<td>[View session work]</td>
</tr>
<tr>
<td>Ali Raza Malik</td>
<td>10/28/2022, 1:09:00 AM</td>
<td>[View session work]</td>
</tr>
<tr>
<td>Ali Raza Malik</td>
<td>10/28/2022, 2:59:00 AM</td>
<td>[View session work]</td>
</tr>
</tbody>
</table>
Peer Learning in CS109

The queue is currently closed. The next peer learning session is starting in 02 secs

When you hit the button we will either match you with a peer who is at a similar conceptual point in the course, or, if a section leader is available you will get to talk to them. Once you sign up, it may take up to 5 mins to find a good pairing. Once you are matched, we will put you into a virtual session with tools for you to collaborate. You should expect to spend around 15 minutes in the session.

Past Connections

<table>
<thead>
<tr>
<th>Collaborators</th>
<th>Date</th>
<th>Links</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ali Raza Malik</td>
<td>10/28/2022, 1:09:00 PM</td>
<td>[View session work]</td>
</tr>
<tr>
<td>Ali Raza Malik</td>
<td>10/28/2022, 1:09:00 AM</td>
<td>[View session work]</td>
</tr>
<tr>
<td>Ali Raza Malik</td>
<td>10/28/2022, 1:05:00 AM</td>
<td>[View session work]</td>
</tr>
<tr>
<td>Ali Raza Malik</td>
<td>10/28/2022, 1:01:00 AM</td>
<td>[View session work]</td>
</tr>
<tr>
<td>Ali Raza Malik</td>
<td>10/27/2022, 8:58:46 PM</td>
<td>[View session work]</td>
</tr>
</tbody>
</table>
Learning Goals

1. Finish conversation on correlations
2. Learn rejection sampling
BAYES NETS!
Where do models come from?
Constructing a Bayesian Network

In a Bayesian Network, each random variable is caused by its parents. Def P(node | parents)

- Node: random variable
- Directed edge: causality

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

Flu

Under-grad

Fever

Tired

$P(T = 1 | F_{lu} = 0, U = 0)$
$P(T = 1 | F_{lu} = 0, U = 1)$
$P(T = 1 | F_{lu} = 1, U = 0)$
$P(T = 1 | F_{lu} = 1, U = 1)$
Make a **Generative** Model

A good probabilistic model is **generative**. It explains the process through which the joint is **created**.
Other applications

Chemical present?

Chemical detected?

Battery failure

Solar panel failure

Electrical system failure

Trajectory deviation

Communication loss
def get_prob_Xi(x, parents):
    # what is the probability that Xi = x
    # given the list parents of assignments to
    # the parents variables Xi

P(U = 1) = 0.8
P(Influenza = 1|Uni = 1) = 0.2
P(Influenza = 1|Uni = 0) = 0.1
P(Tired = 1|Uni = 0, Influenza = 0) = 0.1
P(Tired = 1|Uni = 1, Influenza = 0) = 0.8
P(Fever = 1|Influenza = 1) = 0.9
P(Fever = 1|Influenza = 0) = 0.05
P(Tired = 1|Uni = 0, Influenza = 1) = 0.9
P(Tired = 1|Uni = 1, Influenza = 1) = 1.0
Bayesian Network Assumption

Order nodes by ancestry

\[ P(\text{Joint}) = \prod_{i} P(x_i | x_{i-1}, \ldots, x_1) = \prod_{i} P(x_i | \text{Values of parents of } X_i) \]

Assume: Once you know the value of the parents of a variable in your network, \( X_i \), any further information about non-descendants will not change your belief in \( X_i \).
End Review
How do people design these?
<table>
<thead>
<tr>
<th></th>
<th>Dance</th>
<th>Folk</th>
<th>Country</th>
<th>Classical Music</th>
<th>Musical</th>
<th>Pop</th>
<th>Rock</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>11</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>12</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>13</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>14</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>16</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>17</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>18</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>19</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>20</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>21</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>22</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>23</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>24</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>26</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>27</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>28</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>29</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>30</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>31</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>32</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>33</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>34</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>35</td>
<td>5</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>
From Correlation to Bayes Net. Alternative!
The art of modelling

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

1. Design this

2. Also design this.

Later in CS109: learn this from data

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

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

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

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

\[
P(T = 1|F_{lu} = 1, U = 0) = 0.9
\]

\[
P(T = 1|F_{lu} = 1, U = 1) = 1.0
\]
Calculate the Covariance / Correlation (new stat!)

\[
\text{Cov}(X, Y) = E[(X - E[X])(Y - E[Y])]
\]

\[
\text{Cov}(X, Y) = E[XY] - E[Y]E[X]
\]
Covariance of Zero Does Not Mean Independence!

\( X \) and \( Y \) are random variables:

\( X \) is -1, 0 or 1 with equal probability

\[
Y = \begin{cases} 
0 & \text{if } X \neq 0 \\
1 & \text{otherwise}
\end{cases}
\]
Covariance of Zero Does Not Mean Independence!

X and Y are random variables with PMF:

<table>
<thead>
<tr>
<th>Y</th>
<th>X</th>
<th>p_X(x)</th>
<th>p_Y(y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-1</td>
<td>1/3</td>
<td>1/3</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>1/3</td>
</tr>
<tr>
<td>1</td>
<td>1/3</td>
<td>1/3</td>
<td>1/3</td>
</tr>
</tbody>
</table>

\[
Y = \begin{cases} 
0 & \text{if } X \neq 0 \\
1 & \text{otherwise} 
\end{cases}
\]

- \( E[X] = -1(1/3) + 0(1/3) + 1(1/3) = 0 \)
- \( E[Y] = 0(2/3) + 1(1/3) = 1/3 \)
- Since \( XY = 0 \), \( E[XY] = 0 \)
- \( \text{Cov}(X, Y) = E[XY] - E[X]E[Y] = 0 - 0 = 0 \)

But, X and Y are clearly dependent!
What is Wrong With This?

Consider the following data:

<table>
<thead>
<tr>
<th>Weight</th>
<th>Height</th>
<th>Weight * Height</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>57</td>
<td>3648</td>
</tr>
<tr>
<td>71</td>
<td>59</td>
<td>4189</td>
</tr>
<tr>
<td>53</td>
<td>49</td>
<td>2597</td>
</tr>
<tr>
<td>67</td>
<td>62</td>
<td>4154</td>
</tr>
<tr>
<td>55</td>
<td>51</td>
<td>2805</td>
</tr>
<tr>
<td>58</td>
<td>50</td>
<td>2900</td>
</tr>
<tr>
<td>77</td>
<td>55</td>
<td>4235</td>
</tr>
<tr>
<td>57</td>
<td>48</td>
<td>2736</td>
</tr>
<tr>
<td>56</td>
<td>42</td>
<td>2352</td>
</tr>
<tr>
<td>51</td>
<td>42</td>
<td>2142</td>
</tr>
<tr>
<td>76</td>
<td>61</td>
<td>4636</td>
</tr>
<tr>
<td>68</td>
<td>57</td>
<td>3876</td>
</tr>
</tbody>
</table>

\[
E[W] = 62.75 \quad E[H] = 52.75 \quad E[W*H] = 3355.83
\]

\[
\]

\[
= 3355.83 - (62.75)(52.75)
\]

\[
= 45.77
\]
Cauchy Schwarz, a great way to normalize!

\[-\text{Std}(X)\text{Std}(Y) \leq \text{Cov}(X, Y) \leq \text{Std}(X)\text{Std}(Y)\]
Correlation is just normalized Covariance

\[ \rho(X, Y) = \frac{\text{Cov}(X, Y)}{\sqrt{\text{Var}(X)\text{Var}(Y)}} \]

It is always true that

\[ \text{Cov}(X, Y) < \sqrt{\text{Var}(X)\text{Var}(Y)} \]
\[ \text{Cov}(X, Y) > -\sqrt{\text{Var}(X)\text{Var}(Y)} \]
Rock Music Vs Oil?

High Correlation

Hubbert Peak Theory

http://www.aei.org/publication/blog/
Tell your friends!

![Graph showing correlation between per capita consumption of cheese and number of people who died by becoming tangled in their bedsheets.](image)

- **Per capita consumption of cheese (US):**
  - 2000: 29.8
  - 2001: 30.1
  - 2002: 30.5
  - 2003: 30.6
  - 2004: 31.3
  - 2005: 31.7
  - 2006: 32.6
  - 2007: 33.1
  - 2008: 32.7
  - 2009: 32.8

- **Number of people who died by becoming tangled in their bedsheets (Deaths US):**
  - 2000: 327
  - 2001: 456
  - 2002: 509
  - 2003: 497
  - 2004: 596
  - 2005: 573
  - 2006: 661
  - 2007: 741
  - 2008: 809
  - 2009: 717

**Correlation:** 0.947091

**Sources:** USDA, CDC, tylervigen.com

---

Chris Piech, CS109, 2021

Stanford University
Divorce Vs Butter?

Correlation: 99%

Divorce rate in Maine per 1,000 people

Source: US Census, USDA, tylerigen.com

Recall: It is a useful starting point
We have models. Need to solve problems
Inference: Algebra
Bayes Nets: 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

Review
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.

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

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

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

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

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

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

\[
= 0.095
\]
Inference via math

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

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

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

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

2. Definition of conditional probability
   \[ \frac{\sum_y P(F_{lu} = 1, U = 1, F_{ev} = y, T = 1)}{\sum_x \sum_y P(F_{lu} = x, U = 1, F_{ev} = y, T = 1)} = 0.122 \]
Rejection sampling algorithm

Step 0:
Have a fully specified Bayesian Network

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

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

$P(T = 1|F_{lu} = 0, U = 0) = 0.1$
$P(T = 1|F_{lu} = 0, U = 1) = 0.8$
$P(T = 1|F_{lu} = 1, U = 0) = 0.9$
$P(T = 1|F_{lu} = 1, U = 1) = 1.0$
Too many possible inference questions one could ask...
N_SAMPLES = 100000

# Program: Joint Sample
# ___________________
# we can answer any probability question
# with multivariate samples from the joint,
# where conditioned variables match

def main():
    obs = getObservation()
    print 'Observation = ', obs

    samples = sampleATon()
    prob = probFluGivenObs(samples, obs)
    print 'Pr(Flu) = ', prob
# Method: Sample A Ton
# ------------------------
# chose N_SAMPLES with likelihood proportional to the joint distribution

def sampleATon():
    samples = []
    for i in range(N_SAMPLES):
        sample = makeSample()
        samples.append(sample)
    return samples
Recall: Probabilistic Model

\[ P(Fl = 1) = 0.1 \]

\[ P(Fev = 1|Flu = 1) = 0.9 \]
\[ P(Fev = 1|Flu = 0) = 0.05 \]

\[ P(U = 1) = 0.8 \]

\[ P(T = 1|Flu = 0, U = 0) = 0.1 \]
\[ P(T = 1|Flu = 0, U = 1) = 0.8 \]
\[ P(T = 1|Flu = 1, U = 0) = 0.9 \]
\[ P(T = 1|Flu = 1, U = 1) = 1.0 \]
# Method: Make Sample
# ------------------
# chose a single sample from the joint distribution
# based on the medical "Probabilistic Graphical Models"

def makeSample():
    # prior on causal factors
    flu = bern(0.1)
    und = bern(0.8)

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

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

    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
# Method: Make Sample
#
# chose a single sample from the joint distribution
# based on the medical "Probabilistic Graphical Models"

def makeSample():
    # prior on causal factors
    flu = bern(0.1)
    und = bern(0.8)

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

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

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

# chose a single sample from the joint distribution
# based on the medical "Probabilistic Graphical Models"

def makeSample():
    # prior on causal factors
    flu = bern(0.1)
    und = bern(0.8)

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

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

    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
# Method: Make Sample
#
# chose a single sample from the joint distribution
# based on the medical "Probabilistic Graphical Models"

def makeSample():
    # prior on causal factors
    flu = bern(0.1)
    und = bern(0.8)

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

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

    # a sample from the joint has an
    # assignment to *all* random variables
    return [flu, und, fev, tir]
N_SAMPLES = 100000

# Program: Joint Sampling
# _________________
# we can answer any problem with multivariate sampling
# where conditioned variables are

def main():
    obs = getObservation()
    print 'Observation'

    samples = sampleAtObserver(obs)
    prob = probFluGivenSamples(samples)
    print 'Pr(Flu) =', prob
Alg #1: Rejection Sampling

N_SAMPLES = 100000

# Program: Joint Sample
# ________________
# we can answer any probability question
# with multivariate samples from the joint,
# where conditioned variables match

def main():
    obs = getObservation()
    print 'Observation = ', obs

    samples = sampleATon()
    prob = probFluGivenObs(samples, obs)
    print 'Pr(Flu) = ', prob
# Method: Probability of Flu Given Observation
# ---------------------
# Calculate the probability of flu given many samples from the joint distribution and a set of observations to condition on.

def probFluGivenObs(samples, obs):
    # reject all samples which don't align with condition
    keepSamples = []
    for sample in samples:
        if checkObsMatch(sample, obs):
            keepSamples.append(sample)

    # from remaining, simply count...
    fluCount = 0
    for sample in keepSamples:
        [flu, und, fev, tir] = sample
        if flu == 1:
            fluCount += 1

    # counting can be so sweet...
    return float(fluCount) / len(keepSamples)
Method: Probability of Flu Given Observation

Calculate the probability of flu given many samples from the joint distribution and a set of observations to condition on.

```python
def probFluGivenObs(samples, obs):
    keepSamples = []
    for sample in samples:
        if checkObsMatch(sample, obs):
            keepSamples.append(sample)

    fluCount = 0
    for sample in keepSamples:
        [flu, und, fev, tir] = sample
        if flu == 1:
            fluCount += 1

    return float(fluCount) / len(keepSamples)
```
# Method: Probability of Flu Given Observation
# -----------------------------
# Calculate the probability of flu given many
# samples from the joint distribution and a set
# of observations to condition on.

def probFluGivenObs(samples, obs):
    # reject all samples which don't align
    # with condition
    keepSamples = []
    for sample in samples:
        if checkObsMatch(sample, obs):
            keepSamples.append(sample)

    # from remaining, simply count...
    fluCount = 0
    for sample in keepSamples:
        [flu, und, fev, tir] = sample
        if flu == 1:
            fluCount += 1

    # counting can be so sweet...
    return float(fluCount) / len(keepSamples)
Algorithm 1: Rejection Sampling

```python
N_SAMPLES = 100000

# Program: Joint Sample
# ________________
# we can answer any probability
# with multivariate sample
# where conditioned var

def main():
    obs = getObservation()
    print('Observation')
    samples = sampleAtObs(obs)
    prob = probFluGiven(samples)
    print('Pr(Flu) =', prob)
    Pr(Flu) = 0.141503173687
```

Let's try it!
To the code!
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 \( P(F_{lu} = 1|U = 1, T = 1) \)?

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:

\[
P(E) = \lim_{{n \to \infty}} \frac{n(E)}{n}
\]

\( n = \# \) of total trials

\( n(E) = \# \) trials where \( E \) occurs
If you can sample enough from the joint distribution, you can answer any probability question.

Each one of these is one joint sample: [Flu, Undergrad, Fever, Tired]

Observation = [None, None, None, None, None]
Pr(Flu | Obs) = 0.10164
What's the matter with rejection sampling?
Probabilistic Model

\[ P(Fl = 1) = 0.1 \]

- **Flu**
  - \[ P(Fev = 1|Flu = 1) = 0.9 \]
  - \[ P(Fev = 1|Flu = 0) = 0.05 \]

- **Undergrad**
  - \[ P(U = 1) = 0.8 \]
  - \[ P(T = 1|Flu = 0, U = 0) = 0.1 \]
  - \[ P(T = 1|Flu = 0, U = 1) = 0.8 \]
  - \[ P(T = 1|Flu = 1, U = 0) = 0.9 \]
  - \[ P(T = 1|Flu = 1, U = 1) = 1.0 \]

- **Fever**

- **Tired**
Probabilistic Model

\[ P(Fl = 1) = 0.1 \]

\[ P(U = 1) = 0.8 \]

\[ P(T = 1|Flu = 0, U = 0) = 0.1 \]
\[ P(T = 1|Flu = 0, U = 1) = 0.8 \]
\[ P(T = 1|Flu = 1, U = 0) = 0.9 \]
\[ P(T = 1|Flu = 1, U = 1) = 1.0 \]

\[ Fev|Flu = 0 \sim N(100.0, 1.81) \]
\[ Fev|Flu = 1 \sim N(98.25, 0.73) \]
The Magic School Bus™
Markov Chain

MCMC

Monte Carlo
Each one of these is one posterior sample:

[Flu, Undergrad, Fever, Tired]

MCMC is a way to sample with conditioned variables fixed.
Many Algorithms

Rejection Sampling

MCMC

Pyro

Idea2Text
What about Code.org?
Computers Couldn’t Understand Code

60,000 students attempted this problem
37,000 unique solutions

Challenge

Student Code

Insight

You need to move and turn in your loop
Computers Couldn’t Understand Code

Feedback F1 Score

Code.org Short Code (10 lines)

Cond. Prob.  Deep Learning  Generative Model  Humans

Stanford University
Generative Model of Handwritten Digits

Lake et al, 2015

A

i) primitives

ii) sub-parts

iii) parts

iv) object template

relation: attached along

relation: attached along

relation: attached at start

type level
token level

B

procedure GENERATETYPE

\( \kappa \leftarrow P(\kappa) \) \hfill \triangleright \text{Sample number of parts}

for \( i = 1 \ldots \kappa \) do

\( n_i \leftarrow P(n_i | \kappa) \) \hfill \triangleright \text{Sample number of sub-parts}

for \( j = 1 \ldots n_i \) do

\( s_{ij} \leftarrow P(s_{ij} | s_{i(1)} \ldots s_{i(j-1)}) \) \hfill \triangleright \text{Sample sub-part sequence}

end for

\( R_i \leftarrow P(R_i | S_1 \ldots S_{i-1}) \) \hfill \triangleright \text{Sample relation}

end for

\( \psi \leftarrow \{ \kappa, R, S \} \)

return @GENERATETOKEN(\psi) \hfill \triangleright \text{Return program}

procedure GENERATETOKEN(\psi)

for \( i = 1 \ldots \kappa \) do

\( S_i^{(m)} \leftarrow P(S_i^{(m)} | S_i) \) \hfill \triangleright \text{Add motor variance}

\( L_i^{(m)} \leftarrow P(L_i^{(m)} | R_i, T_i^{(m)} \ldots T_{i-1}^{(m)}) \) \hfill \triangleright \text{Sample part’s start location}

\( T_i^{(m)} \leftarrow f(L_i^{(m)}, s_i^{(m)}) \) \hfill \triangleright \text{Compose a part’s trajectory}

end for

\( A^{(m)} \leftarrow P(A^{(m)}) \) \hfill \triangleright \text{Sample affine transform}

\( I^{(m)} \leftarrow P(I^{(m)} | T^{(m)}, A^{(m)}) \) \hfill \triangleright \text{Sample image}

return I^{(m)}
Inference. Given a character, infer generation.
Human Level. And More!
Generative Model of Grading

Lake et al, 2015

Muke Wu, Ali Malik, Noah Goodman, Chris Piech, 2019

Outstanding Student paper award, AAAI 2019

Chris Piech, CS109, 2021
Computers Couldn’t Understand Code

Code.org Short Code (10 lines)

Feedback F1 Score

Cond. Prob. Deep Learning Generative Model Humans
Idea 2 Text

Generative model

Decision process

Output solution

Inference model
What is the Generative Model for Binomial Questions?

class DeclareExpTask(Decision):
    def renderCode(self):
        explicit = self.getChoice('explicitRv')
        if explicit:
            return self.expand('DeclareExplicitExpTask')
        else:
            return self.expand('DeclareSubtleExpTask')

TEMPLATES = {
    'standard':{
        'template': 'what is the expected number of {weight:5}
    },
    'v2':{
        'template': 'what is the expectation of {success:5}
    },
    'v3':{
        'template': 'what is the average number of {success:5}
    },
}

class DeclareSubtleExpTask(Decision):
    def registerChoices(self):
        self.addChoice('expStyle1', gu.makeChoicesFromTemplate
    def renderCode(self):
        tempVars = {
            'successes': self.getState('successesStr')
        }
        key = self.getChoice('expStyle1')
template = TEMPLATES[key]['template']

Answer:
You are flipping a coin 50 times. The probability of a head on each coin-<br>flip is 1/2. What is the probability that the number of heads is 21?

Answer:
Let $X$ be the number of heads.
$X \sim \text{Bin}(n = 50, p = 1/2)$
$P(X = 21) = (n \choose 21) p^{21} (1 - p)^{n - 21}$

Answer:
You are trying to mine bitcoins. You try 100 times. The probability of a mining a bitcoin on each attempt is 3/25. What is the probability that the number of bitcoins mined is 99?

Answer:
Let $X$ be the number of bitcoins mined.
$X \sim \text{Bin}(n = 100, p = 3/25)$
$P(X = 99) = (n \choose 99) p^{99} (1 - p)^{n - 99}$

Answer:
You are running in an election. The number of votes for you can be represented by a random variable $X$. $X \sim \text{Bin}(n = 100, p = 1/2)$. What is the probability that $X$ is equal to 67?

Answer:
Let $X$ be the number of votes for you.
$X \sim \text{Bin}(n = 100, p = 1/2)$
$P(X = 6) = (n \choose 6) p^{6} (1 - p)^{n - 6}$

Answer:
A ball hits a series of 10 pins where it can bounce either right or left. The probability of a right on each pin hit is 0.5. What is the probability that the number of right hits is greater than 7?

Answer:
Let $X$ be the number of rights.
$X \sim \text{Bin}(n = 10, p = 0.5)$
$P(X > 7) = \sum_{i=8}^{10} (n \choose i) p^{i} (1 - p)^{n - i}$
What haven’t we talked about?
Inference via math

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

1. Learn this from data

2. Learn this from data

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

\[ P(T = 1|F_{lu} = 0, U = 0) = 0.1 \]
\[ P(T = 1|F_{lu} = 0, U = 1) = 0.8 \]
\[ P(T = 1|F_{lu} = 1, U = 0) = 0.9 \]
\[ P(T = 1|F_{lu} = 1, U = 1) = 1.0 \]
See you at the midterm