General Inference

CS109, Stanford University

Light Midterm Reflection

Learning Goals

1. Finish conversation on correlations
2. Learn rejection sampling
3. Maybe get to the beta? (What's a beta?)

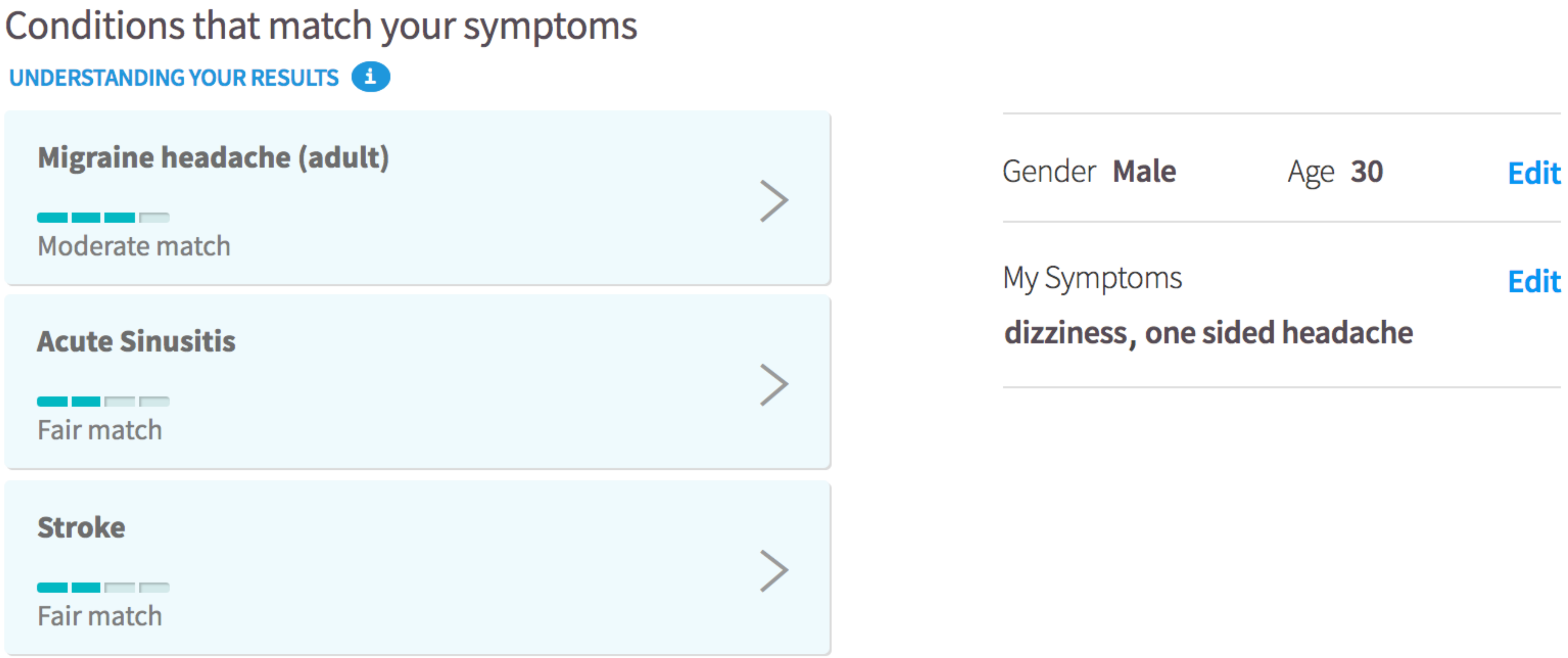
BAYES NETS!

Where do models come from?

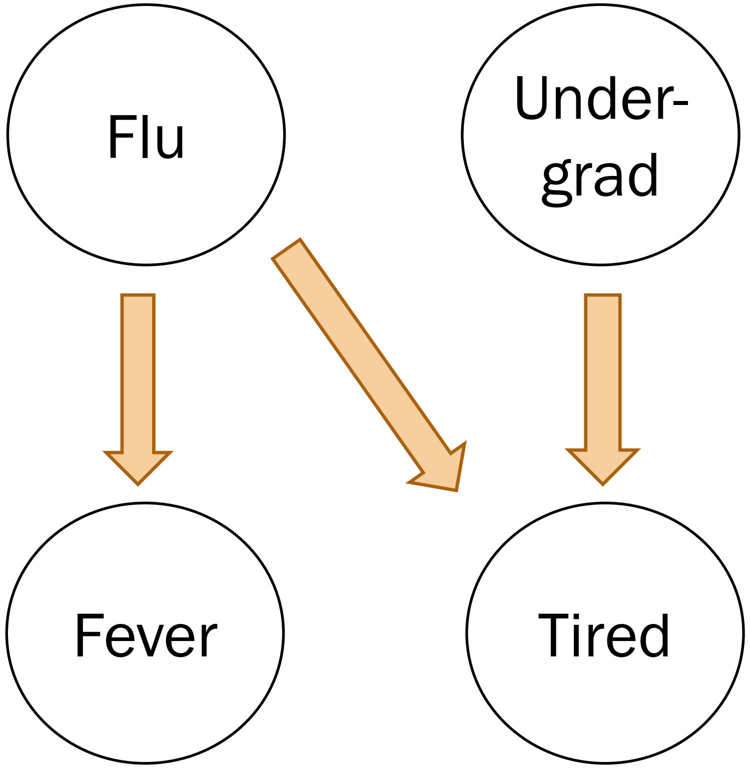
Multiple Random Variables. Start of Digital Revolution

WebMD HEALTH NOW

Multiple Random Variables. Start of Digital Revolution



Constructing a Bayesian Network



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

In a Bayesian Network,

Each random variable is caused by its parents. Def P(node | parents)

- Node: random variable
- Directed edge: causality

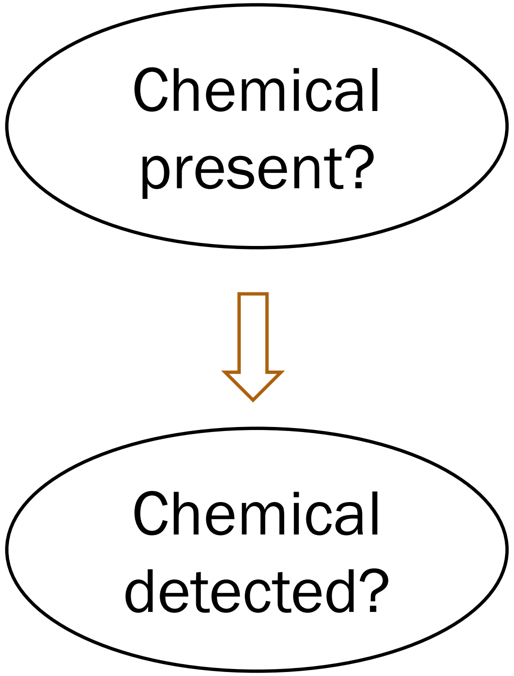
Examples:

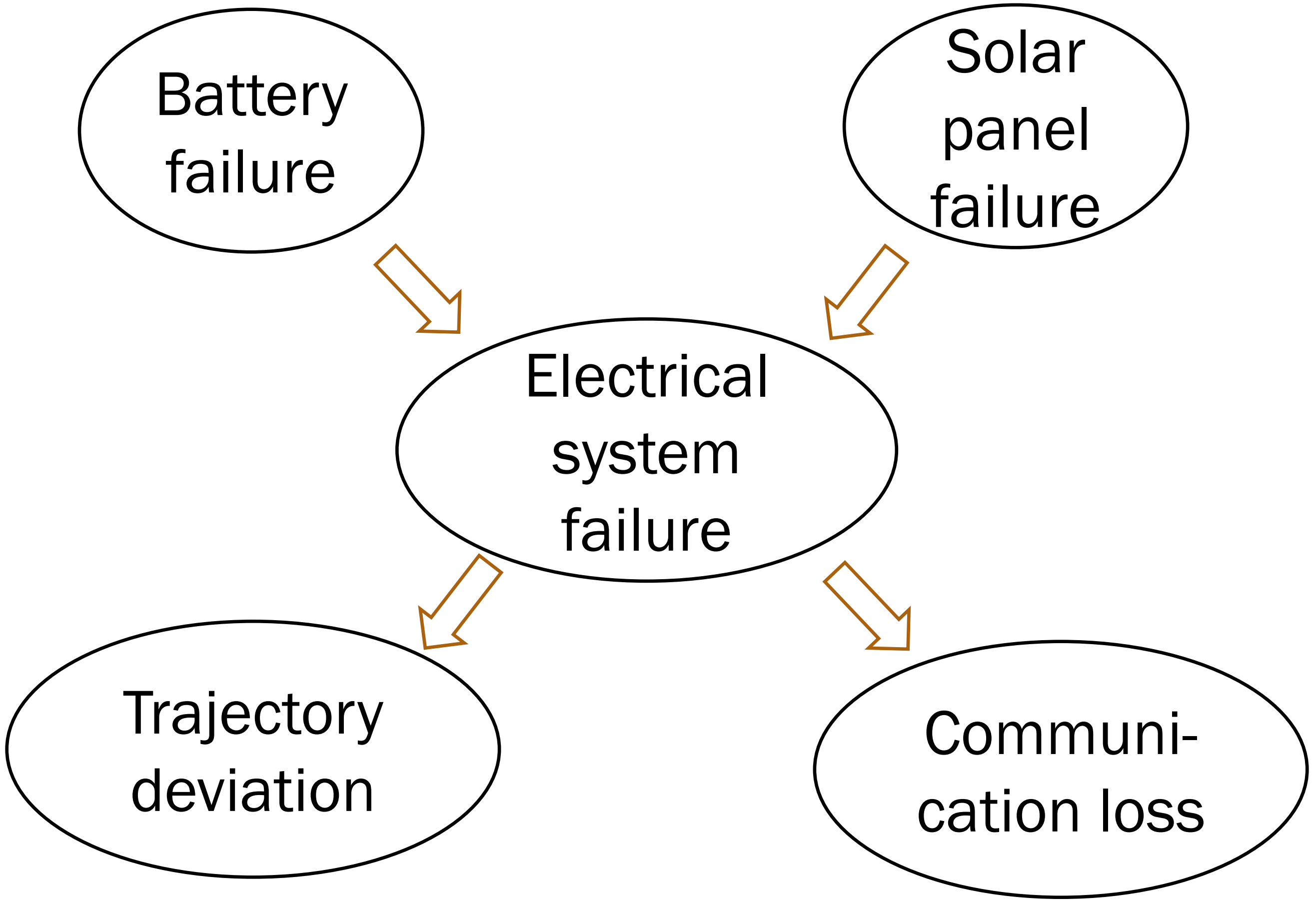
- $P(F_{lu} = 1)$
- $P(U = 0)$
- $P(F_{ev} = 1|F_{lu} = 1)$, $P(F_{ev} = 1|F_{lu} = 0)$
- $P(T = 1|F_{lu} = 0, U = 0)$ …

Make a *Generative* Model

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

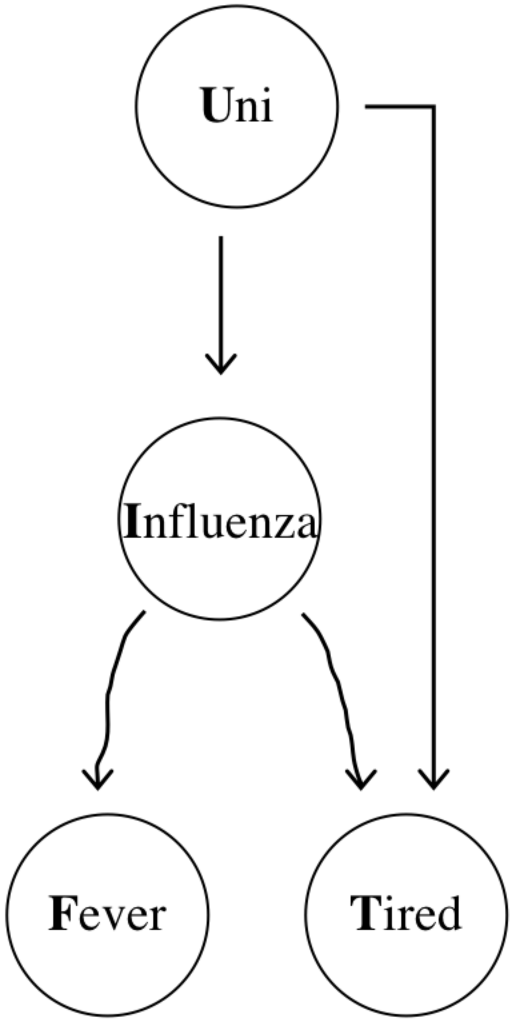
Other applications





Bayesian Network

Simple Disease Model




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

$P(\text{Uni} = 1) = 0.8$

$P(\text{Influenza} = 1|\text{Uni} = 1) = 0.2$

$P(\text{Influenza} = 1|\text{Uni} = 0) = 0.1$

$P(\text{Tired} = 1|\text{Uni} = 0, \text{Influenza} = 0) = 0.1$

$P(\text{Tired} = 1|\text{Uni} = 1, \text{Influenza} = 0) = 0.8$

$P(\text{Fever} = 1|\text{Influenza} = 1) = 0.9$

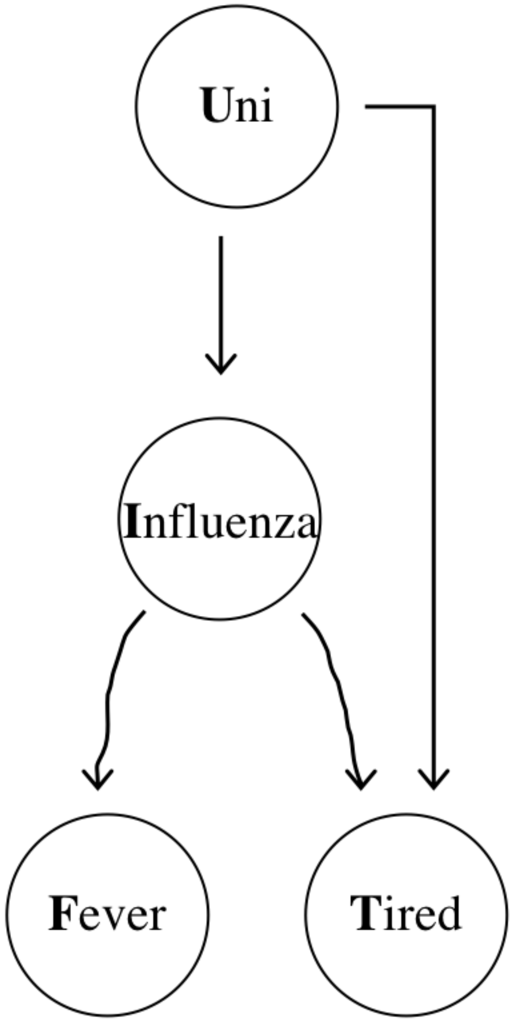
$P(\text{Fever} = 1|\text{Influenza} = 0) = 0.05$

$P(\text{Tired} = 1|\text{Uni} = 0, \text{Influenza} = 1) = 0.9$

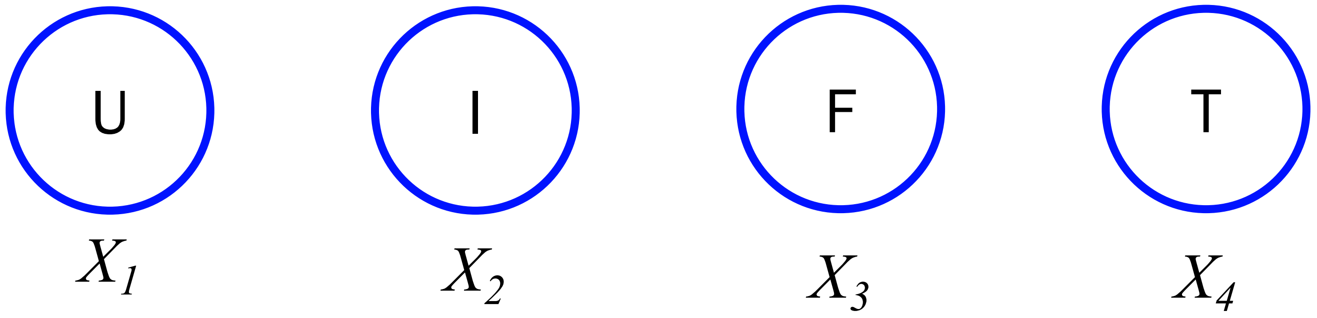
$P(\text{Tired} = 1|\text{Uni} = 1, \text{Influenza} = 1) = 1.0$

Bayesian Network Assumption

Simple Disease Model



Order nodes by ancestry



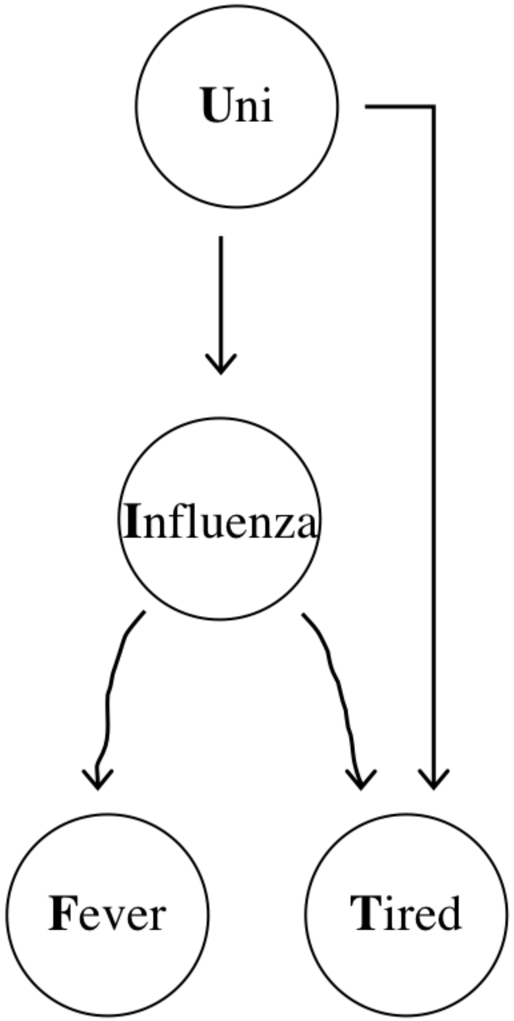
$$P(\text{Joint}) = \prod_i P(x_i | x_{i-1}, \ldots, x_1)$$

$$= \prod_i P(x_i | \text{Values of parents of } X_i)$$

Assume: Once you know the value of the parents of a variable in your network, X_i, any further information about non-descendents will not change your belief in X_i.

Bayesian Network

Simple Disease Model



Let's go ahead and figure this one out together!

P(Fever = 0, Influenza = 1, Uni = 0, Tired = 0)

$P(\text{Uni} = 1) = 0.8$

$P(\text{Influenza} = 1|\text{Uni} = 1) = 0.2$

$P(\text{Influenza} = 1|\text{Uni} = 0) = 0.1$

$P(\text{Tired} = 1|\text{Uni} = 0, \text{Influenza} = 0) = 0.1$

$P(\text{Tired} = 1|\text{Uni} = 1, \text{Influenza} = 0) = 0.8$

$P(\text{Fever} = 1|\text{Influenza} = 1) = 0.9$

$P(\text{Fever} = 1|\text{Influenza} = 0) = 0.05$

$P(\text{Tired} = 1|\text{Uni} = 0, \text{Influenza} = 1) = 0.9$

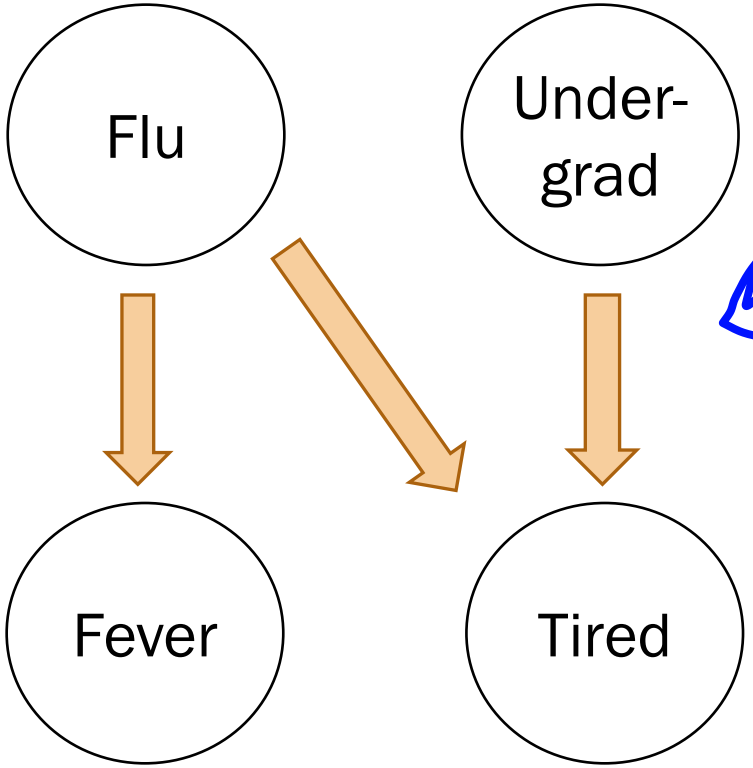
$P(\text{Tired} = 1|\text{Uni} = 1, \text{Influenza} = 1) = 1.0$

End Review

How do people design these?

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

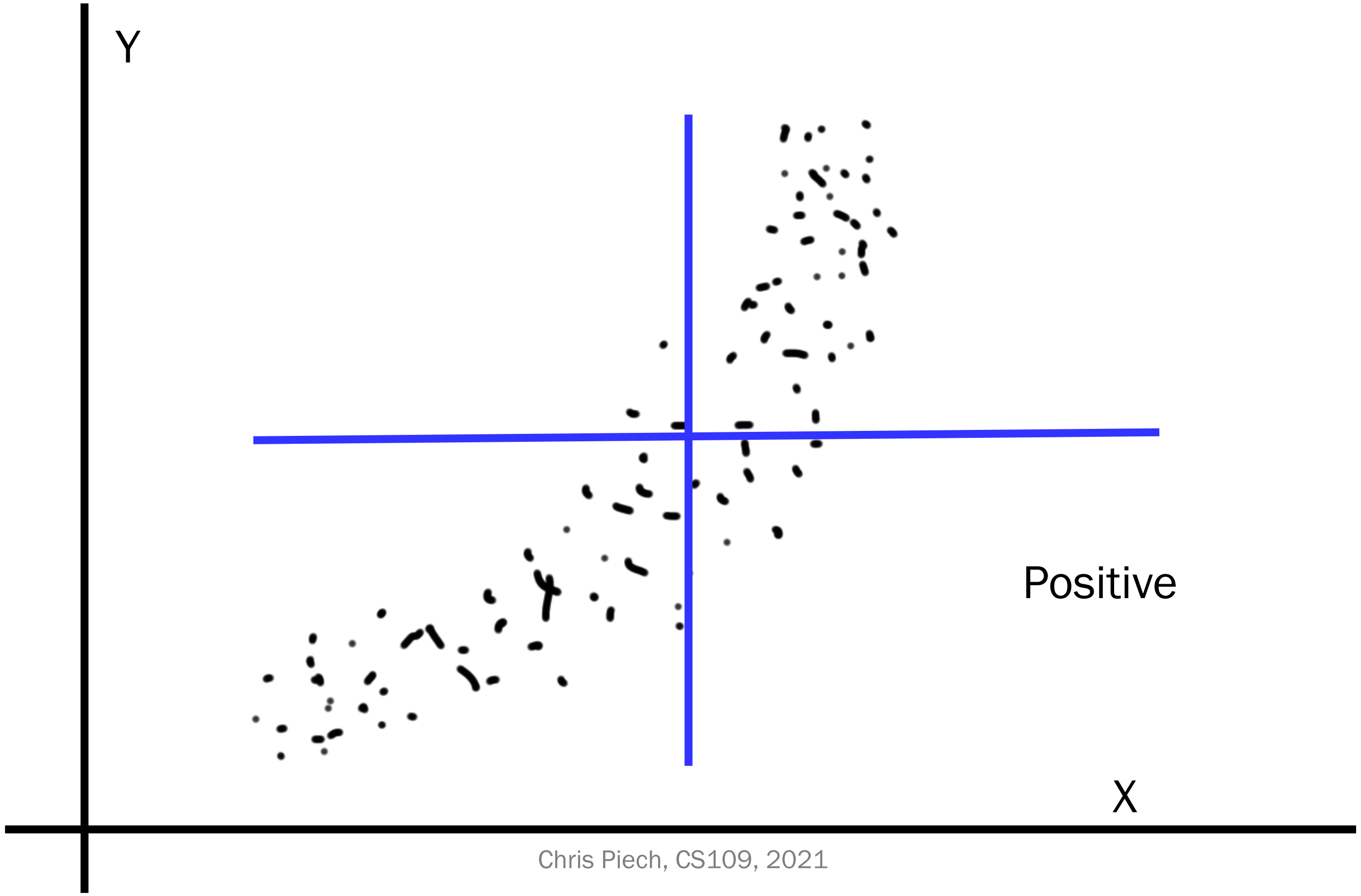
$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$
$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$

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

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:

Y \ X	-1	0	1	$p_Y(y)$
0	1/3	0	1/3	2/3
1	0	1/3	0	1/3
$p_X(x)$	1/3	1/3	1/3	1

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

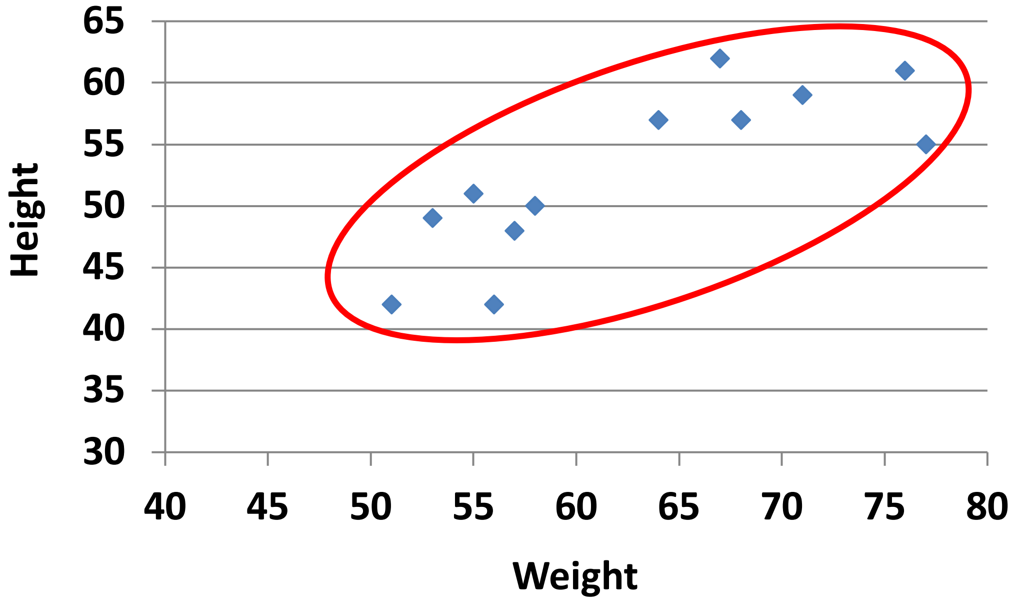
- E[X] = -1(1/3) + 0(1/3) + 1(1/3) = 0
- E[Y] = 0(2/3) + 1(1/3) = 1/3
- Since XY = 0, E[XY] = 0
- Cov(X, Y) = E[XY] – E[X]E[Y] = 0 – 0 = 0

But, X and Y are clearly dependent!

What is Wrong With This?

Consider the following data:

Weight	Height	Weight * Height
64	57	3648
71	59	4189
53	49	2597
67	62	4154
55	51	2805
58	50	2900
77	55	4235
57	48	2736
56	42	2352
51	42	2142
76	61	4636
68	57	3876



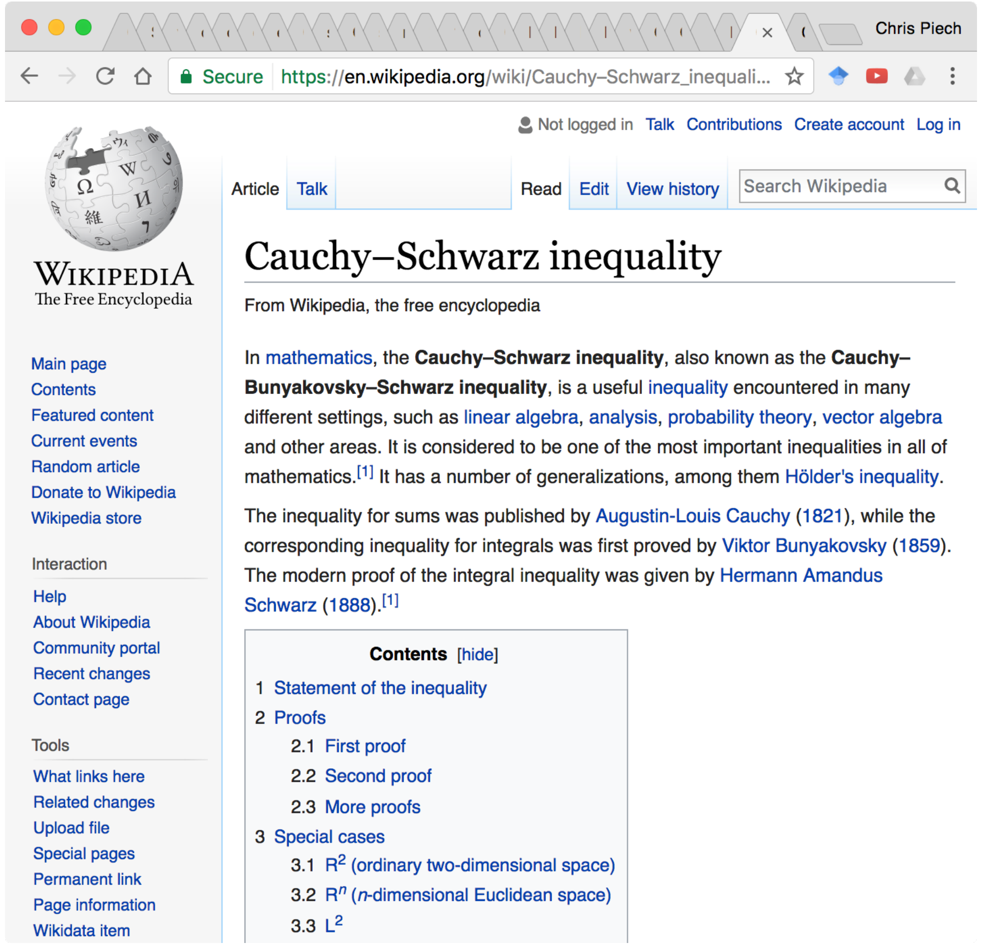
E[W] = 62.75 **E[H] = 52.75** **E[W*H] = 3355.83**

Cov(W, H) = E[W*H] – E[W]E[H]

= 3355.83 – (62.75)(52.75)

= 45.77

Cauchy Schwarz, a great way to normalize!



$$-\mathrm{Std}(X)\mathrm{Std}(Y) \leq \mathrm{Cov}(X,Y) \leq \mathrm{Std}(X)\mathrm{Std}(Y)$$

Just in case you wanted to see the proof....

Probability theory [edit]

Let X and Y be random variables. Then the covariance inequality[17][18] is given by:

$$\operatorname{Var}(X) \geq \frac{\operatorname{Cov}(X,Y)^2}{\operatorname{Var}(Y)}.$$

After defining an inner product on the set of random variables using the expectation of their product,

$$\langle X, Y \rangle := \operatorname{E}(XY),$$

the Cauchy–Schwarz inequality becomes

$$|\operatorname{E}(XY)|^2 \leq \operatorname{E}(X^2)\operatorname{E}(Y^2).$$

To prove the covariance inequality using the Cauchy–Schwarz inequality, let $\mu = \operatorname{E}(X)$ and $\nu = \operatorname{E}(Y)$, then

$$\begin{aligned}
|\operatorname{Cov}(X,Y)|^2 &= |\operatorname{E}((X-\mu)(Y-\nu))|^2 \\
&= |\langle X-\mu, Y-\nu \rangle|^2 \\
&\leq \langle X-\mu, X-\mu \rangle \langle Y-\nu, Y-\nu \rangle \\
&= \operatorname{E}\left((X-\mu)^2\right)\operatorname{E}\left((Y-\nu)^2\right) \\
&= \operatorname{Var}(X)\operatorname{Var}(Y),
\end{aligned}$$

where Var denotes variance and Cov denotes covariance.

Correlation is just normalized Covariance

Correlation

Covariance

$$\rho(X,Y) = \frac{\mathrm{Cov}(X,Y)}{\sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)}}$$

It is always true that

$$\mathrm{Cov}(X,Y) < \sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)}$$

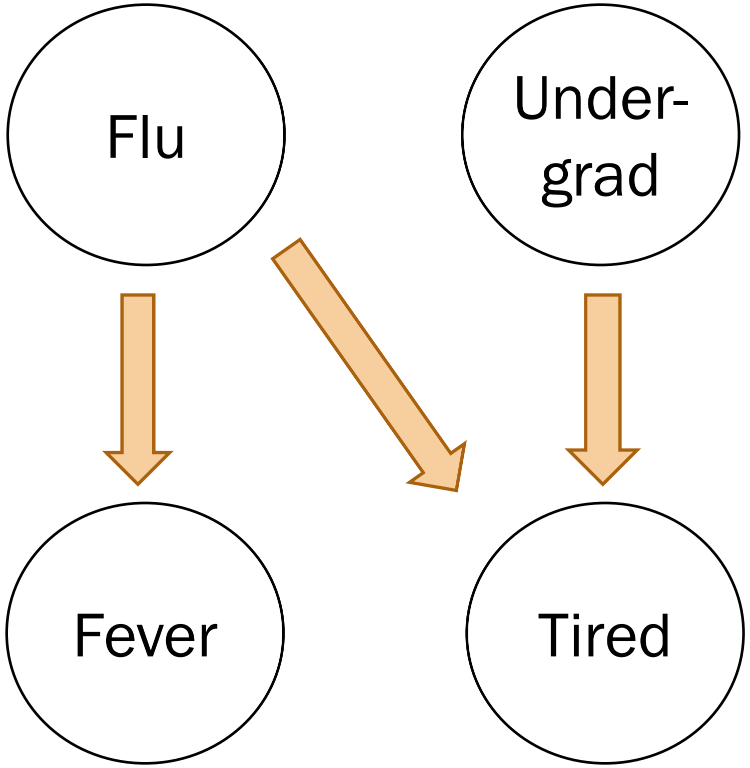
$$\mathrm{Cov}(X,Y) > -\sqrt{\mathrm{Var}(X)\mathrm{Var}(Y)}$$

We have models. Need to solve problems

Inference: Algebra

Bayes Nets: Conditional independence

Review



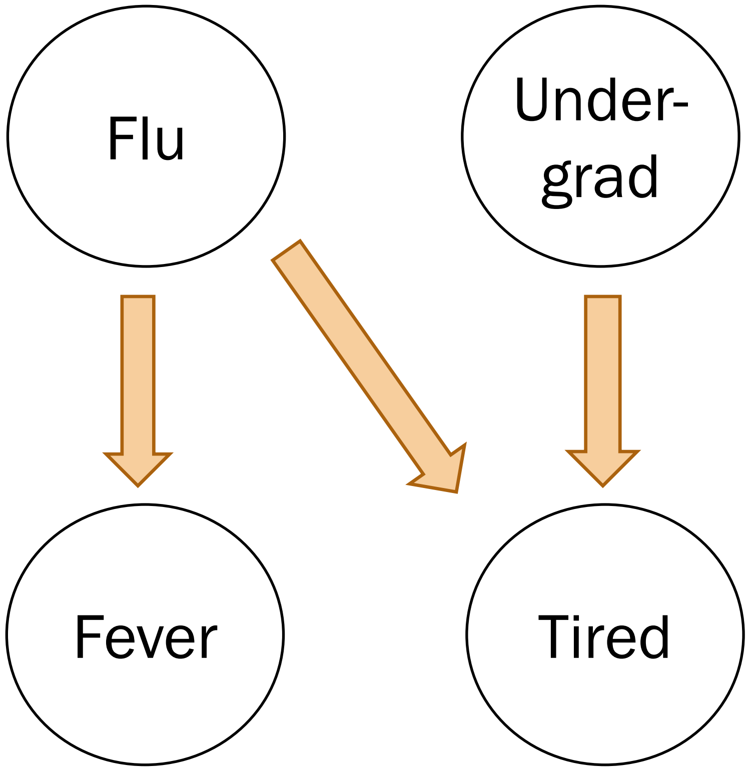
In a Bayesian Network,

Each random variable is conditionally independent of its non-descendants, given its parents.

- Node: random variable
- Directed edge: conditional dependency

Inference via math

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



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

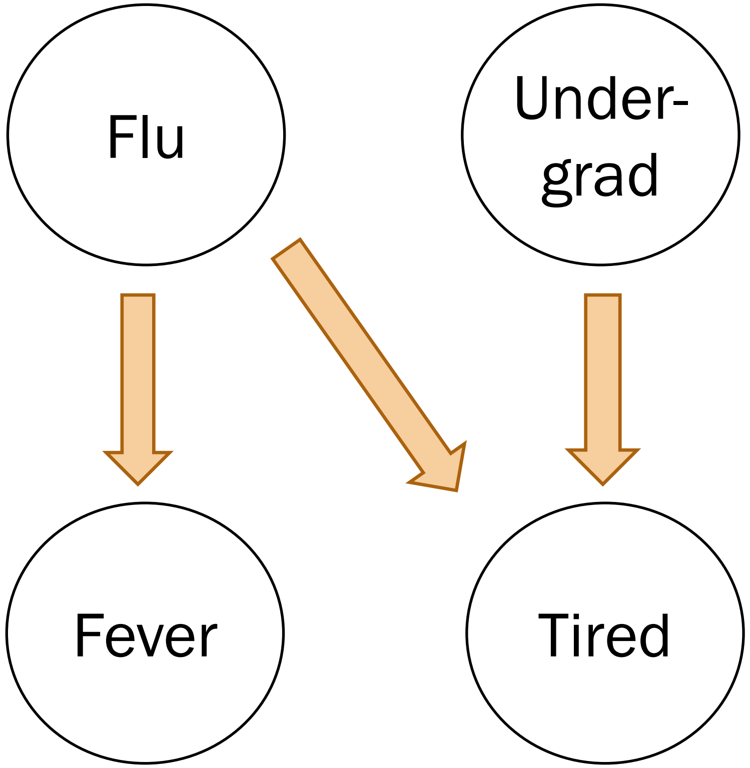
$P(T = 1 | F_{lu} = 0, U = 0) = 0.1$
$P(T = 1 | F_{lu} = 0, U = 1) = 0.8$
$P(T = 1 | F_{lu} = 1, U = 0) = 0.9$
$P(T = 1 | F_{lu} = 1, U = 1) = 1.0$

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

   Compute joint probabilities using chain rule.

Inference via math

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



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

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

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

1. Compute joint probabilities

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

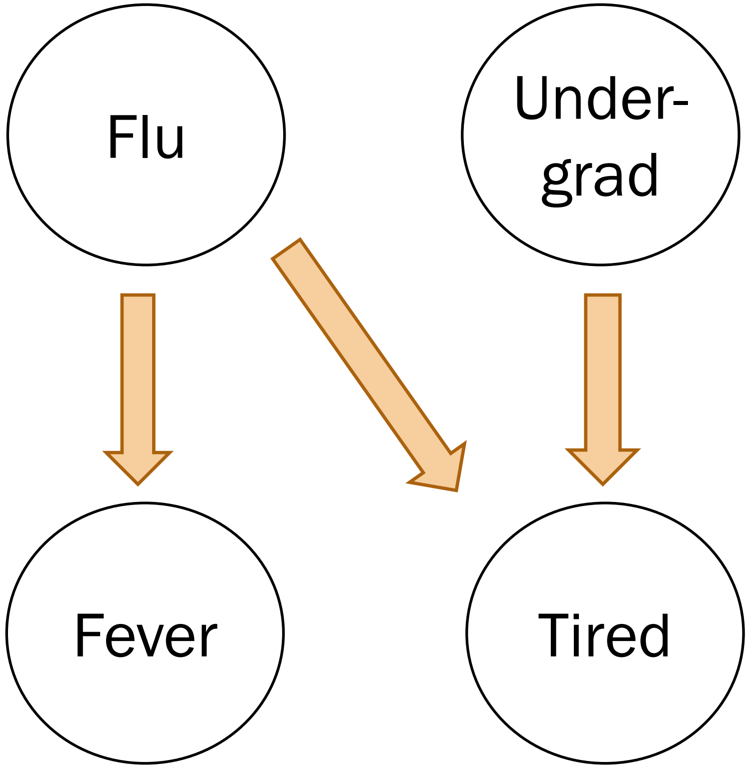
2. Definition of conditional probability

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

$$=0.095$$

Inference via math

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



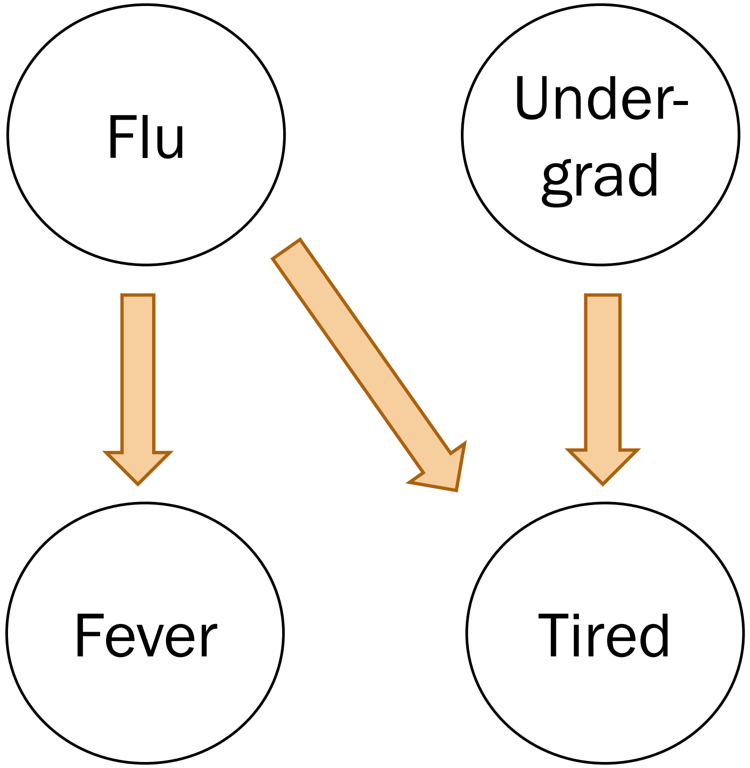
$P(F_{ev} = 1 | F_{lu} = 1) = 0.9$
$P(F_{ev} = 1 | F_{lu} = 0) = 0.05$

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

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

Inference via math

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



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

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

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

   1. Compute joint probabilities

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

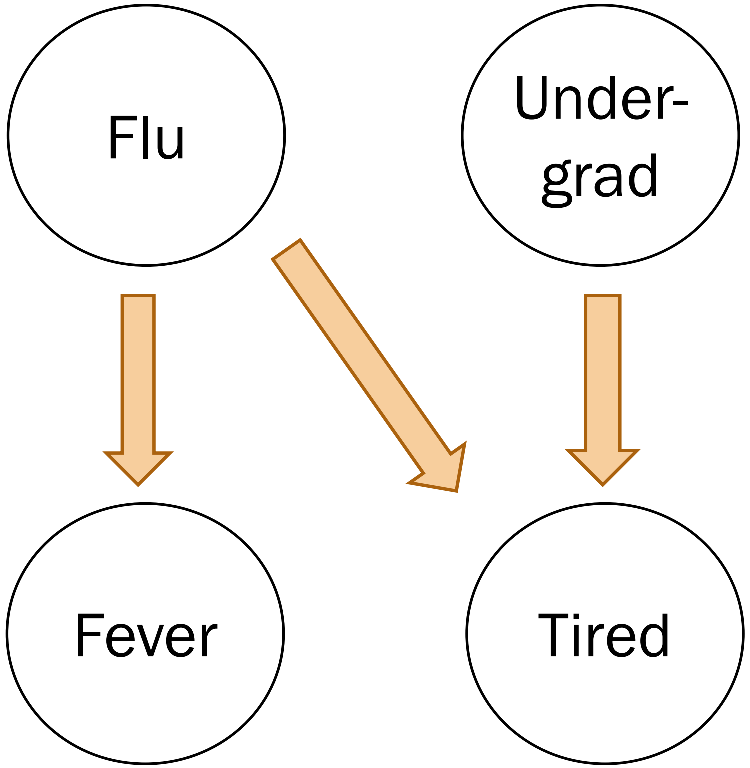
   2. Definition of conditional probability

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

$$= 0.122$$

Rejection sampling algorithm

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



Step 0:

Have a fully specified Bayesian Network

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

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

Alg #0: Straight Math

Too many possible inference questions one could ask…

Alg #1: Rejection Sampling

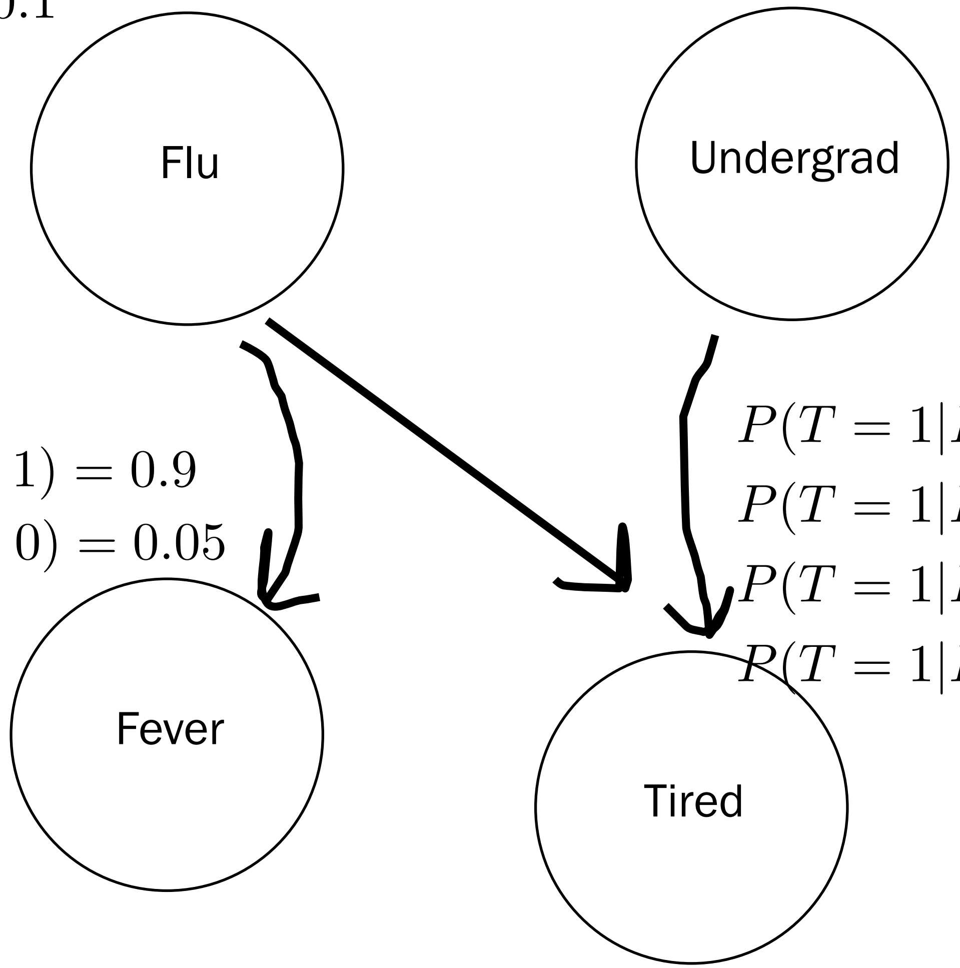
```
N_SAMPLES = 100000

# 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
```

```python
# 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



$$P(Fl = 1) = 0.1$$

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

$$P(Fev = 1|Flu = 1) = 0.9$$
$$P(Fev = 1|Flu = 0) = 0.05$$

$$P(T = 1|Flu = 0, U = 0) = 0.1$$
$$P(T = 1|Flu = 0, U = 1) = 0.8$$
$$P(T = 1|Flu = 1, U = 0) = 0.9$$
$$P(T = 1|Flu = 1, U = 1) = 1.0$$

```python
# Method: Make Sample
# --------------------
# chose a single sample from the joint distribut
# based on the medical "Probabilistic Graphical I
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]
```

```python
# Method: Make Sample
# -------------------
# chose a single sample from the joint distribut
# based on the medical "Probabilistic Graphical
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]
```

```python
# Method: Make Sample
# ------------------
# chose a single sample from the joint distribut
# based on the medical "Probabilistic Graphical 
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]
```

```python
# Method: Make Sample
# -------------------
# chose a single sample from the joint distribut
# based on the medical "Probabilistic Graphical 
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]
```

Alg #1: Rejection Sampling

```
N_SAMPLES = 100000

# Program: Joint Sample
# ----------------------
# we can answer any pro
# with multivariate sam
# where conditioned var
def main():
    obs = getObservatio
    print 'Observation

    samples = sampleATo
    prob = probFluGiven
    print 'Pr(Flu) = ',
```

webMd — -bash — 30×20

```
[0, 1, 0, 1]
[1, 1, 1, 1]
[0, 1, 0, 1]
[0, 1, 0, 0]
[0, 1, 0, 0]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 0, 0, 0]
[0, 0, 0, 0]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 0]
[0, 0, 0, 0]
[0, 1, 0, 1]
[0, 1, 0, 1]
[1, 1, 0, 1]
```

Alg #1: Rejection Sampling

```python
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
```

```python
# Method: Probability of Flu Given Observation
# -------------------------
# Calculate the probability of flu given many
# samples from the joint distribution and a set
# of ovservations 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)
```

```python
# Method: Probability of Flu Given Observation
# ----------------------------
# Calculate the probability of flu given many
# samples from the joint distribution and a set
# of ovservations 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)
```

```python
# Method: Probability of Flu Given Observation
# ---------------------------
# Calculate the probability of flu given many
# samples from the joint distribution and a set
# of ovservations 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)
```

Alg #1: Rejection Sampling

```
N_SAMPLES = 100000

# Program: Joint Sample
# ----------------------
# we can answer any pro
# with multivariate sam
# where conditioned var
def main():
    obs = getObservatio
    print 'Observation 
    
    samples = sampleATo
    prob = probFluGiven
    print 'Pr(Flu) = ',
```

webMd — -bash — 30×20

```
[0, 1, 0, 1]
[0, 1, 0, 0]
[0, 1, 0, 0]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 0, 0, 0]
[0, 0, 0, 0]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 0]
[0, 0, 0, 0]
[0, 1, 0, 1]
[0, 1, 0, 1]
[1, 1, 0, 1]
Pr(Flu) =  0.141503173687
> 
```

Lets try it!

BACK TO THE CODE

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

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

Recall our definition of probability as a frequency:

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

n = # of total trials

$n(E)$ = # trials where E occurs

webMd — -bash — 39×20

```
[0, 1, 1, 0]
[1, 0, 1, 1]
[0, 1, 0, 1]
[0, 1, 0, 0]
[0, 1, 0, 0]
[0, 1, 1, 0]
[1, 1, 1, 1]
[0, 1, 0, 0]
[0, 0, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 1]
[0, 1, 0, 0]
[0, 1, 0, 1]
[0, 1, 0, 0]
[0, 0, 0, 0]
[0, 0, 0, 1]
Observation =  [None, None, None, None]
Pr(Flu | Obs) =  0.10164
> 
```

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]

Where are we in CS109?

Which video are you more likely to like?

Davie504

10,000 50

Not Davie504

10 0

Which drug should you give if you are uncertain about *p*?

Drug A

Drug B

Which one do you give to a patient?

Philosophical Ponderings:
You ask about the probability of rain tomorrow.

Person A: My leg itches when it rains and its kind of itchy…. Uh, p = .80

Person B: I have done complex calculations and have seen 10,451 days like tomorrow… p = 0.80

What is the difference between the two estimates?

“Those who are able to represent what they do not know make better decisions”

- CS109

Today we are going to learn something unintuitive, beautiful and useful

Let's play a game!

Flip a plate 5 times. If you get heads 3 times you win

Credit: Rembrandt via Dall E

$$P(X = 3) = \binom{5}{3} \cdot \frac{1}{2}^3 \cdot \frac{1}{2}^2$$
$$= 0.3125$$

What if you don't know a probability?

What if you don't know a probability?

What is your belief that you flip a heads on my coin?

The parameter p to a binomial can be a random variable

Coffee out of 10 Bevs……What is your Belief of the P(Coffee)?

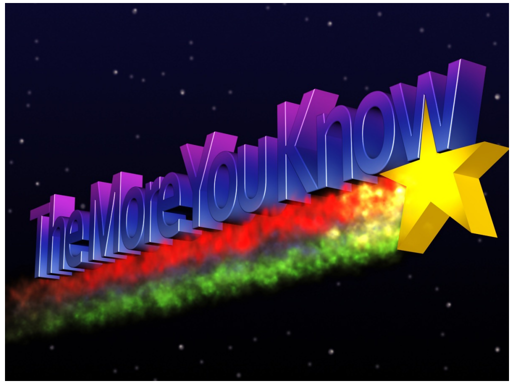












9 Heads out of 10 Flips. What is your Belief in p?

9 Heads out of 10 Flips. What is your Belief in p?

Let X be our belief about the probability of heads:

Binomial

$$\overset{f}{\cancel{P}}(X = x | H = 9, T = 1) = \frac{\boxed{P(H = 9, T = 1 | X = x)}\,\boxed{f(X = x)}}{P(H = 9, T = 1)}$$

Uniform?

9 Heads out of 10 Flips. What is your Belief in p?

Let X be our belief about the probability of heads:

$$\cancel{P}\, f(X = x | H = 9, T = 1)$$

Binomial

$$= \frac{\boxed{P(H = 9, T = 1 | X = x)}\,\boxed{f(X = x)}}{P(H = 9, T = 1)}$$

Uniform?

$$= \frac{\binom{10}{9} x^9 (1 - x)^1}{P(H = 9, T = 1)}$$

9 Heads out of 10 Flips. What is your Belief in p?

Let X be our belief about the probability of heads:

$$\cancel{P}f(X = x | H = 9, T = 1)$$

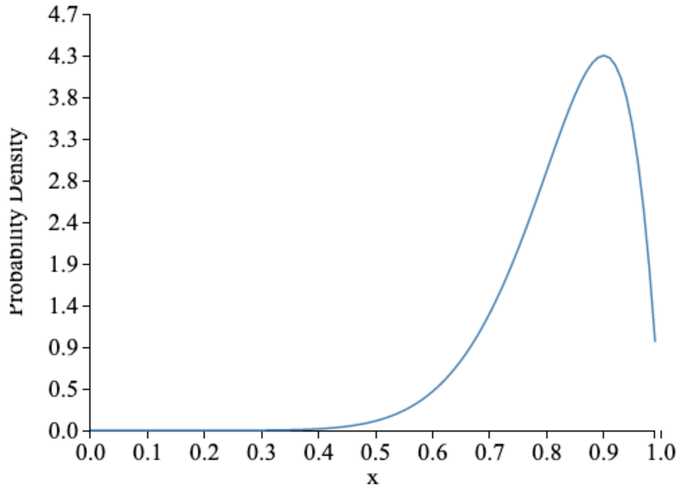
Binomial

$$= \frac{\boxed{P(H = 9, T = 1 | X = x)}\boxed{f(X = x)}}{P(H = 9, T = 1)}$$

Uniform?

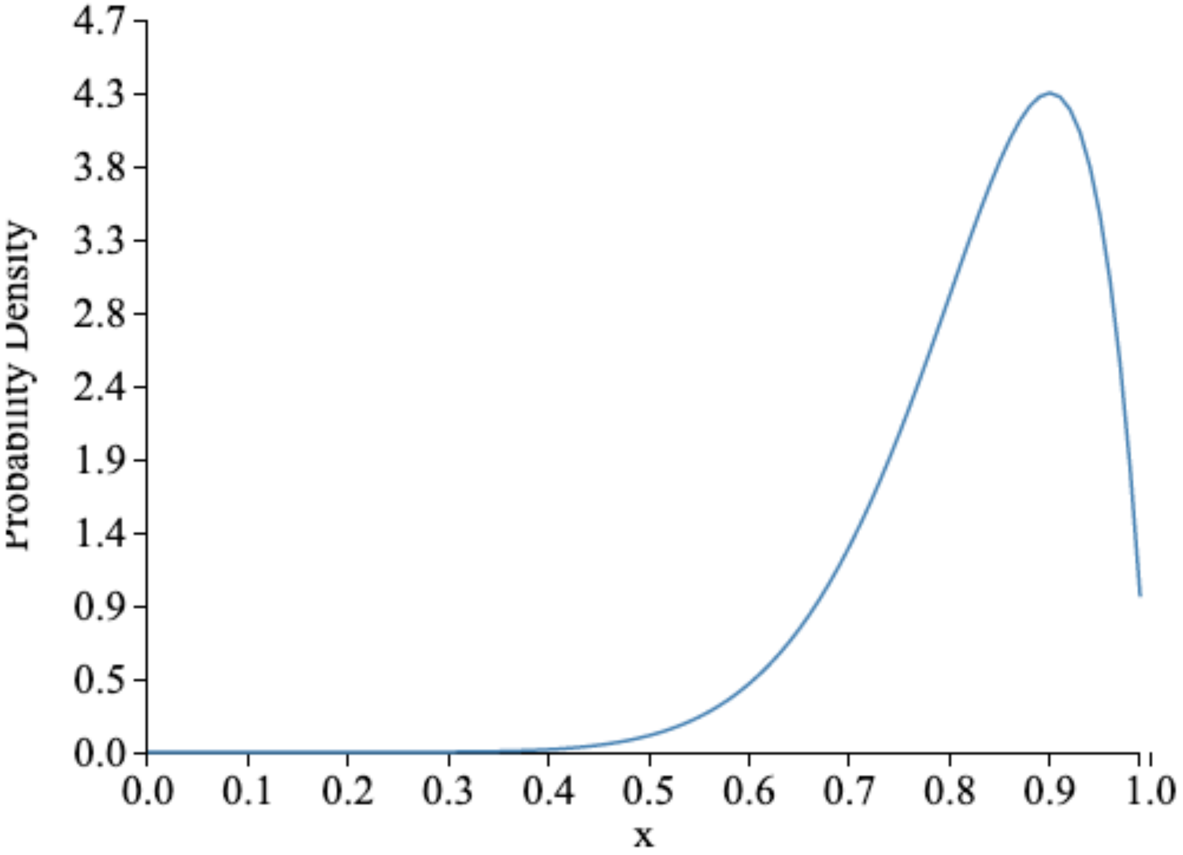
$$= \frac{\binom{10}{9} x^9 (1 - x)^1}{P(H = 9, T = 1)}$$

$$= K \cdot x^9 (1 - x)^1$$



9 Heads out of 10 Flips. What is your Belief in p?

$$\cancel{P}\,f(X = x | H = 9, T = 1)$$



Flip a coin with unknown probability

Flip a coin (n + m) times, comes up with n heads

- We don't know probability X that coin comes up heads

Frequentist (never prior)

$$X = \lim_{n+m\to\infty} \frac{n}{n+m}$$
$$\approx \frac{n}{n+m}$$

X is (often) a single value

Bayesian (prior is great)

$$f_{X|N}(x|n) = \frac{P(N=n|X=x) f_X(x)}{P(N=n)}$$

X is a random variable. Leads to a belief distribution which captures confidence

Flip a coin with unknown probability!

Flip a coin (n + m) times, comes up with n heads

- We don't know probability X that coin comes up heads
- Our belief before flipping coins is that: X ~ Uni(0, 1)
- Let N = number of heads
- Given X = x, coin flips independent: (N | X) ~ Bin(n + m, x)

$$f_{X|N}(x|n) = \frac{P(N = n|X = x) f_X(x)}{P(N = n)}$$

Bayesian "posterior" probability distribution

Bayesian "prior" probability distribution

Flip a coin with unknown probability!

Flip a coin (n + m) times, comes up with n heads

- We don't know probability X that coin comes up heads
- Our belief before flipping coins is that: X ~ Uni(0, 1)
- Let N = number of heads
- Given X = x, coin flips independent: (N | X) ~ Bin(n + m, x)

$$f_{X|N}(x|n) = \frac{\boxed{P(N = n|X = x)}\boxed{f_X(x)}}{P(N = n)}$$

1

Binomial

$$= \frac{\binom{n+m}{n} x^n (1-x)^m}{P(N = n)}$$

Move terms around

$$= \frac{\binom{n+m}{n}}{P(N = n)} x^n (1-x)^m$$

$$= \frac{1}{c} \cdot x^n (1-x)^m \quad \text{where } c = \int_0^1 x^n (1-x)^m \, dx$$

Flip a coin with unknown probability!

If you start with a X ~ Uni(0, 1) prior over probability, and observe:

n "successes" and

m "failures"…

Your new belief about the probability is:

$$f_X(x) = \frac{1}{c} \cdot x^n (1-x)^m$$

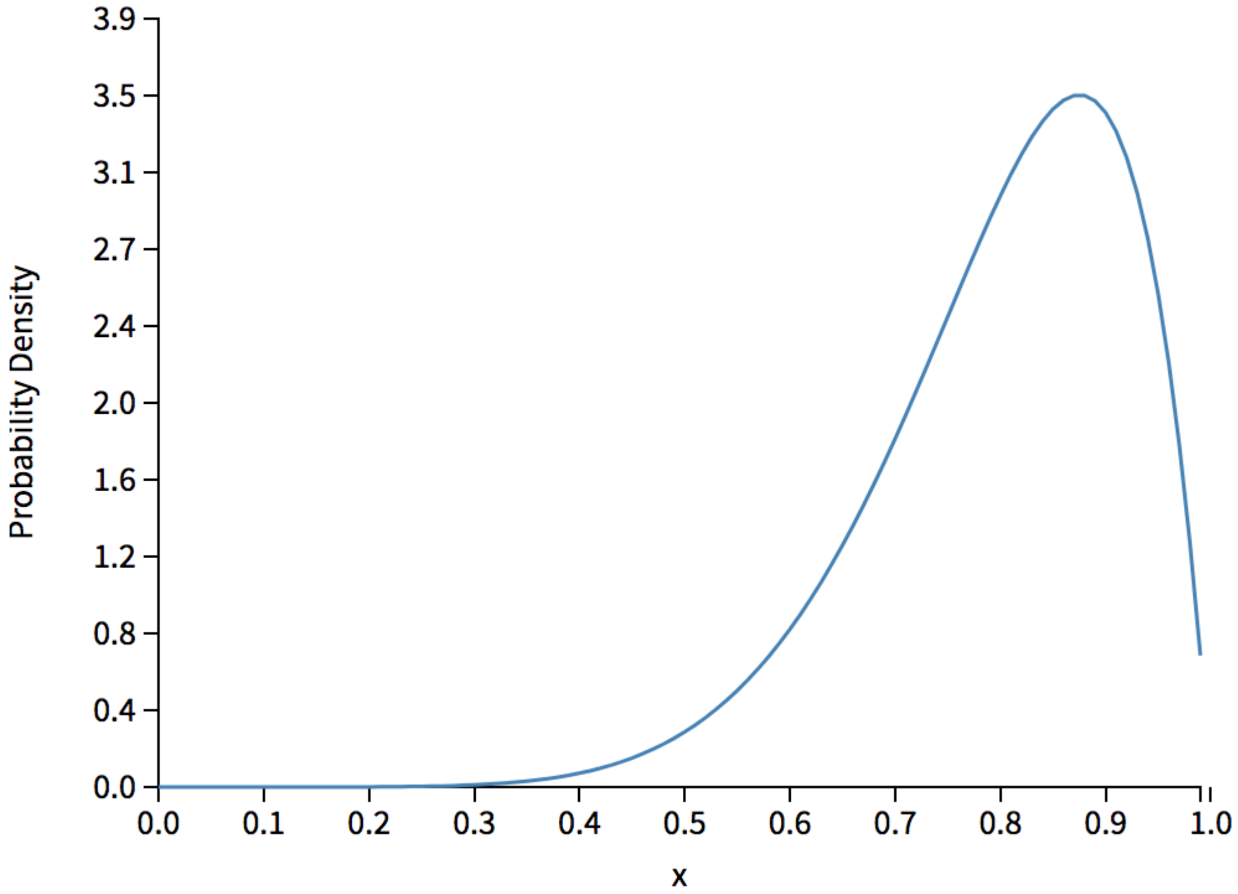
where $c = \int_0^1 x^n (1-x)^m$

Belief after 7 success and 1 fail

$$f_X(x) = \frac{1}{c} \cdot x^n (1-x)^m$$

$n = 7$

$m = 1$



Equivalently!

If you start with a $X \sim \text{Uni}(0, 1)$ prior over probability, and observe:

let a = num "successes" + 1

let b = num "failures" + 1

Your new belief about the probability is:

$$f_X(x) = \frac{1}{c} \cdot x^{a-1}(1-x)^{b-1}$$

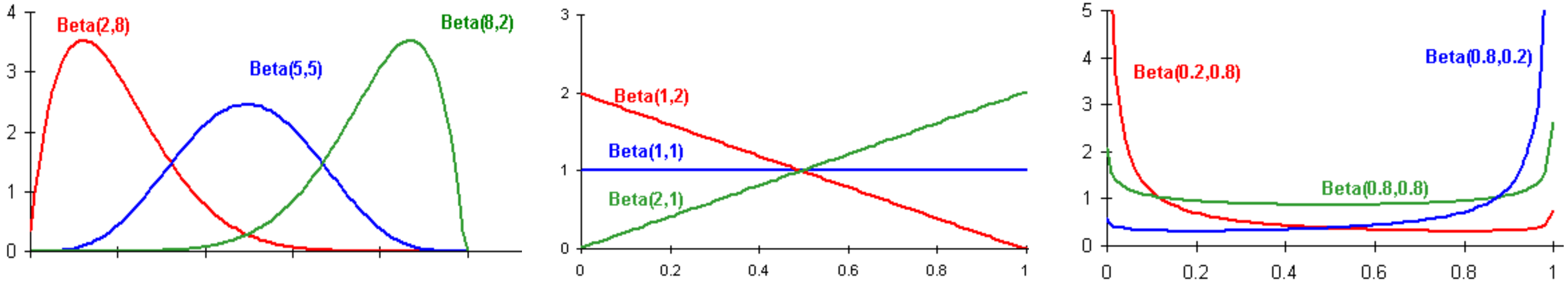
where $$c = \int_0^1 x^{a-1}(1-x)^{b-1}$$

Beta Random Variable

X is a **Beta Random Variable**: X ~ Beta(*a*, *b*)

- Probability Density Function (PDF): (where *a*, *b* > 0)

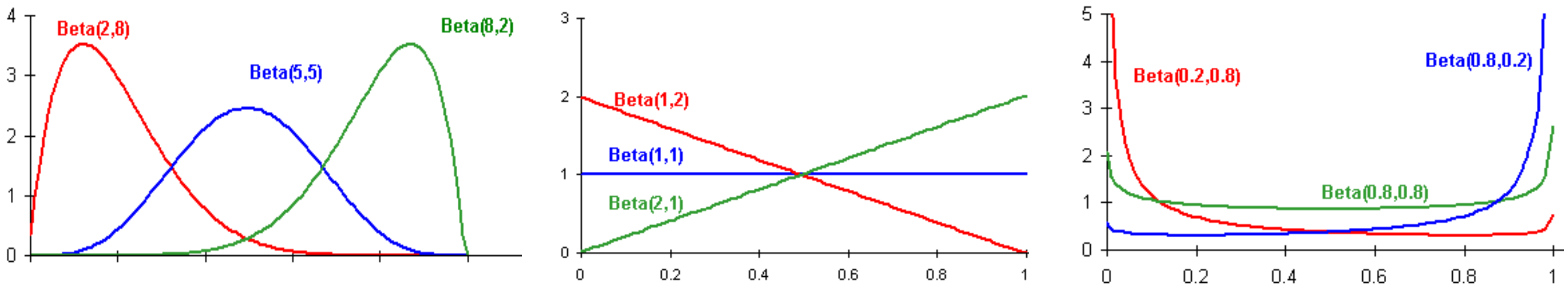
$$f(x) = \begin{cases} \frac{1}{B(a,b)} x^{a-1}(1-x)^{b-1} & 0 < x < 1 \\ 0 & \text{otherwise} \end{cases} \qquad B(a,b) = \int_0^1 x^{a-1}(1-x)^{b-1}\,dx$$



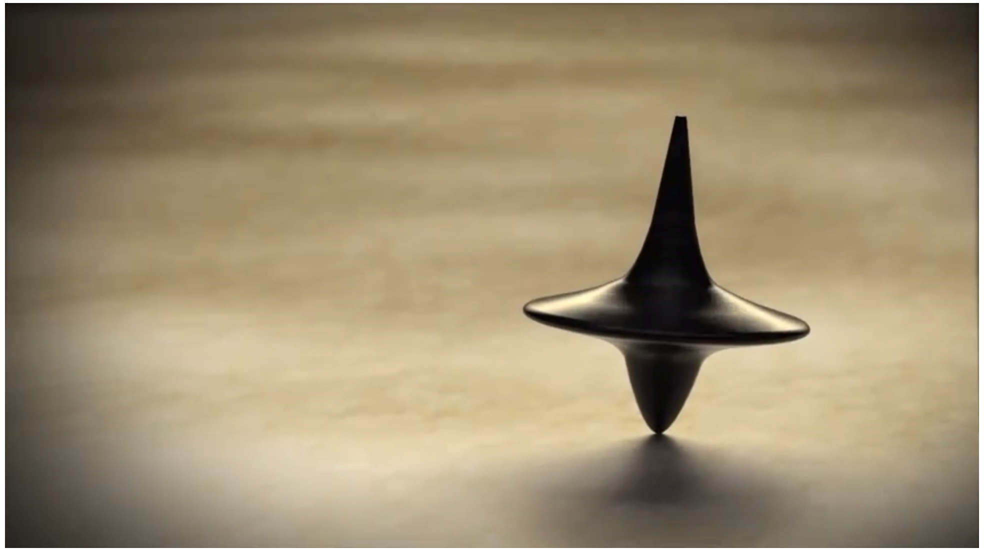
- Symmetric when *a* = *b*

$$E[X] = \frac{a}{a+b} \qquad Var(X) = \frac{ab}{(a+b)^2(a+b+1)}$$

Beta is the Random Variable for Probabilities



Used to represent a distributed belief of a probability



Beta Parameters *can* come from experiments:

a = “successes” + 1

b = “failures” + 1

And with that, we’ll end for today!

See you Friday!