Linear Dynamical Systems

Matt Barren

CS3750 Advanced Machine Learning

Introduction

- Consider the two following problems in a time/sequence data domain
  1. Predicting the next observation
  2. Inferencing the true value in a noisy environment

Problem 1: Predicting the Next Observation

- **Goal:** Want to model an instance, x_n, based on some arbitrary number of prior observations that occur at regular time intervals, such that

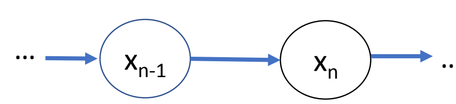
$$x_n = \sum_{i=0}^{?} b_{n-i}\, x_{n-1} + \sigma_{\mathrm{n}}, \qquad \sigma_n \sim N(0, \Sigma)$$

  - x_i is a continuous multivariate vector
  - b_i is a learned vector
  - σ_n is additive Gaussian noise
  - ? we would like to be an arbitrary value
- What is a suitable model?
  - AR(p) model?
    - Would need to define set number of P lags

AR(1): Connecting to LDS

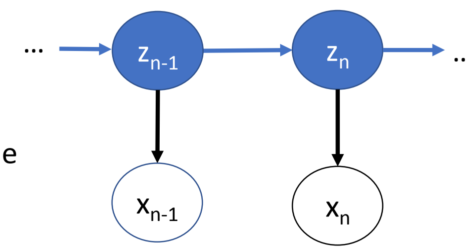
- Consider an AR(1) written as a

$$x_n = b x_{n-1} + \sigma_{\mathrm{n}}, \qquad \sigma_n \sim N(0, \Sigma)$$



AR(1): Connecting to LDS

- ~~Consider an AR(1) written as a~~
  ~~$x_n = bx_{n-1} + \sigma_n, \qquad \sigma_n \sim N(0, \sigma_n)$~~
- **Move to a state space model and assume linear Gaussian**
  - Compute the current observation based on the latent observation
    $x_n = Cz_n + w_n, \qquad w \sim N(w|0, \Sigma)$
  - Get the current latent state, z_n, from the previous latent state, z_{n-1}
    $z_n = Az_{n-1} + v_n, \qquad v \sim N(v|0, \Gamma)$



Solution – Linear Dynamical Systems(LDS)

- AR Case:
  - Move from defining a model based on prior observations to a state space model
  - Latent states incorporate signals from past observations

Problem 2: Inferencing from Noise

- **Goal:** to measure an unknown z at regular time intervals from a noisy sensor producing an observation x, such that

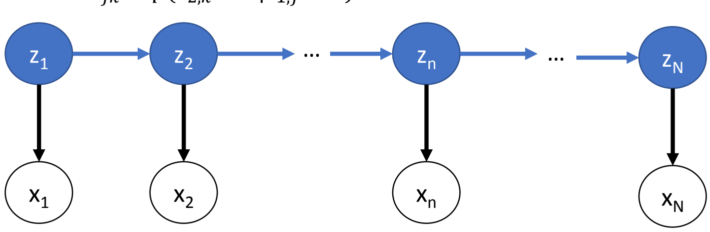
$$z_n = Bx_n + \sigma_\mathrm{n}, \qquad \sigma_n \sim N(0, \sigma)$$

  - z_n, x_n are continuous multivariate vectors
  - B is a learned probability matrix
  - σ_n is additive white noise
- What is a suitable model?
  - A HMM where
    - the **observations** are **continuous** (can be done with an HMM)
    - but the **latent states** are only **discrete**

HMM: Connecting to LDS

- Consider a Hidden Markov Model

$$A_{jk} = p(z_{2,k} = 1 | z_{1,j} = 1)$$

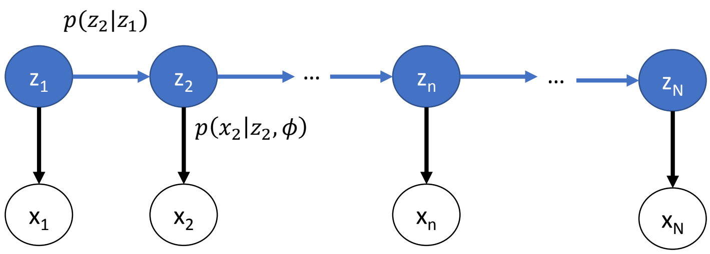


Where
Z: is a set of discrete latent RVs
X: is a set of continuous RVs
A: a matrix of discrete transition probabilities

$$A = \left[\begin{pmatrix} A_{00} & \cdots & A_{k0} \\ \vdots & \ddots & \vdots \\ A_{0k} & \cdots & A_{kk} \end{pmatrix}\right]$$

HMM: Connecting to LDS

- Consider a Hidden Markov Model

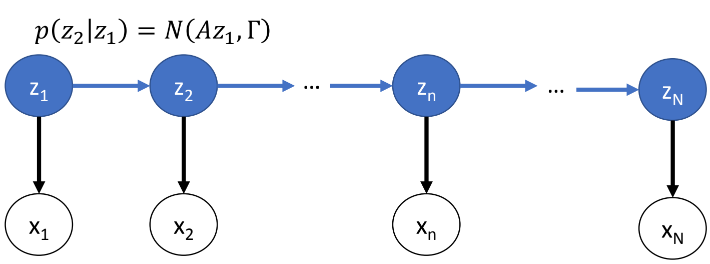


Where
Z: is a set of discrete latent RVs
X: is a set of continuous RVs
A: a matrix of discrete transition probabilities
$\boldsymbol{\phi}$**:** a matrix of emission probabilities

$$p(x_n|z_n, \phi) = \prod_{k=1}^{K} p(x_n|\phi_k)^{z_{nk}}$$

HMM: Connecting to LDS

- Extending **transitions** to LDS

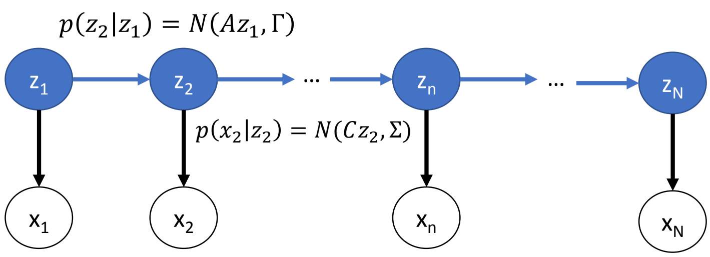


Where
X, Z: are multivariate linear Gaussians
$\boldsymbol{N(z_n|Az_{n-1}, \Gamma)}$ **:** Gaussian transitions to the next state

$$p(z_n|z_{n-1}) = N(z_n|Az_{n-1}, \Gamma)$$

HMM: Connecting to LDS

- Extending **emissions** to LDS



$$p(x_n|z_n) = N(x_n|Cz_n, \Sigma)$$

Where
X, Z: are multivariate linear Gaussians
$\boldsymbol{N(z_n|Az_{n-1}, \Gamma)}$ **:** Gaussian **transitions** to the next state
$\boldsymbol{N(x_n|Cz_n, \Sigma)}$ **:** Gaussian **emissions** of an observation

Solution – Linear Dynamical Systems(LDS)

- AR Case:
  - Move from defining a model based on prior observations to a state-space model
  - Latent states incorporate signals from past observations
- HMM Case:
  - the observations and **latent states** become **continuous**
- In both instances, we will assume a Gaussian-linear model
  - **Why**? This will become clear when considering efficiency (along with many other reasons)

What is left to discuss?

- **What tasks are appropriate for LDS?**
- **Formalization of LDS parameters ($\boldsymbol{\theta}$)**
- Inferencing in LDS
- Maximizing the Likelihood through EM
  - E-Step: Evaluation of local marginal and joint posterior distributions of the latent variables
  - M-Step: Maximizing the parameters, θ
- Extensions and Applications of LDS
- LDS Packages

What can we do with an LDS model?

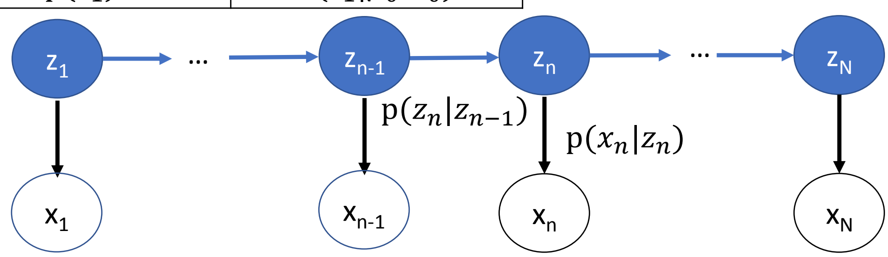
1. **Filtering:** Find the distribution of z_n given the current and all previous time observations, $p(z_n|x_1, \dots x_n)$
   - aka Kalman Filtering
2. **Prediction:** given time stamps $\mathrm{y} = \mathrm{n} + \delta$, find the observed and latent at distributions at time y, $p(z_y|x_1, \dots x_n)$ and $p(x_y|x_1, \dots x_n)$
3. **Smoothing:** For a time stamp n, find the distributions of z_n given all observations, $p(z_n|x_1, \dots x_N)$
   - aka Kalman Smoothing
4. **EM Parameter Estimation:** find $\theta = \{A, C, \mu_0, \Gamma, \Sigma, \mathrm{V}_0\}$ and the likelihood of the model of the given model $\mathrm{p}(\mathrm{X}|\theta)$

Linear Dynamical System Parameters

- HMM and LDS both use shared parameters across transition and emissions

Equivalent Conditional Distribution Representation		
Transition	$p(z_n \| z_{n-1}, A, \Gamma)$	$N(z_n \| Az_{n-1}, \Gamma)$
Emission	$p(x_n \| z_n, C, \Sigma)$	$N(x_n \| Cz_n, \Sigma)$
Initial Trans.	$p(z_1)$	$N(z_1 \| \mu_0, V_0)$

$\theta = \{\boldsymbol{A}, \boldsymbol{\Gamma}, \boldsymbol{C}, \boldsymbol{\Sigma}, \boldsymbol{\mu_0}, \boldsymbol{V_0}\}$



Linear Dynamical Systems Parameters

- HMM and LDS both use shared parameters across transition and emissions

Equivalent Conditional Distribution Representation		
Transition	$p(z_n \| z_{n-1})$	$N(z_n \| Az_{n-1}, \Gamma)$
Emission	$p(x_n \| z_n)$	$N(x_n \| Cz_n, \Sigma)$
Initial Trans.	$p(z_1)$	$N(z_1 \| \mu_0, V_0)$

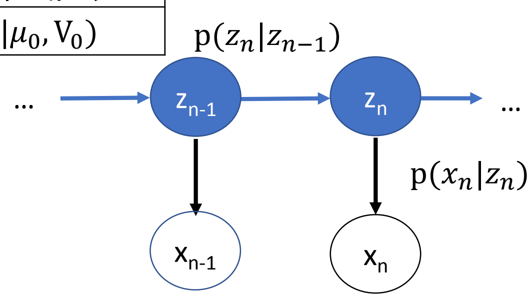
$\theta = \{\boldsymbol{A}, \boldsymbol{\Gamma}, \boldsymbol{C}, \boldsymbol{\Sigma}, \boldsymbol{\mu_0}, \boldsymbol{V_0}\}$

Equivalent Representation as noisy linear Gaussian equations

$\boldsymbol{z_n} = \boldsymbol{A z_{n-1}} + \boldsymbol{w_n} \qquad \boldsymbol{w} \sim \boldsymbol{N}(\boldsymbol{w}|\boldsymbol{0}, \boldsymbol{\Gamma})$

$\boldsymbol{x_n} = \boldsymbol{C z_n} + \boldsymbol{v_n} \qquad \boldsymbol{v} \sim \boldsymbol{N}(\boldsymbol{v}|\boldsymbol{0}, \boldsymbol{\Sigma})$

$\boldsymbol{z_1} = \boldsymbol{\mu_0} + \boldsymbol{u} \qquad \mathbf{u} \sim \boldsymbol{N}(\boldsymbol{u}|\boldsymbol{0}, \boldsymbol{V_0})$



What is left to discuss?

- ~~What tasks are appropriate for LDS?~~
- ~~Formalization of LDS parameters, θ~~
- **Inferencing and Prediction in LDS**
- Learning with EM
  - E-Step: Evaluation of local marginal and joint posterior distributions of the latent variables
  - M-Step: Maximizing the parameters, θ
- Extensions and Applications of LDS
- LDS Packages

Inferencing: Kalman Filter

- Find the **joint distribution** of the last latent variable, z_n, conditioned on the observation sequence up to time *n*
- Given θ, Find $\alpha(z_n) = p(x_1, \dots, x_n, z_n)$
- Consider the posterior distribution for latent variable z_n

$$p(z_n|X) = \frac{p(X|z_n)p(z_n)}{p(X)}$$

Kalman Filter, $\alpha(z_n)$

$$= \frac{p(x_1, \dots, x_n, \dots, x_N, z_n)}{p(X)} = \frac{p(x_1, \dots, x_n, z_n)p(x_{n+1}, \dots x_N|z_n)}{p(X)}$$

Inferencing: Kalman Filter

$$\alpha(z_n) = p(x_1, \ldots, x_{n-1}, x_n, z_n)$$

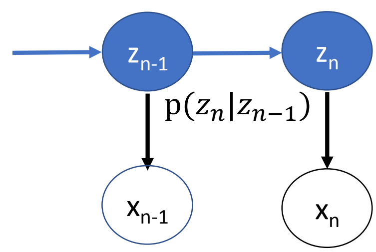
- We can separate x_N from other observations

$= p(x_n|z_n)p(x_1, \ldots, x_{n-1}, z_n)$

- Based on our model graph, knowing $p(z_{n-1})$ will d-separate z_n from the remainder of the graph

Side Note: In HMM, this is a summation. Why?

$= p(x_n|z_n)\int p(z_n|z_{n-1})p(z_{n-1})dz_{n-1}$



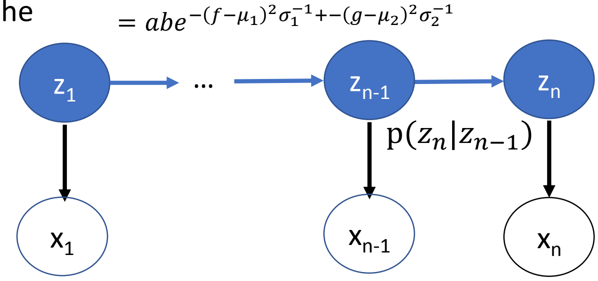
Inferencing: Kalman Filter

- Quick recap so far:
  - Defined the posterior distribution, $p(z_n|X)$, by
    1. Using Bayes Theorem find $\alpha(z_n)$
    2. D-separating to place $p(z_n|X)$ in terms of z_{n-1} and x_n
  - If $p(z_{n-1})$ were not Gaussian (or in the exponential family), this would be intractable
    - Due to parameter growth
  - Since it is Gaussian, the result is a change in parameter values only

$$p(x_1, \ldots, x_n z_n) = p(x_n|z_n)\int p(z_n|z_{n-1})p(z_{n-1})dz_{n-1}$$

Gaussians are closed under multiplication

$$p_1(f)p_2(g)$$
$$= ae^{-(f-\mu_1)^2\sigma_1^{-1}} * be^{-(g-\mu_2)^2\sigma_2^{-1}}$$
$$= abe^{-(f-\mu_1)^2\sigma_1^{-1}+-(g-\mu_2)^2\sigma_2^{-1}}$$



Inferencing: Kalman Filter

$$p(x_1, \dots, x_n z_n) = p(x_n|z_n) \int p(z_n|z_{n-1}) p(z_{n-1}) dz_{n-1}$$

- In practice, since the sequence can be very long, scaling factors, c_n are used to avoid numerical underflow

$$c_n = p(x_1 \dots x_n) = p(x_n|x_1 \dots x_{n-1})$$

- Given $\alpha(z_n) = p(z_n, x_1, \dots x_n)$, and $\hat{\alpha}(z_n) = p(z_n|x_1, \dots x_n)$

$$\alpha(z_n) = p(x_1, \dots x_n) p(z_n|x_1, \dots, x_n)$$

$$\alpha(z_n) = \prod_{i=1}^{n} c_n \, \hat{\alpha}(z_n)$$

- Turning it into a recursive equation of $\hat{\alpha}$

$$\prod_{i=1}^{n} c_n \, \hat{\alpha}(z_n) = p(x_n|z_n) \int p(z_n|z_{n-1}) \prod_{i=1}^{n-1} c_{n-1} \, \hat{\alpha}(z_{n-1}) dz_{n-1}$$

$$c_n \hat{\alpha}(z_n) = p(x_n|z_n) \int p(z_n|z_{n-1}) \hat{\alpha}(z_{n-1}) dz_{n-1}$$

Inferencing: Kalman Filter

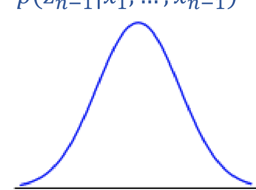
$$\hat{\alpha}(z_n) = p(x_n|z_n) \int p(z_n|z_{n-1}) p(z_{n-1}) dz_{n-1}$$

- Replacing with the Gaussian form equivalent

$$c_n \hat{\alpha}(z_n) = \mathrm{N}(x_n|Cz_n, \Sigma) \int N(z_n|Az_{n-1}, \Gamma) N(z_{n-1}|\mu_n, V_n) dz_{n-1}$$

Parametrized similar to the initial state, z_0

$p(z_{n-1}|x_1, \dots, x_{n-1})$

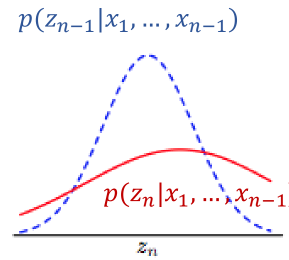


- Evaluating just the Integral

$$\int \frac{1}{(2\pi)^{\frac{p}{2}} |\Gamma|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(z_n - Az_{n-1})^T \Gamma^{-1} (z_n - Az_{n-1})\right\}$$

$$\frac{1}{(2\pi)^{\frac{p}{2}} |\mathrm{V_n}|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(z_{n-1} - \mu_{n-1})^T \mathrm{V_n}^{-1} (z_{n-1} - \mu_{n-1})\right\} dz_{n-1}$$

Inferencing: Kalman Filter



- After integration and applying the rule of conditioning, we get the marginalized normal distribution of the latent state

$$c_n\hat{\alpha}(z_n) = \mathrm{N}(x_n|Cz_n,\Sigma)N(z_n|A\mu_{n-1},\mathrm{P}_{n-1}),$$

where $\mathrm{P}_{n-1} = AV_{n-1}A^T + \Gamma$

- Note $p(z_n|x_1,\dots,x_{n-1})$ with the transition variance given by $p(z_n|z_{n-1})$ is broader compared to $p(z_{n-1}|x_1,\dots,x_{n-1})$

Inferencing: Kalman Filter

$p(z_n|X)$

- $c_n\hat{\alpha}(z_n) = \underbrace{N(x_n|Cz_n,\Sigma)}_{p(x_n|z_n)}\underbrace{N(z_n|A\mu_{n-1},\mathrm{P}_{n-1})}_{p(z_n|x_1,\dots,x_{n-1})}$
- Using the rule of conditioning $[p(a|b)p(b) = p(a)p(b|a)]$

$$c_n\hat{\alpha}(z_n) = N\left(x_n|AC\mu_{n-1},\mathrm{CP}_{n-1}\mathrm{C}^{\mathrm{T}}+\Sigma\right)$$

$$N(z_n|A\mu_{n-1}+K(x_n-CA\mu_{n-1}),(I-K_nC)P_{n-1})$$

where K_n is the Kalman Gain Matrix $K_n = P_{n-1}C^T(CP_{n-1}C^T+\Sigma)$

- Equivalently in linear Gaussian notation we get

$$\mu_n = A\mu_{n-1} + K_n(x_n - CA\mu_n), \qquad V_N = (I-K_nC)P_{n-1}$$

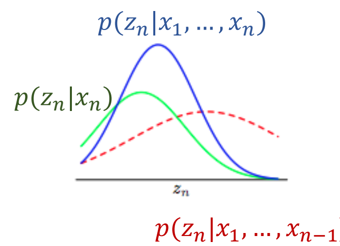
$$c_n = N\left(x_n|AC\mu_{n-1},\mathrm{CP}_{n-1}\mathrm{C}^{\mathrm{T}}+\Sigma\right)$$

Inferencing: Kalman Filter

$\mu_N = \boldsymbol{A\mu_{n-1}} + K_n(x_n - CA\mu_n), \quad V_n = (I - K_nC)P_{n-1}$

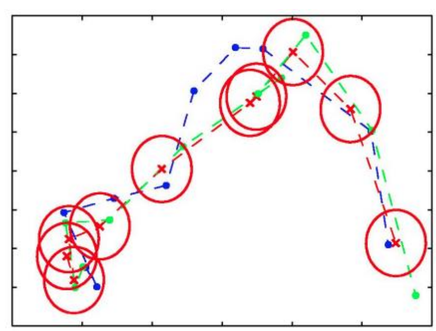
$c_{n=} N\left(x_n \middle| AC\mu_{n-1}, \mathrm{CP_{n-1}C^T} + \Sigma\right)$

- Examining μ_N,
  - $\boldsymbol{A\mu_{n-1}}$ is projection of the new mean using the transition probability matrix and the prior mean
  - $CA\mu_n$ the predicted x_N (the new mean applied to the emission probability matrix)
  - $x_n - CA\mu_n$ a correction of the observation by the prediction
  - K_n the coefficient of the correction (Kalman Gain Matrix)
- $p(z_n|x_n)$ contribution of the emission density with the current observation has tightened $p(z_n|x_1, \dots, x_n)$



Kalman Filtering in Tracking

- Following an object in a graphical 2D space
- What are the red x's and circles?
  - Means and variances at each time step
- **Kalman Filtering** is just successive applications of a current observation and the prior local latent marginal posterior to the next marginal posterior



Blue Dots: True position

Green Dots: Noisy Observations

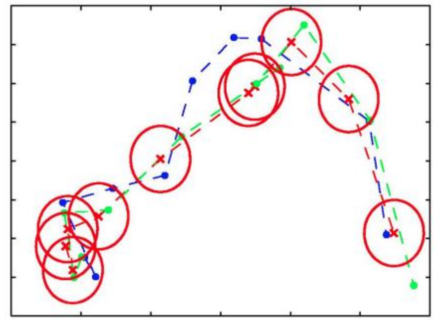
Tracking: Extending to Prediction

- How do we predict a new hidden state at time $N+1$?
$$P(z_{N+1}|x_1, \dots x_N, z_1, \dots, z_N)$$
- Assuming $P(z_N)$ is known, $N(z_N|\mu_N, V_N)$

$P(z_{N+1}|x_1, \dots x_N, z_1, \dots, z_N)$

$p(z_{N+1}) = \int p(z_{N+1}|z_N)p(z_N)dz_N$

- Solve for the next latent state by marginalizing over the prior
- Still using forward propagation without $p(z_{N+1}|x_{N+1})$



Blue Dots: True position

Green Dots: Noisy Observations

What is left to discuss?

- ~~What tasks are appropriate for LDS?~~
- ~~Formalization of LDS parameters, θ~~
- ~~Inferencing and Prediction in LDS~~
- **Learning with EM**
  - **M-Step: Maximizing the parameters, $\boldsymbol{\theta}$**
  - **E-Step: Evaluation of local posteriors**
- Extensions and Applications of LDS
- LDS Packages

Learning with EM

- Consider the complete-data log likelihood function

$$\ln p(X,Z|\theta) = \ln p(z_1|\mu_0, V_0) + \sum_{n=2}^{N} \ln p(z_n|z_{n-1}, A, \Gamma) + \sum_{n=1}^{N} \ln p(x_n|z_n, C, \Sigma)$$

- **E-Step** Take the expectation w.r.t. to the posterior distribution

$$Q(\theta, \theta^{old}) = \mathbb{E}_{z|\theta^{old}}[\ln p(X,Z|\theta)]$$

- **M-Step** Then maximize w.r.t. to the parameters θ
  - Take the the derivative of parameter and solve for it equal to 0
  - $\theta = \{A, C, \mu_0, \Gamma, \Sigma, V_0\}$

Learning with EM: E-Step

- Consider the complete-data log likelihood function

$$\ln p(X,Z|\theta) = \ln p(z_1|\mu_0, V_0) + \sum_{n=2}^{N} \ln p(z_n|z_{n-1}, A, \Gamma) + \sum_{n=1}^{N} \ln p(x_n|z_n, C, \Sigma)$$

- Take the expectation w.r.t. to the posterior distribution

$$\mathbb{E}_{z|\theta}[\ln p(X,Z|\theta)]$$

- Depends on the following quantities (We don't know these yet?)
  - $\mathbb{E}[z_n] = \hat{\mu}_n,$
  - $\mathbb{E}[z_n z_{n-1}^T] = J_{n-1}\hat{V}_n + \hat{\mu}_n \hat{\mu}_{n-1}^T$
  - $\mathbb{E}[z_n z_n^T] = \hat{V}_n + \hat{\mu}_n \hat{\mu}_n^T$

Find them with local posterior distributions

Find the marginal posterior distribution of the latent variables

$$\gamma(z_n) = p(z_n|X, \theta_t)$$

Joint posterior distribution of successive latent variables

$$\xi(z_n, z_{n-1}) = p(z_n, z_{n-1}|X, \theta_t)$$

E-Step: Marginal Posteriors

- Does this look familiar?

$$p(z_n|X) = \gamma(z_n) = \frac{p(x_1, \dots, x_n, z_n)p(x_{n+1}, \dots, x_N|z_n)}{p(X)}$$

- Forward and backward recursions (can find sum-product equivalent)

$$= \frac{\alpha(z_n)\beta(z_n)}{p(X)}$$

where $\alpha(z_n) = p(x_1, \dots, x_n, z_n)$ and $\beta(z_n) = p(x_{n+1}, \dots, x_N|z_n)$

Kalman Filter (we know this!) — Kalman Smoother (find this)

E-Step: Backward Recursion

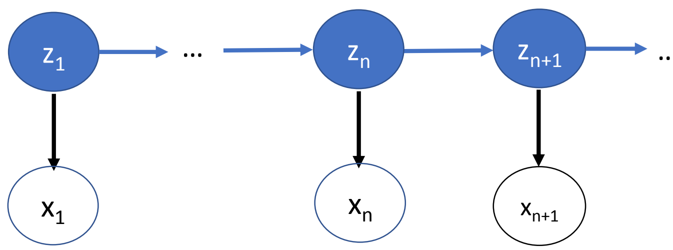
$$\gamma(z_n) = \frac{\alpha(z_n)\beta(z_n)}{p(X)} = \hat{\alpha}(z_n)\hat{\beta}(z_n) = N\left(z_n\middle|\hat{\mu}_n, \hat{V}_n\right)$$

- $\gamma(z_n)$ must be Gaussian since it is the products of Gaussians
- Formulation difference between HMM $\alpha - \beta$ and LDS $\alpha - \gamma$
  - They are equivalent in their result
  - $\alpha - \gamma$ requires α is performed first
- What role does this play in a real-time prediction?
  - This is needed to learn the parameters, θ, that are used when applying forward recursions

E-Step: Backward Recursion

1. First derive the backward recursive equation, $\beta(z_n)$

$$\beta(z_n) = p(x_{n+1}, \dots, x_N | z_n)$$
$$= \int p(x_{n+1}, \dots, x_N, z_{n+1} | z_n) dz_{n+1}$$
$$= \int p(x_{n+1}, \dots, x_N | z_{n+1}) p(z_{n+1} | z_n) dz_{n+1}$$
$$= \int p(x_{n+2}, \dots, x_N | z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n) dz_{n+1}$$



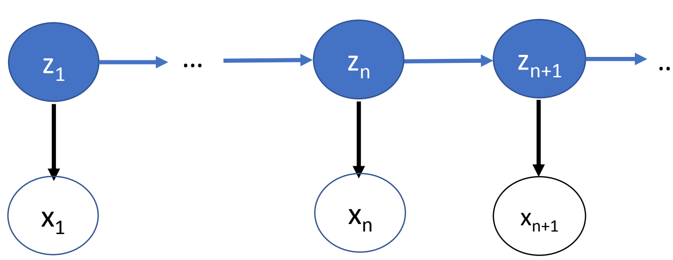
E-Step: Backward Recursion

1. First derive the backward recursive equation, $\beta(z_n)$, cont'd

$$= \int p(x_{n+2}, \dots, x_N | z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n) dz_{n+1}$$
$$= \int \beta(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n) dz_{n+1}$$

- Similarly apply the next scaling factor to $\hat{\beta}(z_n)$

$$c_{n+1} \hat{\beta}(z_n) = \int \hat{\beta}(z_{n+1}) p(x_{n+1} | z_{n+1}) p(z_{n+1} | z_n) dz_{n+1}$$

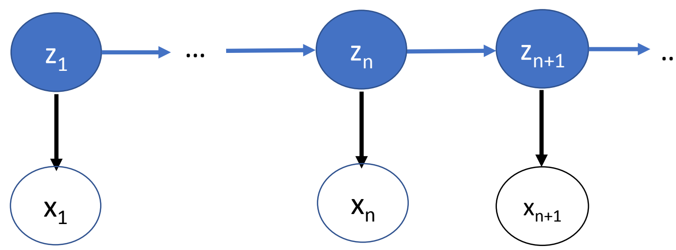


E-Step: Backward Recursion

2. Change it into a problem defined by $\gamma(z_n)$ by multiplying $\hat{\alpha}(z_n)$ on both sides
   - $c_{n+1}\hat{\alpha}(z_n)\hat{\beta}(z_n) = \hat{\alpha}(z_n)\int \hat{\beta}(z_{n+1})p(x_{n+1}|z_{n+1})p(z_{n+1}|z_n)dz_{n+1}$

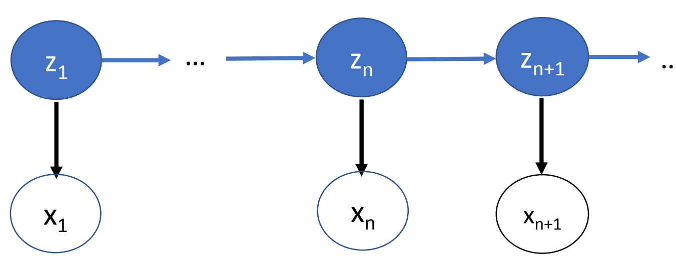
$\beta(z_n) = p(x_{n+1}, \dots, x_N|z_n)$

$\alpha(z_n) = p(x_1, \dots, x_n, z_n)$



E-Step: Backward Recursion

3. Substituting $\hat{\alpha}(z_{n+1})$ for $\hat{\beta}(z_{n+1})$ and applying the same Gaussian conditioning as with forward recursion we get
   - $\hat{\mu}_n = \mu_n + J_n(\hat{\mu}_{n+1} - A\mu_n)$, $\quad \hat{\mathrm{V}}_\mathrm{n} = \mathrm{V}_\mathrm{n} + \mathrm{J}_\mathrm{n}(\hat{\mathrm{V}}_{n+1} - P_n)J_n^T$
   - Where $J_n^T = V_n A^T (P_n)^{-1}$



E-Step: Joint Pairwise Posterior Distributions

$$\xi(z_n, z_{n-1}) = p(z_n, z_{n-1}|X)$$
$$\xi(z_n, z_{n-1}) = (c_n)^{-1}\hat{\alpha}(z_{n-1})p(x_n|z_n)p(z_n|z_{n-1})\hat{\beta}(z_n)$$

- Inserting the Conditional Distributions, and substituting $\hat{\gamma}(z_n)$ for $\hat{\beta}(z_n)$

$$\xi(z_n, z_{n-1}) = N(z_{n-1}|\mu_{n-1}, V_{n-1})N(z_n|Az_{n-1}, \Gamma)N(x_n|Cz_n, \Sigma)\frac{N(z_n|\hat{\mu}_n, \hat{V}_n)}{c_n N(z_n|u_n, V_n)}$$

Similar to **forward/backward recursion**, solve through the rule of Gaussian conditioning

$$\mu[z_n, z_{n-1}] = \hat{\mu}_n\hat{\mu}_{n-1}^T\ , cov[z_n, z_{n-1}] = J_{n-1}\hat{V}_n$$

Learning with EM

- Consider the complete-data log likelihood function

$$\ln p(X, Z|\theta) = \ln p(z_1|\mu_0, V_0) + \sum_{n=2}^{N} \ln p(z_n|z_{n-1}, A, \Gamma) + \sum_{n=1}^{N} \ln p(x_n|z_n, C, \Sigma)$$

- **E-Step** Take the expectation w.r.t. to the posterior distribution

$$Q(\theta, \theta^{old}) = \mathbb{E}_{Z|\theta^{old}}[\ln p(X, Z|\theta)]$$

- **M-Step** Then maximize w.r.t. to the parameters θ
  - Take the the derivative of parameter and solve for it equal to 0
  - $\theta = \{A, C, \ \mu_0, \Gamma, \Sigma, V_0\}$

M-Step: Initial State μ_0, V_0

$$Q\left(\theta, \theta^{old}\right) = \mathbb{E}_{Z|\theta^{\text{old}}}[\ln p(z_1|\mu_0, V_0)\ ]$$

- Expanding the distribution of $p(z_1|\mu_0, V_0)$ and pushing in the expectation

$$Q\left(\theta, \theta^{old}\right) = -\frac{1}{2}\ln|V_0| - \mathbb{E}_{Z|\theta^{old}}[\frac{1}{2}(z_1 - \mu_0)^T V_0^{-1}(z_1 - \mu_0)] + const$$

$$\frac{\partial Q}{\partial \sigma_0} = (\mathbb{E}[z_1] - \mu_0)V_0^{-1} = 0 \qquad \frac{\partial Q}{\partial V_0^{-1}} = \frac{1}{2}V_0 - \frac{1}{2}(\mathbb{E}[z_1 z_1^T] - \mathbb{E}[z_1]\mu_0^T - \mu_0\mathbb{E}[z_1]^T + \mu_0\mu_0^T$$

$$\mu_0^{new} = \mathbb{E}[z_1] \qquad V_0^{new} = \mathbb{E}[z_1 z_1^T] - \mathbb{E}[z_1\ ]\mathbb{E}[z_1]^T$$

M-Step: Transitions A, Γ

$$Q\left(\theta, \theta^{old}\right) = \mathbb{E}_{Z|\theta^{\text{old}}}[\sum_{n=2}^{N} \ln p(z_n|z_{n-1}, A, \Gamma)]$$

$$Q\left(\theta, \theta^{old}\right) = -\frac{N-1}{2}\ln|\Gamma| - \mathbb{E}_{Z|\theta^{old}}[\frac{1}{2}\sum_{i=0}^{n}(z_n - Az_{n-1})^T\Gamma^{-1}\ (z_n - Az_{n-1})] + const$$

$$\frac{\partial Q}{\partial A} = -\sum_{n=2}^{N}\Gamma^{-1}\,\mathbb{E}[z_n z_{n-1}^T] + \sum_{n=2}^{N}\Gamma^{-1}\,A\mathbb{E}[z_{n-1}z_{n-1}^T] = 0$$

$$\mathrm{A}^{\text{New}} = (\textstyle\sum_{n=2}^{N}\mathbb{E}[z_n z_{n-1}^T])(\sum_{n=2}^{N}\mathbb{E}[z_{n-1}z_{n-1}^T])^{-1}$$

M-Step: Transitions A, Γ

$$Q(\theta, \theta^{old}) = \mathbb{E}_{Z|\theta^{\text{old}}}[\sum_{n=2}^{N} \ln p(z_n|z_{n-1}, A, \Gamma)]$$

$$Q(\theta, \theta^{old}) = -\frac{N-1}{2}\ln|\Gamma| - \mathbb{E}_{Z|\theta^{old}}[\frac{1}{2}\sum_{i=0}^{n}(z_n - Az_{n-1})^T\Gamma^{-1}(z_n - Az_{n-1})] + const$$

$$\frac{\partial Q}{\partial \Gamma} = \frac{N-1}{2}\Gamma - \frac{1}{2}\sum_{n=2}^{N}(\mathbb{E}[z_n z_n^T] - A\mathbb{E}[z_{n-1}z_n^T] - \mathbb{E}[z_n z_{n-1}^T]A^T + A\mathbb{E}[z_{n-1}z_{n-1}^T]A^T) = 0$$

$$\Gamma^{New} = \frac{1}{N-1}\sum_{n=2}^{N}\{\mathbb{E}[z_n z_n^T] - A\mathbb{E}[z_{n-1}z_n^T] - \mathbb{E}[z_n z_{n-1}^T]A^T + A\mathbb{E}[z_{n-1}z_{n-1}^T]A^T\}$$

M-Step: Emissions C, Σ

$$Q(\theta, \theta^{old}) = \mathbb{E}_{Z|\theta^{\text{old}}}[\sum_{n=1}^{N} \ln p(x_n|z_n, C, \Gamma)]$$

$$Q(\theta, \theta^{old}) = -\frac{N-1}{2}\ln|\Sigma| - \mathbb{E}_{Z|\theta^{old}}[\frac{1}{2}\sum_{n=1}^{n}(x_n - Cz_n)^T\Sigma^{-1}(x_n - Cz_n)] + const$$

$$\frac{\partial Q}{\partial C} = -\sum_{n=1}^{N}\Sigma^{-1}x_n\mathbb{E}[z_n]^T + \sum_{n=1}^{N}\Sigma^{-1}C\mathbb{E}[z_n z_n^T] = 0$$

$$\mathrm{C}^{\mathrm{New}} = (\textstyle\sum_{n=1}^{N} x_n\mathbb{E}[z_{n-1}^T])(\mathbb{E}[z_n z_{\mathrm{n}}^T])^{-1}$$

M-Step: Emissions C, $\boldsymbol{\Sigma}$

$$Q(\theta, \theta^{old}) = \mathbb{E}_{Z|\theta^{\text{old}}}[\sum_{n=1}^{N} \ln p(z_n|z_{n-1}, C, \Sigma)]$$

$$Q(\theta, \theta^{old}) = -\frac{N-1}{2}\ln|\Sigma| - \mathbb{E}_{Z|\theta^{old}}[\frac{1}{2}\sum_{n=1}^{n}(x_n - Cz_n)^T\Sigma^{-1}(x_n - Cz_n)] + const$$

$$\frac{\partial Q}{\partial \mathrm{C}} = \frac{N-1}{2}\mathrm{C} - \frac{1}{2}\sum_{n=2}^{N}(x_n x_n^T - C\mathbb{E}[z_n]x_n^T - x_n^T\mathbb{E}[z_n^T]C + C\mathbb{E}[z_{n-1}z_{n-1}^T]C^T) = 0$$

$$\Sigma^{New} = \frac{1}{N-1}\sum_{n=2}^{N}\{x_n x_n^T - C\mathbb{E}[z_n]x_n^T - x_n^T\mathbb{E}[z_n^T]C + C\mathbb{E}[z_{n-1}z_{n-1}^T]C^T\}$$

Evaluating the Model Likelihood

- The model parameters can be evaluated based on the following likelihood function

$p(\boldsymbol{X}|\boldsymbol{\theta}) = \prod_{n=1}^{N} c_n$, where $c_n = p(x_n|x_1, \ldots x_{n-1})$ and

$$\hat{\alpha}_n = \frac{\alpha_n}{p(x_n|x_1, \ldots, x_n)}$$

- Continue updating the parameters until some threshold of likelihood gain is achieved

What is left to discuss?

- ~~What tasks are appropriate for LDS?~~
- ~~Formalization of LDS parameters, θ~~
- ~~Inferencing and Prediction in LDS~~
- ~~Learning with EM~~
  - ~~M-Step: Maximizing the parameters, θ~~
  - ~~E-Step: Evaluation of local posteriors~~
- **Applications of LDS**
- **LDS Packages**

LDS Extensions

- Learning Stable Linear Dynamical Systems (Boots)
  - Parameters are not guaranteed to converge in LDS
  - Given a long set of sequences this can be a problem
  - Provide a parameter constraint that forces the largest eigenvector to be equal to 1 at each EM step
- Learning Linear Dynamical Systems from Multivariate Time Series: A Matrix Factorization Based Framework (Liu, Hauskrecht)
  - Parameters are learned through a sequence of matrix factorizations (*instead of EM*)

Applications

- Polyphonic Sound Event Tracking Using Linear Dynamical Systems (Benetos, Lafay, Lagrange, Plumbley)
  - Tracking overlapping sound events
  - 4D spectral template dictionary of frequency
    - Dictionary of frequency, sound event class, exemplar inc sound state
  - Estimation of distinguishing office sounds

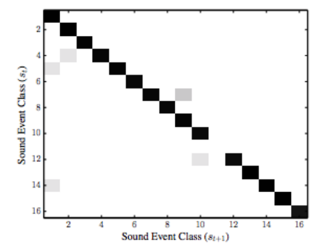


Fig. 4. The LDS transition matrix **A** trained on sequences of office sounds. Sound class indices 1-16 are listed in subsection IV-A.

Python Packages

- Pylds – supports linear Gaussian state dynamics
  - https://github.com/mattjj/pylds
  - Message passing code in Cython
  - Uses BLAS and LAPACK routines linked to the scipy build
  - Supports linear Gaussian state dynamics

References

1. Bishop, C. M. (2016). *Pattern Recognition and Machine Learning*. S.l.: Springer-Verlag New York.
2. From Hidden Markov Models to Linear Dynamical Systems. (1999). Thomas P. Minka. *Tech. Rep. 531, Vision and Modeling Group of Media Lab, MIT*. Retrieved from http://www.stat.columbia.edu/~liam/teaching/neurostat-fall17/papers/hmm/minka-lds-techreport.pdf
3. Linear Time Series Analysis and Its Applications. (n.d.). *Analysis of Financial Time Series,* 22-78. doi:10.1002/0471264105.ch2
4. Boots, B. (n.d.). Learning Stable Linear Dynamical Systems. *Machine Learning Department Carnegie Mellon University*. Retrieved from https://www.ml.cmu.edu/research/dap-papers/dap_boots.pdf.
5. Liu, Z., & Hauskrecht, M. (2016). Learning Linear Dynamical Systems from Multivariate Time Series: A Matrix Factorization Based Framework. *Proceedings of the 2016 SIAM International Conference on Data Mining*. doi:10.1137/1.9781611974348.91
6. Ghahramani, Z. & Hinton, E. (1996) Parameter Estimation for Linear Dynamical Systems.
7. Zafeiriou, S. (n.d.). Linear Dynamical Systems (Kalman filter). Retrieved from https://ibug.doc.ic.ac.uk/media/uploads/documents/kalman_filter.pdf

- Some additional slides related to expectation
  - These focus on conditioning Gaussians by using partition matrices (which is discussed in Chapter 2 of Bishop)

Expectation – $\alpha(z_n)$ Forward Recursion

Inserting the Conditional Distributions

$$\alpha(z_n) = p(x_n|z_n)\int \alpha(z_{n-1})p(z_n|z_{n-1})dz_{n-1}$$

$$c_n N(z_n|\mu_n, V_n) = N(x_n|Cz_n, \Sigma)\int N(z_n|Az_{n-1}, \Gamma)N(z_{n-1}|\mu_{n-1}, V_{n-1})dz_{n-1}$$

- μ_{n-1}, V_{n-1} are known values from the last time step
- Want to find parameters μ_n , V_n, and c_n, the parameters of the posterior marginal distribution

Gaussian Conditioning and Affine Transformations

Gaussian Conditioning and Affine Transformations

- Given $p(x) = N(x|\mu_x, V_x), p(y|x) = N(y|A\mu_x, V_y)$
- The Gaussian random vector can be partitioned
- $\begin{bmatrix} x \\ y \end{bmatrix} \sim N(\begin{bmatrix} m_x \\ Am_x \end{bmatrix}, \begin{bmatrix} V_x & V_x A^T \\ AV_x & AV_x A^T + V_y \end{bmatrix})$
- Covariance partitioning is of the form $\begin{bmatrix} \Sigma_{xx} & \Sigma_{yx} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix}$
- Solving for $p(y)$, $\mathrm{p(x|y)}$ with an Affine transformation, where $p(x)\mathrm{p(y|x)} = \mathrm{p(x|y)p(y)}$

Gaussian Conditioning and Affine Transformations

- Solving for $p(y)$, $\mathrm{p(x|y)}$ with an Affine transformation

$p(y) = N(Am_x, AV_x A^T + V_y)$ this being Σ_{yy}

$p(x|y) = N(m_x + K(y - Am_x), (I - KA)V_x)$ this being Σ_{yy}

where $K = V_x A^T \left(AV_x A^T + V_y\right)^{-1}$, or $K = \Sigma_{\mathrm{yx}} \Sigma_{yy}^{-1}$

We will employ this same strategy twice to solve for $\boldsymbol{\mu_n}$, $\boldsymbol{V_n}$, $\boldsymbol{and\ c_n}$

Expectation – $\alpha(z_n)$ Forward Recursion

Evaluating the integral: $\int N(z_n|Az_{n-1},\Gamma)N(z_{n-1}|\mu_{n-1},V_{n-1})dz_{n-1}$

$\begin{bmatrix} z_{n-1} \\ z_n \end{bmatrix} \sim N(\begin{bmatrix} \mu_{n-1} \\ A\mu_{n-1} \end{bmatrix}, \begin{bmatrix} V_{n-1} & V_{n-1}A^T \\ AV_{n-1} & AV_{n-1}A^T + \Gamma \end{bmatrix})$

$\int p(z_{n-1}|z_n)p(z_n)dz_{n-1}$

$\int N(z_n|A\mu_{n-1},\mathrm{P}_{\mathrm{n-1}})N(z_{n-1}|K(z_n - A\mu_{n-1}),(I-KA)V_{n-1})dz_{n-1}$

where $P_{n-1} = AV_{n-1}A^T + \Gamma$

After integration

$N(z_n|A\mu_{n-1},\mathrm{P}_{\mathrm{n-1}})$

Note: the mean and variance of the prior step has been passed forward

Expectation – $\alpha(z_n)$ Forward Recursion

Evaluating: $N(x_n|Cz_n,\Sigma)N(z_{n-1}|\mu_{n-1},V_{n-1})$

$\begin{bmatrix} z_n \\ x_n \end{bmatrix} \sim N(\begin{bmatrix} A\mu_{n-1} \\ CA\mu_{n-1} \end{bmatrix}, \begin{bmatrix} P_{n-1} & P_{n-1}C^T \\ CP_{n-1} & CP_{n-1}C^T + \Sigma \end{bmatrix})$

$$c_n\hat{\alpha}_n = p(x_n)p(z_n|x_n)$$

$N\left(z_n \middle| CA\mu_{n-1},\mathrm{CP}_{\mathrm{n-1}}\mathrm{C}^{\mathrm{T}} + \Sigma\right)$

$N(z_n|A\mu_{n-1} + K(x_n - CA\mu_{n-1}),(I-K_nC)P_{n-1})$

where $\mathrm{K} = CP_{n-1}(CP_{n-1}C^T + \Sigma)^{-1}$, or the Kalman Gain Matrix