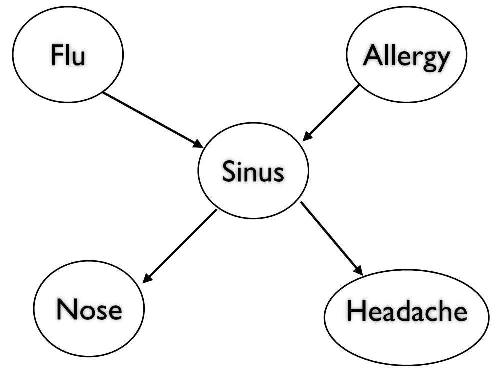
Bayesian Networks

George Konidaris
gdk@cs.duke.edu

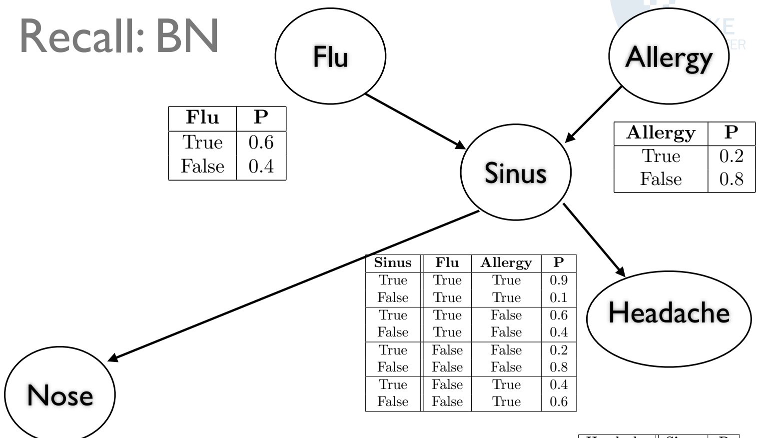
DUKE COMPUTER SCIENCE

Spring 2015

Recall: Bayesian Network



Recall: BN



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

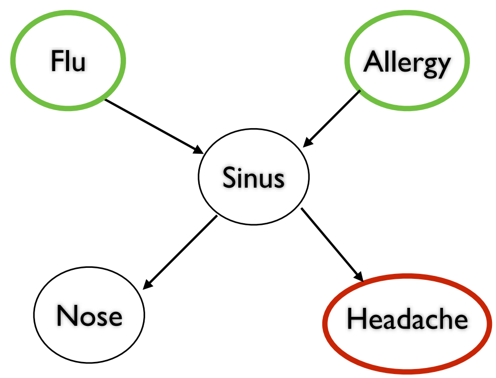
Nose	Sinus	P
True	True	0.8
False	True	0.2
True	False	0.3
False	False	0.7

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

joint: 32 (31) entries

Inference

Given A compute P(B | A).



Last Time: Variable Elimination

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

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

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

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

Sampling

Bayesian networks are generative models:
- Describe a **probability distribution**.
- Can **draw samples** from that distribution.
- This is like a *stochastic simulation*.
- Computationally expensive, but *easy to code!*

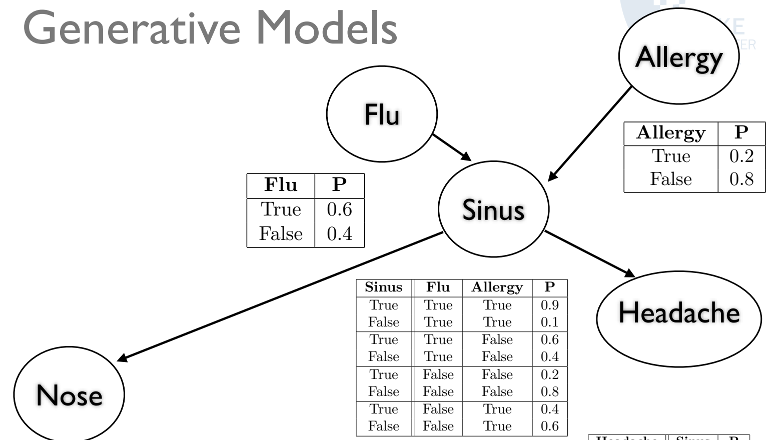
Generative Models

Widely used methodology in machine learning (later).

Describe a generative process for the data.
- Each variable is generated by a distribution
- Can generate more data.

Natural way to include domain knowledge.

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

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.*

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*.
- But what if *b* doesn't happen often?
- What is *b* involves many variables?

Sampling the Conditional

Two broad approaches.

Rejection sampling:

- Sample, throw away when mismatch occurs. (B != b)

Importance sampling:

- Bias the sampling process to get more "hits".
- Use a reweighing trick to unbias probabilities.

Sampling

Properties of sampling:

- Slow.
- *Always* works.
- *Always* applicable.
- **Computers are getting faster.**

Bayes Nets

High-level thoughts.

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

There are multiple algorithms for inference; you can choose whichever you like.

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

Probability Distributions

If you have a discrete RV, probability distribution is a table:

Flu	**P**
True	0.6
False	0.4

What if you have a real-valued random variable?

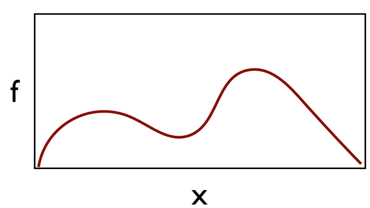
- Temperature tomorrow
- Rainfall
- Number of votes in election
- Height

PDFs

Continuous probabilities described by **probability density function** f(x).

PDF is about density, not probability.

- Non-negative.
- $\int_X f(x) = 1$ integrates to 1
- f(x) might be greater than 1.



PDFs

Can't ask *P(x = 0.0014245)?*

The probability of a single real-valued number is zero.

Instead we can ask for a *range*:

$$P(a \leq X \leq b) = \int_a^b f(x)dx$$

Distributions

Distributions usually specified by a PDF *type* or *family*.

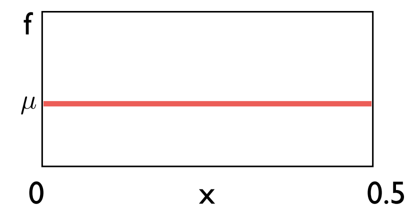
Each family is a *parametrized function* describing the PDF.

Get a specific distribution by fixing the parameters.

Uniform Distribution

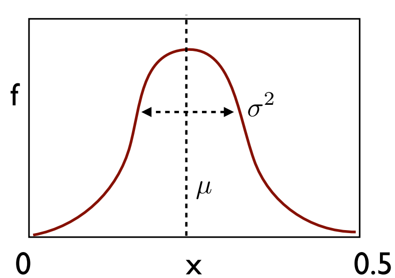
For example, uniform distribution over [0, 0.5].

Parameter: mean.



Gaussian (Normal)

A *mean* + an exponential drop-off, characterized by *variance*.



$$f(x, \mu, \sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

PDFs

When dealing with a real-valued variable, two steps:

- Specifying the family of distribution.
- Specifying the values of the parameters.

Conditioning on a discrete variable just means picking from a discrete number of parameter settings.

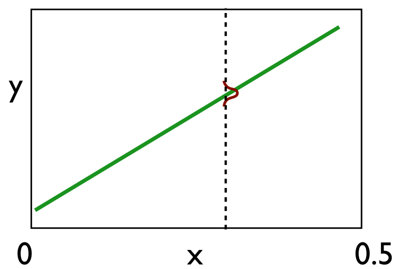
μ_A	σ^2_A	**B**
0.5	0.02	True
0.1	0.06	False

PDFs

Conditioning on real-valued RV:
- Parameters function of RV

Linear regression:

$$f(x) = w \cdot x + \epsilon$$
$$y \sim N(w \cdot x, \sigma^2)$$



Parametrized Forms

Many machine learning algorithms start with parametrized, generative models.

Find PDFs / CPTs such that *probability they generated the data is maximized.*

There are also *non-parametric forms*: describe the PDF directly from the data itself, not a function.