Stat 521A
Lecture 4

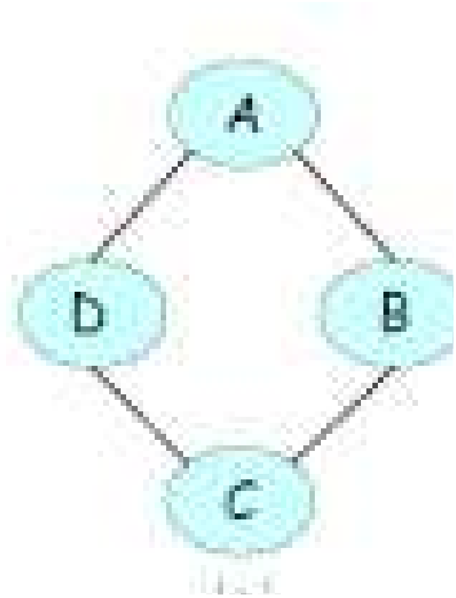
Admin

- CS auditors: please turn in your form to Joyce Poon, who will pass it to Laks for signing

Outline

- Aside on canonical parameterization (ex 4.4.14)
- Structured factors (4.4.1.2)
- Structured CPDs (5.2-5.6)
- Temporal models (6.2)

Degrees of freedom of a UGM



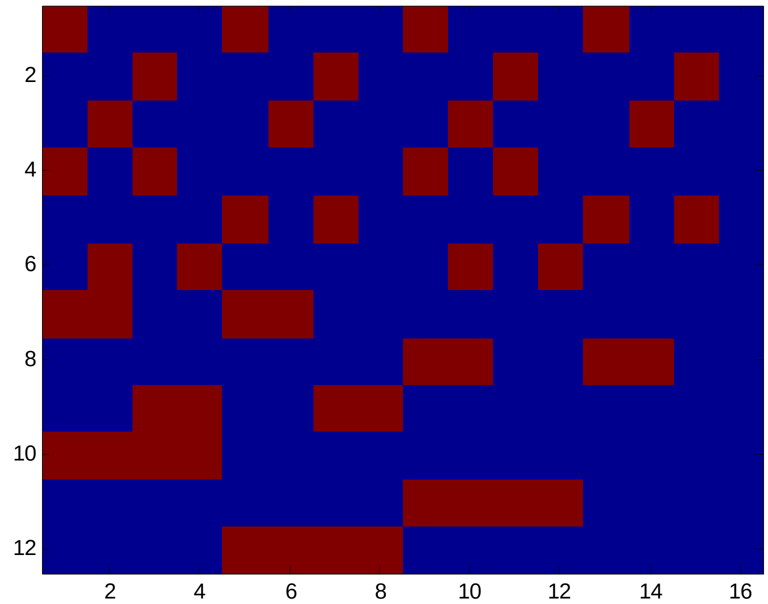
Why do we just need 8 numbers to uniquely parameterize the distribution?

Eg a^1, b^1, c^1, d^1, (a^1,b^1), (b^1,c^1), (c^1,d^1), (a^1,d^1)

Num params = rank of feature matrix

- Let F(n, i)=1 iff i'th bit vector turns on n'th feature
- Each feature specifies a value for every pair of nodes connected by an edge, and hence is a vector in R^{16}. 4 edges, 3 unique settings = 12 rows.

Rank = 8



Eg a^1, b^1, c^1, d^1, (a^1,b^1), (b^1,c^1), (c^1,d^1), (a^1,d^1)

Rank of feature matrix

- edges = {[1 2], [1 3], [2 4], [3 4]};
- ndx = 1;
- F = zeros(0, 2^4);
- for e=1:length(edges)
-   s = edges{e}(1); t = edges{e}(2);
-   for j=1:2
-     for k=1:2
-       if j==2 && k==2, continue; end
-       for x=1:16
-         xv= ind2subv([2 2 2 2], x);
-         if xv(s)==j && xv(t)==k
-           F(ndx,x)=1;
-         end
-       end
-       ndx = ndx + 1;
-     end
-   end
- end
- rank(F)

Log-linear factors

- A factor defined on m discrete rv's with K states needs K^m parameters.
- Imagine a factor on triples of letters. Instead of having 26^3 numbers, we can define binary features that only turn on for certain values, eg $f_{ing}(x) = 1$ iff x_1='I',x_2='n',x_3='g'. This has weight ω_{ing}. We define

$$\phi_c(\mathbf{x}_c) = \exp(\sum_{i=1}^{k} w_{c,i} f_{c,i}(\mathbf{x}_c))$$

Tables are a special case

		$x_2 = 0$	$x_2 = 1$
x_1	0	e^{θ_1}	1
	1	1	1

$$f_1(x_1, x_2) = \delta(x_1 = 0, x_2 = 0)$$

		$x_2 = 0$	$x_2 = 1$
x_1	0	1	e^{θ_2}
	1	1	1

$$f_2(x_1, x_2) = \delta(x_1 = 0, x_2 = 1)$$

		$x_2 = 0$	$x_2 = 1$
x_1	0	1	1
	1	e^{θ_3}	1

$$f_3(x_1, x_2) = \delta(x_1 = 1, x_2 = 0)$$

		$x_2 = 0$	$x_2 = 1$
x_1	0	1	1
	1	1	e^{θ_4}

$$f_4(x_1, x_2) = \delta(x_1 = 1, x_2 = 1)$$

		$x_2 = 0$	$x_2 = 1$
x_1	0	e^{θ_1}	e^{θ_2}
	1	e^{θ_3}	e^{θ_4}

$$\varphi_{12}(x_1, x_2) = e^{\theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 + \theta_4 f_4}$$

Jordan, fig 19.1

CRF features

- Typical features used in a CRF model for language processing (X=words, Y=labels)
- $F_1(Y_t, X_t, X_{t-1}, X_t+1) = I(X_{t-1}=\text{"New"}, X_t=\text{"York"}, X_{t+1}=\text{"Times"}, Y_t=\text{"Object"})$
- $F_2(Y_t, X_t, X_{t-1}, X_t+1) = I(X_{t-1}=\text{"New"}, X_t=\text{"York"}, X_{t+1} \neq \text{"Times"}, Y_t=\text{"Place"})$
- Models often have ~100k manually specified features.
- Common to use L1 regularization to sparsify.
- Can also perform feature induction, by eg greedily creating conjunctions or disjunctions

Exponential family (maxent) models

- Combining all the local potentials

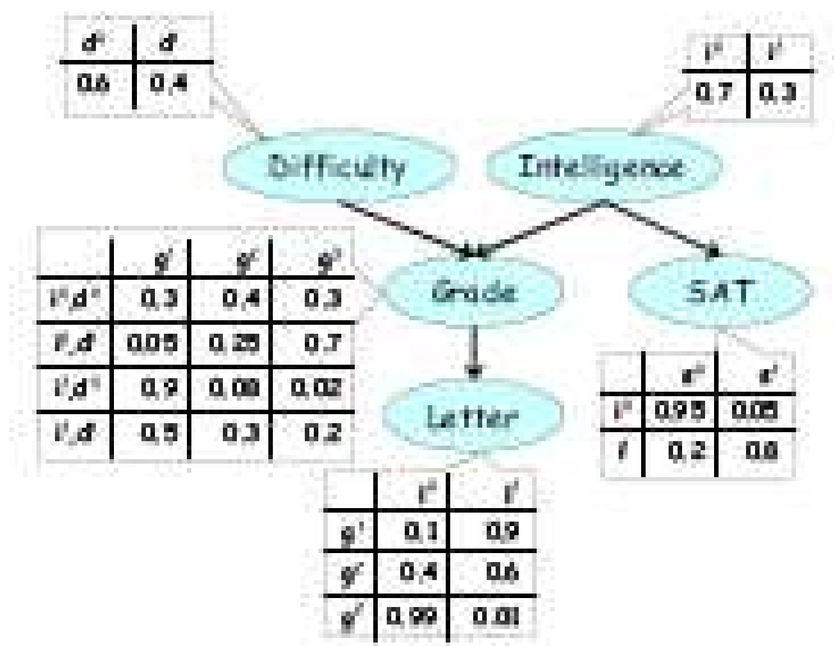
$$
\begin{aligned}
p(\mathbf{x}) &= \frac{1}{Z} \prod_c \phi_c(\mathbf{x}_c) \\
\phi_c(\mathbf{x}_c) &= \exp(\sum_{i=1}^{k} w_{c,i} f_{c,i}(\mathbf{x}_c)) \\
p(\mathbf{x}) &= \frac{1}{Z} \exp(\sum_i w_i f_i(\mathbf{x}_{c_i}))
\end{aligned}
$$

DAGs are a special case where each $\phi_c(x_c) = p(X_i|Pa(X_i))$ sums to 1, so Z=1

See ch 8

Tabular CPDs

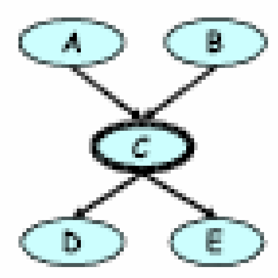
- If all nodes are discrete and have K values, we can represent p(X_i|Pa(X_i)) as a table, with one row per conditioning case (K^#pa), and K columns which sum to 1
- If K and/or #pa is large, this is too many parameters, so we seek more parsimonious representations.



Deterministic CPDs

- In some cases, the child is a deterministic function of the parents, eg bloodtype is determined by the 2 alleles
- Deterministic nodes often denoted by double-ringed oval.
- Determinism can imply additional (non-graphical) independencies
- Eg $D \perp E \mid A,B$ since $C = fn(A,B)$

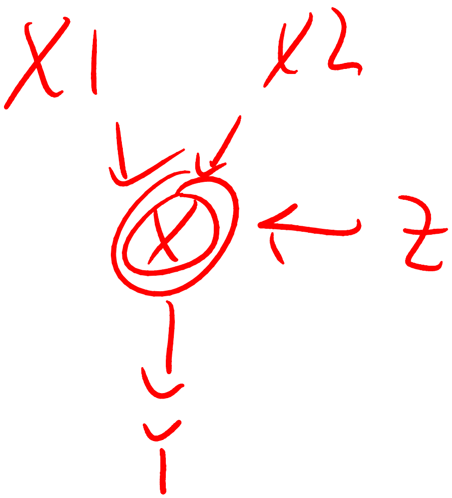
Det-sep



Context specific independence (CSI)

- Sometimes, the set of edges which are "active" depends on the value of the nodes
- Eg Y is a noisy observation of object X1, or X2. Z specifies the identity of the measurement. Let X =multiplexer(X1,X2, Z). Then $X2 \perp Y \mid Z=1$. So our

  posterior on X2 is not affected by the measurement. (Data association ambiguity)

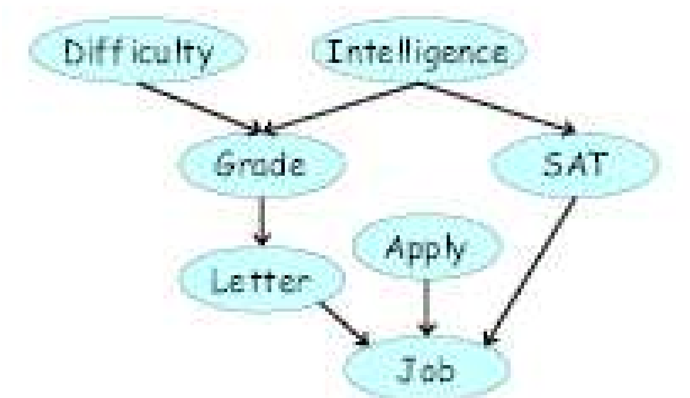


Contingently acyclic BNs

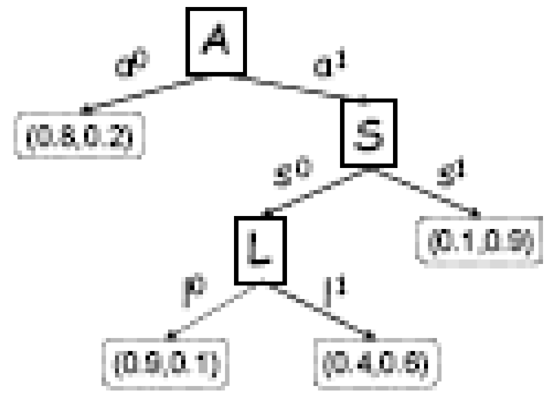
- Sometimes we can define a directed graph with cycles, but where some of the edges are not active for a given setting of certain variables C.
- If we can guarantee that the graph is a DAG for each context C=c, the result is a mixture of differently structured BNs.
- This is called a Bayesian multinet.

Tree-structured CPDs

- Different parents can be rendered irrelevant, depending on the values



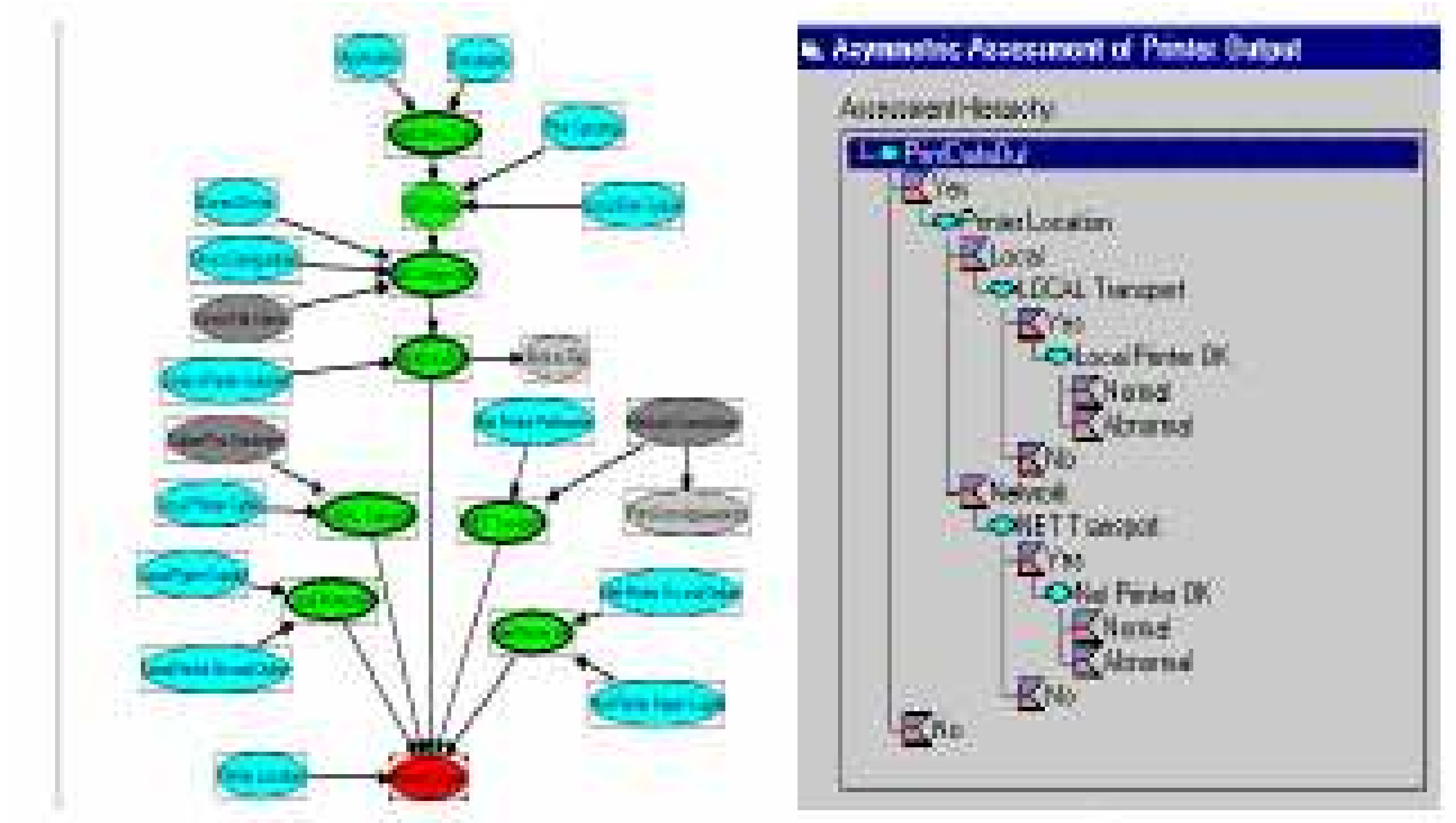
P(J|A,S,L)



Eg. J | S,L if A=0 since we go down left branch of tree

Printer fault diagnosis in MS windows

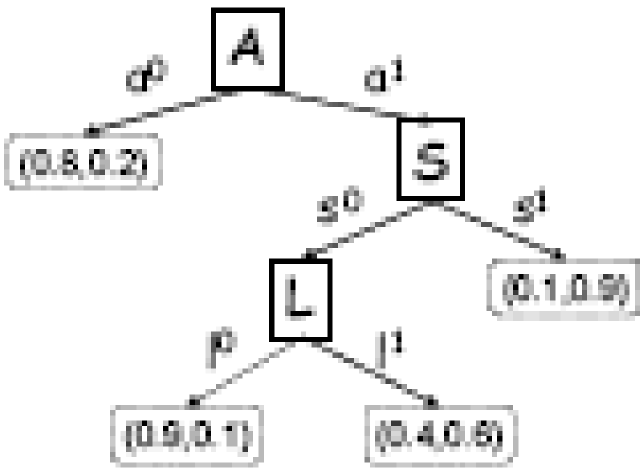
- Uses tree structured CPDs, since different sets of variables are relevant in different contexts



(a) (b)

Rule-structured CPDs

- Specify a pattern and a value

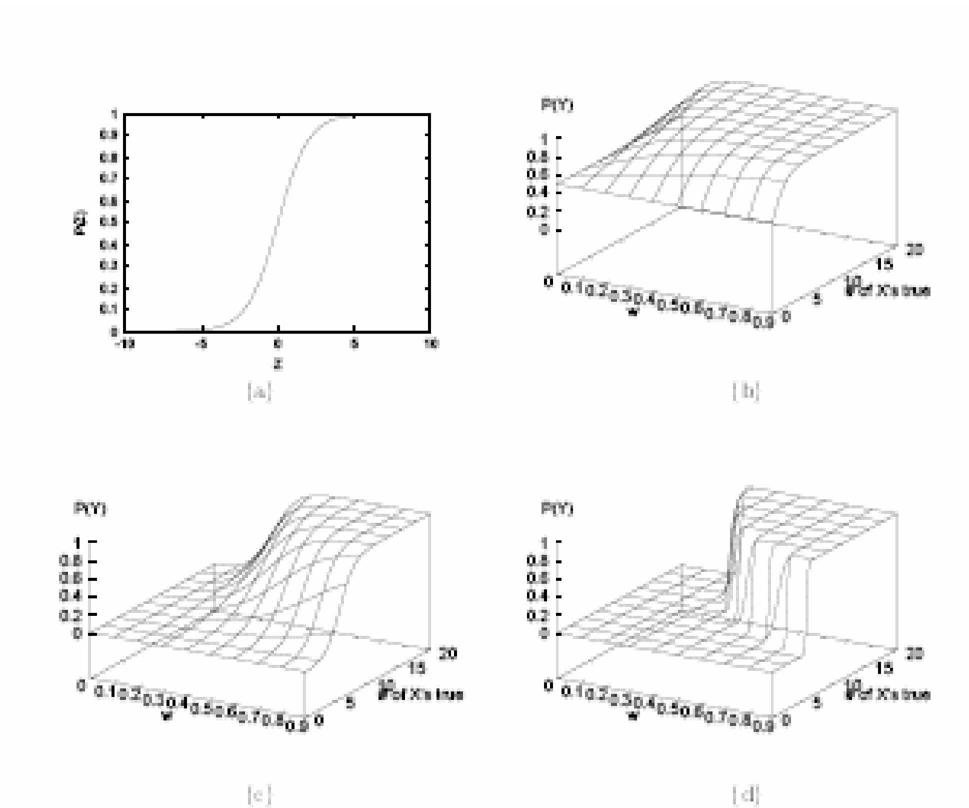


$$
\left\{
\begin{array}{l}
\rho_1 : \langle a^0, f^0 ; 0.8 \rangle \\
\rho_2 : \langle a^0, f^1 ; 0.2 \rangle \\
\rho_3 : \langle a^1, s^0, l^0, f^0 ; 0.9 \rangle \\
\rho_4 : \langle a^1, s^0, l^0, f^1 ; 0.1 \rangle \\
\rho_5 : \langle a^1, s^0, l^1, f^0 ; 0.4 \rangle \\
\rho_6 : \langle a^1, s^0, l^1, f^1 ; 0.6 \rangle \\
\rho_7 : \langle a^1, s^1, f^0 ; 0.1 \rangle \\
\rho_8 : \langle a^1, s^1, f^1 ; 0.9 \rangle
\end{array}
\right\}
$$

Logistic regression (sigmoid BNs)

- Suppose all nodes are binary. We can use logreg CPDs

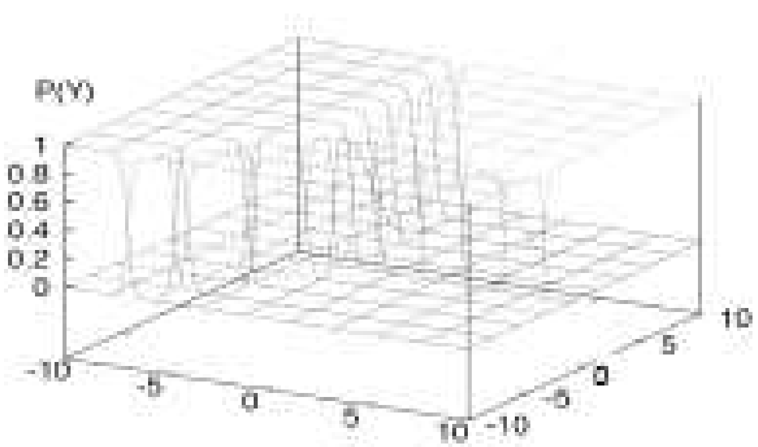
$$p(y=1|\mathbf{x}) = \sigma(w_0 + \sum_{i=1}^{k} w_i x_i) \qquad \sigma(u) = \frac{1}{1+e^{-u}}$$



Multinomial logreg

- If Y is K-ary, and the parents are binary or cts, we can use a softmax function

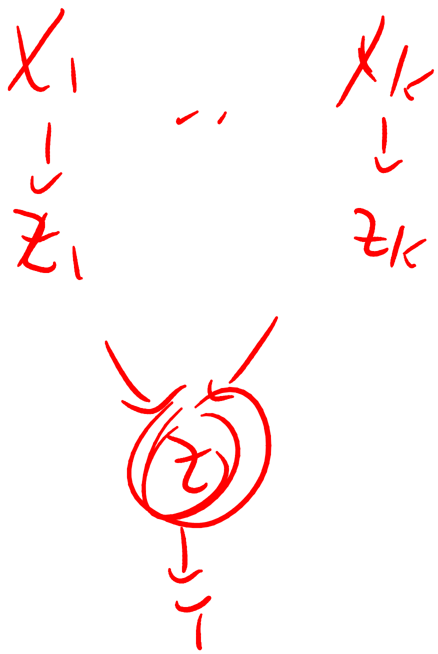
$$p(y = j|\mathbf{x}) = \frac{\exp(\mathbf{w}_j^T \mathbf{x})}{\sum_{j'=1}^K \exp(\mathbf{w}_{j'}^T \mathbf{x})}$$



For K-ary parents, use 1-of-K encoding

Independence of causal influence

- We can model the effects of many parents by assuming that each parent is corrupted by independent noise, and the results are deterministically combined via a simple function such as OR or MAX



Noisy-or model

- Each Xi in {0,1} gets passed through a noisy wire to produce Zi in {0,1}. 0 maps to 0, 1 maps to 0 wp w_i (failure probability). λ_i=1-w_i is the prob. that Xi alone turns on Y.
- The Zi's are combined in an OR to produce Z. Then Y=Z.
- The only way Y can be off is if all Zi's are off, which means all the wires for Xi st Xi=1 independently failed:

$$p(y=0|\mathbf{x}) = \prod_{i:x_i=1} w_i = \prod_{i=1}^{k} w_i^{x_i}$$

$$p(y=1|\mathbf{x}) = 1 - p(y=0|\mathbf{x})$$

Popular in cogsci models of causality

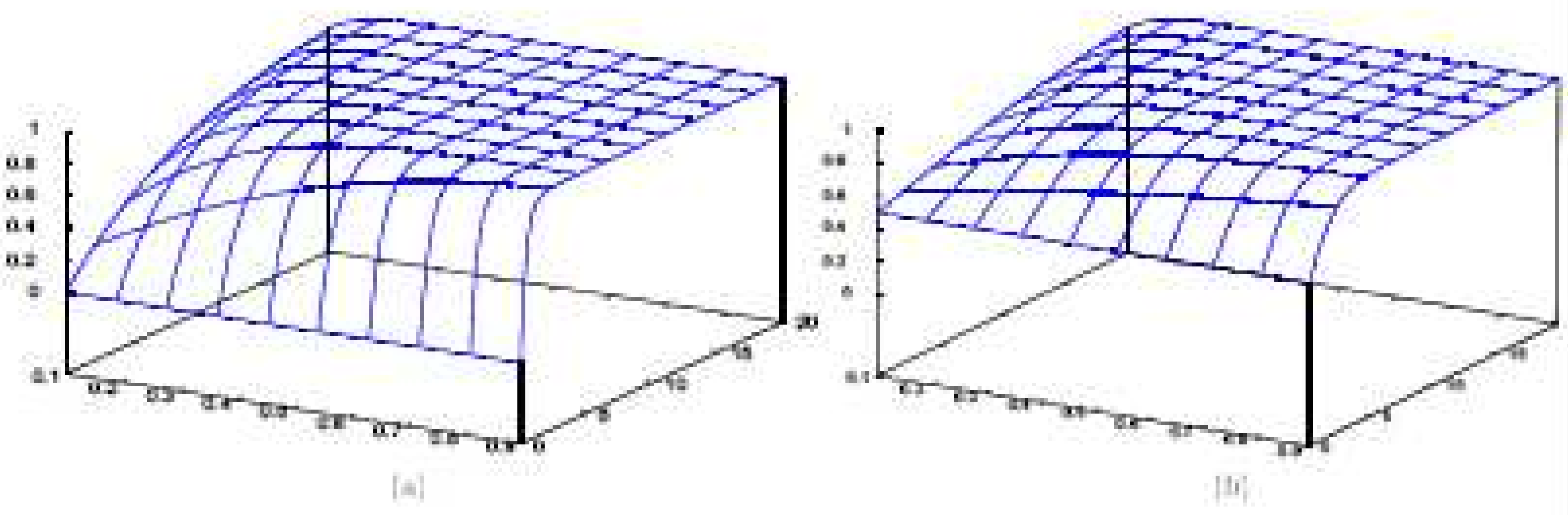
Example

- P(fever=0|cold=1, flu=0, malaria=0)=0.6
- P(fever=0|cold=0, flu=1, malaria=0)=0.2
- P(fever=0|cold=0, flu=0, malaria=1)=0.1

Cold	Flu	Malaria	p(Fever=1)	p(Fever=0)
0	0	0	0.0	1.0
0	0	1	0.0	0.1
0	1	0	0.8	0.2
0	1	1	0.98	$0.02 = 0.2 \times 0.1$
1	0	0	0.4	0.6
1	0	1	0.94	$0.0.6 = 0.6 \times 0.1$
1	1	0	0.88	$0.12 = 0.6 \times 0.2$
1	1	1	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

Leak nodes

- If Y=0 and all Xi=0, the CPD assigns 0 probability to this event. To prevent this, we add a leak node, X0=1, which is always on, to model "any other cause". The leak can fail wp w0.

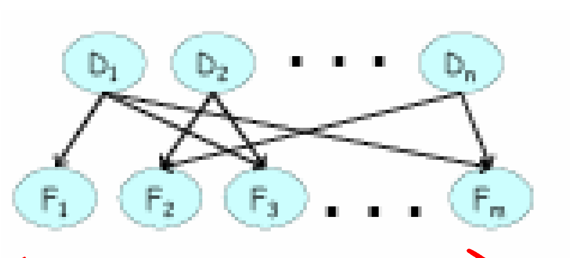


W0=1 W0=0.5

BN20 networks

- In medical diagnosis, it is common to construct 2 layered bipartite networks of binary nodes, mapping diseases to symptoms (findings).
- Because of the large number of parents, the child nodes use noisy-or.
- Conditional on F, the diseases D are correlated.
- The QMR-DT network is a standard testbed for evaluating approximate inference algorithms.

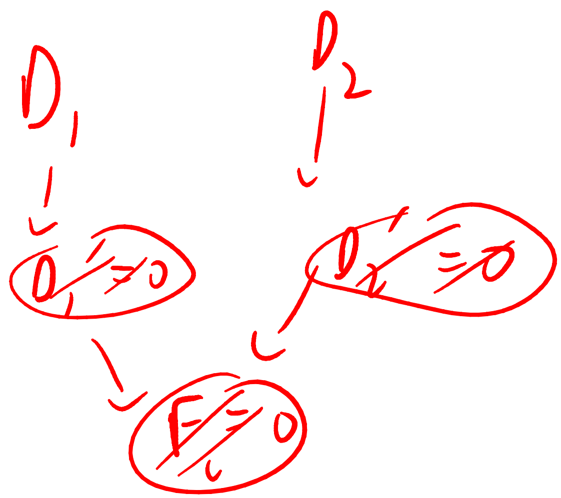
500



4000

Negative findings

- If Fi=1, the disease parents fight to explain the finding. Hence they become fully correlated.
- But if Fi=0, the parents are independent! Hence the p(Fi=0|Pa(Fi)) likelihood fully factorizes, and does not make inference harder (homework).



$D_1 \perp D_2 \mid F_i = 0$

Conditional linear Gaussian CPDs

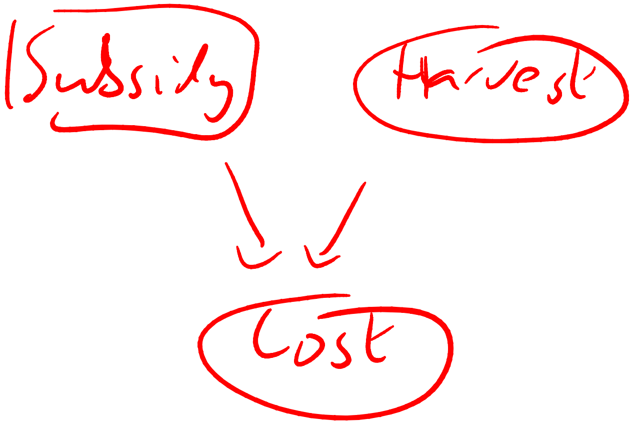
- If Y is continuous and all the parents are cts we can define

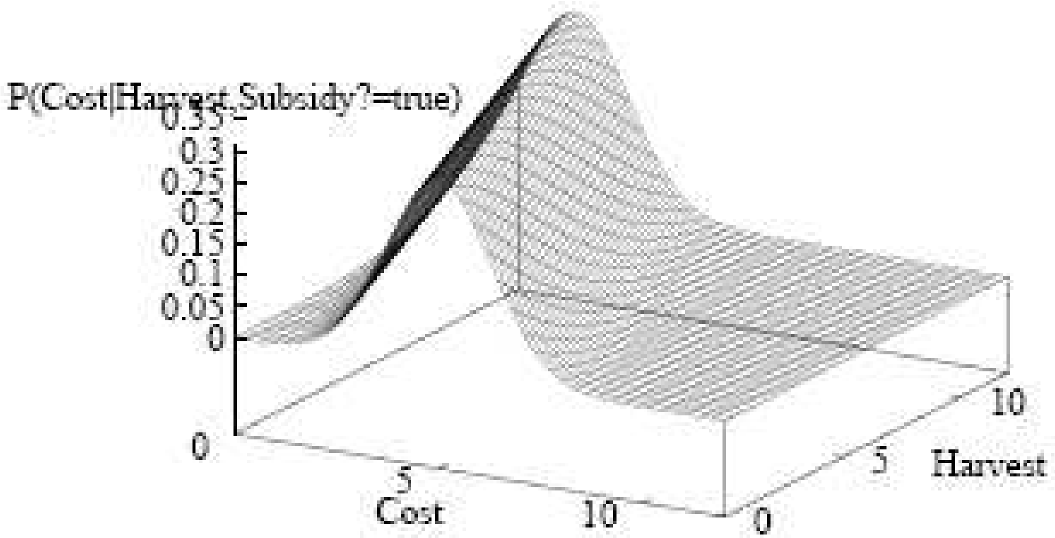
$$p(y|\mathbf{x}) = \mathcal{N}(y|\mathbf{x}^T\mathbf{w}, \sigma^2)$$

- Networks of linear Gaussian CPDs define a joint multivariate Gaussian (see ch 7)
- For discrete parents u, we can use 1-of-K and LG, or we can use a different set of parameters for each discrete setting (CLG). The resulting distribution is a mixture of Gaussians, where each discrete setting defines a mixture component.

$$p(y|\mathbf{x}, \mathbf{u} = k) = \mathcal{N}(y|\mathbf{x}^T\mathbf{w}_k, \sigma_k^2)$$

Example of CLG network

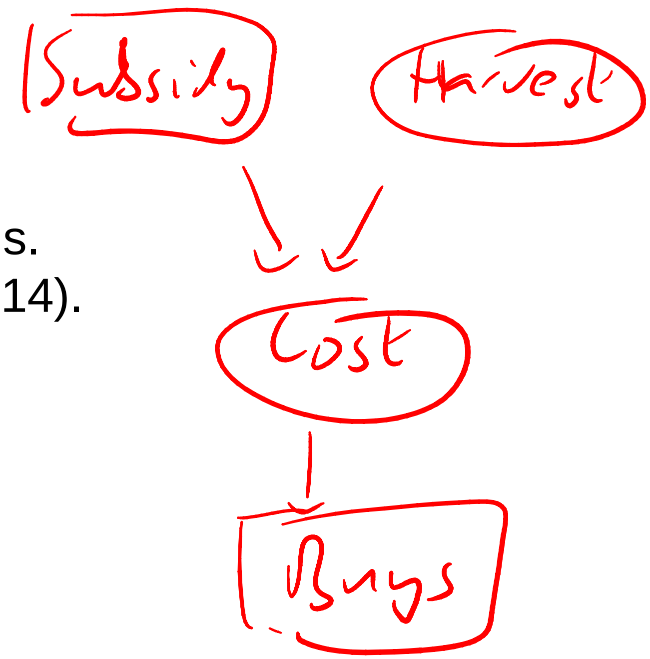




Russell & Norvig, p502

Hybrid network

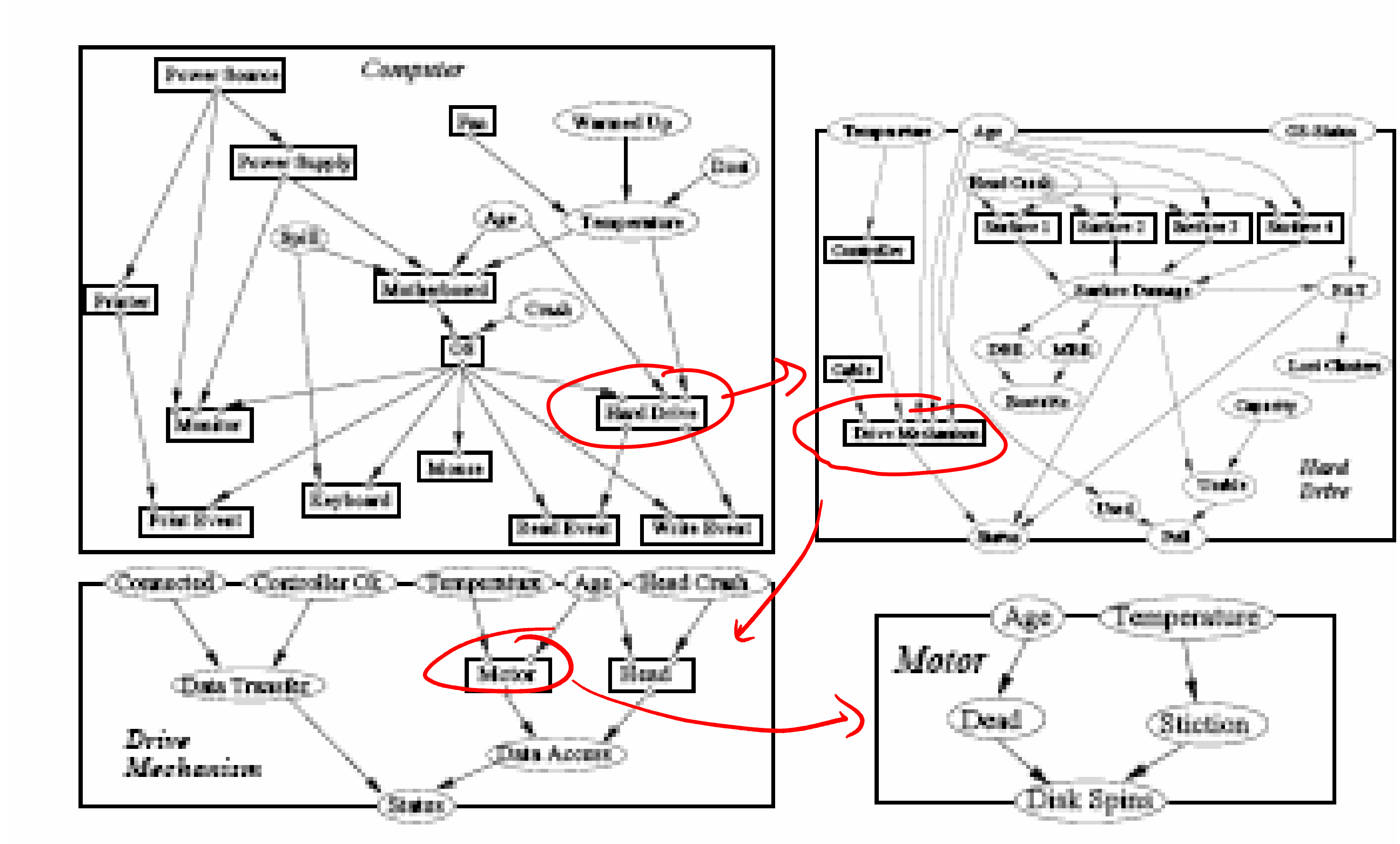
P(buys=1|cost) = logreg or probit.
Joint distribution is no longer mixture of Gaussians.
Closed-form inference no longer possible (see ch14).



Russell & Norvig, p502

Encapsulated BNs

- We can embed a BN inside a CPD, and “hide” the internal nodes using an interface layer.
- This, combined with parameter tying, yields OOBN.



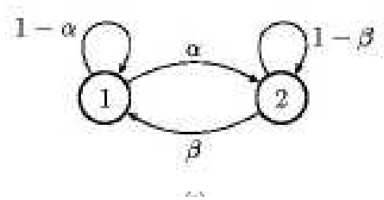
Markov chains

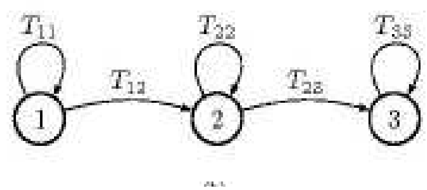
- We can define a distribution over a semi-infinite sequence X_1, X_2, … by using a discrete-time Markov chain with tied parameters (stationary)

$$
\begin{aligned}
p(\mathbf{x}|\boldsymbol{\theta}) &= p(x_1|\pi) \prod_{t=T}^{\infty} p(X_t|X_{t-1}, A) \\
A(i,j) &= p(X_t = j | X_{t-1} = i)
\end{aligned}
$$

State transition diagram

Picture of the stochastic finite state automaton





$$T = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}$$

$$T = \begin{pmatrix} T_{11} & T_{12} & 0 \\ 0 & T_{22} & T_{23} \\ 0 & 0 & 1 \end{pmatrix}$$

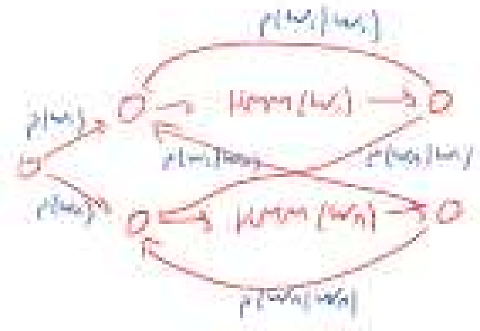
Hidden Markov Models

- An HMM is a function of a Markov chain.
- We observe V_t, hidden state is H_t in {1,…,K}
- $P(H_t=j|H_{t-1}=i)$ is the transition model
- $P(V_t|H_t=j)$ is the observation model (eg mixture of Gaussians)

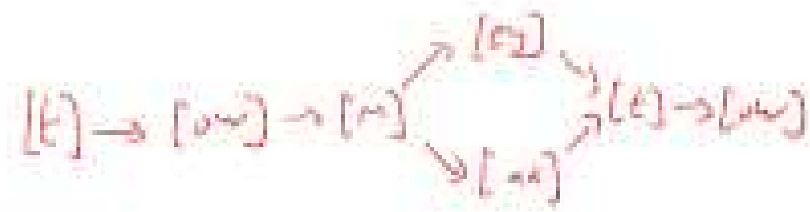
$$\begin{tikzcd} H_1 \arrow[r] \arrow[d] & H_2 \arrow[r] \arrow[d] & \cdots \\ V_1 & V_2 & \end{tikzcd}$$

HMMs for speech recognition

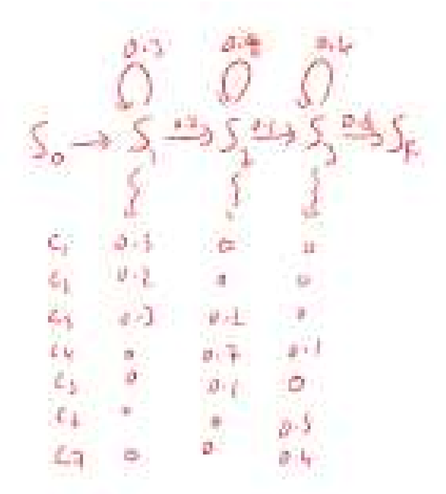
Bigram model of words

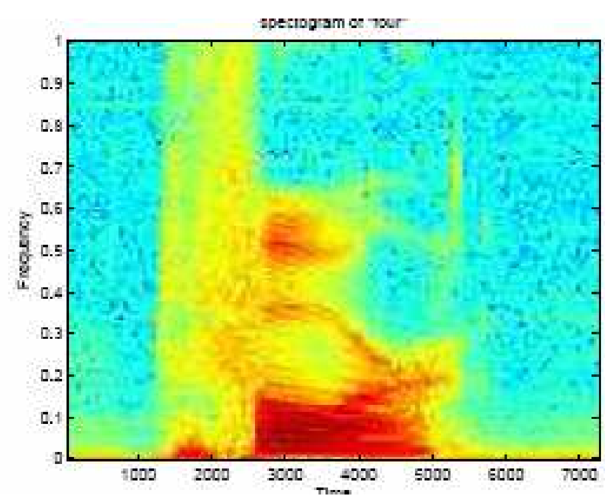


Pronunciation model : word -> phonemes



Acoustic model: phonemes -> observations





State space models

- Same graph (CI assumptions) as HMM, but now X and Y are real-valued vectors
- Special case: linear dynamical system (LDS)

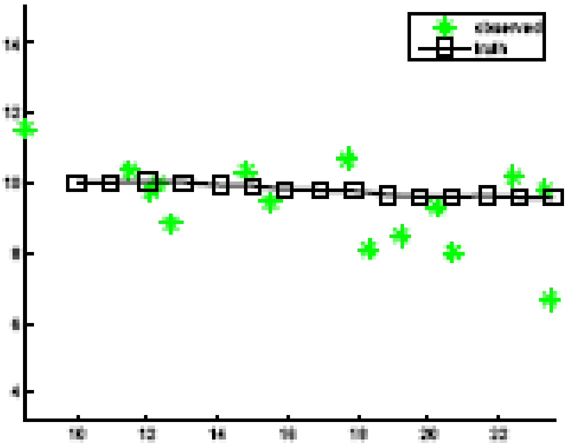
$$
\begin{aligned}
p(\mathbf{x}_t|\mathbf{x}_{t-1}) &= \mathcal{N}(\mathbf{x}_t|\mathbf{A}\mathbf{x}_{t-1}, \mathbf{Q}) \\
p(\mathbf{y}_t|\mathbf{x}_t) &= \mathcal{N}(\mathbf{y}_t|\mathbf{H}\mathbf{x}_t, \mathbf{R})
\end{aligned}
$$

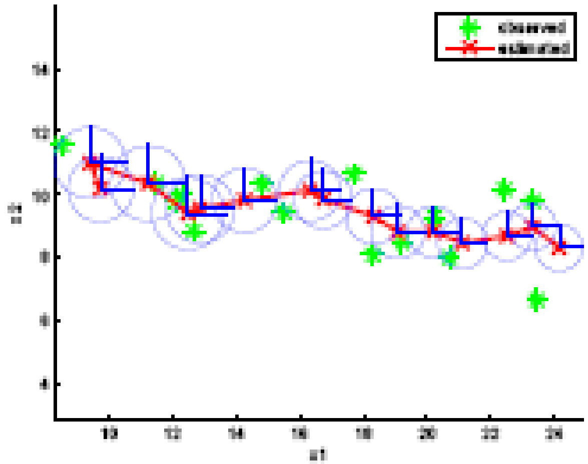
$$
\begin{aligned}
\mathbf{x}_t &= \mathbf{A}\mathbf{x}_{t-1} + \mathcal{N}(\mathbf{0}, \mathbf{Q}) \\
\mathbf{y}_t &= \mathbf{H}\mathbf{x}_t + \mathcal{N}(\mathbf{0}, \mathbf{R})
\end{aligned}
$$

Example: tracking in 2D

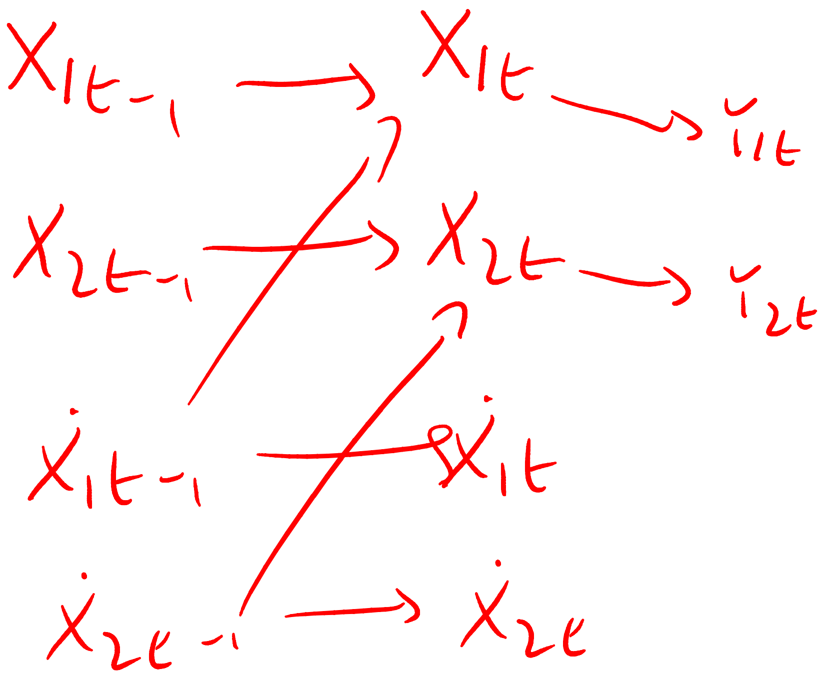
$$
\begin{pmatrix} x_{1t} \\ x_{2t} \\ \dot{x}_{1t} \\ \dot{x}_{2t} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} x_{1t-1} \\ x_{2t-1} \\ \dot{x}_{1t-1} \\ \dot{x}_{2t-1} \end{pmatrix} + \begin{pmatrix} w_{1t} \\ w_{2t} \\ w_{3t} \\ w_{4t} \end{pmatrix}
$$

$$
\begin{pmatrix} y_{1t} \\ y_{2t} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} x_{1t} \\ x_{2t} \\ \dot{x}_{1t} \\ \dot{x}_{2t} \end{pmatrix} + \begin{pmatrix} v_{1t} \\ v_{2t} \\ v_{3t} \\ v_{4t} \end{pmatrix}
$$





LDS as DGM

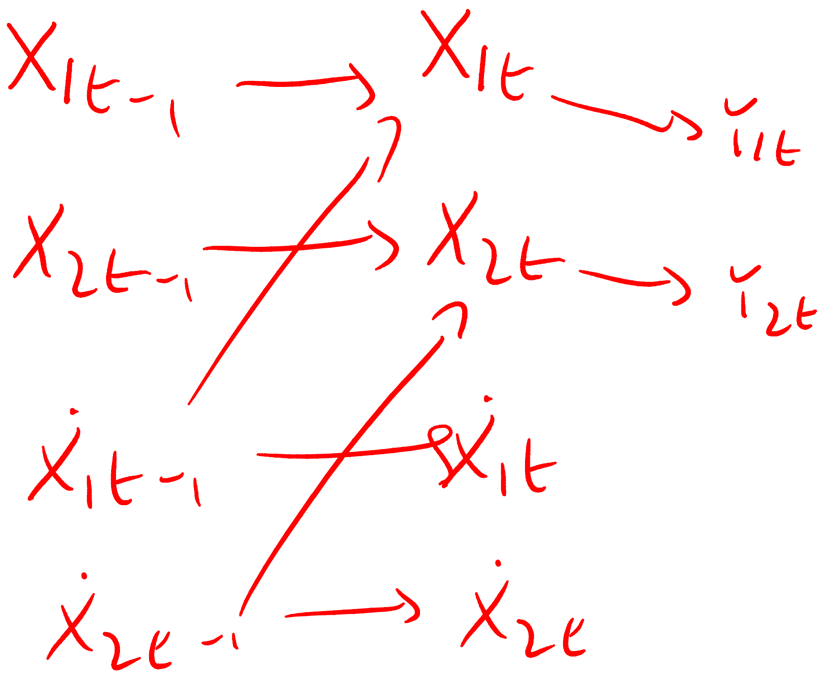


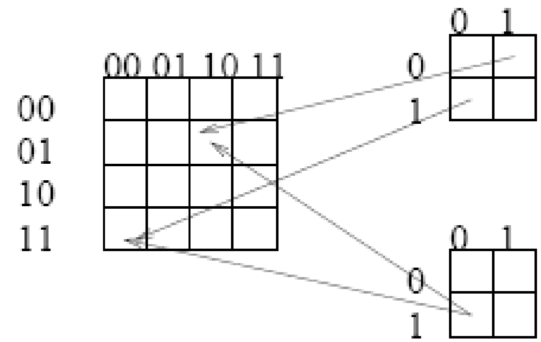
$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

For linear Gaussian systems, sparse matrices = sparse graphs

Dynamic Bayes Nets





P(X1(t),X2(t) | X1(t-1),X2(t-1))

If the variables are discrete, the transition matrix of the compound model (all 4 variables) is not sparse or structured. So the graph structure is crucial.

See ch 15