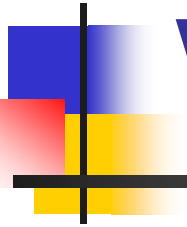
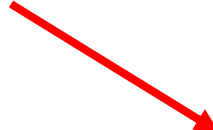


Probabilistic Reasoning wrt Time





Decision Theoretic Agents

- Introduction to Probability [Ch13]
- Belief networks [Ch14]
-  Dynamic Belief Networks [Ch15]
 - Foundations
 - Markov Chains (Classification)
 - Hidden Markov Models (HMM)
 - Kalman Filter
 - General: Dynamic Belief Networks (DBN)
 - Applications
 - Future Work, Extensions, ...
- Single Decision [Ch16]
- Sequential Decisions [Ch17]
- Game Theory [Ch 17.6 – 17.7]

Markovian Models

- In general, X_{t+1} depends on everything ! ... X_t, X_{t-1}, \dots
- Markovian means...

Future is independent of the past once you know the present.

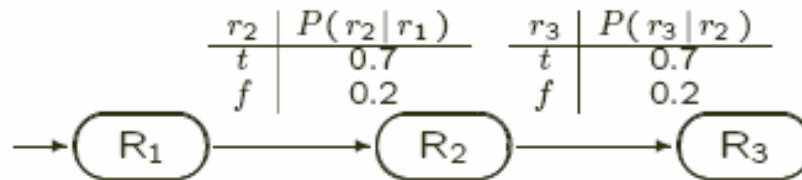
$$P(X_{t+1} | X_t, X_{t-1}, \dots) = P(X_{t+1} | X_t)$$

- Markov Chain: "state" (everything important) is visible

$$P(x_{t+1} | x_t, \langle \text{everything} \rangle) = P(x_{t+1} | x_t)$$

- Eg: First-Order Markov Chain

1. Random Walk along x axis, changing x-position ± 1 at each time
2. Predicting rain



- Stationarity:

$$P(x_2 | x_1) = P(x_3 | x_2) = \dots = P(x_{t+1} | x_t)$$

- Hidden Markov Model: State information not visible

Using Markov Chain, for Classification

- Two classes of DNA...
different di-nucleotide distribution

$$a_{i,j}^+ = P(x_i \mapsto x_j | +) = p^+(X_{t+1} = x_j | X_t = x_i)$$

$$a_{i,j}^- = P(x_i \mapsto x_j | -) = p^-(X_{t+1} = x_j | X_t = x_i)$$

	+	A	C	G	T	-	A	C	G	T
A		0.180	0.274	0.426	0.120	A	0.300	0.205	0.285	0.210
C		0.171	0.368	0.274	0.188	C	0.322	0.298	0.078	0.302
G		0.161	0.339	0.375	0.125	G	0.248	0.246	0.298	0.208
T		0.079	0.355	0.384	0.182	T	0.177	0.239	0.292	0.292

- Use this to classify a nucleotide sequence
 $x = \langle \text{ACATTGACCA...} \rangle$

A: $P(x | +) =$

$$p^+(x_1 |) p^+(x_2 | x_1) p^+(x_3 | x_2) \dots p^+(x_k | x_{k-1}) = \prod_{i=1}^k p^+(x_i | x_{i-1}) = \prod_{i=1}^k a_{x_i | x_{i-1}}^+$$

using Markov properties

Using Markov Chain, for Classification

$$a_{i,j}^+ = P(x_i \mapsto x_j | +) = p^+(X_{t+1} = x_j | X_t = x_i)$$

$$a_{i,j}^- = P(x_i \mapsto x_j | -) = p^-(X_{t+1} = x_j | X_t = x_i)$$

+	A	C	G	T
A	0.180	0.274	0.426	0.120
C	0.171	0.368	0.274	0.188
G	0.161	0.339	0.375	0.125
T	0.079	0.355	0.384	0.182

-	A	C	G	T
A	0.300	0.205	0.285	0.210
C	0.322	0.298	0.078	0.302
G	0.248	0.246	0.298	0.208
T	0.177	0.239	0.292	0.292

- Is $x = \langle \text{ACATTGACCAT} \rangle$ positive?

$$\begin{aligned} P(x | +) &= p^+(x_1 |) p^+(x_2 | x_1) p^+(x_3 | x_2) \dots p^+(x_k | x_{k-1}) \\ &= p^+(A) p^+(C | A) p^+(A | C) \dots p^+(T | A) \\ &= 0.25 \times 0.274 \times 0.171 \times \dots \times 0.355 \end{aligned}$$

$$\begin{aligned} P(x | -) &= p^-(x_1 |) p^-(x_2 | x_1) p^-(x_3 | x_2) \dots p^-(x_k | x_{k-1}) \\ &= p^-(A) p^-(C | A) p^-(A | C) \dots p^-(T | A) \\ &= 0.25 \times 0.205 \times 0.322 \times \dots \times 0.239 \end{aligned}$$

- Pick larger: + if $p(x | +) > p(x | -)$

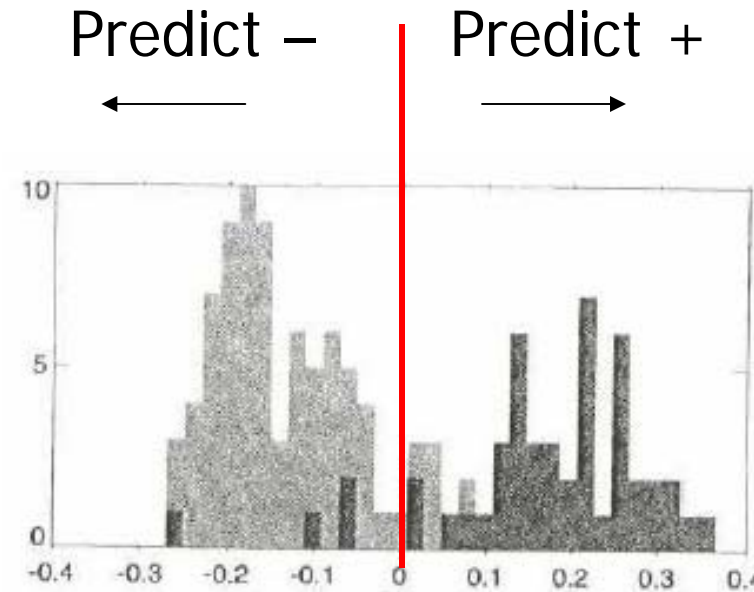
Results (Markov Chain)

- $$S(x) = \log \frac{P(x|+)}{P(x|-)} = \sum_{i=1}^k \log \frac{a_{x_{i-1},x_i}^+}{a_{x_{i-1},x_i}^-} = \sum_{i=1}^k \beta_{x_{i-1},x_i}$$

where $\beta_{i,j} = \log \frac{a_{ij}^+}{a_{ij}^-}$

β	A	C	G	T
A	-0.740	0.419	0.580	-0.803
C	-0.913	0.302	1.812	-0.685
G	-0.624	0.461	0.331	-0.730
T	-1.169	0.573	0.393	-0.679

- Results over 48 sequences:
- Here: everything is visible
- Sometimes, can't see the "states"

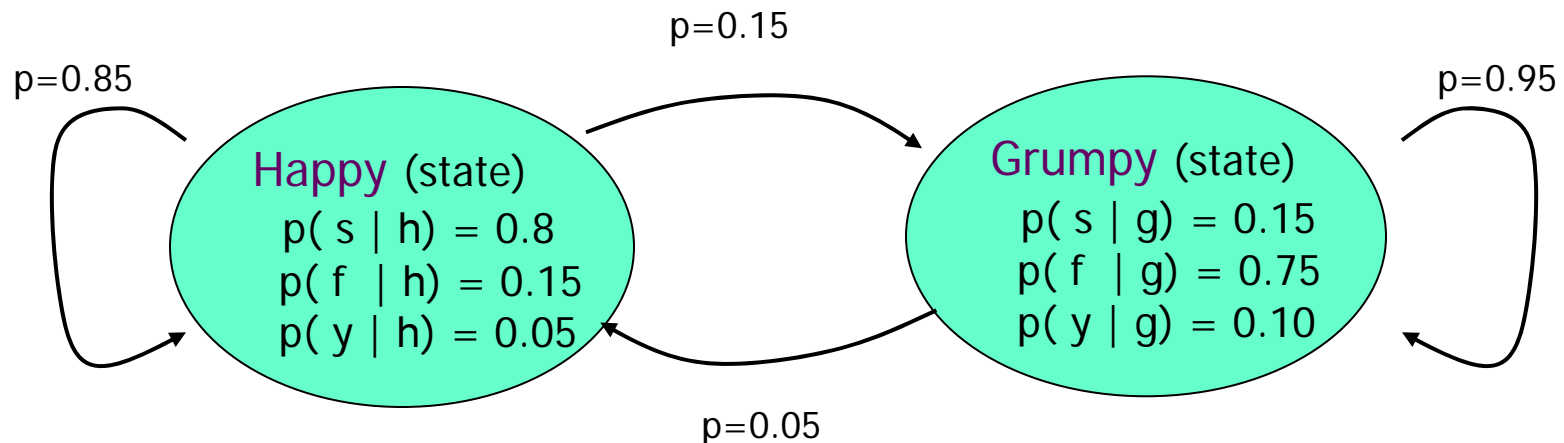


Phydeaux, the Dog

- Sometimes: *Grumpy*
- Sometimes: *Happy*
- But hides emotional state...
Only observations:
 $\{ \textit{slobbers}, \textit{frowns}, \textit{yelps} \}$

Known Correlations

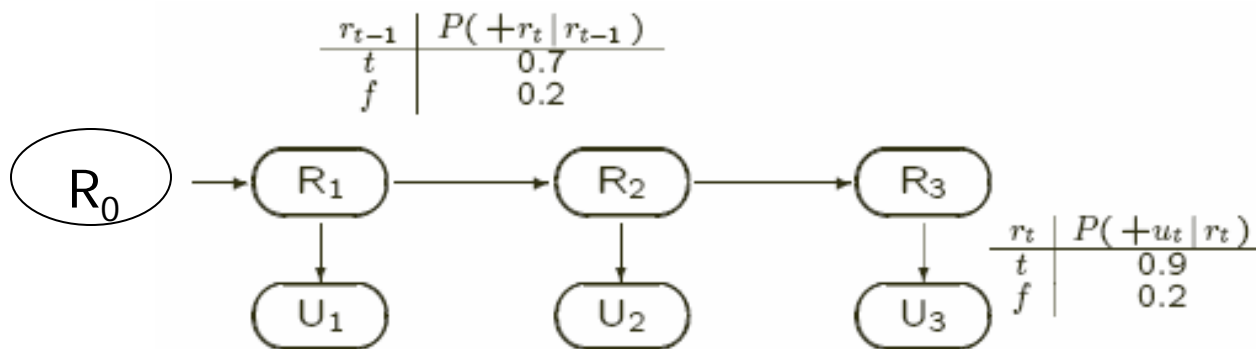
- State $\{ G, H \}$ to Observations $\{ s, f, y \}$
- State $\{ G, H \}$ on day t to state $\{ G, H \}$ on day $t+1$



- Challenge: Given observation sequence: $\langle s, s, f, y, y, f, \dots \rangle$
what were Phydeaux's states? ?? $\langle H, H, H, G, G, G, \dots \rangle$

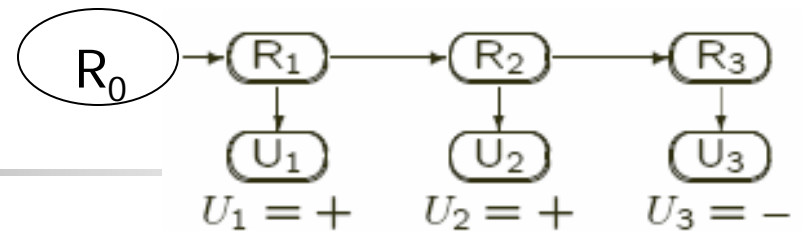
Umbrella+Rain Situation

- State: $X_t \in \{ +rain, -rain \}$
- Observation: $E_t \in \{ +umbrella, -umbrella \}$
- Simple Belief Net:



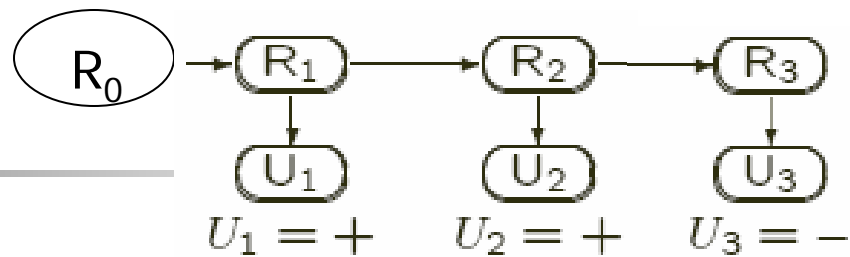
- Note: $Umbrella_t$ depends only on $Rain_t$
 $Rain_t$ depends only on $Rain_{t-1}$

HMM Tasks



1. Filtering / Monitoring: $\mathbf{P}(X_t | e_{1:t})$
 - What is $\mathbf{P}(R_3 = + | U_1 = +, U_2 = +, U_3 = -)$?
 - Need distr. current state to make rational decisions
2. Prediction: $\mathbf{P}(X_{t+k} | e_{1:t})$
 - What is $\mathbf{P}(R_5 = - | U_1 = +, U_2 = +, U_3 = -)$?
 - Use to evaluate possible courses of actions
3. Smoothing / Hindsight: $\mathbf{P}(X_{t-k} | e_{1:t})$
 - What is $\mathbf{P}(R_1 = - | U_1 = +, U_2 = +, U_3 = -)$?
4. Likelihood: $\mathbf{P}(e_{1:t})$
 - What is $\mathbf{P}(U_1 = +, U_2 = +, U_3 = -)$?
 - For comparing different models ... classification
5. Most likely expl'n: $\mathbf{argmax}_{x_{1:t}} \{ \mathbf{P}(x_{1:t} | e_{1:t}) \}$
 - Given $\langle U_1 = +, U_2 = +, U_3 = - \rangle$, what is most likely value for $\langle R_1, R_2, R_3 \rangle$?
 - Compute assignments, for DNA, sounds, ...

1. Filtering



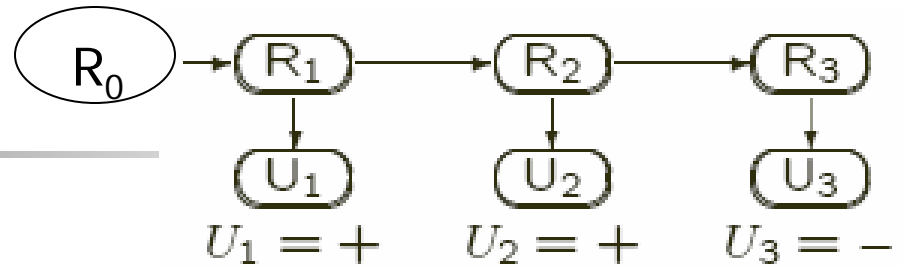
- At time 3: have

- $P(R_2 | u_{1:2}) = \langle P(+r_2 | ++), P(-r_2 | ++) \rangle$
- ... then observe $u_3 = -$

- $$\begin{aligned}
 P(R_3 | u_{1:3}) &= P(R_3 | u_{1:2}, u_3) \\
 &= 1/P(u_{1:3}) P(u_3 | R_3, u_{1:2}) P(R_3 | u_{1:2}) \\
 &= 1/P(u_{1:3}) P(u_3 | R_3) P(R_3 | e_{1:2})
 \end{aligned}$$

- $$\begin{aligned}
 P(R_3 | e_{1:2}) &= \sum_{r_2} P(R_3, r_2 | e_{1:2}) \\
 &= \sum_{r_2} P(R_3 | r_2, e_{1:2}) P(r_2 | e_{1:2}) \\
 &= \sum_{r_2} P(R_3 | r_2) P(r_2 | e_{1:2})
 \end{aligned}$$

1. Filtering



- At time t :
 - have $P(X_t | e_{1:t})$
 - ... then update from e_{t+1}

- $P(X_{t+1} | e_{1:t+1}) =$

$$\alpha P(e_{t+1} | X_{t+1}) \sum_{x_t} P(X_{t+1} | x_t) P(x_t | e_{1:t})$$

Emission Prob's

Transition Prob's

distribution wrt time t

- Called "**Forward Algorithm**"



$P(x_t, e_{1:t})$ vs $P(x_t | e_{1:t})$

To compute $P(X_t=a | e_{1:t})$:

Just compute

$$\langle P(X_t=1, e_{1:t}), \dots, P(X_t=k, e_{1:t}) \rangle$$

1. Compute $P(e_{1:t}) = \sum_i P(X_t=i, e_{1:t})$
2. Return $P(X_t=a | e_{1:t})$
 $= P(X_t=a, e_{1:t}) / P(e_{1:t})$
 $= P(X_t=a, e_{1:t}) / \sum_i P(X_t=i, e_{1:t})$

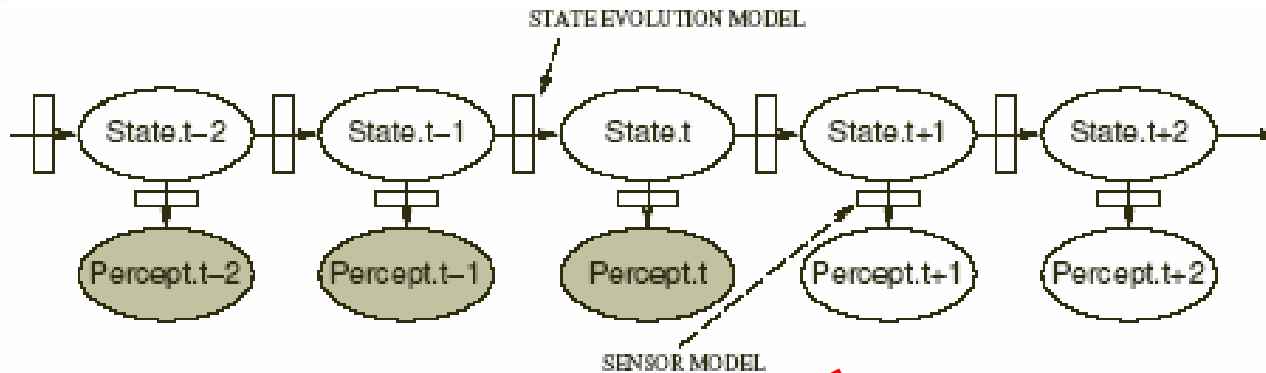
Normalizing constant: $\alpha = 1 / P(e_{1:t})$

Filtering – Forward Algorithm

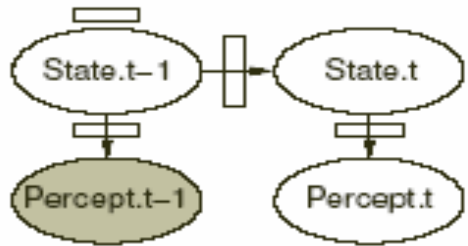
- Let $f_{1:t} = P(X_t | e_{1:t})$
 $= \langle P(X_t = 1 | e_{1:t}), \dots, P(X_t = r | e_{1:t}) \rangle$
 $f_{1:t+1}(\mathbf{x}_{t+1}) = P(x_{t+1} | e_{1:t+1})$
 $= \alpha P(e_{t+1} | x_{t+1}) \sum_{x_t} P(X_{t+1} | x_t) \mathbf{f}_{1:t}(\mathbf{x}_t)$
- $f_{1:t+1} = \alpha \textit{Forward}(f_{1:t+1}, e_{t+1})$
- Update (for discrete state variables):
Constant time & Constant space!

Detached!

Filtering Process

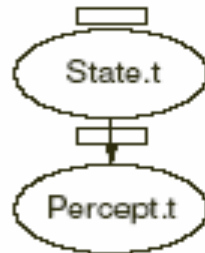


(a) Prediction



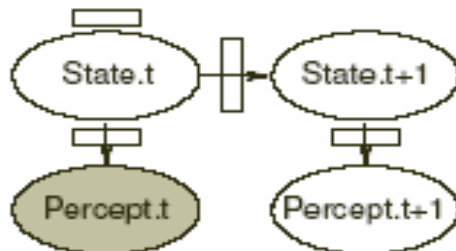
State.t from State.t-1

(b) Rollup



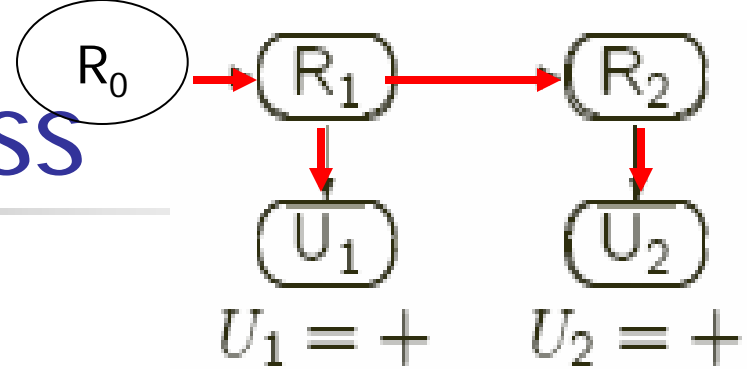
State.t from Percept.t

(c) Estimation



State.t+1 from State.t

Forward() Process



- Given: $P(R_0) = \langle 0.5, 0.5 \rangle$
Evidence $\langle U_1 = +, U_2 = + \rangle$:
- Predict state distribution** (before evidence)

$$P(R_1) = \sum_{r_0} P(R_1 | r_0) P(r_0)$$

$$= \langle 0.7, 0.3 \rangle \times 0.5 + \langle 0.2, 0.8 \rangle \times 0.5 = \langle 0.45, 0.55 \rangle$$
- Incorporate** "Day 1 evidence" $+u_1$:

$$P(R_1 | +u_1) = \alpha P(+u_1 | R_1) P(R_1)$$

$$= \alpha \langle 0.9, 0.2 \rangle .* \langle 0.45, 0.55 \rangle = \alpha \langle 0.405, 0.11 \rangle \approx \langle 0.786, 0.214 \rangle$$
- Predict** (from $t = 1$ to $t = 2$, before new evidence)

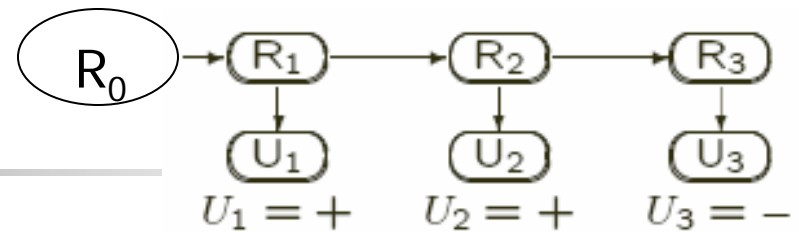
$$P(R_2 | +u_1) = \sum_{r_1} P(R_2 | r_1) P(r_1 | +u_1)$$

$$= \langle 0.7, 0.3 \rangle 0.786 + \langle 0.2, 0.8 \rangle 0.214 \approx \langle 0.593, 0.407 \rangle$$
- Incorporate** "Day 2 evidence" $+u_2$:

$$P(R_2 | +u_1, +u_2) = P(+u_2 | R_2) P(R_2 | +u_1) =$$

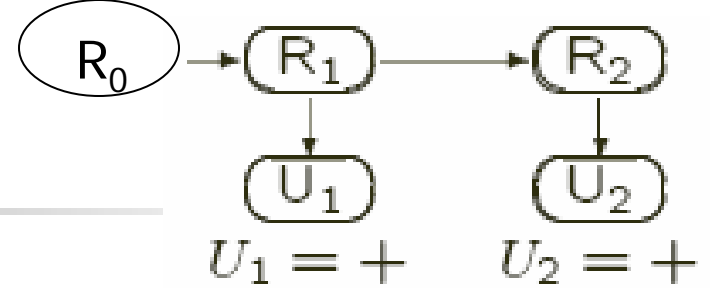
$$\alpha \langle 0.9, 0.2 \rangle .* \langle 0.609, 0.391 \rangle = \alpha \langle 0.533, 0.081 \rangle \approx \langle 0.868, 0.132 \rangle$$

HMM Tasks



1. Filtering / Monitoring: $\mathbf{P}(X_t | e_{1:t})$
 - What is $\mathbf{P}(R_3 = + | U_1 = +, U_2 = +, U_3 = -)$?
 - Need distr. current state to make rational decisions
2. Prediction: $\mathbf{P}(X_{t+k} | e_{1:t})$
 - What is $\mathbf{P}(R_5 = - | U_1 = +, U_2 = +, U_3 = -)$?
 - Use to evaluate possible courses of actions
3. Smoothing / Hindsight: $\mathbf{P}(X_{t-k} | e_{1:t})$
 - What is $\mathbf{P}(R_1 = - | U_1 = +, U_2 = +, U_3 = -)$?
4. Likelihood: $\mathbf{P}(e_{1:t})$
 - What is $\mathbf{P}(U_1 = +, U_2 = +, U_3 = -)$?
 - For comparing different models ... classification
5. Most likely expl'n: $\mathbf{argmax}_{x_{1:t}} \{ \mathbf{P}(x_{1:t} | e_{1:t}) \}$
 - Given $\langle U_1 = +, U_2 = +, U_3 = - \rangle$, what is most likely value for $\langle R_1, R_2, R_3 \rangle$?
 - Compute assignments, for DNA, sounds, . . .

4. Likelihood



- How to compute **likelihood** $P(e_{1:t})$?

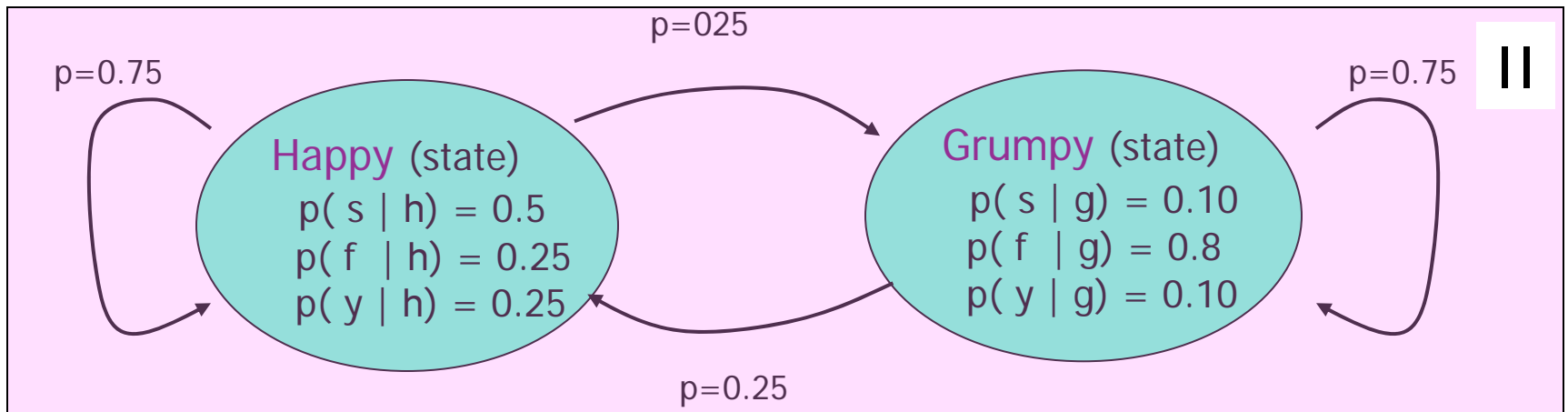
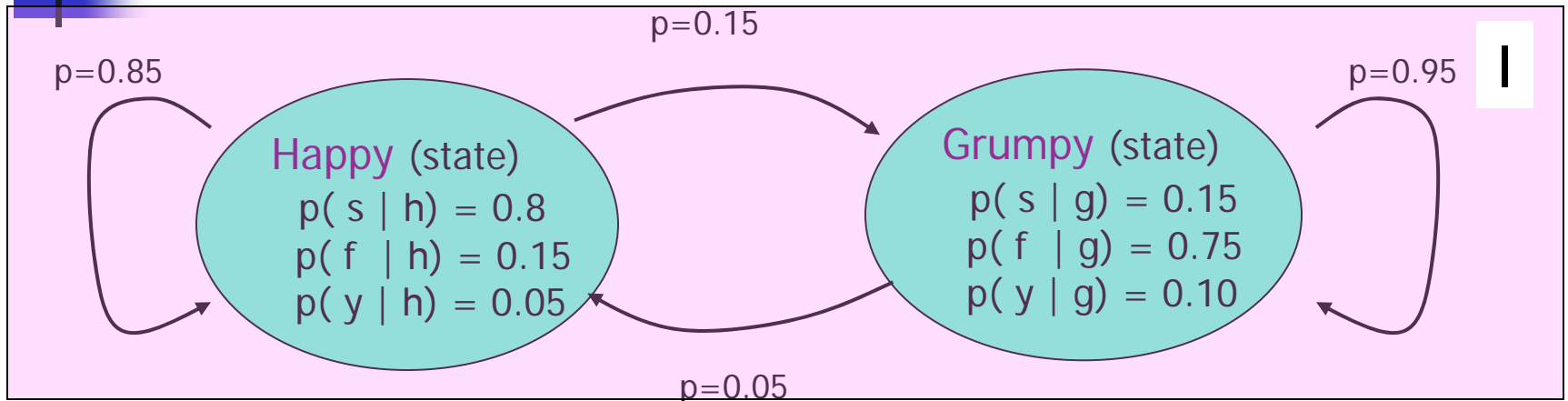
- Let $L_{1:t} = P(X_t, e_{1:t})$

$$\begin{aligned} L_{1:t+1} &= P(X_{t+1}, e_{1:t+1}) = \sum_{x_t} P(x_t, X_{t+1}, e_{1:t}, e_{t+1}) \\ &= \sum_{x_t} P(e_{t+1} | X_{t+1}, x_t, e_{1:t}) P(X_{t+1} | x_t, e_{1:t}) P(x_t, e_{1:t}) \\ &= P(e_{t+1} | X_{t+1}) \sum_{x_t} P(X_{t+1} | x_t) L_{1:t}(x_t) \end{aligned}$$

- Note: Same *Forward()* algorithm!!
- To compute actual likelihood:

$$P(e_{1:t}) = \sum_{x_t} P(X_t = x_t, e_{1:t}) = \sum_{x_t} L_{1:t}(x_t)$$

Best Model of Phydeaux?



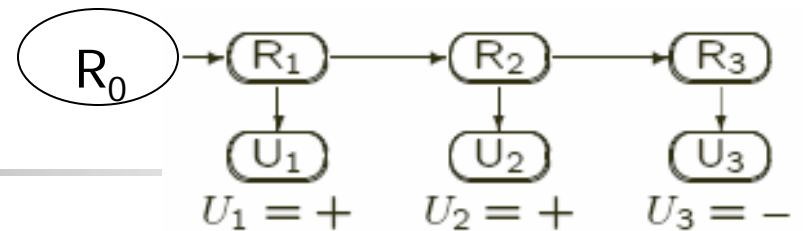
■ Challenge: Given observation sequence: $\langle s, s, f, y, y, f, \dots \rangle$
 which model of Phydeaux is "correct"?? Want $P_I(\mathbf{e})$ vs $P_{II}(\mathbf{e})$



Use HMMs to Classify Words in Speech Recognition

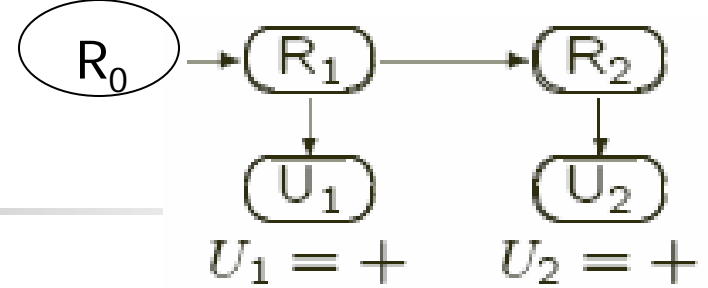
- Use one HMM for each word
 - hmm_j for j^{th} word
- Convert acoustic signal to sequence of fixed duration frames (eg, 60ms)
(Assumes know start/end of each word in speech signal)
- Map each frame to nearest “codebook” frame (discrete symbol x_t)
 - $e_{1:T} = \langle e_1, \dots, e_n \rangle$
- To classify sequence of frames $e_{1:T}$
 - 1. Compute $P(e_{1:T} | hmm_j)$ likelihood $e_{1:T}$ generated by each word hmm_j
 - 2. Return $\text{argmax}_j \{ P(e_{1:T} | hmm_j) \}$
word#j whose hmm_j gave highest likelihood

HMM Tasks



1. Filtering / Monitoring: $\mathbf{P}(\mathbf{X}_t \mid \mathbf{e}_{1:t})$
 - What is $\mathbf{P}(R_3 = + \mid U_1 = +, U_2 = +, U_3 = -)$?
 - Need distr. over current state to make rational decisions
2. Prediction: $\mathbf{P}(\mathbf{X}_{t+k} \mid \mathbf{e}_{1:t})$
 - What is $\mathbf{P}(R_5 = - \mid U_1 = +, U_2 = +, U_3 = -)$?
 - Use to evaluate possible courses of actions
3. Smoothing / Hindsight: $\mathbf{P}(\mathbf{X}_{t-k} \mid \mathbf{e}_{1:t})$
 - What is $\mathbf{P}(R_1 = - \mid U_1 = +, U_2 = +, U_3 = -)$?
4. Likelihood: $\mathbf{P}(\mathbf{e}_{1:t})$
 - What is $\mathbf{P}(U_1 = +, U_2 = +, U_3 = -)$?
 - For comparing different models ... classification
5. Most likely expl'n: $\mathbf{argmax}_{\mathbf{x}_{1:t}} \{ \mathbf{P}(\mathbf{x}_{1:t} \mid \mathbf{e}_{1:t}) \}$
 - Given $\langle U_1 = +, U_2 = +, U_3 = - \rangle$, what is most likely value for $\langle R_1, R_2, R_3 \rangle$?
 - Compute assignments, for DNA, sounds, ...

2. Prediction



- Already have 1 step prediction

Prediction (from $t = 1$ to $t = 2$, before new evidence)

$$P(R_2 \mid +u_1) = \sum_{r_1} P(R_2 \mid r_1) P(r_1 \mid +u_1) = \dots \approx \langle 0.627, 0.373 \rangle$$

- **Prediction** \equiv **filtering w/o incorporating new evidence**

Using transition info, but not observation info

$$P(X_{t+k+1} \mid e_{1:t}) = \sum_{x_{t+k}} P(X_{t+k+1} \mid x_{t+k}) P(x_{t+k} \mid e_{1:t})$$

- Converge to stationary distribution $P(Y \mid e)$

fixed-point: $P(Y \mid e) = \sum_x P(Y \mid x) P(x \mid e)$

here $\langle 0.5, 0.5 \rangle$

Mixing time \approx #steps until reach fixed point

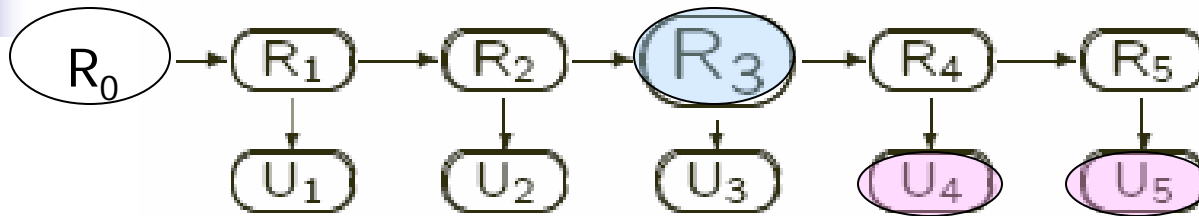
\Rightarrow Prediction meaningless unless $k \approx$ mixing-time

More "mixing" in transitions

\Rightarrow shorter mixing time,

harder to predict future

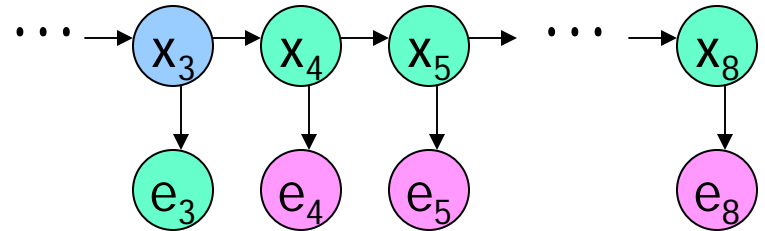
3. Smoothing / Hindsight



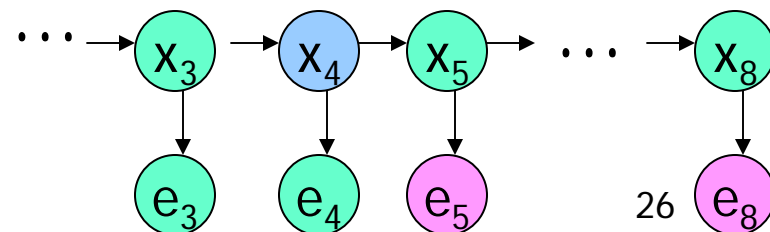
- Given $\langle +u_1, +u_2, -u_3, +u_4, -u_5 \rangle$, what is best estimate of r_3 ?
 $P(R_3 \mid +u_1, +u_2, -u_3, +u_4, -u_5)$
- Let $f_{1:k} = P(X_k \mid e_{1:k})$ $b_{k+1:t} = P(e_{k+1:t} \mid X_k)$
 $P(X_k \mid e_{1:t}) = P(X_k \mid e_{1:k}, e_{k+1:t})$
 $= \alpha P(X_k \mid e_{1:k}) P(e_{k+1:t} \mid X_k, e_{1:k})$
 $= \alpha P(X_k \mid e_{1:k}) P(e_{k+1:t} \mid X_k)$
 $= \alpha \quad f_{1:k} \quad b_{k+1:t}$
- Recursive computation for $f_{1:k}$... go forward: $1, 2, 3, \dots, k$
- Recursive computation for $b_{1:k}$... go backward: $T, T-1, \dots, k+1$

Smoothing – Backward Algorithm

$$\mathbf{b}_{4:8}(\mathbf{x}_3) = P(e_{4:8} \mid \mathbf{x}_3)$$



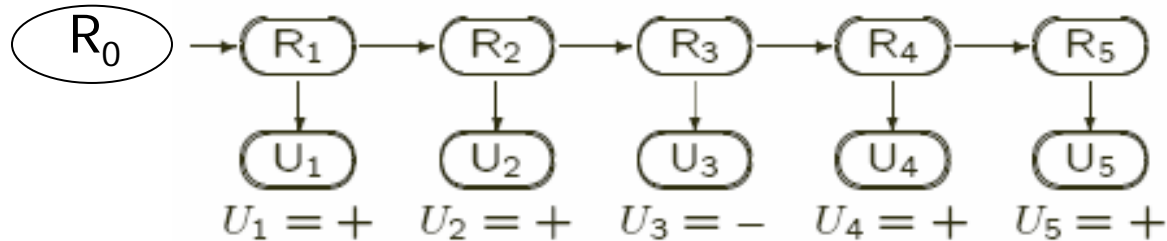
$$\begin{aligned} \mathbf{b}_{4:8}(\mathbf{x}_3) &= P(e_{4:8} \mid \mathbf{x}_3) \\ &= \sum_{\mathbf{x}_4} P(e_{4:8} \mid \mathbf{x}_3, \mathbf{x}_4) P(\mathbf{x}_4 \mid \mathbf{x}_3) \\ &= \sum_{\mathbf{x}_4} P(e_{4:8} \mid \mathbf{x}_4) P(\mathbf{x}_4 \mid \mathbf{x}_3) \\ &= \sum_{\mathbf{x}_4} P(\mathbf{e}_4, \mathbf{e}_{5:8} \mid \mathbf{x}_4) P(\mathbf{x}_4 \mid \mathbf{x}_3) \\ &= \sum_{\mathbf{x}_4} P(e_4 \mid \mathbf{x}_4) P(e_{5:8} \mid \mathbf{x}_4) P(\mathbf{x}_4 \mid \mathbf{x}_3) \\ &= \sum_{\mathbf{x}_4} P(e_4 \mid \mathbf{x}_4) \mathbf{b}_{5:8}(\mathbf{x}_4) P(\mathbf{x}_4 \mid \mathbf{x}_3) \end{aligned}$$



Smoothing – Backward Algorithm

- $\mathbf{b}_{k+1:t}(\mathbf{x}_k) = P(e_{k+1:t} \mid \mathbf{x}_k)$
 - $= \sum_{x_{k+1}} P(e_{k+1:t} \mid \mathbf{x}_k, x_{k+1}) P(x_{k+1} \mid \mathbf{x}_k)$
 - $= \sum_{x_{k+1}} P(e_{k+1:t} \mid x_{k+1}) P(x_{k+1} \mid \mathbf{x}_k)$
 - $= \sum_{x_{k+1}} P(e_{k+1}, e_{k+2:t} \mid x_{k+1}) P(x_{k+1} \mid \mathbf{x}_k)$
 - $= \sum_{x_{k+1}} P(e_{k+1} \mid x_{k+1}) P(e_{k+2:t} \mid x_{k+1}) P(x_{k+1} \mid \mathbf{x}_k)$
 - $= \sum_{x_{k+1}} P(e_{k+1} \mid x_{k+1}) \mathbf{b}_{k+2:t}(\mathbf{x}_{k+1}) P(x_{k+1} \mid \mathbf{x}_k)$
- So $\mathbf{b}_{k+1:t} = \textit{Backward}(\mathbf{b}_{k+1:t}, e_{k+2:t})$
- Initialize: $\mathbf{b}_{t+1:t}(x_t) = P(e_{t+1:t} \mid x_t) = 1$
- **“Forward-Backward Algorithm”**
 - Just **polytree belief net inference!**
- Fixed-lag smoothing $\langle P(X_t \mid e_{1:t+k}) \rangle_t$

5. Most Likely Explanation



- Given $\langle +u_1, +u_2, -u_3, +u_4, +u_5 \rangle$, which is most likely rain-sequence: Perhaps
 - ? $\langle +r_1, +r_2, +r_3, +r_4, +r_5 \rangle$
but forgot on day#3?
 - ? $\langle +r_1, +r_2, -r_3, -r_4, +r_5 \rangle$
but was too cautious on day#4?
 - ? ... 2^5 possibilities !
- ? Idea: Just use “**3. Smoothing**” ?

Use "Smoothing" for MLE ?

- ? Idea: Use "**3. Smoothing**" ?

For $i = 1..5$

Compute $P(R_1 | \mathbf{u})$

Let $r_i^* = \operatorname{argmax}_r \{ P(R_i = r | \mathbf{u}) \}$

Return $\langle r_1^*, \dots, r_5^* \rangle$

- Wrong! Just local... ignores interactions!
- Eg: Suppose

$P(x_{t+1} = 1 | x_t = 0) = 0.0$ [ie, *no* transitions]

$P(e_t = 1 | x_t = 1) \gg P(e_t = 0 | x_t = 1)$

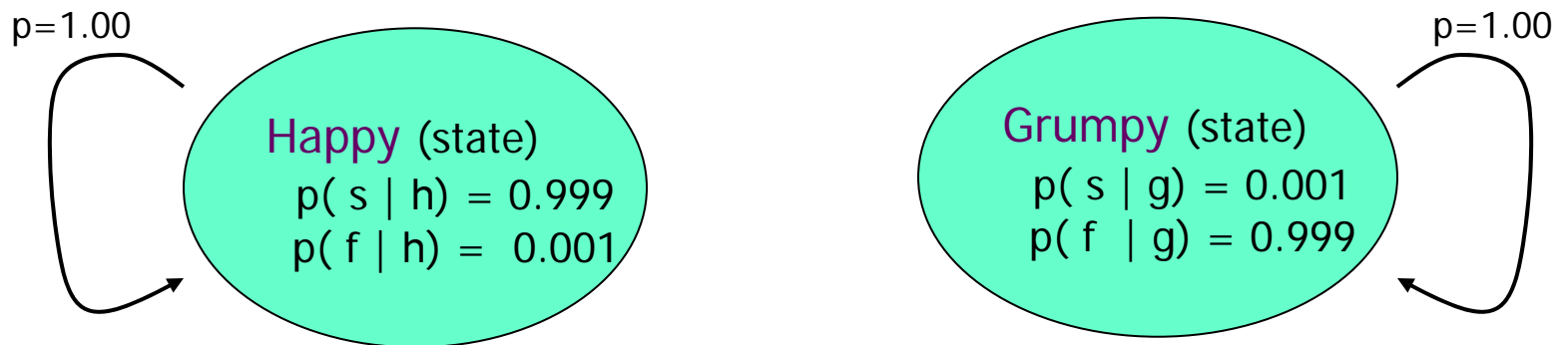
$P(e_t = 0 | x_t = 0) \gg P(e_t = 1 | x_t = 0)$

Given $\mathbf{e} = \langle 1, 0, 1 \rangle$, tempting to say $\mathbf{x} = \langle 1, 0, 1 \rangle$

... but this has **0** prob of occurring!!

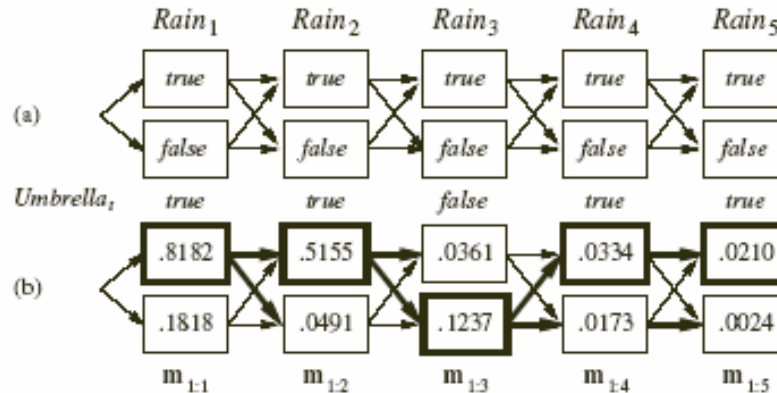
- Better: Path through states ... **dynamic program**

Need to consider ALL States



- Observe $\langle s, f, s \rangle$
- Predict $\langle H, G, H \rangle$
- But *0 chance of occurring!!*
- Only possible sequences:
 - $\langle H, H, H \rangle$
 - $\langle G, G, G \rangle$

MLE: Dynamic Program

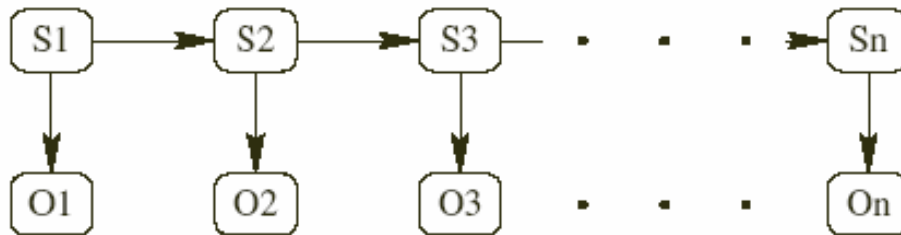


- Recursively, for each $X_k = x_k$:
 - compute prob of most likely path to each x_k
 - $m_{1:t}(X_t) = \max_{x_1, \dots, x_{t-1}} P(x_1, \dots, x_{t-1}, X_t | e_{1:t})$
- $m_{1:t+1}(X_{t+1}) = \max_{x_1, \dots, x_t} P(x_{1:t}, X_{t+1} | e_{1:t+1})$
 $= P(e_{1:t+1} | X_{t+1}) \max_{x_t} [P(X_{t+1} | x_t) \max_{x_{1:t-1}} P(x_{1:t-1}, x_t | e_{1:t})]$
 $= P(e_{1:t+1} | X_{t+1}) \max_{x_t} [P(X_{t+1} | x_t) \mathbf{m}_{1:t}(x_t)]$

MLE – con't

- $\mathbf{m}_{1:t+1} = \max_{x_1, \dots, x_t} P(x_{1:t}, X_{t+1} | e_{1:t+1})$
 $= P(e_{1:t+1} | X_{t+1}) \max_{x_t} P(X_{t+1} | x_t) \mathbf{m}_{1:t}$
- Just like *Filtering* except
 - Replace $f_{1:t} = P(X_t | e_{1:t})$
with $\mathbf{m}_{1:t} = \max_{x_{1:t-1}} P(x_{1:t-1}, X_t | e_{1:t})$
 - Replace \sum_{x_t} with \max_{x_t}
- To recover actual optimal-states x_k^*
... keep back-pointers!
- **Viterbi Algorithm**
- Linear time, linear space

Most Likely Sequence | DNA



- Observe only output values

- $\langle g c c t a \rangle$

- $E_1 = g, E_2 = c, E_3 = c, E_4 = t, E_5 = a$

- Want to determine:

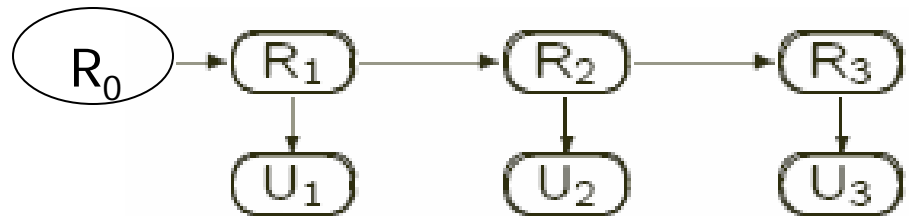
Most likely sequence of STATES

- $X_{1:5} = \langle e e i i i \rangle$

$X_1 = e, X_2 = e, X_3 = i, X_4 = i, X_5 = i$

(e for exon, i for intron)

Comments on HMMs



- Results hold for
 - ANY Markov model
 - with arbitrary hidden state
- HMM is special:
 - *single* discrete state variable
 - *single* discrete observation variable

per time

⇒ can use matrices

$$\mathbf{T} = P(\mathbf{X}_t | \mathbf{X}_{t-1}) = \begin{pmatrix} 0.7 & 0.3 \\ 0.3 & 0.7 \end{pmatrix}$$
$$T_{ij} = P(\mathbf{X}_t = j | \mathbf{X}_{t-1} = i)$$

$$\mathbf{O}^{(1)} = \begin{pmatrix} 0.9 & 0.0 \\ 0.0 & 0.2 \end{pmatrix}$$

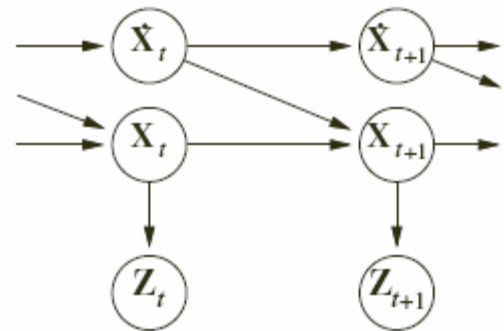
$$O_{ii}^{(t)} = P(e_t | \mathbf{X}_t = i)$$

$$f_{1:t+1} = \alpha \mathbf{O}^{t+1} \mathbf{T}^\top f_{1:t}$$

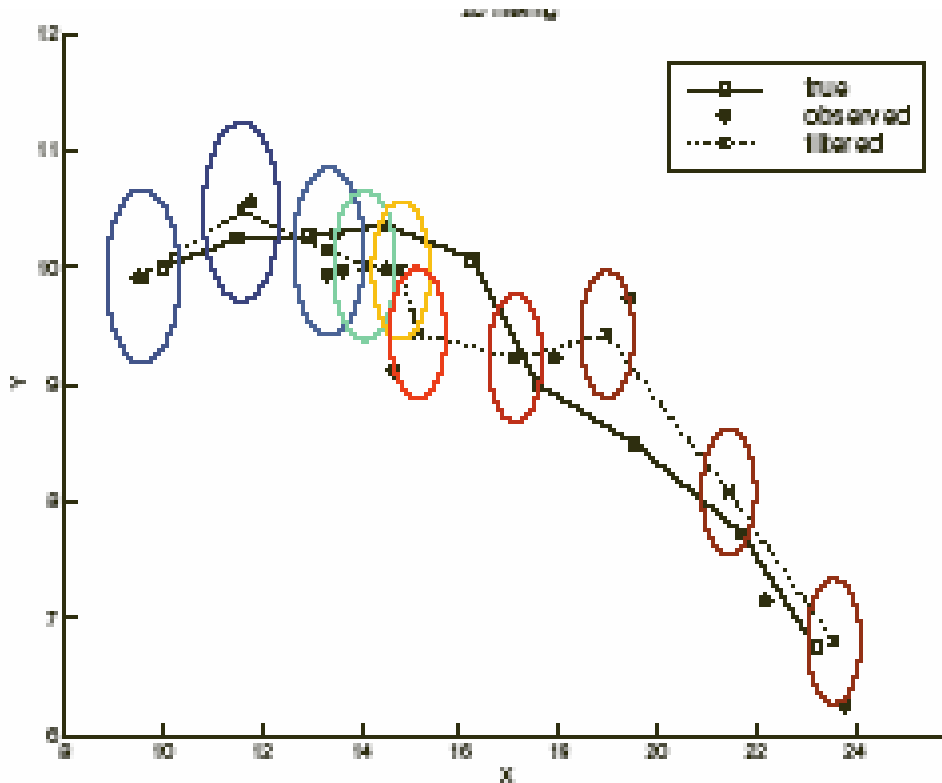
...

Kalman Filters

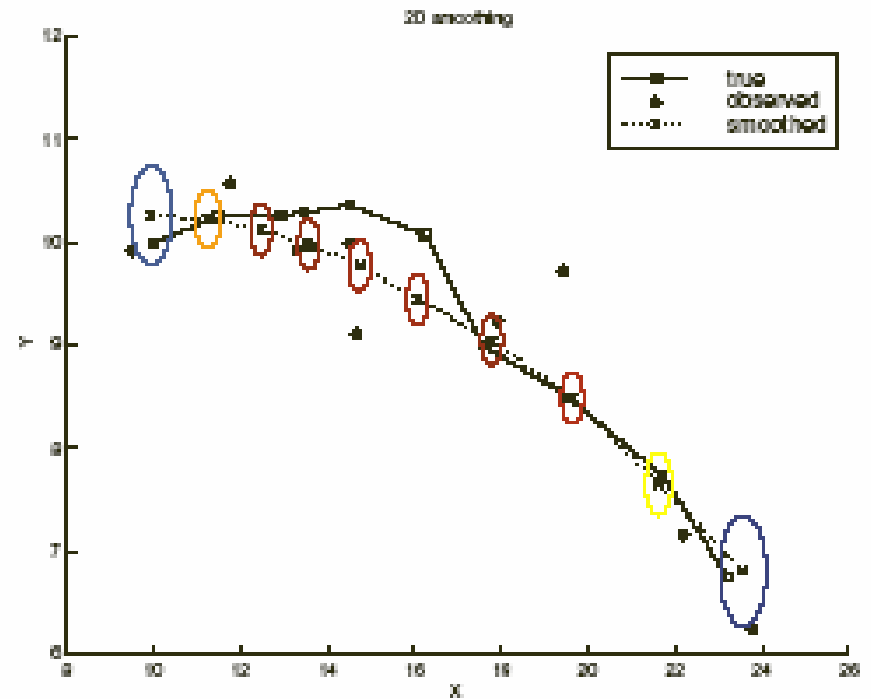
- Tracking a bird in flight, based on (noisy) sensors
Given observations
 ("estimates" of its position/velocity)
predict its future position, . . .
- X_t = TruePosition @time t
 \dot{X}_t = TrueVelocity @time t
 Z_t = MeasuredPosition @time t
- Observation model: $P(Z_t | X_t)$ $Z_t \sim N(X_t, \sigma_t^2)$
Transition model: $P(X_{t+1} | X_t, \dot{X}_t)$
 $X_{t+1} \sim N(X_t + \dot{X}_t, \sigma_t^2)$
- Everything stays Gaussian!
 ... for Filtering, Smoothing, ...



Tracking Object in X-Y Plane



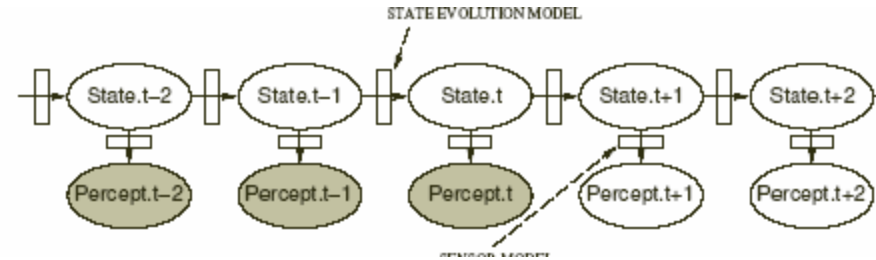
Tracking



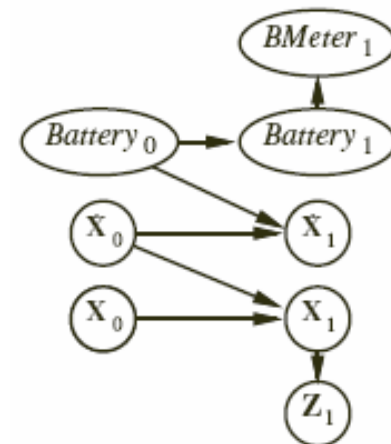
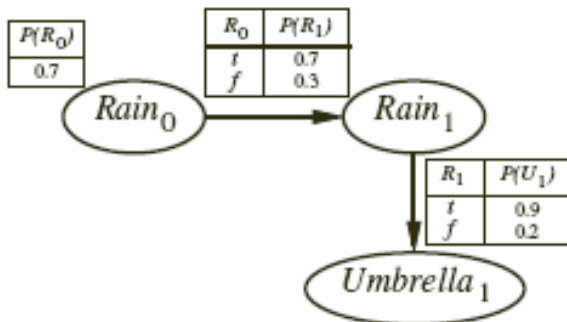
Smoothing

Dynamic Belief Network

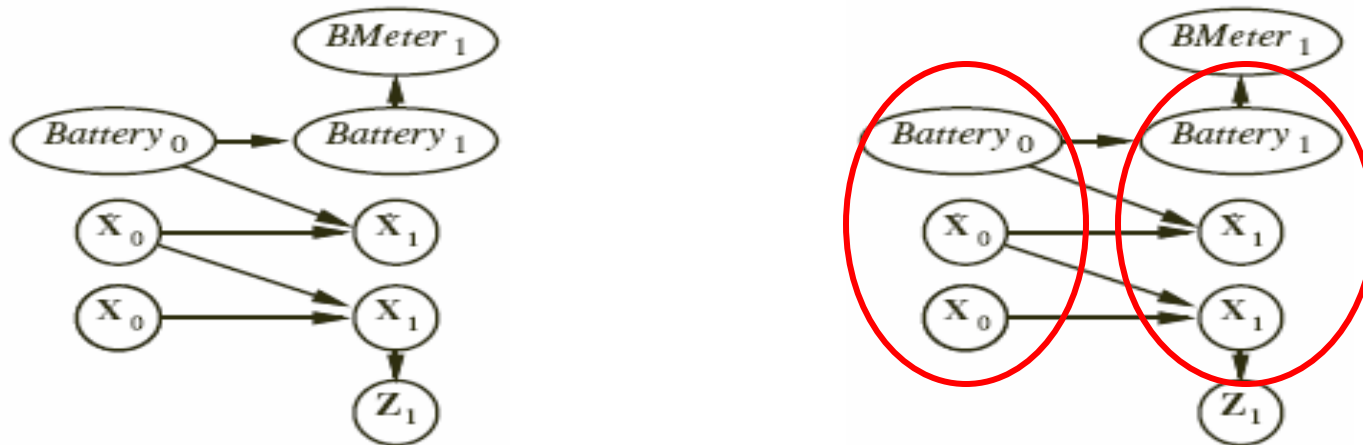
- At each time slice:
 - description of state
 - description of observation



- If 1 var for state, 1 var for obs
 ⇒ HMM
- But can have > 1 variable for state/observation!



Advantage of Dynamic BN

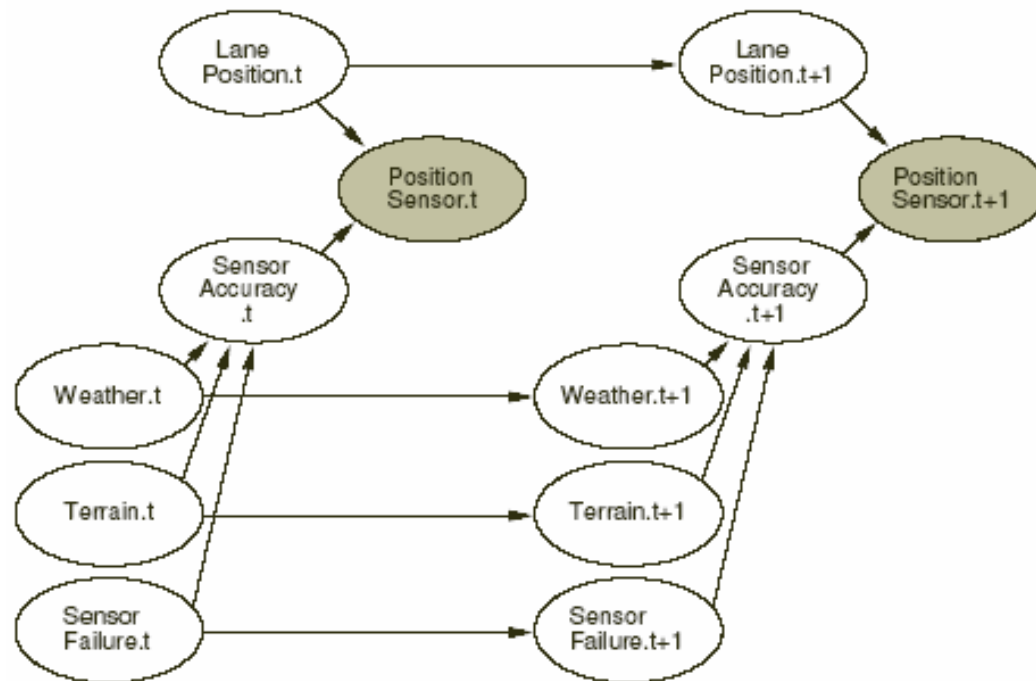


- Why not view DBN as HMM ?
 - ... just "bundle"
 - the observable variables $\{BMeter, Z\}$ into 1 meganode
 - the latent variables $\{X, X', Battery\}$ into 1 meganode
- Answer: Spse $|X|=10$; $|X'|=10$; $|Battery|=10$, $|BMeter|=10$, $|Z|=10$
 Now:
 - CPTables: $Battery \rightarrow Bmeter$: 10×10 ; $X \rightarrow Z$: 10×10
 $X', Battery_t \rightarrow Battery_{t+1}$: $10 \times 10 \times 10$; $X_t, X'_t \rightarrow X'_{t+1}$: $10 \times 10 \times 10$
 - Total: 2,200 values

As simple HMM:

- CPTable for Transition Probability: $10 \times 10 \times 10 \times 10 \times 10 \times 10 = 1M$!
- CPTable for Emission Probability: $10 \times 10 \times 10 \times 10 \times 10 = 100K$

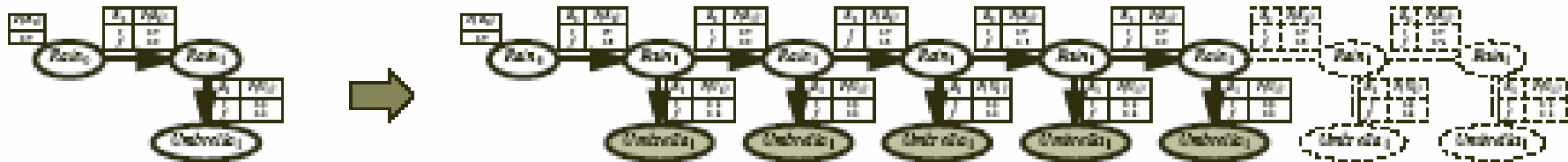
Representing State as GRAPH of Random Variables



... reduces complexity of representing
 $P(X' | X, A)$ and $P(E | X)$

Inference in DBNs

- As DBN is Belief Net,
 - can use std BeliefNet Inference alg
 - ... after unrolling



- Filtering

$$\begin{aligned}
 \mathbf{f}_{1:t+1}(\mathbf{x}_{t+1}) &= P(x_{t+1} \mid e_{1:t+1}) \\
 &= P(e_{t+1} \mid x_{t+1}) \sum_{x_t} P(x_{t+1} \mid x_t) \mathbf{f}_{1:t}(\mathbf{x}_t)
 \end{aligned}$$

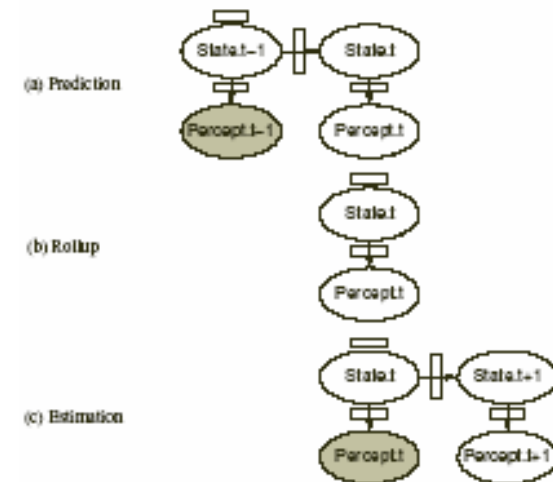
Sums out state variable x_{t-1}
 corresponds to Variable Elimination
 (with this temporal ordering of vars)

Actual DBN Algorithm (Filtering)

- DBN alg: just keep 2 slices in memory

$$\langle X_{t-1}, e_{t-1} \rangle + \langle X_t, e_t \rangle$$

$$f_{1:t+1} = \alpha \text{Forward}(f_{1:t+1}, e_{t+1})$$



- Constant per-update time, per-update space
- BUT...
 - as Evidence is CHILDREN, parents become COUPLED!
 \Rightarrow constant = $O(d^n)$
as factor involves all state variables!

Approximate Algorithms

- Could try. . .
 - likelihood weighting, MCMC, . . .
 - ... but still problems
- Use set of TUPLES themselves as approx'n!
 - Focus on high-probability instances
 - ... tuples \approx posterior distribution . . .
- Particle Filtering

Draw N tuples, $\{d_1^{(0)}, \dots, d_N^{(0)}\}$, from $P(\mathbf{X}_0)$

For $j = 0..Bored$

For $i = 1..N$

Draw x_i from $P(\mathbf{X}_{t+1} | d_i^{(t)})$

Compute weight $w_i = P(e_{t+1} | d_i^{(t+1)})$

Let $\{d_i^{(t+1)}\}_i$ be N tuples drawn from $\{[x_i, t+1, w_i]\}_i$

Particle Filtering

Draw N tuples, $\{d_1^{(0)}, \dots, d_N^{(0)}\}$, from $P(\mathbf{X}_0)$

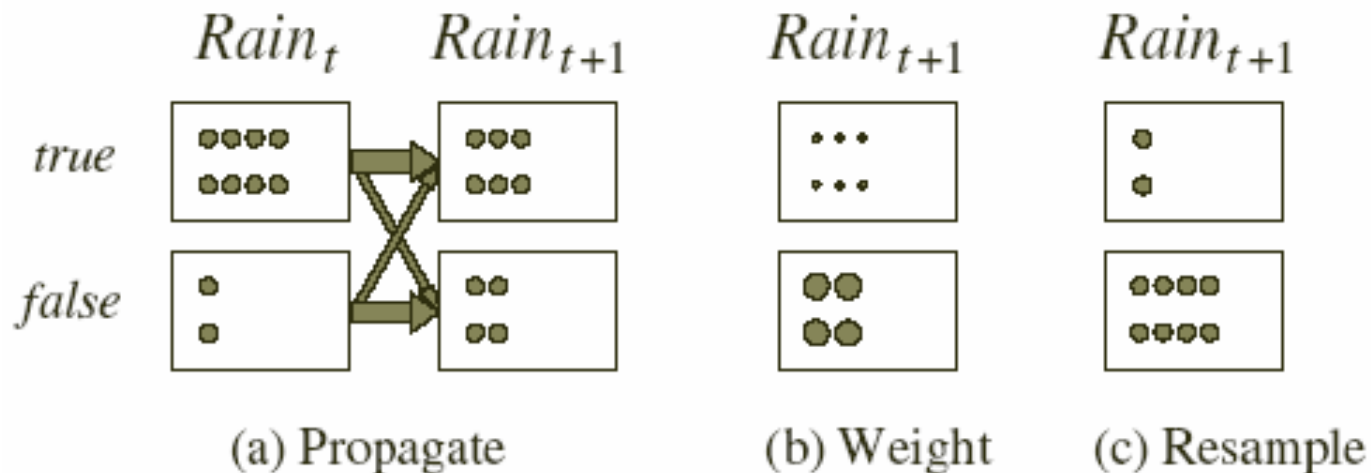
For $j = 0..Bored$

For $i = 1..N$

Draw x_i from $P(\mathbf{X}_{t+1} | d_i^{(t)})$

Compute weight $w_i = P(e_{t+1} | d_i^{(t+1)})$

Let $\{d_i^{(t+1)}\}_i$ be N tuples drawn from $\{[x_i^{t+1}, w_i]\}_i$





Hierarchical HMMs

- Can construct hierarchy of HMM's:
 - Each Sentence-HMM generates string of word-HMMs (I.e., each "hidden state" is a possible word)
 - Each word-HMM generates strings of phoneme-HMMs (I.e., each "hidden state" is a possible phoneme)
 - Each phoneme-HMM generates strings of speech frames
- "Compile" hierarchy into *frame-level HMM* that finds
whole sentence most likely to have been spoken
- MLE – computed by Viterbi algorithm

Beyond First-Order

- Recall First-Order Markov Chain

- Random Walk along x axis,
- changing x-position 1 at each time

- What if position x_t depends on x_{t-1} , x_{t-2} ?

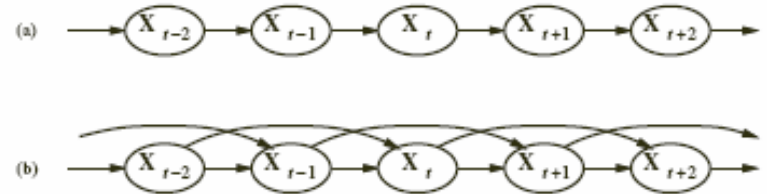
- (Ie, need velocity, as well as position)
- 2nd-order Markov Chain

[Can make any process into 1st-order Markov,
by expanding state

Eg, to deal with power being consumed,
could have BatteryLevel in state

. . . in the limit: "state" \equiv "all history"]

- Interpolated Markov Model (GLIMMER)



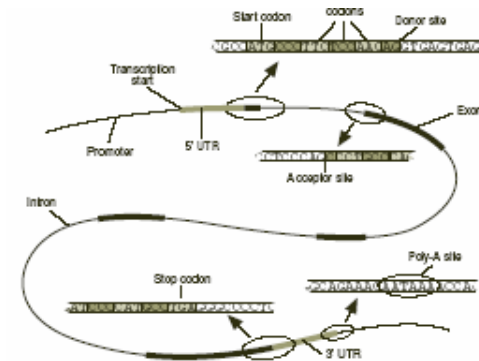
Computational Biology: Find Region of Interest in DNA

- Segment DNA into

- Exon vs Intron vs Intergenic Region
- StartCodon, DonorSite, AcceptorSite, StopCodon
- Techniques: NN, DecisionTrees, HMMs

- Identify "motif"

- "Significant Nucleotide Sequence"
- Intron/Exon boundary
- Sites: Promoter, Enhancer,
Transcription factor binding, Splice cite
- CRP Binding site (or LexA binding site, or ...)

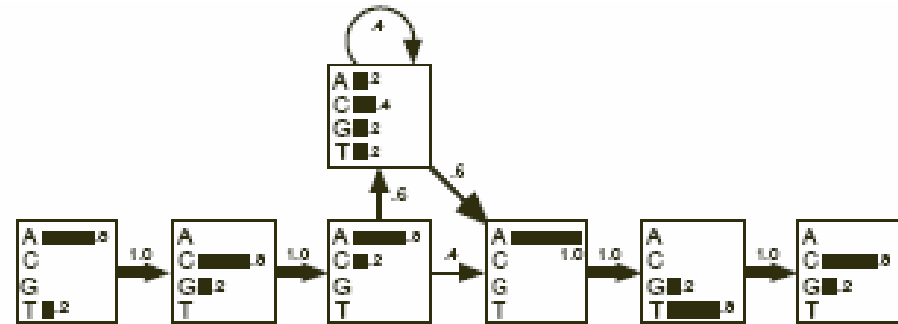




HMM's in Biological Sequence Data

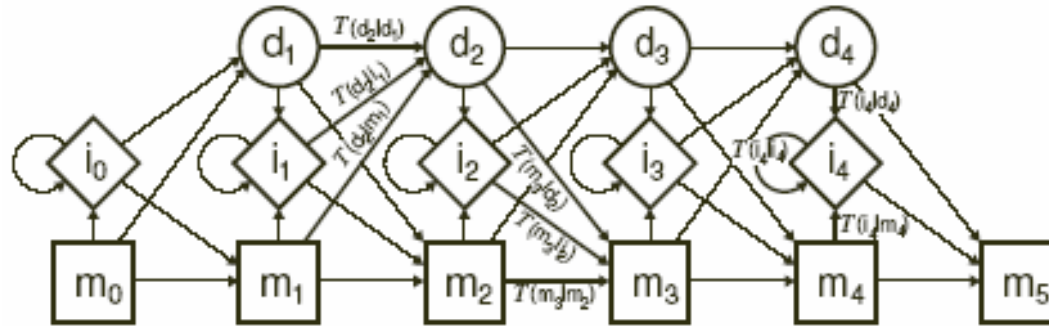
- Given collection of similar genes
(eg, same function, but different animals)
find new genes in other organisms that are similar.
[Ex: Globins (hemoglobin, myoglobin)]
Use "4. Likelihood" alg
- Given collection of similar genes,
align them to one another
(identify where mutations have occurred:
insertions, deletions, replacements)
Useful for studying evolution and discovering
functionally important parts
Use "5. MLE" alg

Simple Hidden Markov Model

$$\left\{ \begin{array}{l} \langle A \ C \ A \ - \ - \ - \ A \ T \ G \rangle \\ \langle T \ C \ A \ A \ C \ T \ A \ T \ C \rangle \\ \langle A \ C \ A \ C \ - \ - \ A \ G \ C \rangle \\ \langle A \ G \ A \ - \ - \ - \ A \ T \ C \rangle \\ \langle A \ C \ C \ G \ - \ - \ A \ T \ C \rangle \end{array} \right\}$$


- Each box is "state"
 - w/prob of "emitting" a letter
- Transition from state to state
 - Bottom Row: standard "emit a letter"
 - Upper Row: insert "extra" letter
(After state3, 3/5 of sequences goto "Insert"
Of 5 transitions from "Insert", 2 goto another insert)
- If no gaps, same as earlier model.

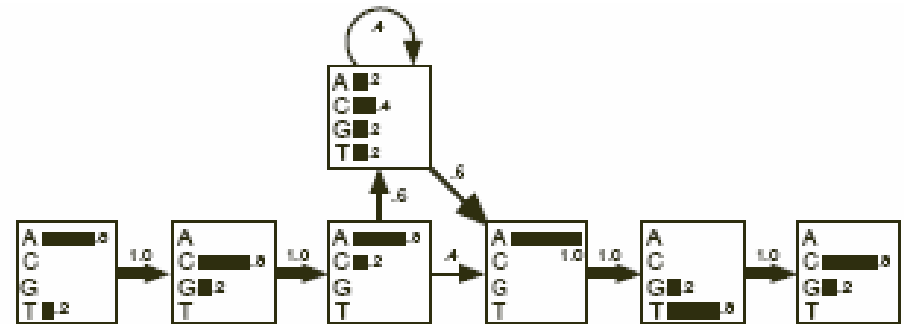
Profile HMM



- Special structure: “profile HMM”
- Main (level 0)
 - For “columns” of alignment
- Insert (level 1)
 - For highly-variable regions
- Delete (level 2)
 - “silent” or “null”

[5] Probability of Sequence wrt HMM

- $$\begin{aligned}
 P_{HMM}(\langle ACACATC \rangle) &= \\
 &P(\text{emit } A | M_1) \times P(M_1 \rightarrow M_2) \times \\
 &P(\text{emit } C | M_2) \times P(M_2 \rightarrow M_3) \times \\
 &P(\text{emit } A | M_3) \times P(M_3 \rightarrow I_3) \times \\
 &P(\text{emit } C | I_3) \times P(I_3 \rightarrow M_4) \times \\
 &P(\text{emit } A | M_4) \times P(M_4 \rightarrow M_5) \times \\
 &P(\text{emit } T | M_5) \times P(M_5 \rightarrow M_6) \times \\
 &P(\text{emit } C | M_6) \\
 &= 0.8 \times 1 \times 0.8 \times 1 \times 0.8 \times 0.6 \times 0.4 \\
 &\quad \times 0.6 \times 1 \times 1 \times 0.8 \times 1 \times 0.8 \\
 &\approx 4.7 \times 10^{-2}
 \end{aligned}$$



- Here, unambiguous. . .
 Only consistent path through HMM is
 $\langle M_1, M_2, M_3, I_3, M_4, M_5, M_6 \rangle$
 In general, several possible paths. . .



Recent applications of HMMs

- Proteins
 - detection of bronectin type III domains in yeast
 - a database of protein domain families
 - protein topology recognition from secondary structure
 - modeling of a protein splicing domain
- Gene finding
 - detection of short protein coding regions and analysis of translation initiation sites in Cyanobacterium
 - characterization of prokaryotic and eukaryotic promoters
 - recognition of branch points
- Also
 - prediction of protein secondary structure
 - modeling an oscillatory pattern in nucleosomes
 - modeling site dependence of evolutionary rates
 - for including evolutionary information in protein secondary structure prediction
- Free packages:
 - hmmer – <http://genome.wustl.edu/eddy/hmm.html>
 - SAM – <http://www.cse.ucsc.edu/research/compbio/sam.html>



Other Applications

Similar approaches work for analyzing

- Proteins (Amino-Acid sequences)
 - Similar composition, similar function, and . . .
- Protein Folding"
 - Protein sequence of a.a.'s
 - "Tertiary structure" \equiv Complete 3D structure
 - "Secondary structure" \equiv Simpler decomposition
 α -helices, β -sheets, (random) coil
- TEMPORAL sequences
 - weather prediction
 - stock-market forecasting
 - ...



Future Research

- Scaling up to handle larger
 {sequences, motifs, DBs }

Learn...

- more accurate descriptions
- in less time (fewer samples, less CPU-time)
- rep'ns that allow more efficient computation
- Exploiting other information
 - facts about a.a.'s (hierarchy?)
 - structural information
 - ...

Summary

- To model temporal events
 - Use rv X_t to model X at time t
 - *Stationary* distribution: $P(X_t)$ same at any time
- Markov Property:
$$P(\mathbf{X}_{t+1} \mid \mathbf{X}_t, \mathbf{X}_{t-1}, \dots) = P(\mathbf{X}_{t+1} \mid \mathbf{X}_t)$$
- Hidden Markov Model:
 - Emission $P(E_t \mid X_t)$; Transition $P(X_{t+1} \mid X_t)$
 - Efficient (linear time!) to predict ...
 - Current state (filtering)
 - Previous state (smoothing)
 - Future state (prediction)
 - Most likely explanation (Viterbi)
- Dynamic Belief Nets – extension of HMM
 - ... mixing ...
- Uses: Speech recognition; Tracking; BioInformatics, ...