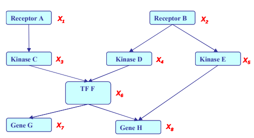
School of Computer Science
Carnegie Mellon

State Space Models

Probabilistic Graphical Models (10-708)

Lecture 13, part II
Nov 5th, 2007

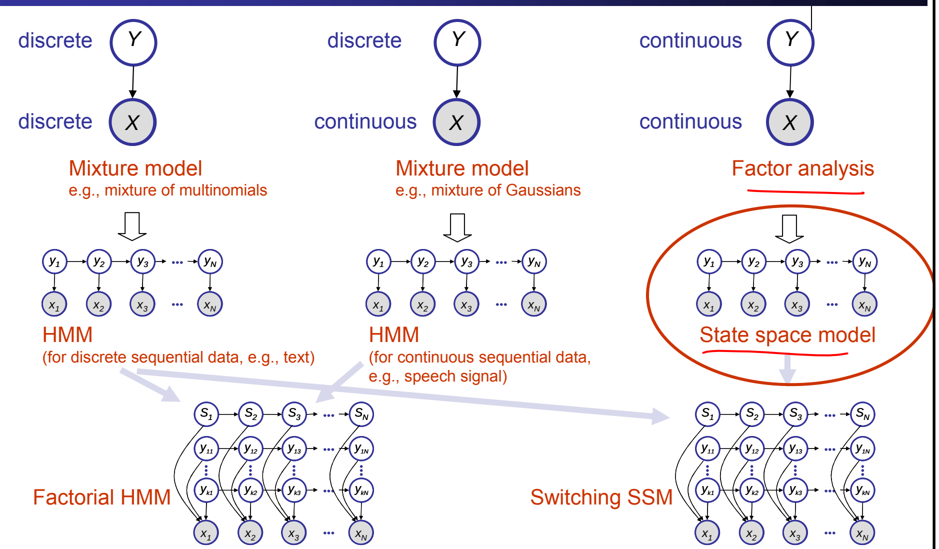
2



Eric Xing

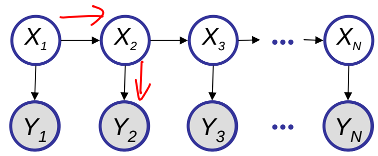
Reading: J-Chap. 15, K&F chapter 19.1 -19.3

A road map to more complex dynamic models



State space models (SSM):

- A sequential FA or a continuous state HMM



$$\mathbf{x}_t = A\mathbf{x}_{t-1} + Gw_t$$
$$\mathbf{y}_t = C\mathbf{x}_t + v_t$$
$$w_t \sim \mathcal{N}(0;Q), \quad v_t \sim \mathcal{N}(0;R)$$
$$\mathbf{x}_0 \sim \mathcal{N}(0;\Sigma_0),$$

This is a linear dynamic system.

- In general,

$$\mathbf{x}_t = f(\mathbf{x}_{t-1}) + Gw_t$$
$$\mathbf{y}_t = g(\cancel{\mathbf{x}_{t-1}}) + v_t \qquad g(\bar{x}_t)+v_t$$

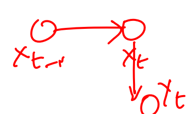
where f is an (arbitrary) dynamic model, and g is an (arbitrary) observation model

LDS for 2D tracking

- Dynamics: new position = *old position* + $\Delta\times$*velocity* + *noise* (constant velocity model, Gaussian noise)

$$\begin{array}{l} \text{positions} \\ \\ \text{velocity} \end{array} \begin{pmatrix} x_t^1 \\ x_t^2 \\ \dot{x}_t^1 \\ \dot{x}_t^2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & \Delta & 0 \\ 0 & 1 & 0 & \Delta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{t-1}^1 \\ x_{t-1}^2 \\ \dot{x}_{t-1}^1 \\ \dot{x}_{t-1}^2 \end{pmatrix} + \text{noise}$$

$$x_t^1 = x_{t-1}^1 + \Delta \dot{x}_{t-1}^1$$



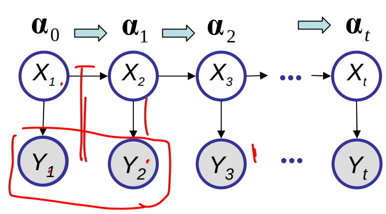
- Observation: project out first two components (we observe Cartesian position of object - linear!)

$$Y \quad \begin{pmatrix} y_t^1 \\ y_t^2 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_t^1 \\ x_t^2 \\ \dot{x}_t^1 \\ \dot{x}_t^2 \end{pmatrix} + \text{noise}$$

The inference problem 1

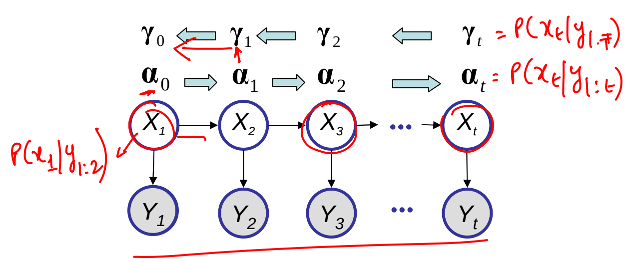
- Filtering → given $\mathbf{y}_1, \ldots, \mathbf{y}_t$, estimate x_t: $P(x_t \mid \mathbf{y}_{1:t}) = \alpha(x_t)$
  - The Kalman filter is a way to perform exact online inference (sequential Bayesian updating) in an LDS.
  - It is the Gaussian analog of the forward algorithm for HMMs:

$$p(\mathrm{X}_t = i \mid \mathbf{y}_{1:t}) = \alpha_t^i \propto p(\mathrm{y}_t \mid \mathrm{X}_t = i) \sum_j p(\mathrm{X}_t = i \mid \mathrm{X}_{t-1} = j)\alpha_{t-1}^j$$



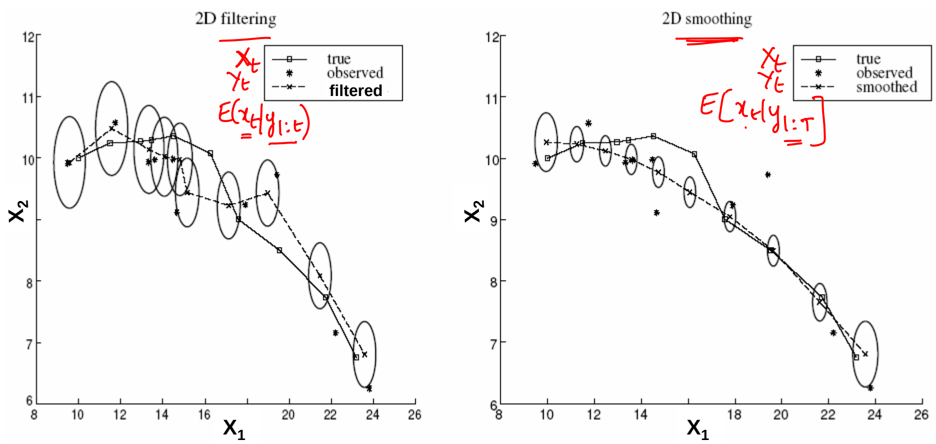
The inference problem 2

- Smoothing → given $\mathbf{y}_1, \ldots, \mathbf{y}_T$, estimate x_t (t<T) $P(x_t \mid y_{1:T}) = \gamma_t(x)$
  - The Rauch-Tung-Strievel smoother is a way to perform exact off-line inference in an LDS. It is the Gaussian analog of the forwards-backwards (alpha-gamma) algorithm: $P(x_t \mid y_{t+1:T})$

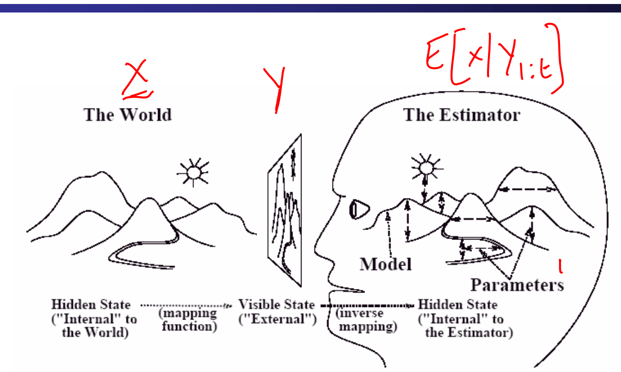


$$p(\mathrm{X}_t = i \mid y_{1:T}) = \gamma_t^i \propto \sum_j \alpha_t^i P(X_{t+1}^j \mid X_i^j)\gamma_{t+1}^j$$

2D tracking



Kalman filtering in the brain?

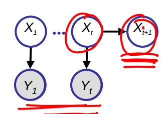


Kalman filtering derivation

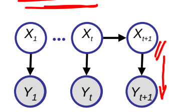
- Since all CPDs are linear Gaussian, the system defines a large multivariate Gaussian.
  - Hence all marginals are Gaussian.
  - Hence we can represent the belief state $p(\mathbf{X}_t|\mathbf{y}_{1:t})$ as a Gaussian
    - mean $\hat{x}_{t|t} \equiv E(X_t \mid \mathbf{y}_1, \ldots, \mathbf{y}_t)$ $\quad \hat{x}_{t|t} := \alpha\, E[x|y_{1:t}]$
    - covariance $P_{t|t} \equiv E(X_t X_t^T \mid \mathbf{y}_1, \ldots, \mathbf{y}_t)$ $\quad P(x_t|y_{1:t}) \quad P(x_t|y_{1:T})$
  - Hence, instead of marginalization for message passing, we will directly estimate the means and covariances of the required marginals
  - It is common to work with the inverse covariance (precision) matrix $P_{t|t}^{-1}$; this is called information form.

Kalman filtering derivation

- Kalman filtering is a recursive procedure to update the belief state: $P(x_t|y_{1:t}) \longrightarrow P(x_{t+1}|y_{1:t+1})$
  - Predict step: compute $p(\mathbf{X}_{t+1}|\mathbf{y}_{1:t})$ from prior belief $p(\mathbf{X}_t|\mathbf{y}_{1:t})$ and dynamical model $p(\mathbf{X}_{t+1}|\mathbf{X}_t)$ --- time update



  - Update step: compute new belief $p(\mathbf{X}_{t+1}|\mathbf{y}_{1:t+1})$ from prediction $p(\mathbf{X}_{t+1}|\mathbf{y}_{1:t})$, observation $\mathbf{y}_{t+1}$ and observation model $p(\mathbf{y}_{t+1}|\mathbf{X}_{t+1})$ --- measurement update

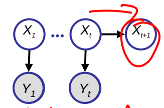


$$P(x_{t+1}|y_{1:t}) = \sum_{x_t} P(x_{t+1}, x_t|y_{1:t}) = \sum_{x_t} P(x_{t+1}|x_t) P(x_t|y_{1:t})$$
$$= \sum_{x_t} P(x_{t+1}|x_t) \cdot P(x_t|y_{1:t})$$

$$P(x_{t+1}|y_{1:t+1}) = P(x_{t+1}|y_{1:t}, y_{t+1})$$
$$\propto P(y_{t+1}|y_{1:t}, x_{t+1})\, P(x_{t+1}|y_{1:t})$$
$$\propto P(y_{t+1}|x_{t+1}) \cdot P(x_{t+1}|y_{1:t})$$

Predict step

- Dynamical Model: $\mathbf{x}_{t+1} = A\mathbf{x}_t + Gw_t, \qquad w_t \sim \mathcal{N}(0; Q)$
  - One step ahead prediction of state:



$E[x_{t+1} \mid y_{1:t}] = \hat{x}_{t+1|t} = E[Ax_t + Gw_t \mid y_{1:t}] = A\hat{x}_{t|t} + 0$

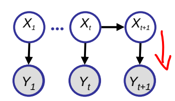
$P_{t+1|t}$: $E[(x_{t+1} - \hat{x}_{t+1|t})(x_{t+1} - \hat{x}_{t+1|t})^T \mid y_{1:t}] = E[(A(x_t - \hat{x}_{t|t}) + Gw_t)(A(x_t - \hat{x}_{t|t}) + Gw_t)^T]$

$p(x_{t+1} \mid y_{1:t})$

$= A\,E[(x_t - \hat{x}_{t|t})(x_t - \hat{x}_{t|t})^T \mid y_{1:t}]A^T + G\,E[w_t w_t^T]G^T \quad y_{1:t}$

$= AP_{t|t}A^T + GQG^T$

- Observation model: $\mathbf{y}_{t+1} = C\mathbf{x}_{t+1} + v_{t+1}, \qquad v_{t+1} \sim \mathcal{N}(0; R)$
  - One step ahead prediction of observation:



$\hat{y}_{t+1|t} = C\hat{x}_{t+1|t}$

$E[(y_{t+1} - \hat{y}_{t+1|t})(y_{t+1} - \hat{y}_{t+1|t})^T \mid y_{1:t}] = E[(C(x_{t+1} - \hat{x}_{t+1|t}) + v_{t+1})(\quad)^T \mid y_{1:t}]$

$= CE[(x_{t+1} - \hat{x}_{t+1|t})(\quad)^T]C^T + E[v_{t+1}v_{t+1}^T]$

$E[(x_{t+1} - \hat{x}_{t+1|t})(y_{t+1} - \hat{y}_{t+1|t}) \mid y_{1:t}] = CP_{t+1|t}C^T + R$

$= P_{t+1|t}C^T$

$\begin{bmatrix} x_{t+1} \mid y_{1:t} \\ y_{t+1} \mid y_{1:t} \end{bmatrix}$

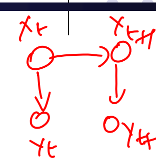
Update step

$p(x_t \mid y_{1:t}) \rightarrow p(x_{t+1} \mid y_{1:t})$

$\downarrow$

$p(x_{t+1} \mid y_{1:t+1})$

- Summarizing results from previous slide, we have $p(\mathbf{X}_{t+1}, \mathbf{Y}_{t+1} \mid \mathbf{y}_{1:t}) \sim \mathcal{N}(m_{t+1}, V_{t+1})$, where



$$m_{t+1} = \begin{pmatrix} \hat{x}_{t+1|t} \\ C\hat{x}_{t+1|t} \end{pmatrix} \begin{matrix} x_{t+1} \\ y_{t+1} \end{matrix}, \qquad V_{t+1} = \begin{pmatrix} P_{t+1|t} & P_{t+1|t}C^T \\ CP_{t+1|t} & CP_{t+1|t}C^T + R \end{pmatrix},$$

- Remember the formulas for conditional Gaussian distributions:

$p(x_{t+1} \mid y_{t+1}, y_{1:t})$

$p(x_{t+1} \mid y_{1:t+1})$

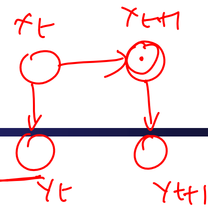
$$p(\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \mid \mu, \Sigma) = \mathcal{N}(\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \mid \begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix})$$

$$p(\mathbf{x}_2) = \mathcal{N}(\mathbf{x}_2 \mid \mathbf{m}_2^m, \mathbf{V}_2^m) \qquad p(\mathbf{x}_1 \mid \mathbf{x}_2) = \mathcal{N}(\mathbf{x}_1 \mid \mathbf{m}_{1|2}, \mathbf{V}_{1|2})$$

$$\mathbf{m}_2^m = \mu_2 \qquad \mathbf{m}_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{x}_2 - \mu_2)$$

$$\mathbf{V}_2^m = \Sigma_{22} \qquad \mathbf{V}_{1|2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$$

Kalman Filter



- Measurement updates:

$$\hat{\mathbf{x}}_{t+1|t+1} = \hat{\mathbf{x}}_{t+1|t} + K_{t+1}(\mathrm{y}_{t+1} - \mathrm{C}\hat{\mathrm{x}}_{t+1|t})$$

$$P_{t+1|t+1} = P_{t+1|t} - K_{t+1}CP_{t+1|t}$$

  - where K_{t+1} is the *Kalman gain matrix*

$$K_{t+1} = P_{t+1|t}C^T(CP_{t+1|t}C^T + R)^{-1}$$

Example of KF in 1D

- Consider noisy observations of a 1D particle doing a random walk:

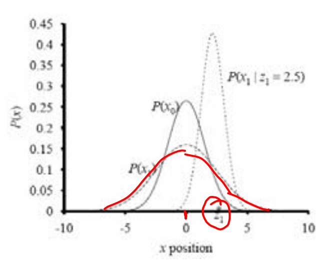
$$x_{t|t-1} = x_{t-1} + w, \quad w \sim \mathcal{N}(0, \sigma_x) \qquad z_t = x_t + v, \quad v \sim \mathcal{N}(0, \sigma_z)$$

- KF equations: $P_{t+1|t} = AP_{t|t}A^T + GQG^T = \sigma_t + \sigma_x \quad , \quad \hat{x}_{t+1|t} = A\hat{x}_{t/t} = \hat{x}_{t/t}$

$$K_{t+1} = P_{t+1|t}C^T(CP_{t+1|t}C^T + R)^{-1} = (\sigma_t + \sigma_x)(\sigma_t + \sigma_x + \sigma_z)$$

$$\hat{x}_{t+1|t+1} = \hat{x}_{t+1|t} + K_{t+1}(z_{t+1} - C\hat{x}_{t+1|t}) = \frac{(\sigma_t + \sigma_x)z_{t+1} + \sigma_z \hat{x}_{t|t}}{\sigma_t + \sigma_x + \sigma_z}$$

$$P_{t+1|t+1} = P_{t+1|t} - KCP_{t+1|t} = \frac{(\sigma_t + \sigma_x)\sigma_z}{\sigma_t + \sigma_x + \sigma_z}$$



KF intuition

- The KF update of the mean is

$$\hat{x}_{t+1|t+1} = \hat{x}_{t+1|t} + K_{t+1}(z_{t+1} - C\hat{x}_{t+1|t}) = \frac{(\sigma_t + \sigma_x)z_{t+1} + \sigma_z \hat{x}_{t|t}}{\sigma_t + \sigma_x + \sigma_z}$$

  - the term $(z_{t+1} - C\hat{x}_{t+1|t})$ is called the *innovation*
- New belief is convex combination of updates from prior and observation, weighted by Kalman Gain matrix:

$$K_{t+1} = P_{t+1|t}C^T(CP_{t+1|t}C^T + R)^{-1} = (\sigma_t + \sigma_x)(\sigma_t + \sigma_x + \sigma_z)$$

- If the observation is unreliable, σ_z (i.e., R) is large so K_{t+1} is small, so we pay more attention to the prediction.
- If the old prior is unreliable (large σ_t) or the process is very unpredictable (large σ_x), we pay more attention to the observation.

KF, RLS and LMS

- The KF update of the mean is

$y_t = C x_t + v_t$

$$\hat{x}_{t+1|t+1} = A\hat{x}_{t|t} + K_{t+1}(y_{t+1} - C\hat{x}_{t+1|t})$$

- Consider the special case where the hidden state is a constant, $x_t = \theta$, but the "observation matrix" C is a time-varying vector, $C = x_t^T$.
  - Hence the observation model at each time slide, $y_t = x_t^T\theta + v_t$, is a linear regression
- We can estimate recursively using the Kalman filter:

$$\hat{\theta}_{t+1} = \hat{\theta}_t + P_{t+1}R^{-1}(y_{t+1} - x_t^T\hat{\theta}_t)x_t$$

This is called the recursive least squares (RLS) algorithm.

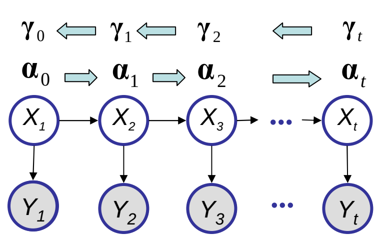
- We can approximate $P_{t+1}R^{-1} \approx \eta_{t+1}$ by a scalar constant. This is called the least mean squares (LMS) algorithm.
- We can adapt η_t online using stochastic approximation theory.

Complexity of one KF step

- Let $X_t \in \mathbb{R}^{N_x}$ and $Y_t \in \mathbb{R}^{N_y}$,
- Computing $P_{t+1|t} = AP_{t|t}A^T + GQG^T$ takes O(N_x^3) time, assuming dense P and dense A.
- Computing $K_{t+1} = P_{t+1|t}C^T(CP_{t+1|t}C^T + R)^{-1}$ takes O(N_y^3) time.
- So overall time is, in general, max {N_x^3,N_y^3}

The inference problem 2

- Smoothing → given $y_1, \ldots, y_T$, estimate x_t (t<T) $P(x_t | y_{1:T})$
  - The Rauch-Tung-Strievel smoother is a way to perform exact off-line inference in an LDS. It is the Gaussian analog of the forwards-backwards (alpha-gamma) algorithm:



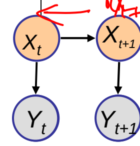
$$p(X_t = i | y_{1:T}) = \gamma_t^i \propto \sum_j \alpha_t^i P(X_{t+1}^j | X_t^j) \gamma_{t+1}^j$$

RTS smoother derivation

$P(x_t|y_{1:T}) = \gamma(x_{t+1}) P(x_{t+1}|x_t)$

- Smoothing → given $\mathbf{y}_1, \ldots, \mathbf{y}_T$, estimate $P(x_t|y_{1:T})$ (t<T)
  - Step 1: joint distribution of x_t and x_{t+1} conditioned on $y_{1:t}$
  - Use $\mathbf{x}_{t+1} = A\mathbf{x}_t + Gw_t;\ w_t \sim \mathcal{N}(0;Q);$

$\gamma(x_{t+1})$



$E[x_t|y_{1:t}] = \hat{x}_{t|t}$ $\quad E[x_{t+1}|y_{1:t}] = \hat{x}_{t+1|t}$

$E[(x_t-\hat x_{t|t})(x_{t+1}-\hat x_{t+1|t})^T|y_{1:t}]$

$= E[(x_t-\hat x_{t|t})(Ax_t+Gw_t - A\hat x_{t|t})^T|y_{1:t}]$

$= E[(x_t-\hat x_{t|t})(A(x_t-\hat x_{t|t})+Gw_t)^T|y_{1:t}]$

$= P_{t|t}A^T$

$E[(x_{t+1}-\hat x_{t+1|t})(x_{t+1}-\hat x_{t+1|t})^T|y_{1:t}] = AP_{t|t}A^T + GQG^T$

$\begin{bmatrix} x_t|y_{1:t} \\ x_{t+1}|y_{1:t}\end{bmatrix}$

RTS smoother derivation

- Following the results from previous slide, we need to derive $p(\mathbf{X}_{t+1},\mathbf{X}_t|\mathbf{y}_{1:t}) \sim \mathcal{N}(m, V)$, where

$$m = \begin{pmatrix}\hat x_{t|t}\\ \hat x_{t+1|t}\end{pmatrix}, \qquad V = \begin{pmatrix} P_{t|t} & P_{t|t}A^T \\ AP_{t|t} & P_{t+1|t}\end{pmatrix},$$

$P(x_t|y_{1:T})$

$\uparrow$

$P(x_t|x_{t+1}; y_{1:t})$

  - all the quantities here are available after a forward KF pass
- Remember the formulas for conditional Gaussian distributions:

$$p(\begin{bmatrix}\mathbf{x}_1\\\mathbf{x}_2\end{bmatrix}|\mu,\Sigma) = \mathcal{N}(\begin{bmatrix}\mathbf{x}_1\\\mathbf{x}_2\end{bmatrix}|\begin{bmatrix}\mu_1\\\mu_2\end{bmatrix},\begin{bmatrix}\Sigma_{11}&\Sigma_{12}\\\Sigma_{21}&\Sigma_{22}\end{bmatrix}),$$

$p(\mathbf{x}_2) = \mathcal{N}(\mathbf{x}_2|\mathbf{m}_2^m, \mathbf{V}_2^m)$

$\mathbf{m}_2^m=\mu_2$

$\mathbf{V}_2^m = \Sigma_{22}$

$p(\mathbf{x}_1|\mathbf{x}_2) = \mathcal{N}(\mathbf{x}_1|\mathbf{m}_{1|2},\mathbf{V}_{1|2})$

$\mathbf{m}_{1|2} = \mu_1+\Sigma_{12}\Sigma_{22}^{-1}(\mathbf{x}_2-\mu_2)$

$\mathbf{V}_{1|2} = \Sigma_{11}-\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}$

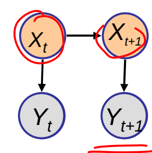
$P(x_t|x_{t+1}; y_{1:t})$

$$E[x_t|x_{t+1},\mathbf{y}_{0:t}] = \hat x_{t|t} + L_t(x_{t+1}-\hat x_{t+1|t})$$
$$\mathrm{Var}[x_t|x_{t+1},\mathbf{y}_{0:t}] = P_{t|t} - L_tP_{t+1|t}L_t^T$$

$L_t = P_{t|t}A^TP_{t+1|t}^{-1}$

RTS smoother derivation

$$E[x_t|x_{t+1}, \mathbf{y}_{0:t}] = \hat{x}_{t|t} + L_t(x_{t+1} - \hat{x}_{t+1|t})$$
$$\mathrm{Var}[x_t|x_{t+1}, \mathbf{y}_{0:t}] = P_{t|t} - L_t P_{t+1|t} L_t^T$$



- Step 2: compute $\hat{x}_{t|T}$ = E[x_t|$\mathbf{y}_{0:T}$] using results above
  - Use $E[x_t|x_{t+1}, \mathbf{y}_{0:T}] = E[x_t|x_{t+1}, \mathbf{y}_{0:t}]$
  - Use E[X|Z] = E[E[X|Y,Z]|Z]

$$\underbrace{E[x_t|y_{0:T}]}_{\hat{x}_{t|T}} = E[E[x_t|x_{t+1}, y_{0:T}] | y_{0:T}]$$
$$= E[E[x_t|x_{t+1}, y_{0:t}] | y_{0:T}]$$
$$= E[\hat{x}_{t|t} + L_t(x_{t+1} - \hat{x}_{t+1|t}) | y_{0:T}]$$
$$= \hat{x}_{t|t} + L_t(\hat{x}_{t+1|T} - \hat{x}_{t+1|t})$$

$$P(x_t|x_{t+1}, y_{1:t}) \longrightarrow P(x_t|y_{1:T})$$

RTS derivation

- Repeat the same process for Variance
  - Refer to Jordan chapter 15
- The RTS smoother results:

$$\hat{x}_{t|T} = \hat{\mathbf{x}}_{t|t} + L_t(\hat{\mathbf{x}}_{t+1|T} - \hat{\mathbf{x}}_{t+1|t})$$

$$P_{t|T} = P_{t|t} + L_t(P_{t+1|T} - P_{t+1|t})L_t^T$$

Learning SSMs

$x_{t+1} = \boxed{A} x_t + \boxed{G} w_t$
$y_t = \boxed{C} x_t + \boxed{v} v_t$

- Complete log likelihood

$$\ell_c(\theta, D) = \sum_n \log p(x_n, y_n) = \sum_n \log p(x_1) + \sum_n \sum_t \log p(x_{n,t} \mid x_{n,t-1}) + \sum_n \sum_t \log p(y_{n,t} \mid x_{n,t})$$
$$= f_1(X_1; \Sigma_0) + f_2\left(\left\{X_t X_{t-1}^T, X_t X_t^T, X_t : \forall t\right\}, A, Q, G\right) + f_3\left(\left\{X_t X_t^T, X_t : \forall t\right\}, C, R\right)$$

- EM
  - E-step: compute $\langle X_t X_{t-1}^T \rangle, \langle X_t X_t^T \rangle, \langle X_t \rangle \Big| y_1, \ldots y_T$

    these quantities can be inferred via KF and RTS filters, etc.,
    e,g., $\langle X_t X_t^T \rangle \equiv \mathrm{var}(X_t X_t^T) + \mathrm{E}(X_t)^2 = P_{t|T} + \hat{x}_{t|T}^2$

  - M-step: MLE using

$$\langle \ell_c(\theta, D) \rangle = f_1(\langle X_1 \rangle; \Sigma_0) + f_2\left(\left\{\langle X_t X_{t-1}^T \rangle, \langle X_t X_t^T \rangle, \langle X_t \rangle : \forall t\right\}, A, Q, G\right) + f_3\left(\left\{\langle X_t X_t^T \rangle, \langle X_t \rangle : \forall t\right\}, C, R\right)$$

    c.f., M-step in factor analysis

Eric Xing 23

Nonlinear systems

- In robotics and other problems, the motion model and the observation model are often nonlinear:

$$x_t = f(x_{t-1}) + w_t \;, \qquad y_t = g(x_t) + v_t$$

- An optimal closed form solution to the filtering problem is no longer possible.
- The nonlinear functions f and g are sometimes represented by neural networks (multi-layer perceptrons or radial basis function networks).
- The parameters of f and g may be learned offline using EM, where we do gradient descent (back propagation) in the M step, c.f. learning a MRF/CRF with hidden nodes.
- Or we may learn the parameters online by adding them to the state space: $x_t' = (x_t, \theta)$. This makes the problem even more nonlinear.

Eric Xing 24

Extended Kalman Filter (EKF)

- The basic idea of the EKF is to linearize f and g using a second order Taylor expansion, and then apply the standard KF.
  - i.e., we approximate a stationary nonlinear system with a non-stationary linear system.

$$x_t = f(\hat{x}_{t-1|t-1}) + A_{\hat{x}_{t-1|t-1}}(x_{t-1} - \hat{x}_{t-1|t-1}) + w_t$$

$$y_t = g(\hat{x}_{t|t-1}) + C_{\hat{x}_{t|t-1}}(x_t - \hat{x}_{t|t-1}) + v_t$$

where $\hat{x}_{t|t-1} = f(\hat{x}_{t-1|t-1})$ and $A_{\hat{x}} \overset{\text{def}}{=} \left.\frac{\partial f}{\partial x}\right|_{\hat{x}}$ and $C_{\hat{x}} \overset{\text{def}}{=} \left.\frac{\partial g}{\partial x}\right|_{\hat{x}}$

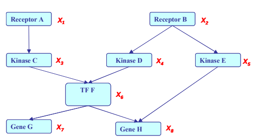
- The noise covariance (Q and R) is not changed, i.e., the additional error due to linearization is not modeled.

School of Computer Science
Carnegie Mellon

Complex Graphical Models

Probabilistic Graphical Models (10-708)

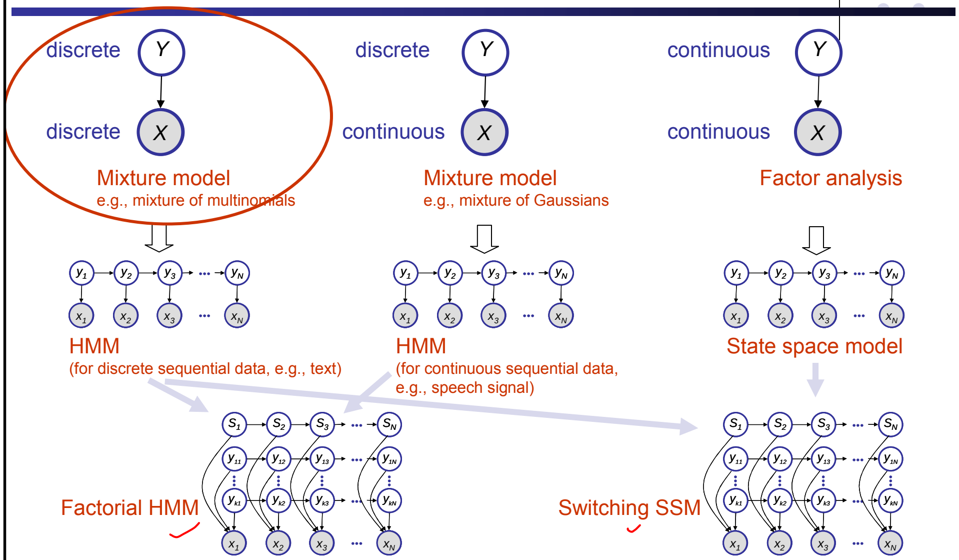
Lecture 14, Nov 5th, 2007



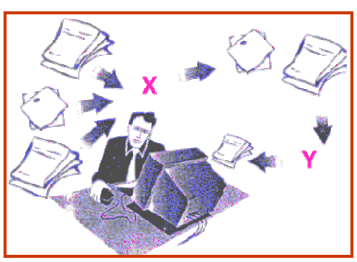
Eric Xing

Reading: K&F chapter 20.1 - 20.3

A road map to more complex dynamic models

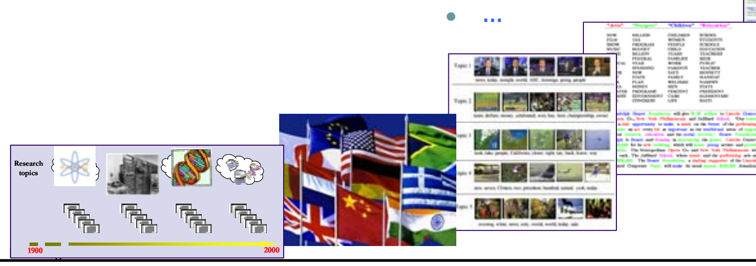


NLP and Data Mining



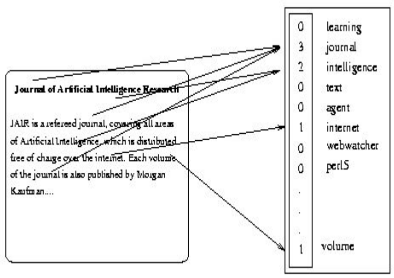
We want:

- Semantic-based search
- infer topics and categorize documents
- Multimedia inference
- Automatic translation
- Predict how topics evolve
- …

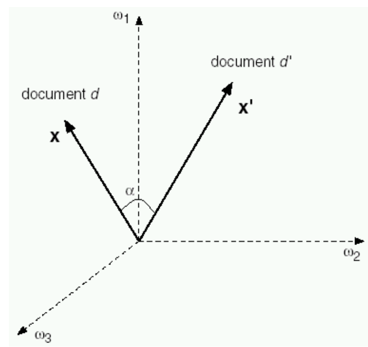


The Vector Space Model

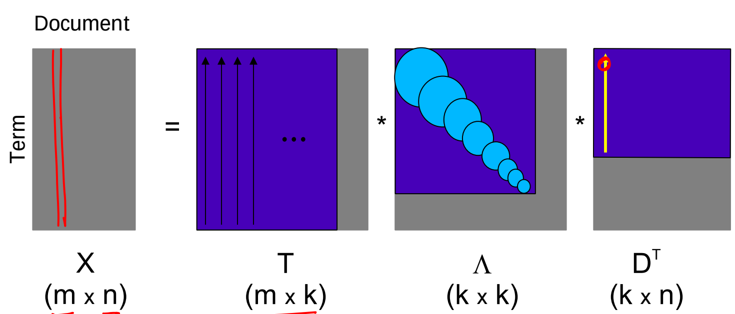
- Represent each document by a high-dimensional vector in the space of words



⇒



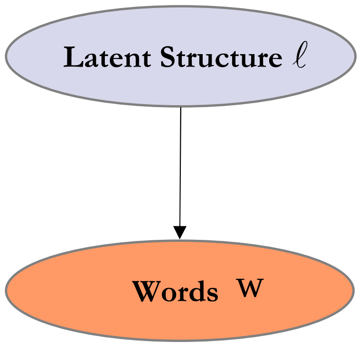
Latent Semantic Indexing



$$\vec{w} = \sum_{k=1}^{K} d_k \lambda_k \vec{T}_k$$

- LSA does not define a properly normalized probability distribution of observed and latent entities
  - Does not support probabilistic reasoning under uncertainty and data fusion

Latent Semantic Structure



Distribution over words

$$P(\mathbf{w}) = \sum_{\ell} P(\mathbf{w}, \ell)$$

Inferring latent structure

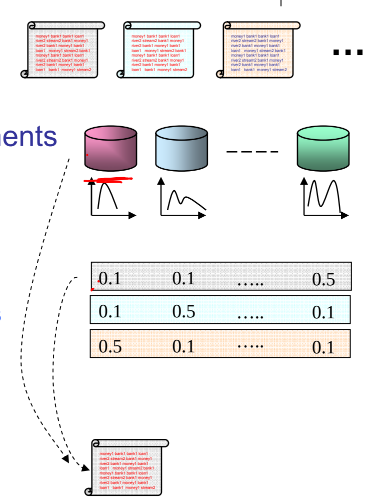
$$P(\ell \mid \mathrm{w}) = \frac{P(\mathrm{w} \mid \ell) P(\ell)}{P(\mathrm{w})}$$

Prediction

$$P(w_{n+1} \mid \mathrm{w}) = \ldots$$

Admixture Models

- Objects are bags of elements
- Mixtures are distributions over elements
- Objects have mixing vector θ
  - Represents each mixtures' contributions
- Object is generated as follows:
  - Pick a mixture component from θ
  - Pick an element from that component



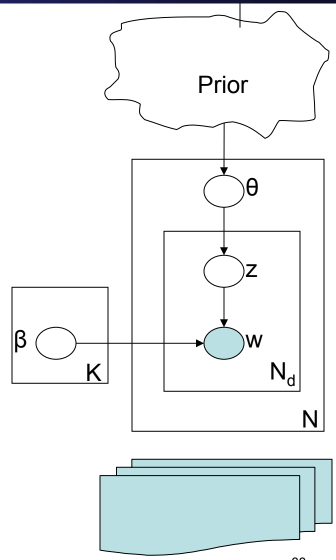
Topic Models =Admixture Models

Generating a document

– *Draw* θ from the prior

For each word n

- Draw z_n from $multinomial(\theta)$
- Draw $w_n \mid z_n, \{\beta_{1:k}\}$ from $multinomial(\beta_{z_n})$



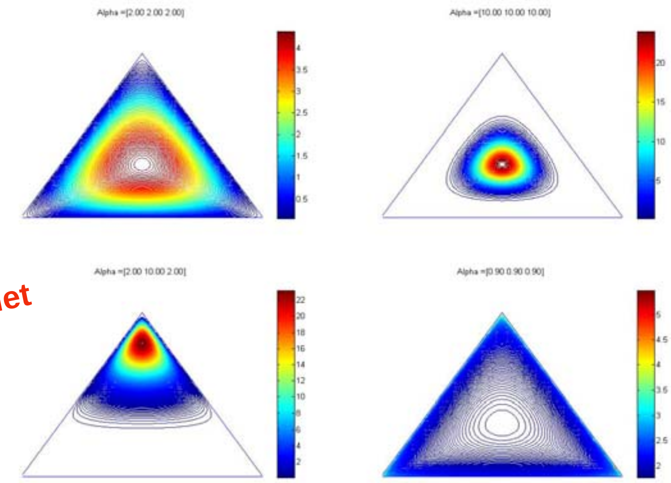
Which prior to use?

Choice of Prior

- Dirichlet (LDA) (Blei et al. 2003)
  - Conjugate prior means efficient inference
  - Can only capture variations in each topic's intensity independently
- Logistic Normal (CTM=LoNTAM) (Blei & Lafferty 2005, Ahmed & Xing 2006)
  - Capture the intuition that some topics are highly correlated and can rise up in intensity together
  - Not a conjugate prior implies hard inference

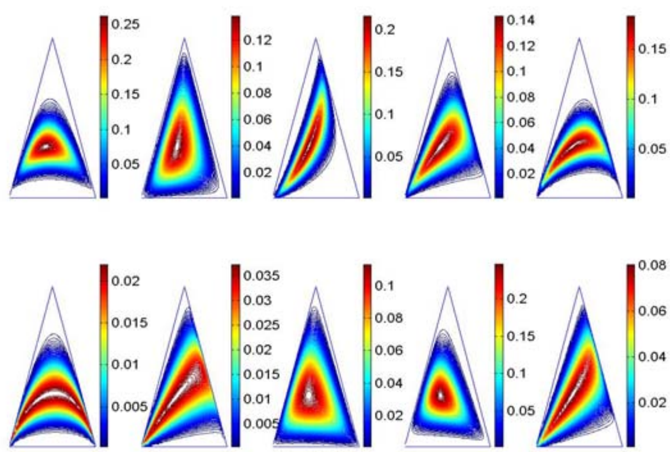
Logistic Normal Vs. Dirichlet

Dirichlet



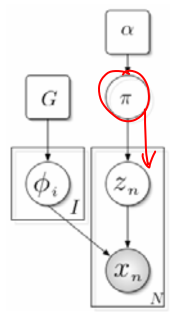
Logistic Normal Vs. Dirichlet

Logistic Normal



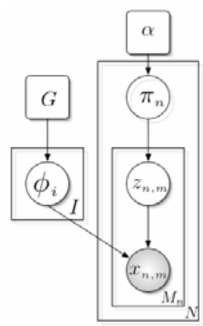
Mixed Membership Model (M^3)

- Mixture versus admixture



A Bayesian mixture model

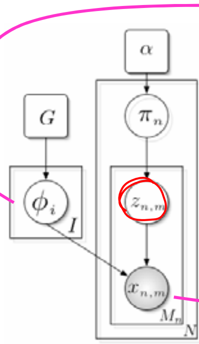
$\Rightarrow$



A Bayesian admixture model: Mixed membership model

Latent Dirichlet Allocation: M^3 in text mining

- A document is a bag of words each generated from a randomly selected topic

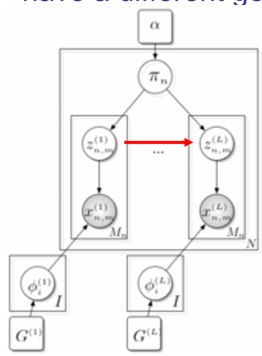


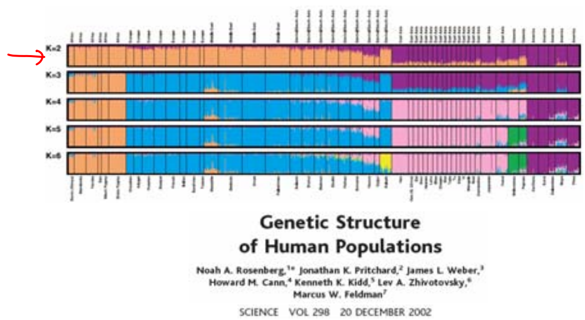
"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

Population admixture: M^3 in genetics

- The genetic materials of each modern individual are inherited from multiple ancestral populations, each DNA locus may have a different generic origin …

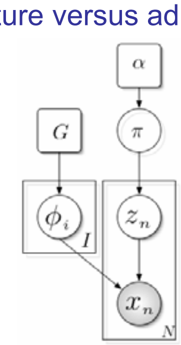




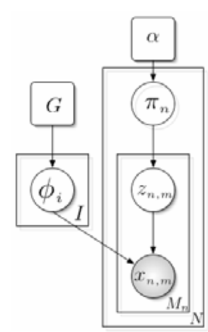
- Ancestral labels may have (e.g., Markovian) dependencies

Inference in Mixed Membership Models

- Mixture versus admixture



$\Rightarrow$



$$p(D) = \sum_{\{z_{n,m}\}} \int \cdots \int \left(\prod_n \left(\prod_m p(x_{n,m} | \phi_{z_n}) p(z_{n,m} | \pi_n) \right) p(\pi_n | \alpha) \right) p(\phi | G) d\pi_1 \cdots d\pi_N d\phi$$

- Inference is very hard in M^3, all hidden variables are coupled and not factorizable!

$$p(\pi_n | D) \sim \sum_{\{z_{n,m}\}} \int \left(\prod_n \left(\prod_m p(x_{n,m} | \phi_{z_n}) p(z_{n,m} | \pi_n) \right) p(\pi_n | \alpha) \right) p(\phi | G) d\pi_{-i} d\phi$$

Approaches to inference

- Exact inference algorithms
  - The elimination algorithm
  - The junction tree algorithms
- Approximate inference techniques
  - Monte Carlo algorithms:
    - Stochastic simulation / sampling methods
    - Markov chain Monte Carlo methods
  - Variational algorithms:
    - Belief propagation
    - Variational inference