Bayesian Networks

George Konidaris
gdk@cs.brown.edu

Fall 2021

Recall

Joint distributions:

- $P(X_1, \ldots, X_n)$.
- *All you (statistically) need to know* about $X_1 \ldots X_n$.
- From it you can infer $P(X_1)$, $P(X_1 \mid Xs)$, etc.

Raining	Cold	Prob.
True	True	0.3
True	False	0.1
False	True	0.4
False	False	0.2

Joint Distributions Are Useful

Classification

- $P(X_1 \mid X_2 \ldots X_n)$ — things you know

thing you want to know

Co-occurrence

- $P(X_a, X_b)$

how likely are these two things together?

Rare event detection

- $P(X_1, \ldots, X_n)$

Independence

If independent, can break JPD into separate tables.

P(A, B) = P(A)P(B)

Raining	**Prob.**
True	0.6
False	0.4

Cold	**Prob.**
True	0.75
False	0.25

x

Raining	**Cold**	**Prob.**
True	True	0.45
True	False	0.15
False	True	0.3
False	False	0.1

Conditional Independence

A and B are **conditionally independent given C** if:

- P(A | B, C) = P(A | C)
- P(A, B | C) = P(A | C) P(B | C)

(recall independence: P(A, B) = P(A)P(B))

This means that, ***if we know C***, we can treat A and B ***as if they were independent***.

A and B might not be independent otherwise!

Example

Consider 3 RVs:

- Temperature
- Humidity
- Season

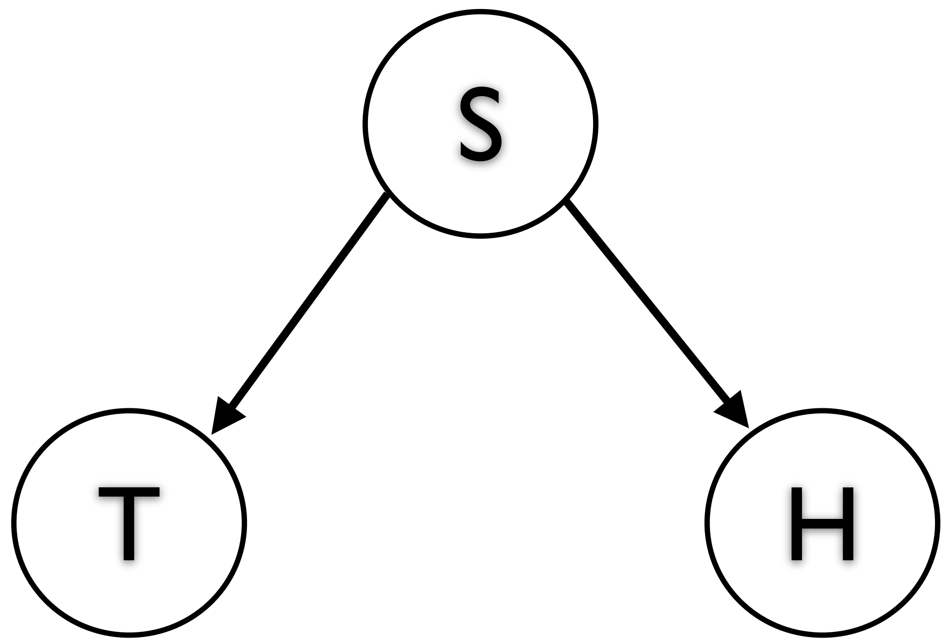
Temperature and humidity are not independent.

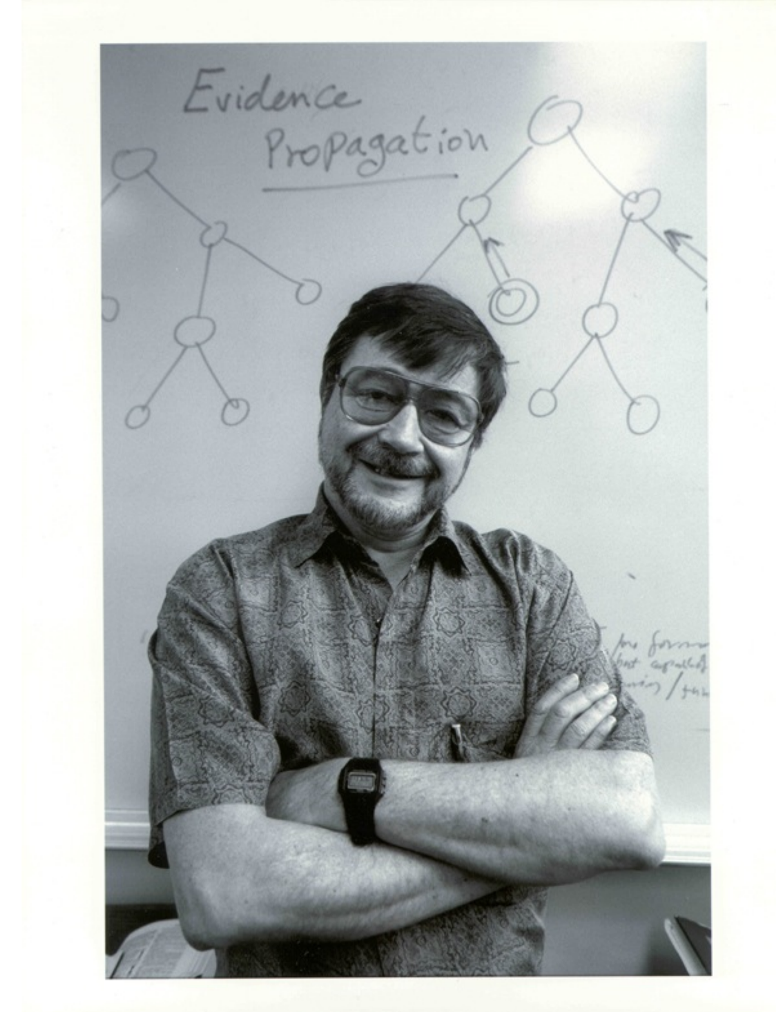
But, they might be, given the season: *the season explains both*, and they become independent of each other.

Bayes Nets

A particular type of graphical model:

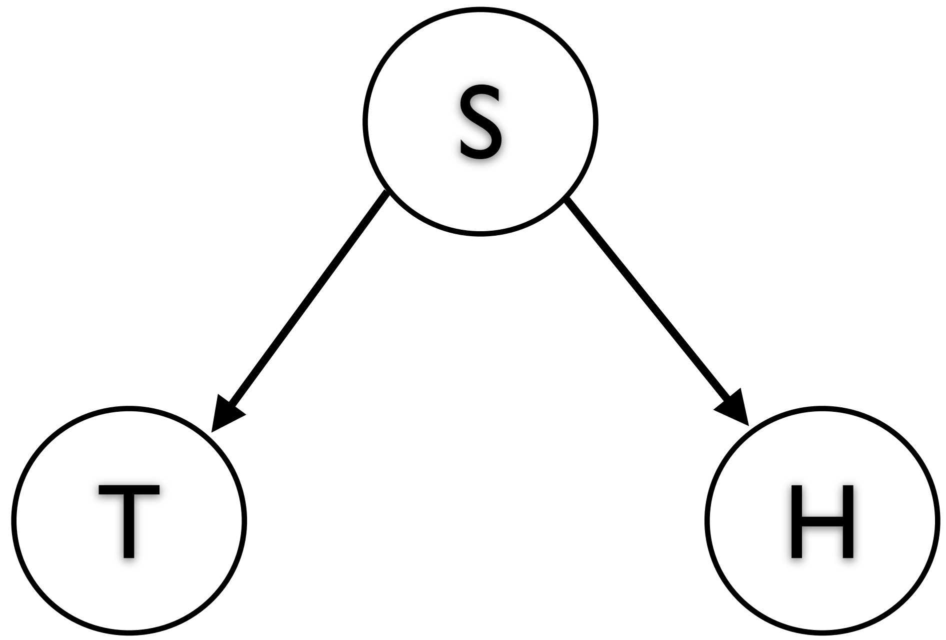
- A directed, acyclic graph.
- A node for each RV.





Given parents, each RV independent of non-descendants.

Bayes Net



JPD decomposes:

$$P(x_1, ..., x_n) = \prod_i P(x_i | \text{parents}(x_i))$$

So for each node, store *conditional probability table* (CPT):

$$P(x_i | \text{parents}(x_i))$$

CPTs

Conditional Probability Table

- Probability distribution over variable given parents.
- One distribution per setting of parents.

conditioning variables

variable of interest

X	Y	Z	P
True	True	True	0.7
False	True	True	0.3
True	True	False	0.2
False	True	False	0.8
True	False	True	0.5
False	False	True	0.5
True	False	False	0.4
False	False	False	0.6

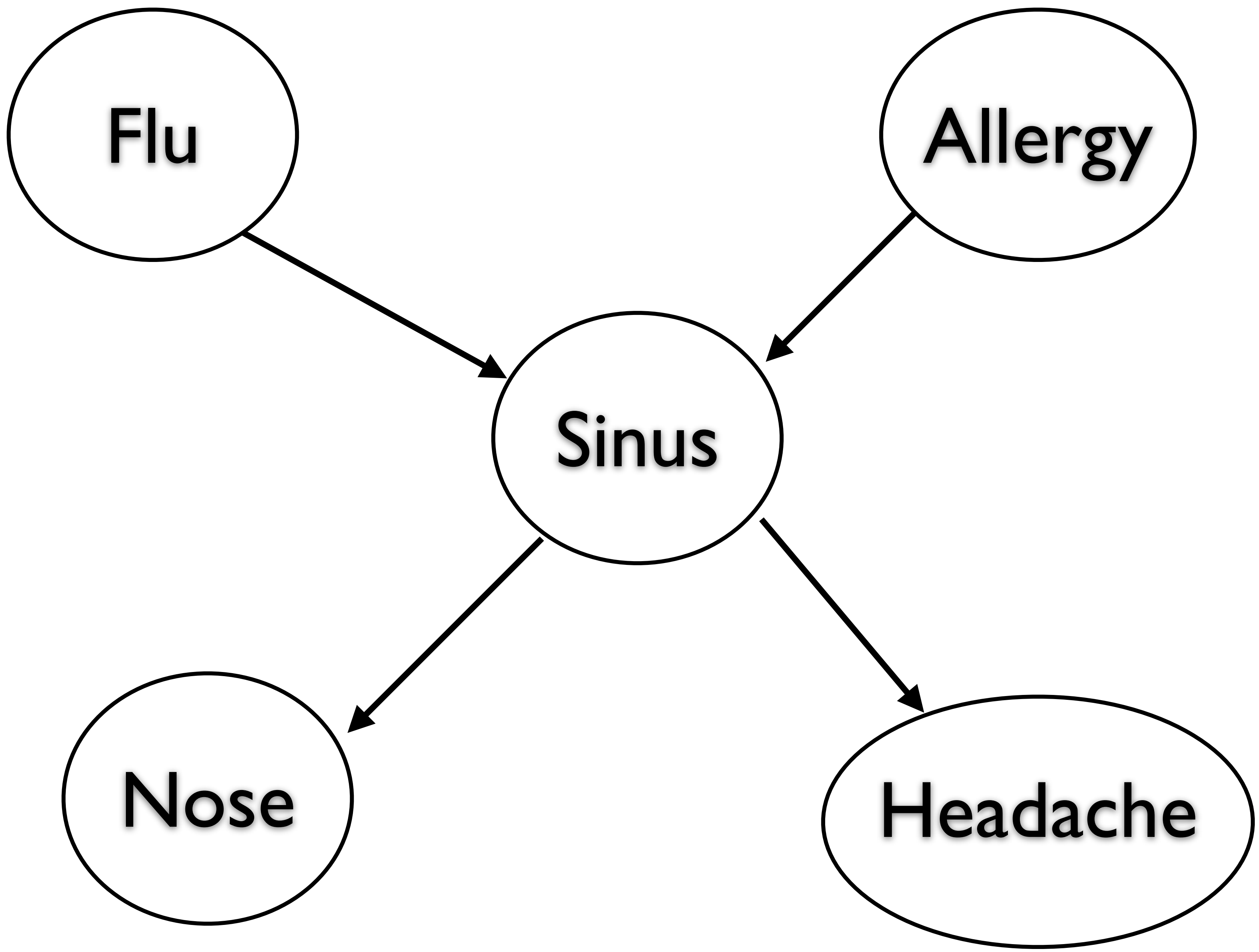
distributions (sum to 1)

Example

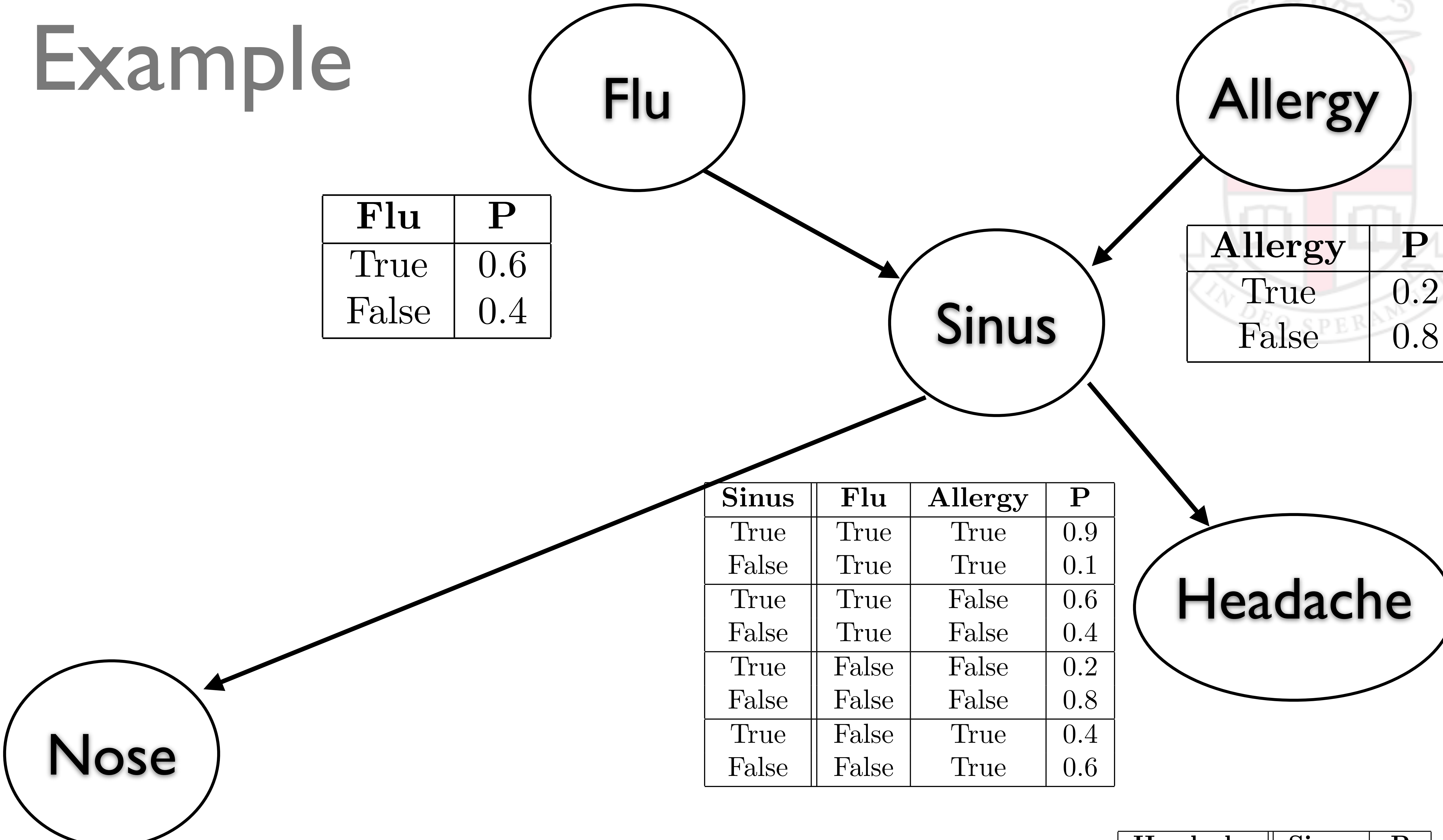
Suppose we know:

- The flu causes sinus inflammation.
- Allergies cause sinus inflammation.
- Sinus inflammation causes a runny nose.
- Sinus inflammation causes headaches.

Example



Example



Flu	P
True	0.6
False	0.4

Allergy	P
True	0.2
False	0.8

Sinus	Flu	Allergy	P
True	True	True	0.9
False	True	True	0.1
True	True	False	0.6
False	True	False	0.4
True	False	False	0.2
False	False	False	0.8
True	False	True	0.4
False	False	True	0.6

Headache	Sinus	P
True	True	0.6
False	True	0.4
True	False	0.5
False	False	0.5

Nose	Sinus	P
True	True	0.8
False	True	0.2
True	False	0.3
False	False	0.7

joint: 32 (31) entries

Uses

Things you can do with a Bayes Net:

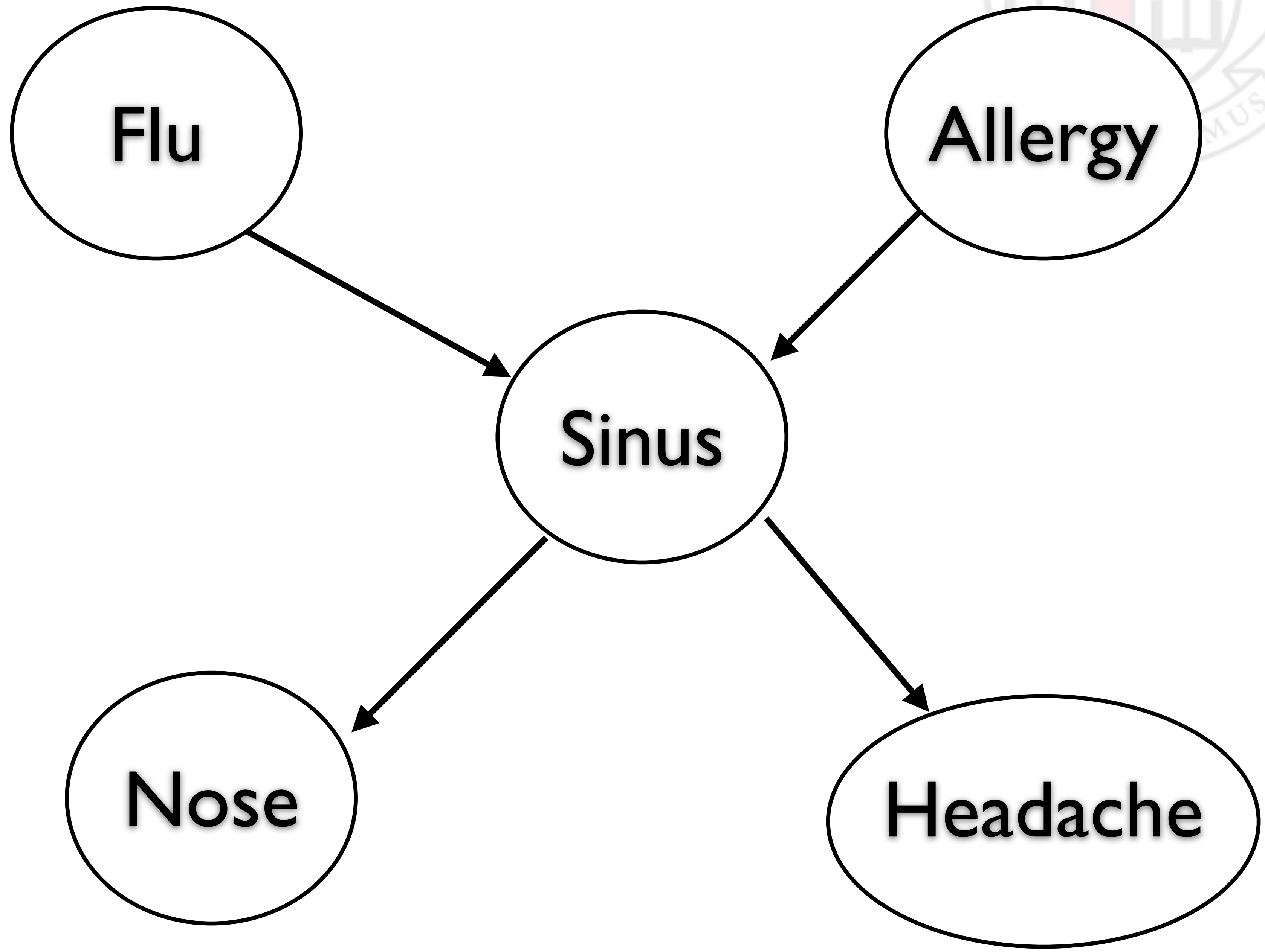
- Inference: given some variables, posterior?
  - (**might be intractable**: NP-hard)
- Learning (fill in CPTs)
- Structure Learning (fill in edges)

Generally:

- Often few parents.
- Inference cost often reasonable.
- Can include domain knowledge.

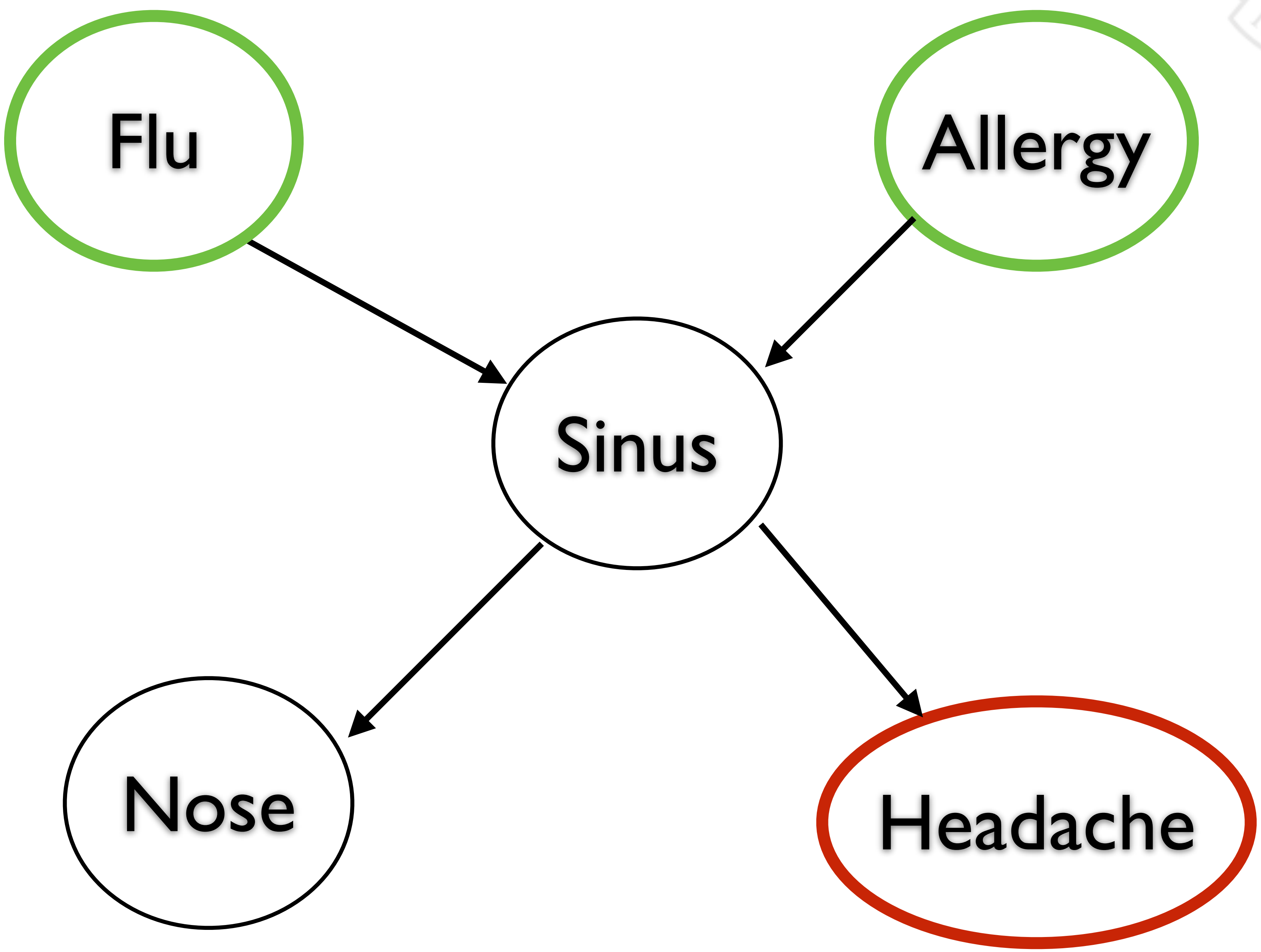
Inference

What is:
P(f | h)?



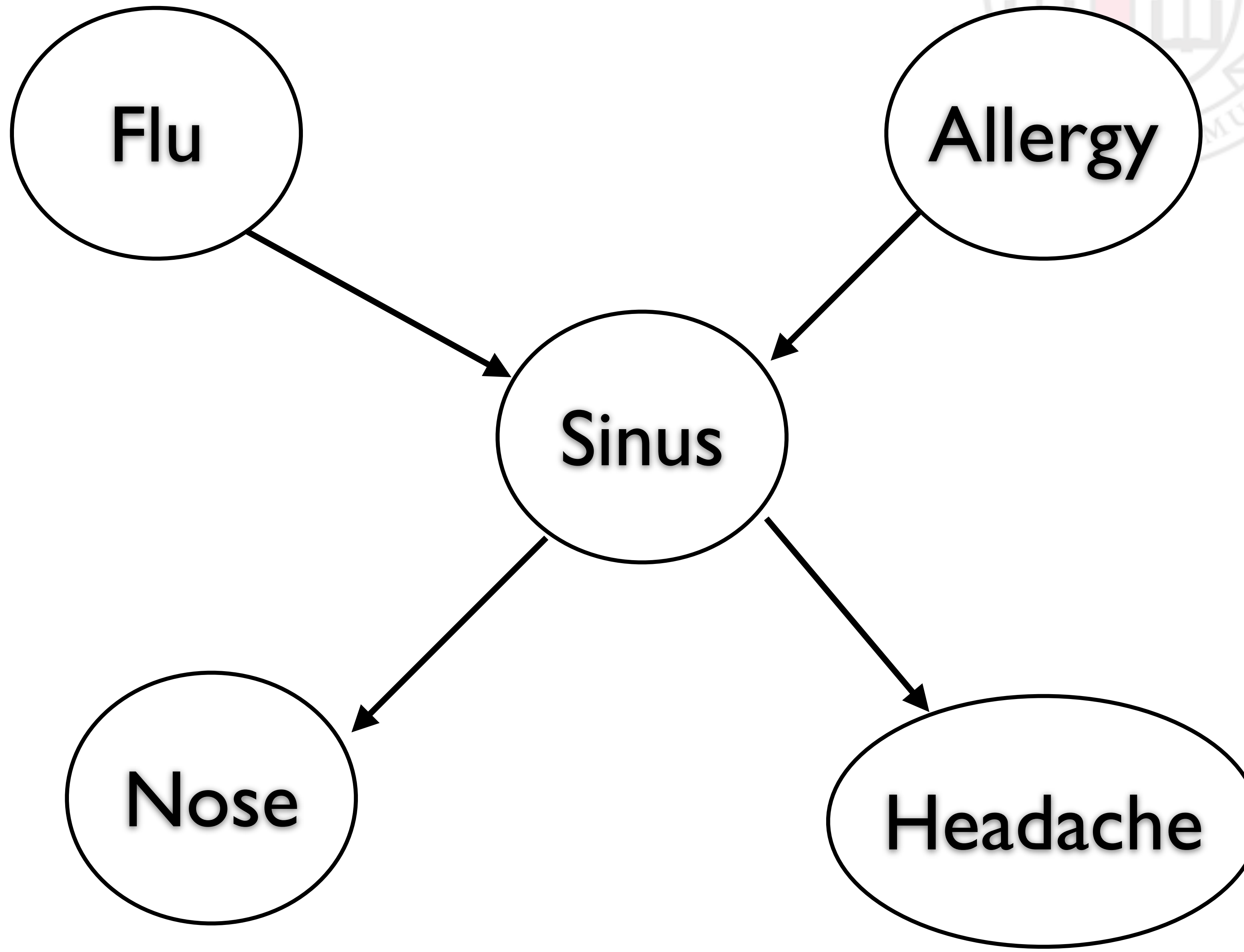
Inference

Given A compute P(B | A).



Inference

What is:
P(F=True | H=True)?



Inference

$$P(f|h) = \frac{P(f,h)}{P(h)} = \frac{\sum_{SAN} P(f,h,S,A,N)}{\sum_{SANF} P(h,S,A,N,F)}$$

identity

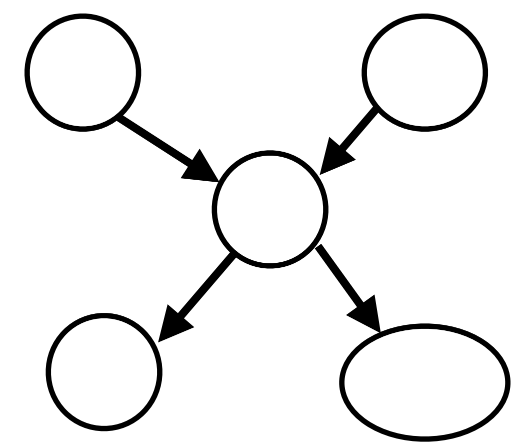
$$P(a) = \sum_{B=T,F} P(a,B)$$

$$P(a) = \sum_{B=T,F} \sum_{C=T,F} P(a,B,C)$$

Inference

$$P(f|h) = \frac{P(f,h)}{P(h)} = \frac{\sum_{SAN} P(f,h,S,A,N)}{\sum_{SANF} P(h,S,A,N,F)}$$

We know from definition of Bayes net:



$$P(h) = \sum_{SANF} P(h,S,A,N,F)$$

$$P(h) = \sum_{SANF} P(h|S)P(N|S)P(S|A,F)P(F)P(A)$$

Variable Elimination

So we have:

$$P(h) = \sum_{SANF} P(h|S)P(N|S)P(S|A,F)P(F)P(A)$$

… we can *eliminate variables* one at a time:
(distributive law)

$$P(h) = \sum_{SN} P(h|S)P(N|S) \sum_{AF} P(S|A,F)P(F)P(A)$$

$$P(h) = \sum_{S} P(h|S) \sum_{N} P(N|S) \sum_{AF} P(S|A,F)P(F)P(A)$$

Variable Elimination

$$P(h) = \sum_{S} P(h|S) \sum_{N} P(N|S) \sum_{AF} P(S|A,F)P(F)P(A)$$

sinus = true

$$0.6 \times \sum_{N} P(N|S=True) \sum_{AF} P(S=True|A,F)P(F)P(A) +$$

$$0.5 \times \sum_{N} P(N|S=False) \sum_{AF} P(S=False|A,F)P(F)P(A)$$

sinus = false

Headache	Sinus	P
True	True	0.6
False	True	0.4
True	False	0.5
False	False	0.5

Variable Elimination

$$P(h) = \sum_{S} P(h|S) \sum_{N} P(N|S) \sum_{AF} P(S|A,F)P(F)P(A)$$

$$0.6 \times [0.8 \times \sum_{AF} P(S = True|A,F)P(F)P(A) +$$

$$0.2 \times \sum_{AF} P(S = True|A,F)P(F)P(A)] +$$

$$0.5 \times [0.3 \times \sum_{AF} P(S = False|A,F)P(F)P(A) +$$

$$0.7 \times \sum_{AF} P(S = False|A,F)P(F)P(A)]$$

Nose	Sinus	P
True	True	0.8
False	True	0.2
True	False	0.3
False	False	0.7

Variable Elimination

Downsides:

- How to simplify? (Hard in general.)
- Computational complexity
- Hard to parallelize

Alternative

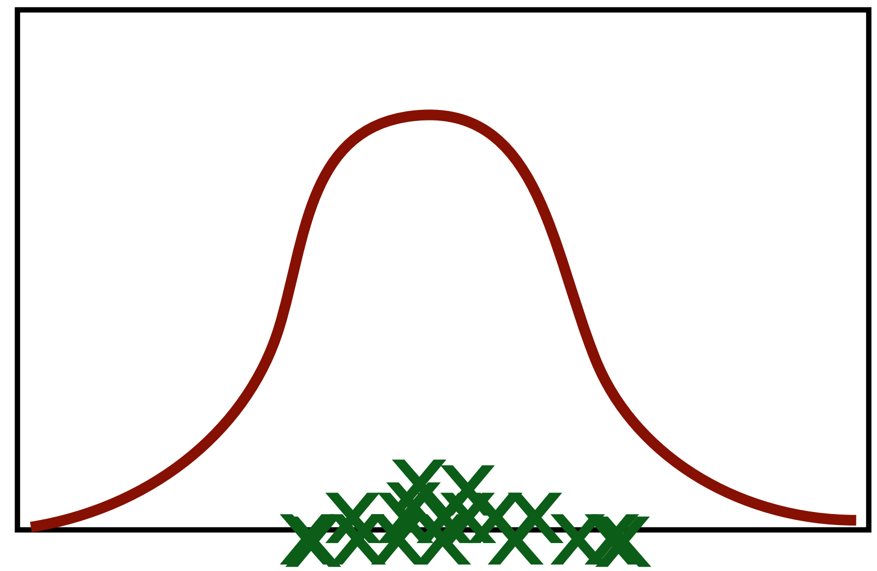
Sampling approaches

- Based on drawing random numbers
- Computationally expensive, but *easy to code!*
- Easy to parallelize

Sampling

What's a sample?

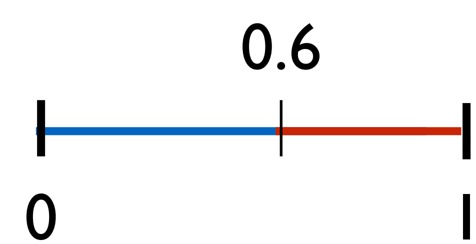
From a distribution:



From a CPT:

Flu	P
True	0.6
False	0.4

F=True
F=True
F=False
F=False
F=True



Generative Models

How do we sample from a Bayes Net?

A Bayes Net is known as a ***generative model***.

Describe a generative process for the data.

- Each variable is generated by a distribution.
- *Describes the structure of that generation.*
- Can generate more data.

Natural way to include domain knowledge via causality.

Sampling the Joint

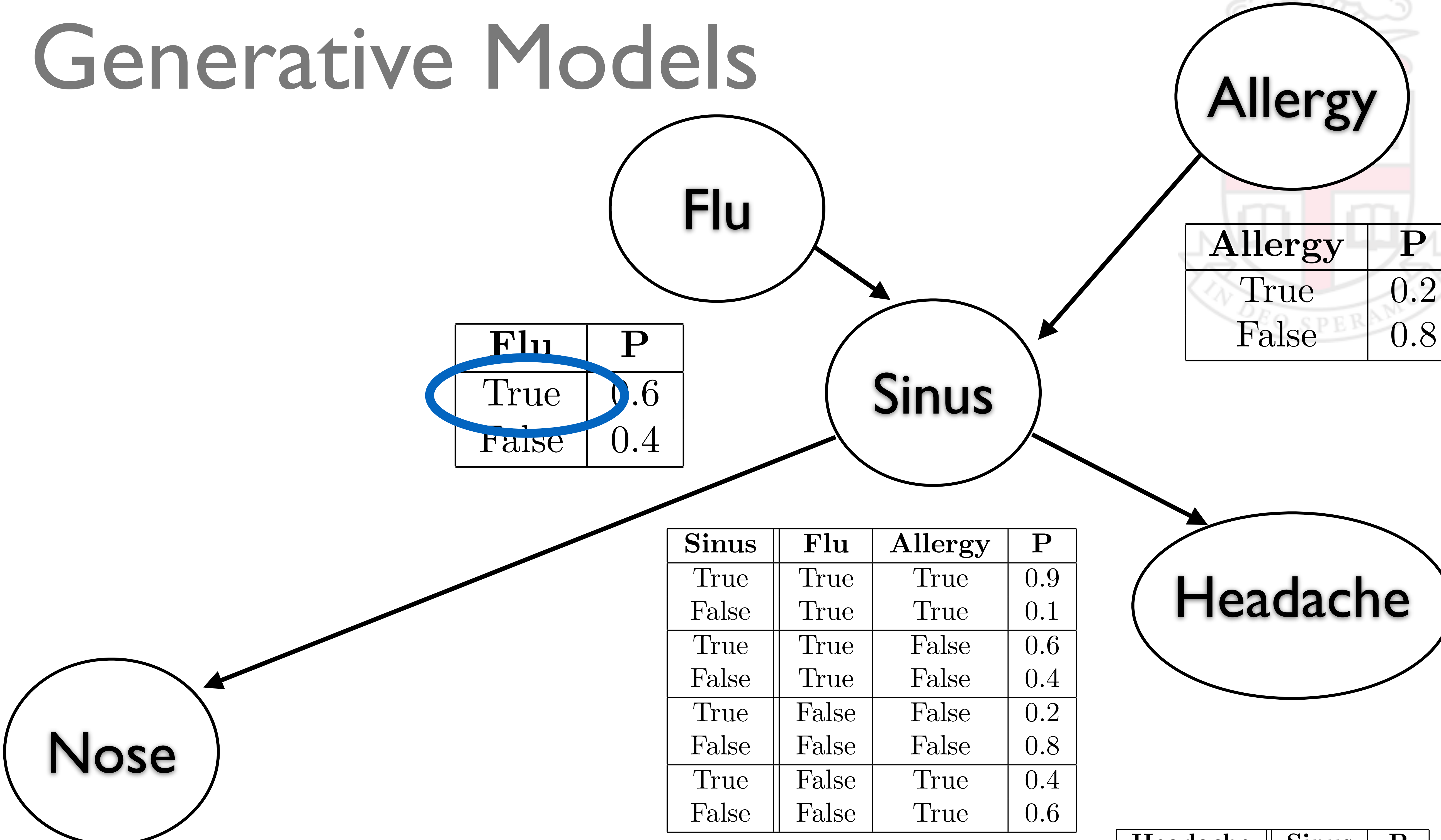
Algorithm for generating samples drawn from the joint distribution:

For each node with no parents:

- Draw sample from marginal distribution.
- Condition children on choice (removes edge)
- Repeat.

Results in artificial data set.
Probability values - *literally just count.*

Generative Models



Allergy	P
True	0.2
False	0.8

Flu	P
True	0.6
False	0.4

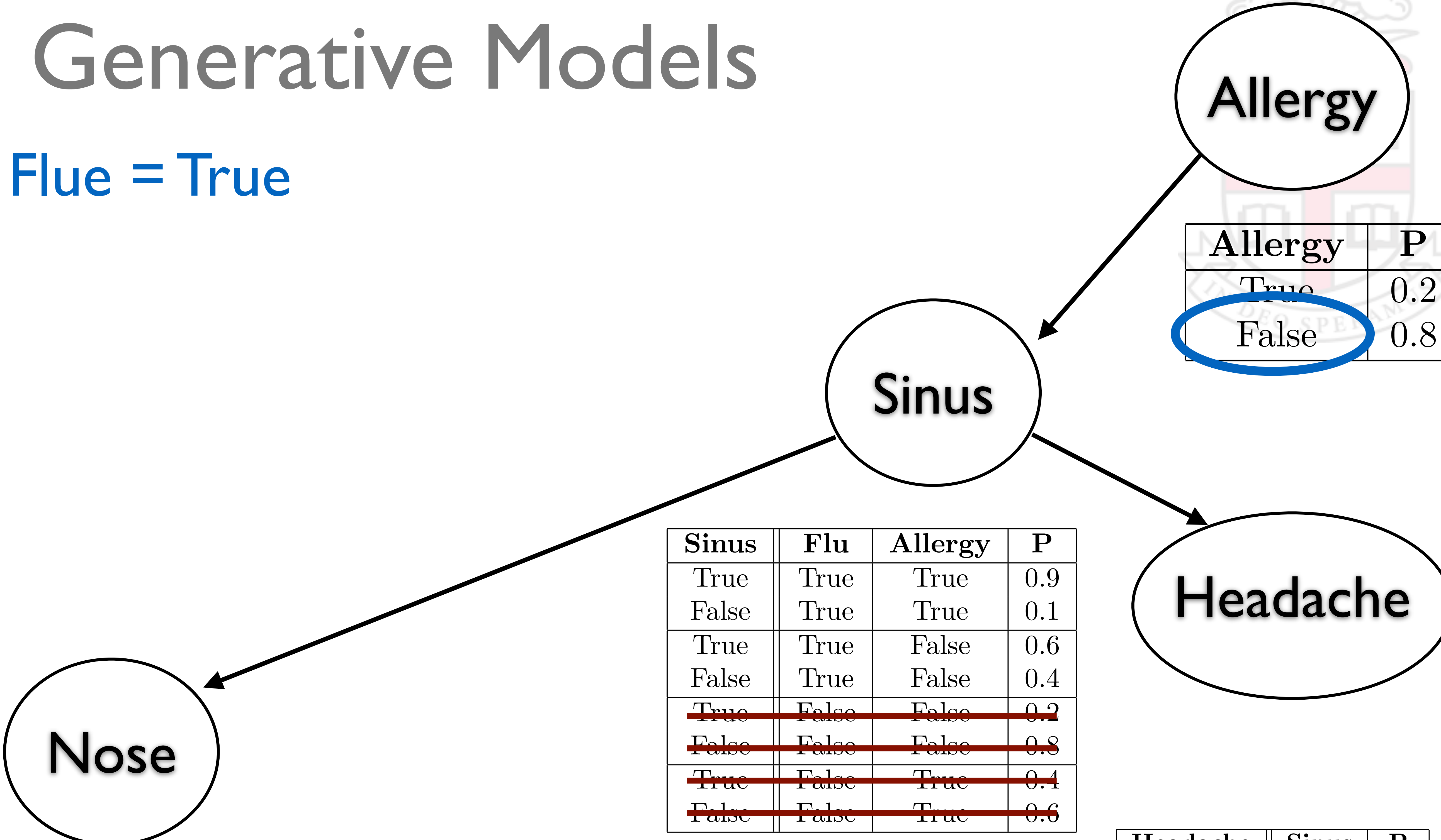
Sinus	Flu	Allergy	P
True	True	True	0.9
False	True	True	0.1
True	True	False	0.6
False	True	False	0.4
True	False	False	0.2
False	False	False	0.8
True	False	True	0.4
False	False	True	0.6

Headache	Sinus	P
True	True	0.6
False	True	0.4
True	False	0.5
False	False	0.5

Nose	Sinus	P
True	True	0.8
False	True	0.2
True	False	0.3
False	False	0.7

Generative Models

Flue = True



Allergy	P
True	0.2
False	0.8

Sinus	Flu	Allergy	P
True	True	True	0.9
False	True	True	0.1
True	True	False	0.6
False	True	False	0.4
~~True~~	~~False~~	~~False~~	~~0.2~~
~~False~~	~~False~~	~~False~~	~~0.8~~
~~True~~	~~False~~	~~True~~	~~0.4~~
~~False~~	~~False~~	~~True~~	~~0.6~~

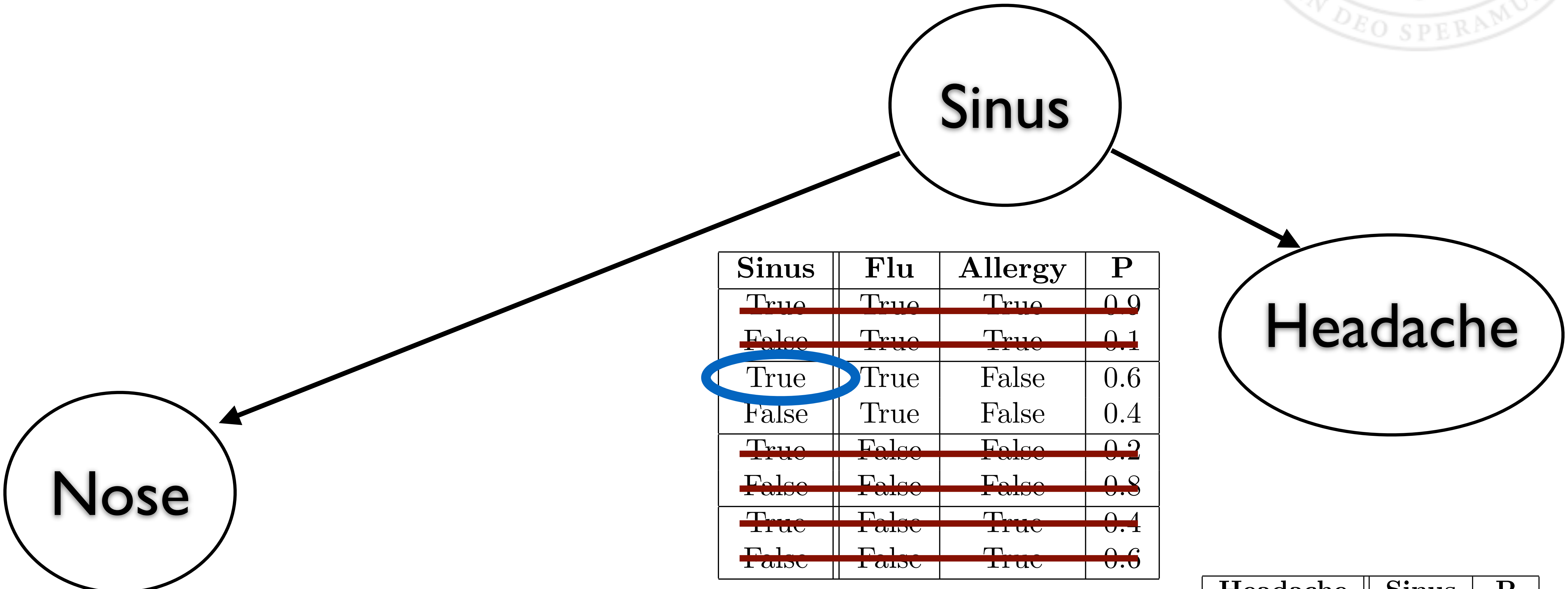
Headache	Sinus	P
True	True	0.6
False	True	0.4
True	False	0.5
False	False	0.5

Nose	Sinus	P
True	True	0.8
False	True	0.2
True	False	0.3
False	False	0.7

Generative Models

Flue = True

Allergy = False



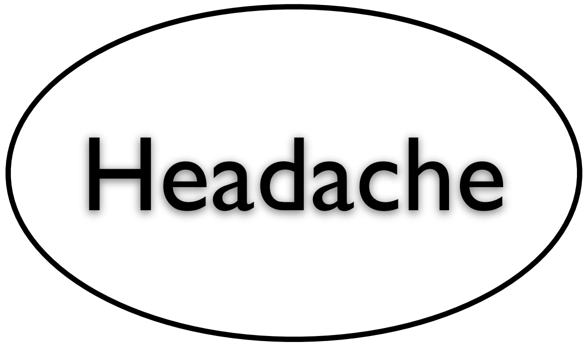
Sinus	Flu	Allergy	P
~~True~~	~~True~~	~~True~~	~~0.9~~
~~False~~	~~True~~	~~True~~	~~0.1~~
True	True	False	0.6
False	True	False	0.4
~~True~~	~~False~~	~~False~~	~~0.2~~
~~False~~	~~False~~	~~False~~	~~0.8~~
~~True~~	~~False~~	~~True~~	~~0.4~~
~~False~~	~~False~~	~~True~~	~~0.6~~

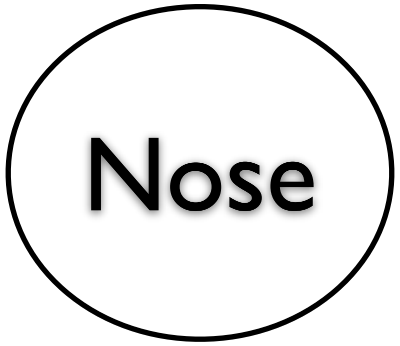
Nose	Sinus	P
True	True	0.8
False	True	0.2
True	False	0.3
False	False	0.7

Headache	Sinus	P
True	True	0.6
False	True	0.4
True	False	0.5
False	False	0.5

Generative Models

Flue = True
Allergy = False
Sinus = True
Nose = True
Headache = False





Nose	Sinus	P
True	True	0.8
False	True	0.2
~~True~~	~~False~~	~~0.3~~
~~False~~	~~False~~	~~0.7~~

Headache	Sinus	P
True	True	0.6
False	True	0.4
~~True~~	~~False~~	~~0.5~~
~~False~~	~~False~~	~~0.5~~

Sampling the Conditional

What if we want to know P(A | B)?

We could use the previous procedure, and just divide the data up based on B.

What if we want P(A | b)?

- Could do the same, just use data with *B*=*b*.
- Throw away the rest of the data.
- ***Rejection sampling.***

Sampling the Conditional

What if b is uncommon?
What if b involves many variables?

Importance sampling:

- Bias the sampling process to get more "hits".
  - New distribution, *Q*.

- Use a reweighing trick to unbias probabilities.
  - Multiply by *P*/*Q* to get probability of sample.

Sampling

Properties of sampling:

- Slow.
- *Always* works.
- *Always* applicable.
- Easy to parallelize.
- **Computers are getting faster.**

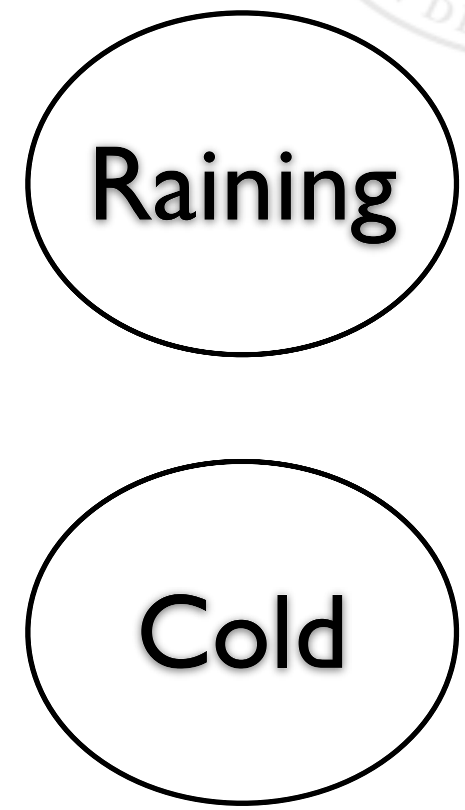
Independence

What does this look like with a Bayes Net?

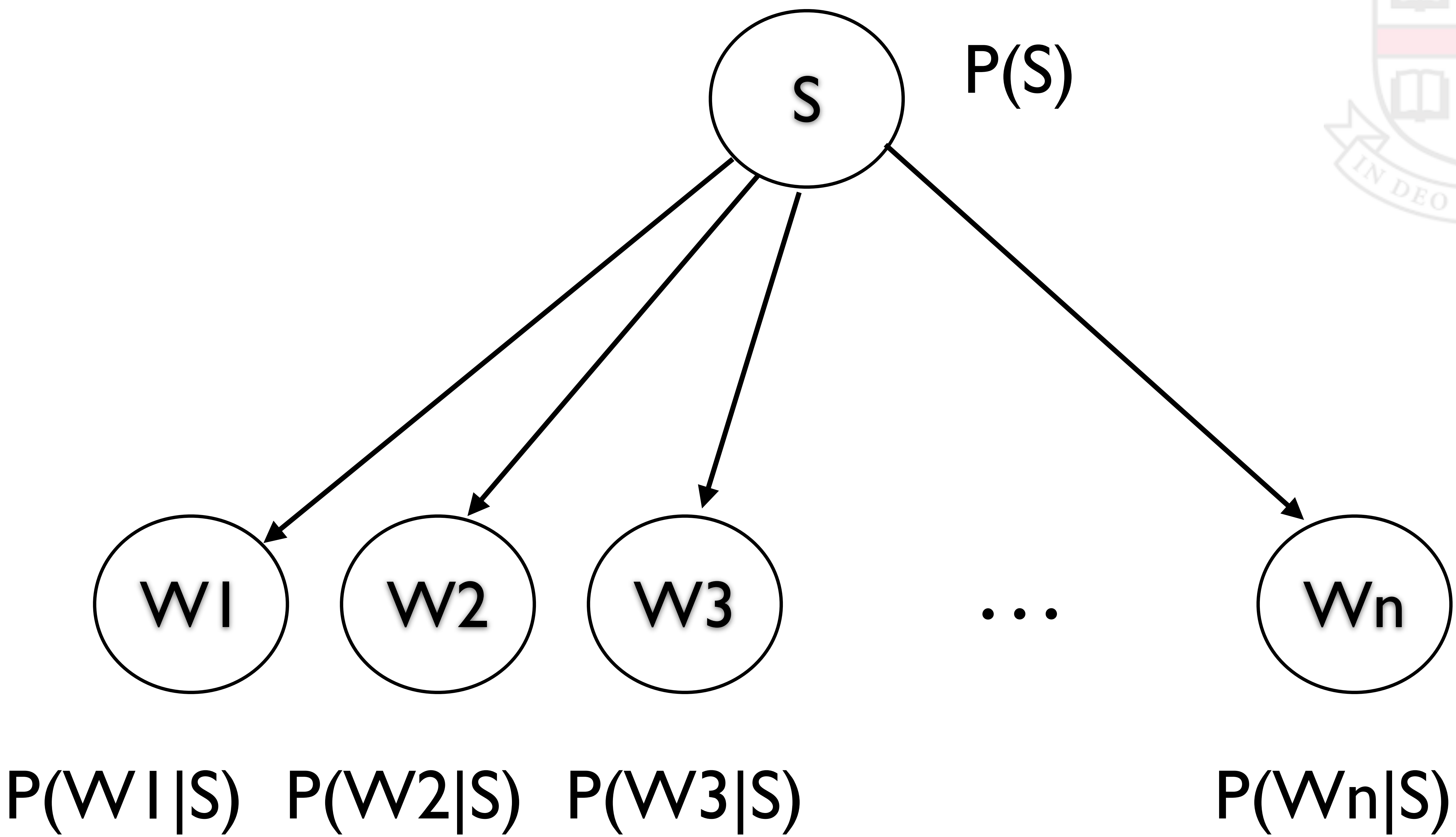
Raining	Prob.
True	0.6
False	0.4

Cold	Prob.
True	0.75
False	0.25

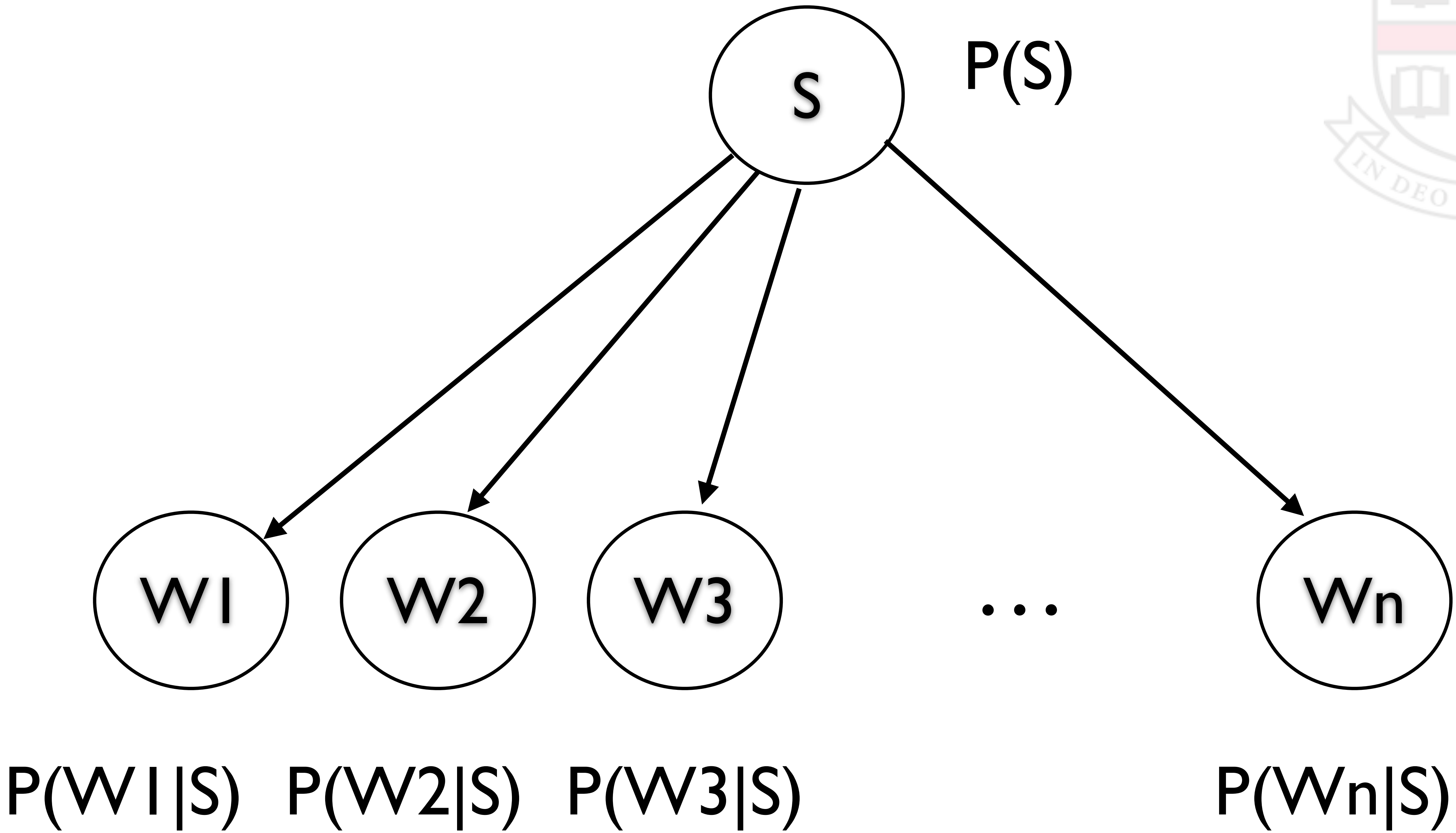
Raining	Cold	Prob.
True	True	0.45
True	False	0.15
False	True	0.3
False	False	0.1



Naive Bayes



Spam Filter (Naive Bayes)



Want P(S | W_1 ... W_n)

Naive Bayes

$$P(S|W_1, ..., W_n) = \frac{P(W_1, ..., W_n|S)P(S)}{P(W_1, ..., W_n)}$$

given

$$P(W_1, ..., W_n|S) = \prod_i P(W_i|S)$$

(from the Bayes Net)

Bayes Nets

Bayes Nets are a ***type of representation***.

Multiple inference algorithms; you can choose!

- AI researchers talk about **models** more than **algorithms**.

Potentially very compressed *but exact*.

- Requires careful construction!

VS

Approximate representation.

- Hope you're not too wrong!

Many, many applications in all areas.