

## Next Week

### This Week

Monday: Bayesian classifiers (read Autoclass paper)

Wednesday: Context specific independence and local structure.

Friday: Normal-Wishart priors

Progress report #2

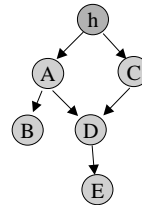
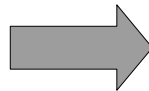
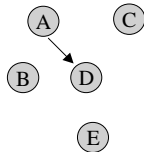
You should be 80% done with the technical portion of your project.

23 April: Papers due (No extensions)

## Optimization over Structure and Parameters (Incomplete Data)

A	B	C	D	E
1	2		0	1
1	1	0		1
0		1	1	
1	1	1	1	2

+



+

$P\{A|h\}$   
 $P\{B|A\}$   
 $P\{C|h\}$   
 $P\{D|A,C\}$   
 $P\{D|A\}$   
 $P\{E|D\}$   
 $P\{h\}$

## Approximating $P\{D|\Theta,G\}$

Why?

Select models based on how well they fit the data

Bayesian score:  $\log P\{D|\Theta,G\} P\{\Theta,G\}$

Approaches

MCMC (exact if you simulate long enough)

Laplace

Full

Block diagonal

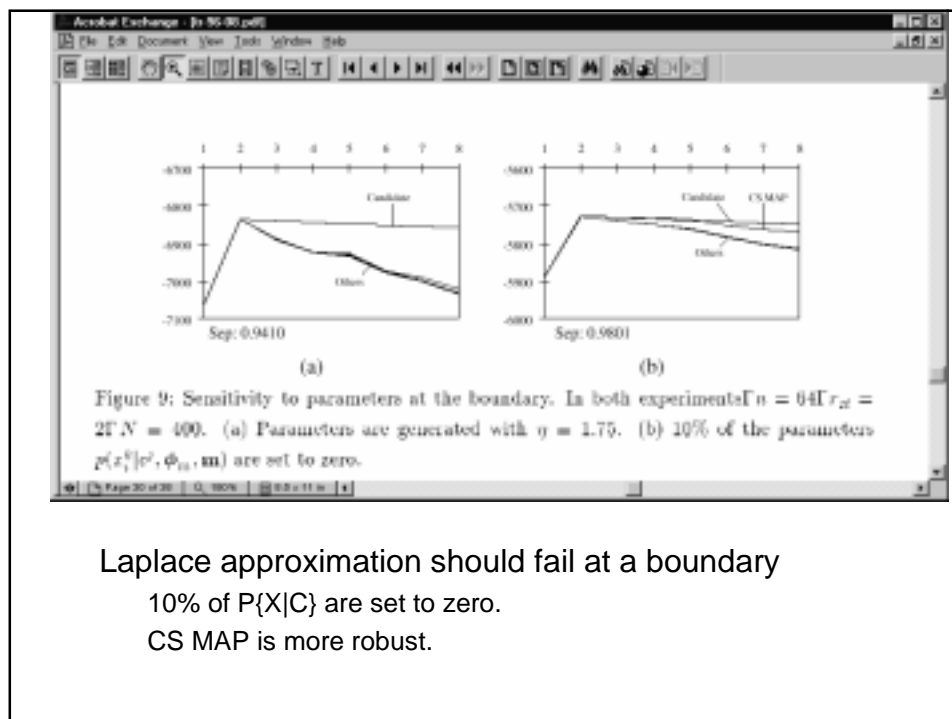
Diagonal

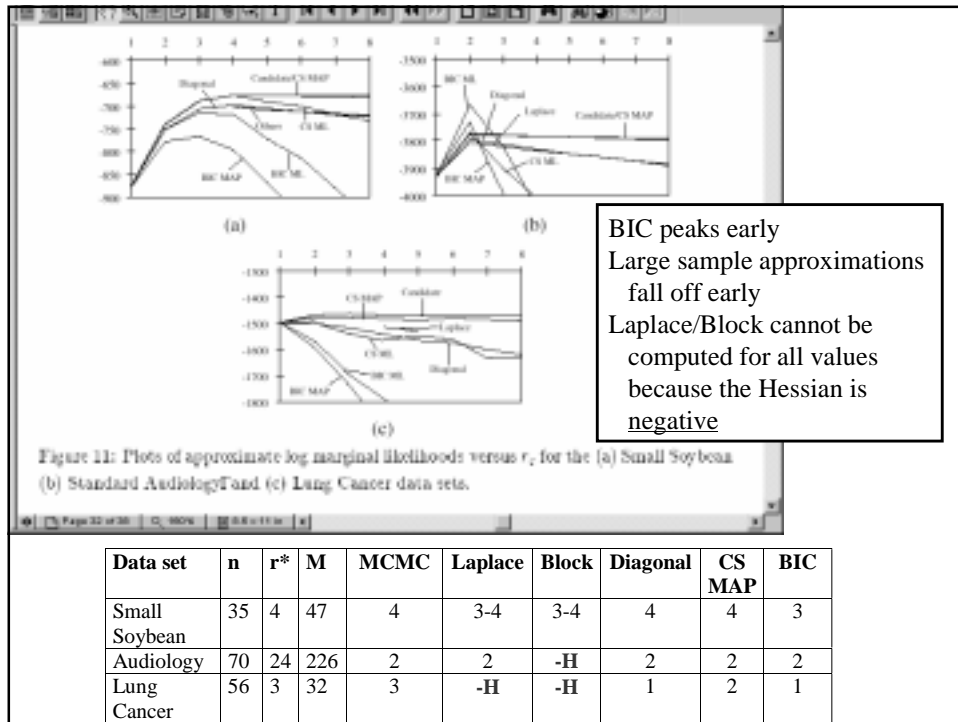
BIC-MAP and BIC-ML

Cheeseman-Stutz CS-MAP, CS-ML

Source:

Chickering and Heckerman, Efficient Approximations for the Marginal Likelihood of Bayesian Networks with Hidden Variables, MSR-TR-96-08





## Approximation Conclusions

### Model selection

All except for BIC/MDL are accurate for model selection.

### Sensitivity to priors

All except BIC/MDL are sensitive to priors

### MAP/ML

CS is more accurate with MAP

BIC/MDL is more accurate with ML

### Accuracy

Cheeseman Stutz tends to be more accurate than other approximations

CS best when MAP is near a boundary.

## Approximation Unknowns

What technique works best for general BN inference?

Good data sets for classification.

No "general" data sets for general BN inference.

Structural EM (Friedman, 97)

## Standard EM

Standard greedy structure search with incomplete data:

Find all of the networks  $C_1 C_2 \dots C_n$  that are adjacent to  $G_n$

Note that there are  $O(n^2)$  successors

“flip the bit on any arc”

Optimize parameters for  $C_1 C_2 \dots C_n$  using “Parametric EM.”

Have to run enough iterations to guarantee good model selection.

Score  $P\{D|G\}$  for  $C_1 C_2 \dots C_n$  using BIC, Laplace, MCMC or CS.

Let  $G_{n+1} = \arg \max_{G' \in C \cup \{G_n\}} (\text{Score}(G'))$

It is only practical to use this algorithm for VERY small problems

## Structural EM

Optimize the parameters  $\Theta_n$  for  $G_n$  using some number of steps of EM.

Complete the data using the expected sufficient statistics given  $\Theta_n$

Pretend that the data is complete and search some number of steps (say k) to find  $G_{n+k}$

Why is this a win?

We only have to run EM a few times.

The same “completion” of the data is used to score several networks.

Thm: Say that  $C_B(D)$  is a completion of the data using B

$$\text{Score}_{MDL}(B'; D) - \text{Score}_{MDL}(B'; D) \geq \text{Score}_{MDL}(B'; C_B(D)) - \text{Score}_{MDL}(B'; C_B(D))$$

## Structure learning with incomplete data

### Big changes:

Need to use EM to optimize the parameters.

Need to approximate  $P\{\Theta|G\}$  in order to compute  $P\{D|G\}$

### Approximations to the likelihood function $P\{D|G\}$ (for clustering)

All approximations underestimate when the problem has reduced dimensionality.

Don't use BIC or MDL: *Underfits the data.*

Of the "cheap" approximations, Cheeseman-Stutz is best.

Open issue: How well do these work for non-clustering problems?

### Structural EM:

It is a good idea to reuse the "completion" derived from one run of EM to score *many* adjacent candidate graphs.

The technique described as paring search time down from **years** to **hours** on large problems (3-4 orders of magnitude).

## CLASSIFICATION

## Classification

Variables:  $X$ : *Features or Attributes*  $C$ : *Class*

### Supervised

Data:  $(C[m], X_1[m], \dots, X_n[m])$

Learn:  $F: X \rightarrow C$

Learn a function that maps observations into a class.

### Unsupervised Classification (Clustering)

Data:  $(X_1[m], \dots, X_n[m])$

Learn:  $\max_{R, \Theta} P\{X\} = \prod_m \sum_{j=1}^R P\{X[m] | C_j[m]\} P\{C_j[m]\}$

Learn a hidden variable that explains "clusters" in the data.

## Supervised Classification

### Large field:

LOTS of research, publically available implementations.

Large set of standard test problems.

UC Irvine Repository:

<http://www.ics.uci.edu/~mlern/MLRepository.html>

### Examples of problems:

#### Breast cancer:

Class: (Recurrent, Non-recurrent)

Features: Digitized image of a fine needle aspiration (33 numeric attributes)

#### Mushrooms:

Class: (Poisonous, Edible)

Features: Color, Spots, Shape, etc.

#### Digits:

Class: (1,2,3,4,5,6,7,8,9,0)

Features: Pixel counts in 4x4 blocks in a 32 x 32 bit map

## Feature/Class distinction

Assignment of variables to “Features” or “Class” is arbitrary

Predict any feature given the other features...

## Supervised classification

### Algorithms:

#### Decision tree approaches

Find the best recursive partition of the sample space

CART

C4.5

#### Generative models

Induce a model and impose a decision rule

Bayes net learning

Naïve Bayes (Idiot Bayes)

Features are conditionally independent given class.

Tree Augmented Naïve Bayes

## Bayes Net-based Classification

Construction:

1. Identify JPD  $P\{C, X_1, \dots, X_n\}$
2. Decision rule:

$$c = \arg \max_{c_j} P\{C = c_j, x_1, \dots, x_n\}$$

Decision rule is optimal to reduce the probability of misclassification.

## Diabetes in Pima Indians

768 - women of Pima Indians heritage (21 or older).

Class:

Class variable (0 or 1): "tested positive for diabetes"

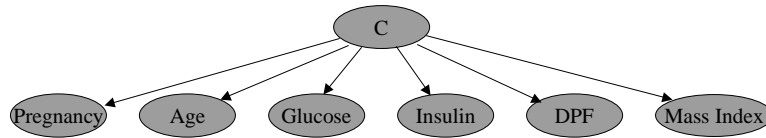
Features:

1. Number of times pregnant
2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test
5. 2-Hour serum insulin ( $\mu$  U/ml)
6. Body mass index (weight in kg/(height in m)<sup>2</sup>)
7. Diabetes pedigree function
8. Age (years)

Not used:

3. Diastolic blood pressure (mm Hg)
4. Triceps skin fold thickness (mm)

## Naïve Bayes Classifier (Duda + Hart, 73)



### Naïve Bayes:

Features are conditionally independent given class

### Learning:

Estimate parameters for  $P\{X_i | C\}$

Surprisingly robust, especially for problems with relatively little data.

Strong prior on structure?

Classification may be correct even though probabilities are not...

## Improving Naïve Bayes

### Feature subset selection

“Wrapper” around NB learning

Pick NB classifier over a subset of the features that has the lowest error rate.

### Model dependencies in data

Data is not necessarily conditionally independent given C

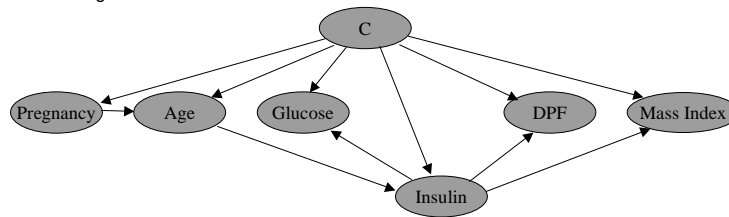
## Tree Augmented Bayes Classifier

[Friedman, Geiger, Goldszmidt; *Machine Learning* 1997]

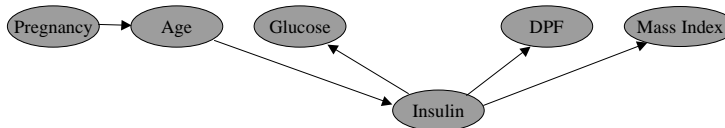
### Unrealistic assumption?

Pregnancy is independent of age given diabetes?

Add "augmentation" arcs.



### Tree Augmented Naïve Bayes (TAN)



## Tree Induction Algorithm (again)

Select tree that maximizes *conditional mutual information*:

$$I_p(X;Y|Z) = \sum_{x,y,z} P\{x,y,z\} \log \frac{P\{x,y|z\}}{P\{x|z\}P\{y|z\}}$$

Algorithm:

- Compute  $I_p(X_i;Y_j|C)$  for every pair of nodes in the network.
- Build a complete undirected graph in which the vertices are  $X_1, \dots, X_n$ . Annotate edges with  $I_p(X_i;Y_j|C)$ .
- Build a maximal weight spanning tree.
- Choose a root node (arbitrary) and direct all edges away from the root. (errata from 3 weeks ago)
- Add an arc from C to each  $X_i$

# Experiments

## N-fold Cross Validation

Split data into N sets

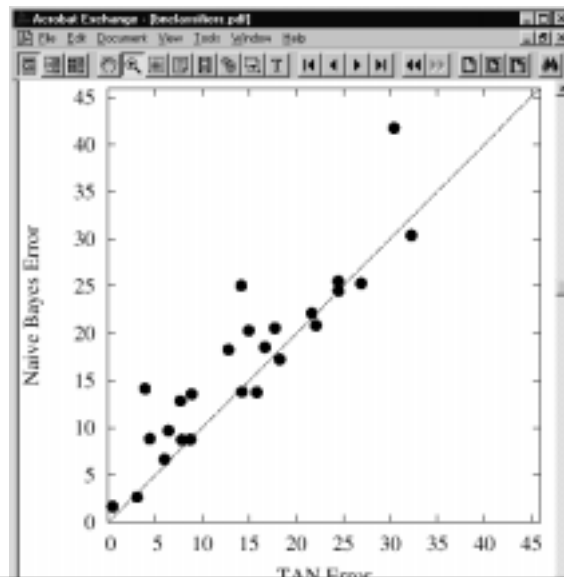
For each run, let one of these sets be testing data (C) and the rest be training data.

For each run,

Train classifier on not(C),  
Test on C

C						run 1
	C					run 2
		C				run 3
			C			
				C		
					C	run N

## TAN vs Naïve Bayes



## TAN vs general Bayes net induction algorithm

Learn a network

Best network is the “best” model for  $P\{C, X_1, \dots, X_n | G\}$

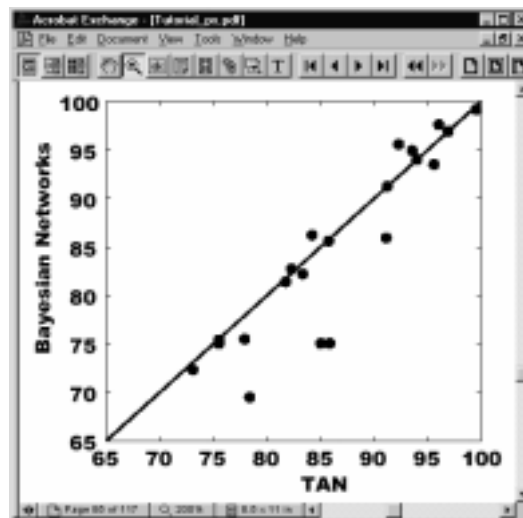
Experiment:

maximize the MDL score

$$\log P\{C, X_1, \dots, X_n | G\} - \frac{d}{2} \log M$$

What do you expect?

## TAN vs BN



## Results

The belief net algorithm performs poorly.

Why?

BN tries to find the most compact model for

$$P\{C, X_1, \dots, X_n | G\}$$

...but we are really interested in  $P\{C | X_1, \dots, X_n\}$

Bayes net induction algorithms optimize JPD accuracy, not classification accuracy

In the Bayes net, there may be few arcs between the classification node and the rest of the network.

Augmented Naïve Bayes: Ensure that there is a connection between every node and the classification node.

## General BN

Optimize BN for  $P\{C | X_1, \dots, X_n\}$

Problem:

$P\{C[m] | X_1[m], \dots, X_n[m]\}$  does not decompose

$$\text{optimize } \sum_X (1 - P\{C_{\max} | X_1, \dots, X_n\}) ?$$

TAN/NB works because arcs are always added.

ANB: Arcs added between C and X for all X

Open question

Criterion for optimality for classification?

Tractable algorithm given this criterion?

Data set	TAN-	AND
1 australian	81.20+-1.21	86.81+-0.42
2 breast	95.92+-0.07	96.42+-1.09
3 chess	92.91+-0.82	94.15+-0.72
4 cleve	81.76+-0.33	80.05+-1.30
5 coral	95.06+-2.51	95.40+-1.60
6 crx	85.76+-1.16	86.37+-0.38
7 diabetes	75.52+-1.11	75.52+-1.06
8 flavr	82.27+-1.89	82.84+-1.79
9 german	73.10+-1.54	73.20+-1.25
10 glass	67.75+-3.43	69.66+-1.85
11 glass2	77.92+-1.11	79.17+-1.71
12 hact	83.33+-2.48	82.69+-2.31
13 hepatitis	91.23+-2.90	88.74+-2.34
14 iris	91.69+-1.25	93.33+-1.05
15 letter	85.86+-0.32	76.69+-0.60
16 lymphography	85.83+-3.09	83.10+-2.19
17 mcfm-3-7-10	91.11+-0.89	96.43+-1.07
18 pima	75.92+-1.27	74.74+-1.23
19 satimage	87.29+-0.75	80.50+-0.89
20 segment	93.55+-0.74	91.17+-1.02
21 shuttle-small	79.53+-0.15	88.91+-0.24
22 soybean-large	92.17+-1.02	92.18+-1.02
23 vehicle	69.63+-2.11	67.35+-1.38
24 vote	93.96+-0.25	89.66+-1.21
25 waveform-21	78.35+-0.89	77.73+-0.61

## Open Issues

How to encode ordered discrete data?

Example: Poisson distribution

Discretize continuous variables?

## Classification

Variables:  $X$ : *Features or Attributes*  $C$ : *Class*

Supervised

Data:  $(C[m], X_1[m], \dots, X_n[m])$

Learn:  $F: X \rightarrow C$

Learn a function that maps observations into a class.

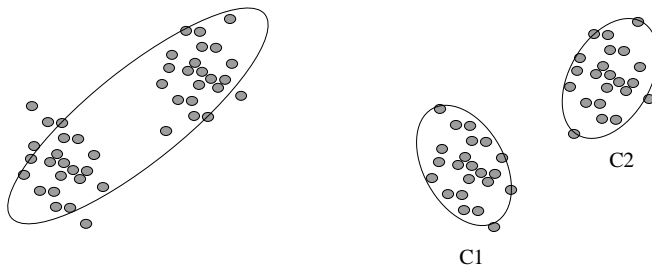
Unsupervised Classification (Clustering)

Data:  $(X_1[m], \dots, X_n[m])$

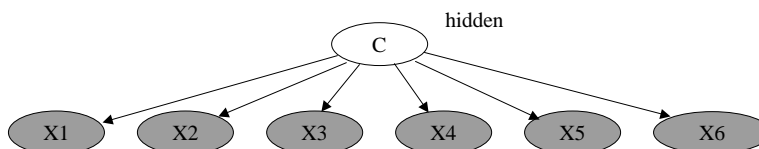
Learn:  $\max_{R, \Theta} P\{X\} = \prod_m \sum_{j=1}^R P\{X[m] | C_j[m]\} P\{C_j[m]\}$

Learn a hidden variable that explains "clusters" in the data.

## Clustering



## Autoclass Assumptions



$$P\{\Theta_C\} = K \prod_{j=1}^{|C|} \theta_j^{1/|C|}$$

Discrete: 
$$P\{\Theta_{X|C}\} = K \prod_{k=1}^{|X|} \theta_{k|C}^{1/|L_k|}$$

$$\hat{\theta}_{X_i=k|C=j} = \frac{E(N_{ijk})}{E(N_{ij})}$$

Gaussian (unknown mean and variance):

$$P\{\mu_{X|C}\} = \frac{1}{\mu_{X|C,\max} - \mu_{X|C,\min}}$$

$$\hat{\mu}_{X_i|C=j} = E(X_{ij})$$

$$P\{\sigma_{X|C}\} = \sigma_{X|C}^{-1} \left[ \log \frac{\sigma_{X|C,\max}}{\sigma_{X|C,\min}} \right]$$

$$\sigma_{X_i|C=j}^2 = \frac{N_j}{N_j + 1} E((X_{ij} - \mu_{ij})^2)$$

## Autoclass C: Application and Distribution Types

Web address:

<http://ic-www.arc.nasa.gov/ic/projects/bayes-group/autoclass/autoclass-c-program.html>

Variable types

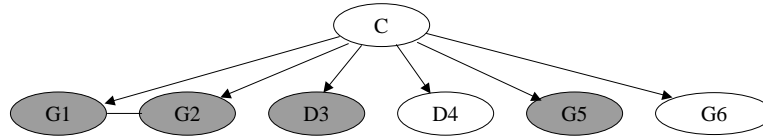
single\_multinomial - discrete attribute multinomial model, including missing values.

single\_normal - real valued attribute model with no missing values.

single\_normal\_missing - real valued attribute model with missing values.

multi\_normal - real valued covariant normal model with no missing values.

## Autoclass C: Distribution Types Illustrated



normal variable combinations  
need to be specified in advance.

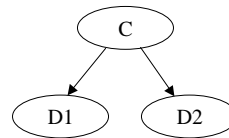
## Autoclass-C Experience

### Dependencies in discrete variables

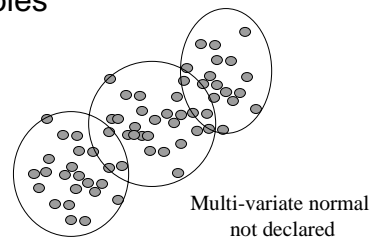
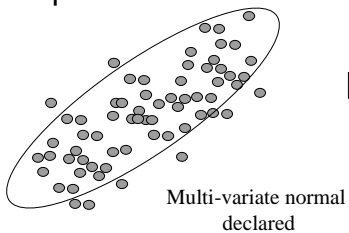
Discrete/Discrete dependency  
in "generator"



Explained through  
classification node.



### Dependencies in normal variables



## Autoclass C Experience

### Turbine Engine Data

Engine parameters  
(fan speed, exhaust gas temperature, altitude, + 65 others)

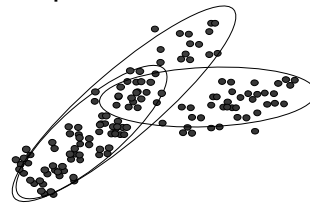
Desired behavior:  
Identify differences in control systems between engines.

Actual behavior:  
Focussed on  
violations of normality assumption

## Autoclass C Experience, continued

### Sensitivity to Gaussian Assumption

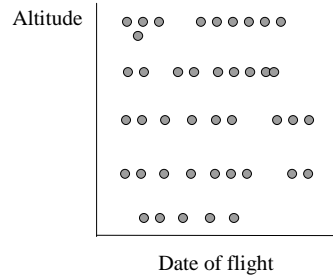
Turbine engine model  
"Desired Clustering"



Turbine engine model  
"Actual Clustering"

FAA flight rules:  
Flights spaced 2000 ft apart  
Alternating layers of east/west  
headings

Moral: Autoclass focusses on  
explaining clusters in data  
without regard to the perceived  
importance of the clusters.



## Local Structure and Learning

### Context Specific Independence (CSI)

(Boutilier, et al, UAI-96)

X and Y are contextually independent given Z and context  $c$  in  $\text{Val}(C)$  iff

$$P\{X | Z, c, Y\} = P\{X | Z, c\} \quad \text{whenever} \quad P\{Y, Z, c\} > 0$$

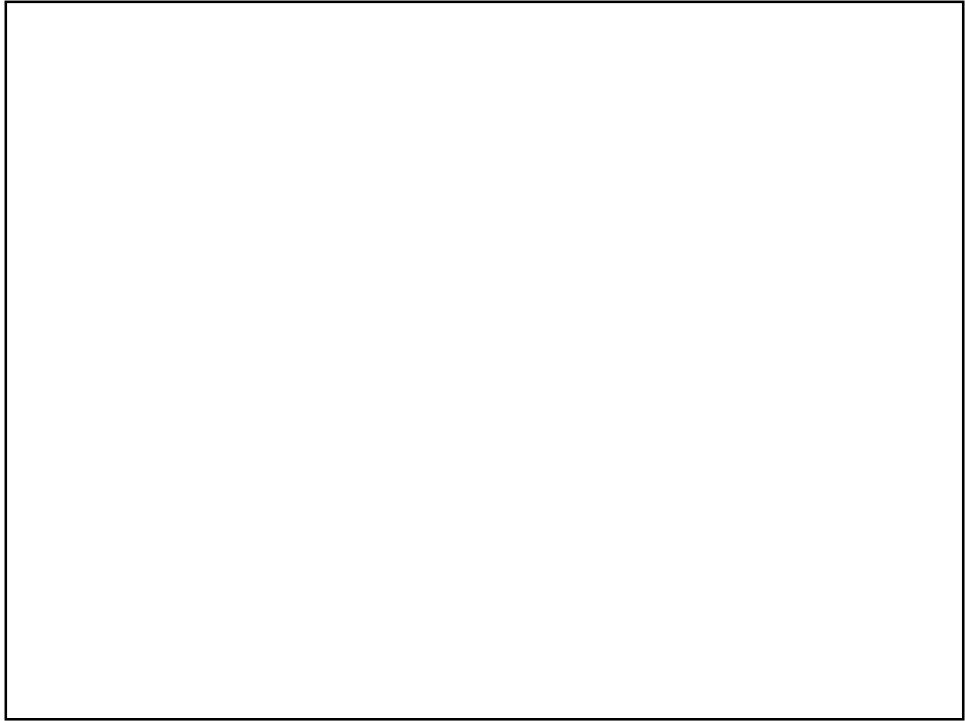
Examples:

P(D)	C=T	C=F
A=T, B=T	0.3	0.1
A=T, B=F	0.5	0.1
A=F, B=T	0.2	0.9
A=F, B=F	0.7	0.9

P(D)	C=T	C=F
A=T, B=T	0.3	0.1
A=T, B=F	0	0
A=F, B=T	0.2	0
A=F, B=F	0	0

Noisy-Or:

$$P\{cause_i | effect = F, cause_j\} = P\{cause_i | effect = F\}$$



## Next Time

Context specific independence and local structure.

Friday: Normal-Wishart priors

Progress report #2

You should be 80% done with the technical portion of your project.

Remember the project is 1/2 of your grade.