

COMS30035, Machine learning: Sequential Data (LDS)

James Cussens

School of Computer Science
University of Bristol

29th October 2024

Acknowledgement

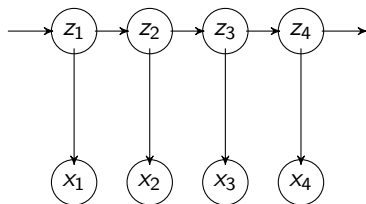
- ▶ These slides are adapted from ones originally created by Edwin Simpson.

Agenda

- ▶ Markov Models
- ▶ Hidden Markov Models
- ▶ EM for HMMs
- ▶ Linear Dynamical Systems

From HMM to LDS

- ▶ HMM assumes discrete latent states.
- ▶ Linear dynamical systems (LDS) assume states have continuous values.
- ▶ Both have the same graphical model:



- ▶ Inference has the same form as for an HMM, but when marginalising z_{n-1} and z_{n+1} , we take integrals instead of sums.

Motivations for LDS

- ▶ Noisy sensors: inferring the true sequence of states from observations with Gaussian noise.
- ▶ Tracking: predicting the next movement and tracing the path from noisy observations.

Transition and Emission Distributions for LDS

- ▶ $p(\mathbf{z}_1) = \mathcal{N}(\mathbf{z}_1 | \boldsymbol{\mu}_0, \mathbf{V}_0);$
- ▶ $p(\mathbf{z}_n | \mathbf{z}_{n-1}) = \mathcal{N}(\mathbf{z}_n | \mathbf{A}\mathbf{z}_{n-1}, \boldsymbol{\Gamma});$
- ▶ $p(\mathbf{x}_n | \mathbf{z}_n) = \mathcal{N}(\mathbf{x}_n | \mathbf{C}\mathbf{z}_n, \boldsymbol{\Sigma}).$
- ▶ Note that the means of both distributions are *linear* functions of the latent states.
- ▶ This choice of distributions ensures that the posteriors are also Gaussians with updated parameters
- ▶ This means that $\mathcal{O}(N)$ inference can still be performed using the sum-product algorithm.

Inference for an LDS

- ▶ *Kalman filter* = forward pass of sum-product for LDS.
- ▶ *Kalman smoother* = backward pass of sum-product for LDS.
- ▶ No need for an analogue of Viterbi: the most likely sequence is given by the individually most probable states, so we get this from the Kalman equations.

Forward Inference (Kalman Filter) for an LDS



$$\alpha(\mathbf{z}_n) = \mathcal{N}(\mathbf{x}_n | \mathbf{C}\mathbf{z}_n, \Sigma) \int \mathcal{N}(\mathbf{z}_n | \mathbf{A}\mathbf{z}_{n-1}, \Gamma) \alpha(\mathbf{z}_{n-1}) d\mathbf{z}_{n-1} \quad (1)$$

- ▶ Normalising results in a Gaussian-distributed variable, whose parameters can be computed efficiently:
 $\hat{\alpha}(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{x}_1, \dots, \mathbf{x}_n) = \mathcal{N}(\mathbf{z}_n | \boldsymbol{\mu}_n, \mathbf{V}_n)$, where
 - ▶ $\boldsymbol{\mu}_n$ is a function of $\boldsymbol{\mu}_{n-1}$, \mathbf{x}_n , \mathbf{A} and \mathbf{C} .
 - ▶ \mathbf{V}_n is a function of \mathbf{V}_{n-1} , Σ , \mathbf{A} , Γ and \mathbf{C} .
- ▶ We can view each forward step as predicting \mathbf{z}_n based on the distribution over \mathbf{z}_{n-1} , then correcting that prediction given the new observation \mathbf{x}_n .
- ▶ For details, see [Bis06, §13.3.1].

Backward Inference (Kalman Smoother) for an LDS

- ▶ Backward pass also follows that of the HMM: messages are passed from the final state to the start of the sequence.
- ▶ The backward messages contain information about future states that affects the posterior distribution at each step n .
- ▶ Since the transition and emission probabilities are all Gaussian, the posterior *responsibilities* are also Gaussian, as are the *state pair* expectations.
- ▶ For details, see [Bis06, §13.3.1].

Learning the Parameters of LDS

- ▶ Kalman filter/smoothers are analogous to the forward-backward algorithm for HMMs.
- ▶ Remember that this algorithm is used for the *E step* of EM.
- ▶ The parameters are optimised in the *M step* as before, by using the responsibilities $\mathbb{E}[\mathbf{z}_n]$, $\mathbb{E}[\mathbf{z}_n \mathbf{z}_n^T]$ and state pair expectations $\mathbb{E}[\mathbf{z}_n \mathbf{z}_{n-1}^T]$.
- ▶ For details, see [Bis06, §13.3.2].

Reading

- ▶ Bishop §13.3 up to §13.3.1
- ▶ Murphy **Book 2** [Mur23] §29.6

Problems and quizzes

- ▶ No problems.
- ▶ Quizzes:
 - ▶ Week 7: Linear Dynamical Systems



Christopher M. Bishop.

Pattern Recognition and Machine Learning.

Springer, 2006.



Kevin P. Murphy.

Probabilistic Machine Learning: Advanced Topics.

MIT Press, 2023.