

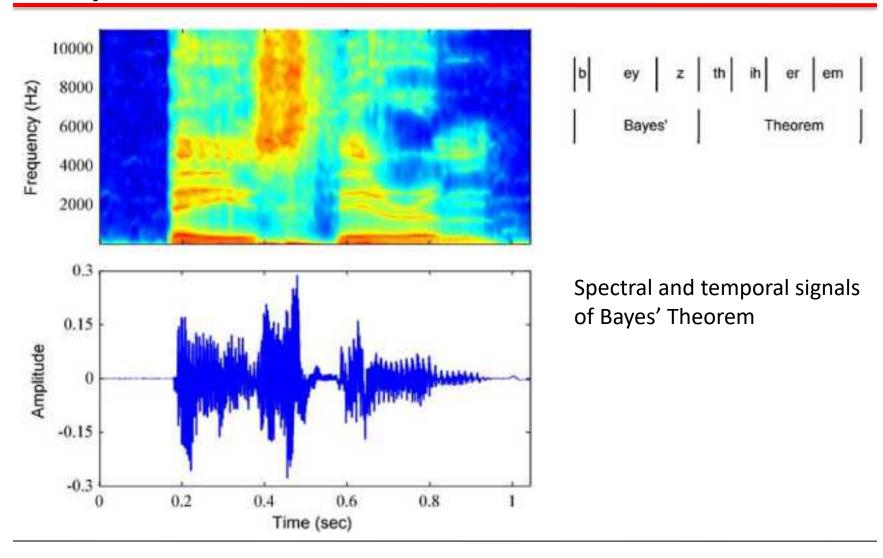
Learning Objectives

- 1. What are hidden Markov models (HMMs)?
- 2、What is the EM scheme for HMMs?
- 3、What are Forward-Backward and Sum-Product Algorithms?
- 4、What are Viterbi and Max-Product Algorithms?
- 5. What are linear dynamic systems?
- 6. What are Kalman and particle filters?
- 7. How to learn linear dynamic system models?
- 8. What are RNN and LSTM?

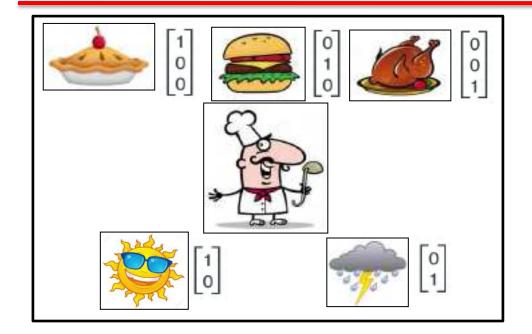
Outlines

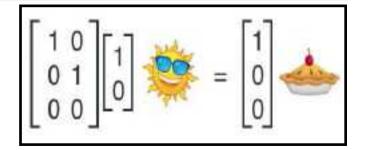
- Hidden Markov Models
- Maximum Likelihood and EM for HMM
- Forward-Backward and Sum-Product Algorithms
- Viterbi and Max-Product Algorithms
- Linear Dynamics Systems
- Kalman Filters and LDS Learning
- RNN and LSTM

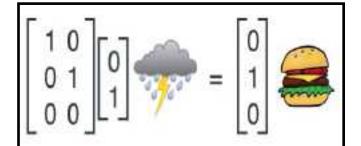
Sequential Data

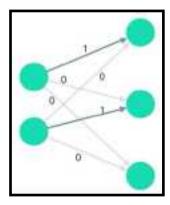


Conditional Data

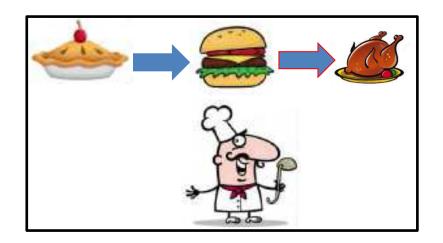


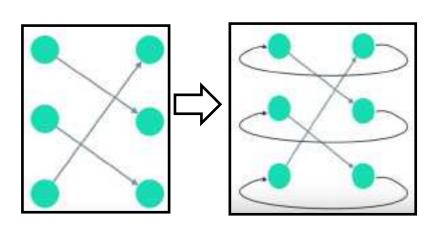


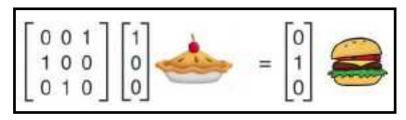


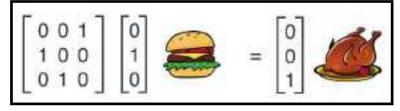


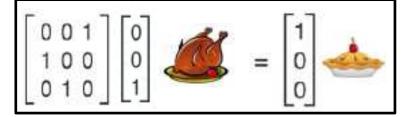
Sequential Data









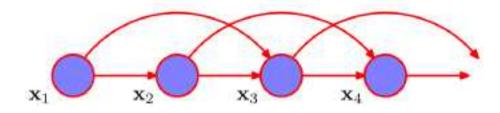


Markov Models

$$p(\mathbf{x}_1, \dots, \mathbf{x}_N) = p(\mathbf{x}_1) \prod_{n=2}^N p(\mathbf{x}_n | \mathbf{x}_{n-1})$$

$$p(\mathbf{x}_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1}) = p(\mathbf{x}_n | \mathbf{x}_{n-1})$$

A second-order Markov chain

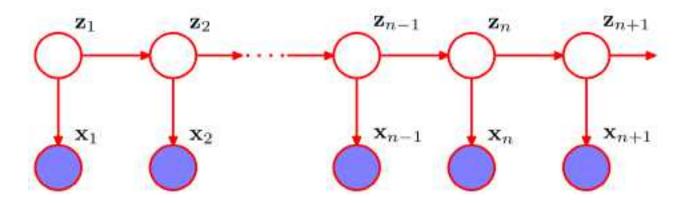


$$p(\mathbf{x}_1,\ldots,\mathbf{x}_N) = p(\mathbf{x}_1)p(\mathbf{x}_2|\mathbf{x}_1)\prod_{n=3}^N p(\mathbf{x}_n|\mathbf{x}_{n-1},\mathbf{x}_{n-2})$$

Hidden Markov Models

Using a Markov chain of latent variables

$$p(\mathbf{x}_1,\ldots,\mathbf{x}_N,\mathbf{z}_1,\ldots,\mathbf{z}_N) = p(\mathbf{z}_1) \left[\prod_{n=2}^N p(\mathbf{z}_n|\mathbf{z}_{n-1}) \right] \prod_{n=1}^N p(\mathbf{x}_n|\mathbf{z}_n)$$



For continuous variables, we can use linear-Gaussian conditional distributions

Latent Markov Model

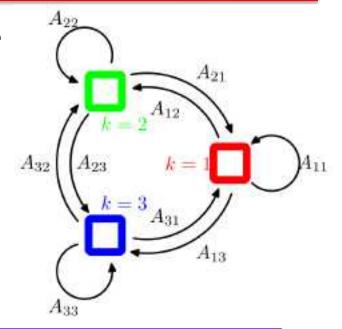
Conditional distribution for latent variable

$$p(\mathbf{z}_n | \mathbf{z}_{n-1, \mathbf{A}}) = \prod_{k=1}^K \prod_{j=1}^K A_{jk}^{z_{n-1, j} z_{nk}}$$

$$p(\mathbf{z}_1|\boldsymbol{\pi}) = \prod_{k=1}^K \pi_k^{z_{1k}}$$

$$\sum_{k} \pi_k = 1$$

A means transition probabilities



As in the case of a standard mixture model, the latent variables are the discrete multinomial variables \mathbf{z}_n using the 1-of-K coding scheme

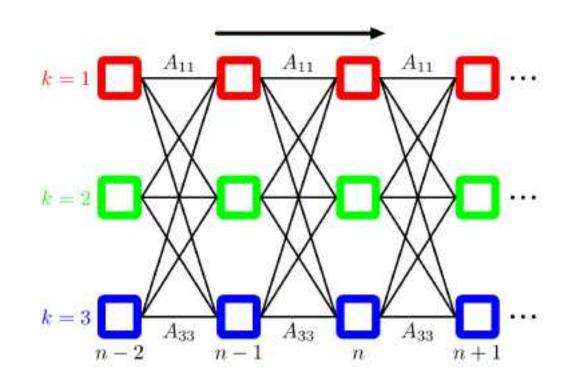
A model whose latent variables have three possible states corresponding to the three boxes. The black lines denote the elements of the transition matrix A_{ik}

Latent State Lattice

Latent states lattice: representing the transitions between latent states

Each column of this diagram corresponds to one of the latent variables \mathbf{z}_n

Each row of this diagram corresponds to one state of the latent variables \mathbf{z}_n



Hidden Markov Models

Getting emission probabilities from latent variable

$$p(\mathbf{x}_n | \mathbf{z}_n, \boldsymbol{\phi}) = \prod_{k=1}^K p(\mathbf{x}_n | \boldsymbol{\phi}_k)^{z_{nk}} \quad p(\mathbf{z}_n | \mathbf{z}_{n-1, \mathbf{A}}) = \prod_{k=1}^K \prod_{j=1}^K A_{jk}^{z_{n-1, j} z_{nk}}$$

$$p(\mathbf{X}, \mathbf{Z} | \boldsymbol{\theta}) = p(\mathbf{z}_1 | \boldsymbol{\pi}) \left[\prod_{n=2}^{N} p(\mathbf{z}_n | \mathbf{z}_{n-1}, \mathbf{A}) \right] \prod_{m=1}^{N} p(\mathbf{x}_m | \mathbf{z}_m, \boldsymbol{\phi})$$

$$\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$$

$$\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_N\}$$

$$\boldsymbol{ heta} = \{ \boldsymbol{\pi}, \mathbf{A}, \boldsymbol{\phi} \}$$

Hidden Markov Model Data Sequence

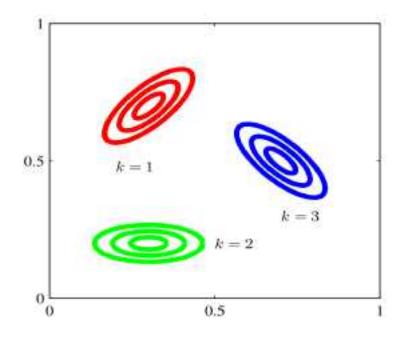
A better understanding of the hidden Markov model

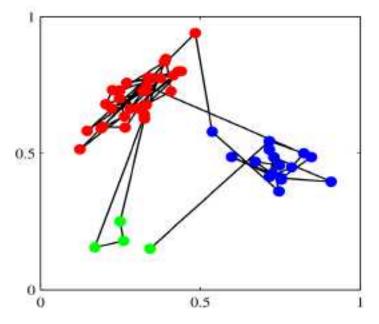
First choose the initial latent variable \mathbf{Z}_1 \mathbf{X}_1 π_k

state *j*

Then, choose the state of the variable \mathbf{z}_2 $p(\mathbf{z}_2|\mathbf{z}_1)$

state k A_{jk} $k = 1, \ldots, K$



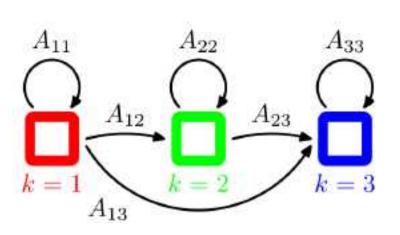


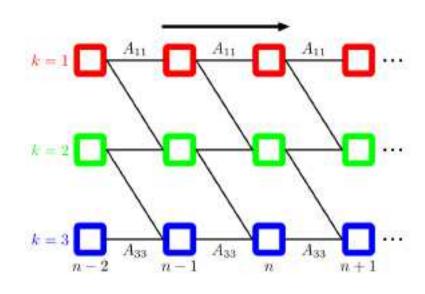
The number of latent variable states is 3; the dimension of emission variable is 2

A sequence of 50 samples

Hidden Markov Model Examples

One Example of variants of the standard HMM model





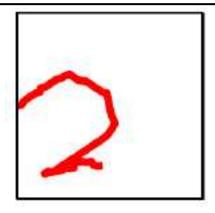
3-state left-to-right hidden Markov model

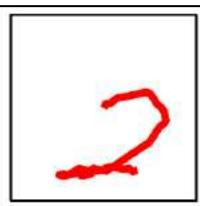
Lattice diagram for a 3-state left- to-right HMM

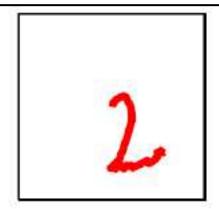
HMM Applications



Synthetic digits using HMMs which are trained by using handwritten digits







Outlines

- Hidden Markov Models
- Maximum Likelihood and EM for HMM
- Forward-Backward and Sum-Product Algorithms
- Viterbi Algorithm
- Linear Dynamics Systems
- Kalman and Particle Filters
- RNN and LSTM

Three Problems for HMMs

□ Evaluation:

Given a HMM model $\theta = \{\pi, \mathbf{A}, \phi\}$, what is likelihood of an observation sequence $\{\mathbf{x}_1, ..., \mathbf{x}_N\}$ generated by that model?

☐ Learning:

What is the most likely HMM model for an observation sequence $\{x_1,...,x_N\}$?

□ Decoding:

Given a HMM model $\theta = \{\pi, \mathbf{A}, \phi\}$, what is the most likely latent sequence $\{\mathbf{z}_1, ..., \mathbf{z}_N\}$ for an observation sequence $\{\mathbf{x}_1, ..., \mathbf{x}_N\}$?

Expectation of Latent Variables

Expectation of latent variables

$$\gamma(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{X}, \boldsymbol{\theta}^{\text{old}}) \qquad \mathbf{X}: \text{ all data samples}$$

$$\xi(\mathbf{z}_{n-1}, \mathbf{z}_n) = p(\mathbf{z}_{n-1}, \mathbf{z}_n | \mathbf{X}, \boldsymbol{\theta}^{\text{old}})$$

$$\gamma(z_{nk}) = \mathbb{E}[z_{nk}] = \sum_{\mathbf{z}} \gamma(\mathbf{z}) z_{nk}$$

$$\xi(z_{n-1,j}, z_{nk}) = \mathbb{E}[z_{n-1,j} z_{nk}] = \sum_{\mathbf{z}} \gamma(\mathbf{z}) z_{n-1,j} z_{nk}$$

Maximum Likelihood of HMMs

Maximum likelihood for the HMM

$$p(\mathbf{X}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

EM algorithm to find an efficient framework

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{\text{old}}) \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

$$\mathbb{I}$$

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) = \sum_{k=1}^{K} \gamma(z_{1k}) \ln \pi_k + \sum_{n=2}^{N} \sum_{j=1}^{K} \sum_{k=1}^{K} \xi(z_{n-1,j}, z_{nk}) \ln A_{jk}$$

$$+ \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(\mathbf{x}_n | \boldsymbol{\phi}_k).$$

EM Learning of HMMs (I)

$$\gamma(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{X}, \boldsymbol{\theta}^{\text{old}})
\xi(\mathbf{z}_{n-1}, \mathbf{z}_n) = p(\mathbf{z}_{n-1}, \mathbf{z}_n | \mathbf{X}, \boldsymbol{\theta}^{\text{old}})
\gamma(z_{nk}) = \mathbb{E}[z_{nk}] = \sum_{\mathbf{z}} \gamma(\mathbf{z}) z_{nk}
\xi(z_{n-1,j}, z_{nk}) = \mathbb{E}[z_{n-1,j} z_{nk}] = \sum_{\mathbf{z}} \gamma(\mathbf{z}) z_{n-1,j} z_{nk}$$

$$p(\mathbf{x}|\mathbf{z}) = \prod_{k=1}^{K} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{z_k}$$

$$\pi_{k} = \frac{\gamma(z_{1k})}{\sum_{j=1}^{K} \gamma(z_{1j})} \qquad \mu_{k} = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_{n}}{\sum_{n=1}^{N} \gamma(z_{nk})}$$

$$A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nk})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nl})} \qquad \Sigma_{k} = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) (\mathbf{x}_{n} - \boldsymbol{\mu}_{k}) (\mathbf{x}_{n} - \boldsymbol{\mu}_{k})^{\mathrm{T}}}{\sum_{n=1}^{N} \gamma(z_{nk})}$$

EM Learning of HMMs (II)

$$\gamma(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{X}, \boldsymbol{\theta}^{\text{old}})
\xi(\mathbf{z}_{n-1}, \mathbf{z}_n) = p(\mathbf{z}_{n-1}, \mathbf{z}_n | \mathbf{X}, \boldsymbol{\theta}^{\text{old}})
\gamma(z_{nk}) = \mathbb{E}[z_{nk}] = \sum_{\mathbf{z}} \gamma(\mathbf{z}) z_{nk}
\xi(z_{n-1,j}, z_{nk}) = \mathbb{E}[z_{n-1,j} z_{nk}] = \sum_{\mathbf{z}} \gamma(\mathbf{z}) z_{n-1,j} z_{nk}$$

$$p(\mathbf{x}|\mathbf{z}) = \prod_{i=1}^{D} \prod_{k=1}^{K} \mu_{ik}^{x_i z_k}$$

$$\pi_{k} = \frac{\gamma(z_{1k})}{\sum_{j=1}^{K} \gamma(z_{1j})}$$

$$A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nk})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nl})}$$

$$\mu_{ik} = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) x_{ni}}{\sum_{n=1}^{N} \gamma(z_{nk})}$$

Outlines

- Hidden Markov Models
- Maximum Likelihood and EM for HMM
- Forward-Backward and Sum-Product Algorithms
- Viterbi and Max-Product Algorithms
- Linear Dynamics Systems
- Kalman Filters and LDS Learning
- RNN and LSTM

Forward Recursion (I)

■ Forward recursion

$$p(\mathbf{X}|\mathbf{z}_{n}) = p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n}|\mathbf{z}_{n})$$

$$p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_{N}|\mathbf{z}_{n})$$

$$p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n-1}|\mathbf{x}_{n}, \mathbf{z}_{n}) = p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n-1}|\mathbf{z}_{n})$$

$$p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n-1}|\mathbf{z}_{n-1}, \mathbf{z}_{n}) = p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n-1}|\mathbf{z}_{n})$$

$$p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n-1}|\mathbf{z}_{n-1}, \mathbf{z}_{n}) = p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n-1}|\mathbf{z}_{n-1})$$

$$p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_{N}|\mathbf{z}_{n}, \mathbf{z}_{n+1}) = p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_{N}|\mathbf{z}_{n+1})$$

$$p(\mathbf{x}_{n+2}, \dots, \mathbf{x}_{N}|\mathbf{z}_{n+1}, \mathbf{x}_{n+1}) = p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n-1}|\mathbf{z}_{n-1})$$

$$p(\mathbf{x}_{n}|\mathbf{z}_{n})p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_{N}|\mathbf{z}_{n})$$

$$p(\mathbf{x}_{n+1}|\mathbf{X}, \mathbf{z}_{N+1}) = p(\mathbf{x}_{1}, \dots, \mathbf{x}_{n-1}|\mathbf{z}_{n-1})$$

$$p(\mathbf{x}_{n}|\mathbf{z}_{n})p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_{N}|\mathbf{z}_{n})$$

$$p(\mathbf{z}_{N+1}|\mathbf{X}, \mathbf{z}_{N+1}) = p(\mathbf{z}_{N+1}|\mathbf{z}_{N})$$

$$p(\mathbf{z}_{N+1}|\mathbf{z}_{N}, \mathbf{X}) = p(\mathbf{z}_{N+1}|\mathbf{z}_{N})$$

$$n - 1$$

$$n$$

$$\alpha(\mathbf{z}_{1}) = p(\mathbf{x}_{1}, \mathbf{z}_{1}) = p(\mathbf{z}_{1})p(\mathbf{x}_{1}|\mathbf{z}_{1}) = \prod_{k=1}^{K} \{\pi_{k}p(\mathbf{x}_{1}|\boldsymbol{\phi}_{k})\}^{z_{1k}}$$

 $\alpha(z_{n-1,1})$

 $\alpha(z_{n,1})$

Forward Recursion (II)

$$\alpha(\mathbf{z}_n) = p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{z}_n)$$

$$= p(\mathbf{x}_1, \dots, \mathbf{x}_n | \mathbf{z}_n) p(\mathbf{z}_n)$$

$$= p(\mathbf{x}_n | \mathbf{z}_n) p(\mathbf{x}_1, \dots, \mathbf{x}_{n-1} | \mathbf{z}_n) p(\mathbf{z}_n)$$

$$= p(\mathbf{x}_n | \mathbf{z}_n) p(\mathbf{x}_1, \dots, \mathbf{x}_{n-1}, \mathbf{z}_n)$$

$$= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} p(\mathbf{x}_1, \dots, \mathbf{x}_{n-1}, \mathbf{z}_{n-1}, \mathbf{z}_n)$$

$$= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} p(\mathbf{x}_1, \dots, \mathbf{x}_{n-1}, \mathbf{z}_n | \mathbf{z}_{n-1}) p(\mathbf{z}_{n-1})$$

$$= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} p(\mathbf{x}_1, \dots, \mathbf{x}_{n-1} | \mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1}) p(\mathbf{z}_{n-1})$$

$$= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} p(\mathbf{x}_1, \dots, \mathbf{x}_{n-1}, \mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1})$$

$$\alpha(\mathbf{z}_n) = p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \alpha(\mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1})$$

Backward Recursion

■ Backward recursion

 \mathbf{z}_{n+1}

$$\beta(\mathbf{z}_n) = p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n)$$

$$= \sum_{\mathbf{z}_{n+1}} p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N, \mathbf{z}_{n+1} | \mathbf{z}_n)$$

$$= \sum_{\mathbf{z}_{n+1}} p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n, \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1} | \mathbf{z}_n)$$

$$= \sum_{\mathbf{z}_{n+1}} p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1} | \mathbf{z}_n)$$

$$= \sum_{\mathbf{z}_{n+1}} p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1} | \mathbf{z}_n)$$

$$= \sum_{\mathbf{z}_{n+1}} p(\mathbf{x}_{n+2}, \dots, \mathbf{x}_N | \mathbf{z}_{n+1}) p(\mathbf{x}_{n+1} | \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1} | \mathbf{z}_n)$$

 $\beta(z_{n+1,1})$

$$\beta(\mathbf{z}_n) = \sum_{\mathbf{z}_{n+1}} \beta(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1} | \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1} | \mathbf{z}_n)$$

Forward-Backward Estimation

The method of evaluating the quantities of $\gamma(z_{nk})$ $\xi(z_{n-1,j},z_{nk})$

$$\gamma(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{X}) = \frac{p(\mathbf{X} | \mathbf{z}_n) p(\mathbf{z}_n)}{p(\mathbf{X})}$$

$$= \frac{p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{z}_n) p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n)}{p(\mathbf{X})} = \frac{\alpha(\mathbf{z}_n) \beta(\mathbf{z}_n)}{p(\mathbf{X})}$$

$$\xi(\mathbf{z}_{n-1}, \mathbf{z}_n) = p(\mathbf{z}_{n-1}, \mathbf{z}_n | \mathbf{X})$$

$$= \frac{\alpha(\mathbf{z}_{n-1})p(\mathbf{x}_n | \mathbf{z}_n)p(\mathbf{z}_n | \mathbf{z}_{n-1})\beta(\mathbf{z}_n)}{p(\mathbf{X})} \quad \alpha(\mathbf{z}_n) \equiv p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{z}_n)$$

$$\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n)$$

$$\alpha(\mathbf{z}_n) \equiv p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{z}_n)$$

 $\beta(\mathbf{z}_n) \equiv p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n)$

$$p(\mathbf{X}) = \sum_{\mathbf{z}_n} \alpha(\mathbf{z}_n) \beta(\mathbf{z}_n) \quad p(\mathbf{X}) = \sum_{\mathbf{z}_N} \alpha(\mathbf{z}_N)$$
 Evaluation

Observation Prediction

$$p(\mathbf{x}_{N+1}|\mathbf{X}) = \sum_{\mathbf{z}_{N+1}} p(\mathbf{x}_{N+1}, \mathbf{z}_{N+1}|\mathbf{X})$$

$$= \sum_{\mathbf{z}_{N+1}} p(\mathbf{x}_{N+1}|\mathbf{z}_{N+1}) p(\mathbf{z}_{N+1}|\mathbf{X})$$

$$= \sum_{\mathbf{z}_{N+1}} p(\mathbf{x}_{N+1}|\mathbf{z}_{N+1}) \sum_{\mathbf{z}_{N}} p(\mathbf{z}_{N+1}, \mathbf{z}_{N}|\mathbf{X})$$

$$= \sum_{\mathbf{z}_{N+1}} p(\mathbf{x}_{N+1}|\mathbf{z}_{N+1}) \sum_{\mathbf{z}_{N}} p(\mathbf{z}_{N+1}|\mathbf{z}_{N}) p(\mathbf{z}_{N}|\mathbf{X})$$

$$= \sum_{\mathbf{z}_{N+1}} p(\mathbf{x}_{N+1}|\mathbf{z}_{N+1}) \sum_{\mathbf{z}_{N}} p(\mathbf{z}_{N+1}|\mathbf{z}_{N}) \frac{p(\mathbf{z}_{N}, \mathbf{X})}{p(\mathbf{X})}$$

$$= \frac{1}{p(\mathbf{X})} \sum_{\mathbf{z}_{N+1}} p(\mathbf{x}_{N+1}|\mathbf{z}_{N+1}) \sum_{\mathbf{z}_{N}} p(\mathbf{z}_{N+1}|\mathbf{z}_{N}) \alpha(\mathbf{z}_{N})$$

Sum-Product v.s. Max-Product

- Sum-Product Algorithm (evaluation)
 - ✓ Compute the joint distribution from the Product
 - ✓ Infer marginal distributions from the Sum

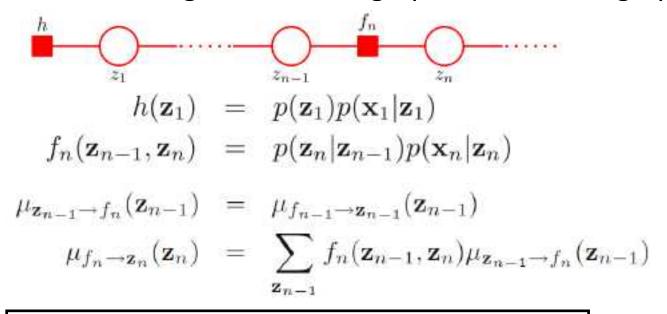
$$p(x_1, x_2) = \sum_{x_3} p(x_1, x_3) p(x_2, x_3)$$

- Max-Product Algorithm (decoding)
 - ✓ Compute the joint distribution from the Product
 - ✓ Perform ML estimation from the Max

$$x_1^* = \max_{x_1} p(x_1, x_3) p(x_2, x_1)$$

Sum-Product for HMMs

Transforming the directed graph into a factor graph

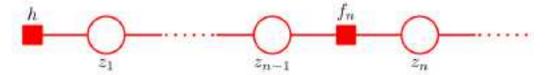


$$\mu_{f_n \to \mathbf{z}_n}(\mathbf{z}_n) = \sum_{\mathbf{z}_{n-1}} f_n(\mathbf{z}_{n-1}, \mathbf{z}_n) \mu_{f_{n-1} \to \mathbf{z}_{n-1}}(\mathbf{z}_{n-1})$$

$$\alpha(\mathbf{z}_n) = \mu_{f_n \to \mathbf{z}_n}(\mathbf{z}_n)$$

Sum-Product for HMMs

☐ Transforming the directed graph into a factor graph



$$\mu_{f_{n+1}\to f_n}(\mathbf{z}_n) = \sum_{\mathbf{z}_{n+1}} f_{n+1}(\mathbf{z}_n, \mathbf{z}_{n+1}) \mu_{f_{n+2}\to f_{n+1}}(\mathbf{z}_{n+1})$$

$$\beta(\mathbf{z}_n) = \mu_{f_{n+1} \to \mathbf{z}_n}(\mathbf{z}_n)$$

$$p(\mathbf{z}_n, \mathbf{X}) = \mu_{f_n \to \mathbf{z}_n}(\mathbf{z}_n) \mu_{f_{n+1} \to \mathbf{z}_n}(\mathbf{z}_n) = \alpha(\mathbf{z}_n) \beta(\mathbf{z}_n)$$

$$\gamma(\mathbf{z}_n) = \frac{p(\mathbf{z}_n, \mathbf{X})}{p(\mathbf{X})} = \frac{\alpha(\mathbf{z}_n)\beta(\mathbf{z}_n)}{p(\mathbf{X})}$$

Scaling of HMMs

■ Scaling factors for computational practice

$$\alpha(\mathbf{z}_n) = p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{z}_n)$$

$$\widehat{\alpha}(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{x}_1, \dots, \mathbf{x}_n) = \frac{\alpha(\mathbf{z}_n)}{p(\mathbf{x}_1, \dots, \mathbf{x}_n)}$$

scaling factors defined by conditional distributions over the observed variables

$$\begin{split} c_n &= p(\mathbf{x}_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1}) \\ c_n \widehat{\alpha}(\mathbf{z}_n) &= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \widehat{\alpha}(\mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1}) \\ \text{similarly define re-scaled variables} & \widehat{\beta}(\mathbf{z}_n) = \frac{p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n)}{p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{x}_1, \dots, \mathbf{x}_n)} \\ c_{n+1} \widehat{\beta}(\mathbf{z}_n) &= \sum \widehat{\beta}(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1} | \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1} | \mathbf{z}_n) \end{split}$$

$$\gamma(\mathbf{z}_n) = \widehat{\alpha}(\mathbf{z}_n)\widehat{\beta}(\mathbf{z}_n)
\xi(\mathbf{z}_{n-1}, \mathbf{z}_n) = c_n^{-1}\widehat{\alpha}(\mathbf{z}_{n-1})p(\mathbf{x}_n|\mathbf{z}_n)p(\mathbf{z}_n|\mathbf{z}_{-1})\widehat{\beta}(\mathbf{z}_n)$$

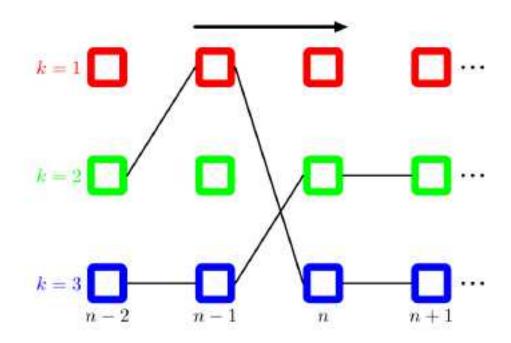
Outlines

- Hidden Markov Models
- Maximum Likelihood and EM for HMM
- Forward-Backward and Sum-Product Algorithms
- Viterbi and Max-Product Algorithms
- Linear Dynamics Systems
- Kalman Filters and LDS Learning
- RNN and LSTM

Latent Sequence Estimation

□ Decoding:

Given a HMM model $\theta = \{\pi, A, \phi\}$, what is the most likely latent sequence $\{\mathbf{z}_1, ..., \mathbf{z}_N\}$ for an observation sequence $\{\mathbf{x}_1, ..., \mathbf{x}_N\}$?



Viterbi Algorithm

$$\omega(\mathbf{z}_n) = \max_{\mathbf{z}_1, \dots, \mathbf{z}_{n-1}} \ln p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{z}_1, \dots, \mathbf{z}_n)$$

$$\omega(\mathbf{z}_{n+1}) = \max_{\mathbf{z}_1, \dots, \mathbf{z}_n} \ln p(\mathbf{x}_1, \dots, \mathbf{x}_n, \mathbf{x}_{n+1}, \mathbf{z}_1, \dots, \mathbf{z}_n, \mathbf{z}_{n+1})$$

$$\omega(\mathbf{z}_{n+1}) = \ln p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) + \max_{\mathbf{z}_n} \left\{ \ln p(\mathbf{z}_{n+1}|\mathbf{z}_n) + \omega(\mathbf{z}_n) \right\}$$

$$\omega(\mathbf{z}_1) = \ln p(\mathbf{z}_1) + \ln p(\mathbf{x}_1|\mathbf{z}_1)$$

Viterbi Algorithm

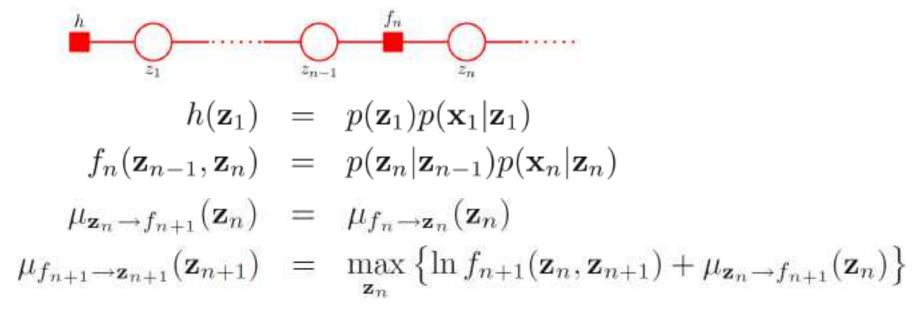
- lacktriangle Note that maximization over \mathbf{z}_n must be performed for each of K possible values of \mathbf{z}_{n+1}
- lacksquare Denote this function by $\psi(k_n)$, where $k \in \{1, \dots, K\}$
- lacksquare Once we find the most probable value of \mathbf{z}_N , we can trackback along the chain

$$k_n^{\max} = \psi(k_{n+1}^{\max})$$

 \blacksquare Reduce the computational cost from $O(K^N)$ to O(KN)

Max-Product for HMMs

Transforming the directed graph into a factor graph



 \square As such, the $f \rightarrow \mathbf{z}$ message will recursively be

$$\omega(\mathbf{z}_{n+1}) = \ln p(\mathbf{x}_{n+1}|\mathbf{z}_{n+1}) + \max_{\mathbf{z}_n} \left\{ \ln p(\mathbf{z}_{n+1}|\mathbf{z}_n) + \omega(\mathbf{z}_n) \right\}$$
$$\omega(\mathbf{z}_n) \equiv \mu_{f_n \to \mathbf{z}_n}(\mathbf{z}_n)$$

Discriminative HMMs

☐ Using discriminative rather than Maximum Likelihood techniques

optimize the cross-entropy of R observation sequences, \mathbf{X}_r , and labels, m_r

$$\sum_{r=1}^{R} \ln p(m_r | \mathbf{X}_r) \iff \sum_{r=1}^{R} \ln \left\{ \frac{p(\mathbf{X}_r | \boldsymbol{\theta}_r) p(m_r)}{\sum_{l=1}^{M} p(\mathbf{X}_r | \boldsymbol{\theta}_l) p(l_r)} \right\} \underset{m=1, \dots, M}{r=1, \dots, R}$$

for each class there is a HMM, $\, heta_{\!m}$

- Weakness of the first-order HMM:
 - ✓ distribution of times for which the system remains in a given state
 - ✓ poor at capturing long-range correlations between the observed variables

Example

☐ Given an HMM and an observation sequence, how to perform evaluation and decoding

Transition A

Emission B

Hidden States Z

Observations X

 $\begin{bmatrix} 0.6 & 0.3 \\ 0.4 & 0.7 \end{bmatrix}$

 $\begin{bmatrix} 0.8 & 0.1 \\ 0.2 & 0.9 \end{bmatrix}$

{bull, bear}

{up, down}

If Z is stationary, then $\pi = [3/7, 4/7]$. We also can assume $\pi = [1/2, 1/2]$

An observation sequence: {up, up, down}

Evaluation (Sum-Product)

$$\alpha(z_1) = p(z_1, x_1) = p(x_1|z_1)p(z_1)$$

$$x_1$$
=up, z_1 =bull or bear

$$= \begin{bmatrix} 0.8 \\ 0.1 \end{bmatrix} \cdot \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 0.8 \times 0.5 \\ 0.1 \times 0.5 \end{bmatrix} = \begin{bmatrix} 0.4 \\ 0.05 \end{bmatrix}$$

$$\alpha(z_2) = p(z_2, x_1, x_2) = p(x_2|z_2) \sum_{z} p(z_2|z_1) \alpha(z_1)$$

$$x_2$$
=up, z_2 =bull or bear

$$\begin{bmatrix} 0.8 \\ 0.1 \end{bmatrix} \cdot \begin{bmatrix} 0.6 \times 0.4 + 0.3 \times 0.05 \\ 0.4 \times 0.4 + 0.7 \times 0.05 \end{bmatrix} = \begin{bmatrix} 0.204 \\ 0.0195 \end{bmatrix}$$

$$\alpha(z_3) = p(z_3, x_1, x_2, x_3) = p(x_3|z_3) \sum_{z_2} p(z_3|z_2) \alpha(z_2)$$

$$= \begin{bmatrix} 0.2 \\ 0.9 \end{bmatrix} \cdot \begin{bmatrix} 0.6 \times 0.204 + 0.3 \times 0.0195 \\ 0.4 \times 0.204 + 0.7 \times 0.0195 \end{bmatrix} = \begin{bmatrix} 0.02565 \\ 0.085725 \end{bmatrix}$$

$$p(x_1, x_2, x_3) = \sum_{z_3} \alpha(z_3) = 0.111375$$

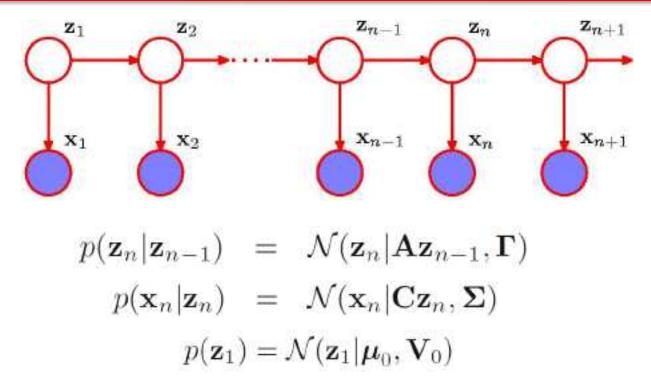
Decoding (Max-Product)

$$\begin{split} \delta(z_1) &= p(z_1, x_1) = p(x_1|z_1)p(z_1) \\ \hline x_1 &= \mathsf{up}, z_1 = \mathsf{bull} \text{ or bear} \\ &= \begin{bmatrix} 0.8 \\ 0.1 \end{bmatrix} \cdot \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 0.8 \times 0.5 \\ 0.1 \times 0.5 \end{bmatrix} = \begin{bmatrix} 0.4 \\ 0.05 \end{bmatrix} \\ \delta(z_2) &= p(z_2, x_1, x_2) = p(x_2|z_2) \max_{z_1} p(z_2|z_1)\delta(z_1) \\ \hline x_2 &= \mathsf{up}, z_2 = \mathsf{bull} \text{ or bear} \\ &= \begin{bmatrix} 0.8 \\ 0.1 \end{bmatrix} \cdot \begin{bmatrix} 0.4 \times 0.6 \\ 0.4 \times 0.4 \end{bmatrix} = \begin{bmatrix} 0.192 \\ 0.016 \end{bmatrix} \\ \phi(z_2) &= \arg\max_{z_1} p(z_2|z_1)\delta(z_1) = \begin{bmatrix} bull \to bull \\ bull \to bear \end{bmatrix} \\ \delta(z_3) &= p(z_3, x_1, x_2, x_3) = p(x_3|z_3) \max_{z_2} p(z_3|z_2)\delta(z_2) \\ \hline x_3 &= \mathsf{down}, z_2 = \mathsf{bull} \text{ or bear} \\ &= \begin{bmatrix} 0.2 \\ 0.9 \end{bmatrix} \cdot \begin{bmatrix} 0.192 \times 0.6 \\ 0.192 \times 0.4 \end{bmatrix} = \begin{bmatrix} 0.02304 \\ 0.06912 \end{bmatrix} \\ \phi(z_3) &= \arg\max_{z_2} p(z_3|z_2)\delta(z_2) = \begin{bmatrix} bull \to bull \\ bull \to bear \end{bmatrix} \\ \phi(z_3) &= \arg\max_{z_2} p(z_3|z_2)\delta(z_2) = \begin{bmatrix} bull \to bull \\ bull \to bear \end{bmatrix} \\ \hline Optimal solution: bull \to bear \end{bmatrix}$$

Outlines

- Hidden Markov Models
- Maximum Likelihood and EM for HMM
- Forward-Backward and Sum-Product Algorithms
- Viterbi and Max-Product Algorithms
- Linear Dynamics Systems
- Kalman and Particle Filters
- RNN and LSTM

Stochastic Linear Dynamical Systems



$$\mathbf{z}_n = \mathbf{A}\mathbf{z}_{n-1} + \mathbf{w}_n \quad \mathbf{w} \sim \mathcal{N}(\mathbf{w}|\mathbf{0}, \mathbf{\Gamma})$$
 $\mathbf{x}_n = \mathbf{C}\mathbf{z}_n + \mathbf{v}_n \quad \mathbf{v} \sim \mathcal{N}(\mathbf{v}|\mathbf{0}, \mathbf{\Sigma})$
 $\mathbf{z}_1 = \boldsymbol{\mu}_0 + \mathbf{u} \quad \mathbf{u} \sim \mathcal{N}(\mathbf{u}|\mathbf{0}, \mathbf{V}_0)$

Inference Problem

☐ Finding the marginal distributions for the latent variables conditional on the observation sequence.

$$\widehat{\alpha}(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{x}_1, \dots, \mathbf{x}_n) = \mathcal{N}(\mathbf{z}_n | \boldsymbol{\mu}_n, \mathbf{V}_n)$$

$$\widehat{\beta}(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{x}_{n+1}, \dots, \mathbf{x}_N)$$

$$\gamma(\mathbf{z}_n) = \widehat{\alpha}(\mathbf{z}_n) \widehat{\beta}(\mathbf{z}_n) = \mathcal{N}(\mathbf{z}_n | \widehat{\boldsymbol{\mu}}_n, \widehat{\mathbf{V}}_n)$$

$$\xi(\mathbf{z}_{n-1}, \mathbf{z}_n) = (c_n)^{-1} \widehat{\alpha}(\mathbf{z}_{n-1}) p(\mathbf{x}_n | \mathbf{z}_n) p(\mathbf{z}_n | \mathbf{z}_{-1}) \widehat{\beta}(\mathbf{z}_n)$$

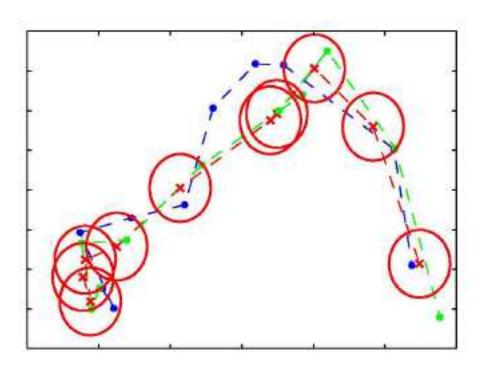
$$c_n = p(\mathbf{x}_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1})$$

Outlines

- Hidden Markov Models
- Maximum Likelihood and EM for HMM
- Forward-Backward and Sum-Product Algorithms
- Viterbi and Max-Product Algorithms
- Linear Dynamics Systems
- Kalman Filters and LDS Learning
- RNN and LSTM

Application: Tracking an Moving Object

☐ One of the most important application of the Kalman filter.



An illustration of a linear dynamical system used to track a moving object.

Blue: \mathbf{Z}_n

Green: X_n

Red: $\mathbf{z}_n | \mathbf{x}_1, \dots, \mathbf{x}_n$

Mean and Variance of Kalman Filter

■ Kalman filter equations

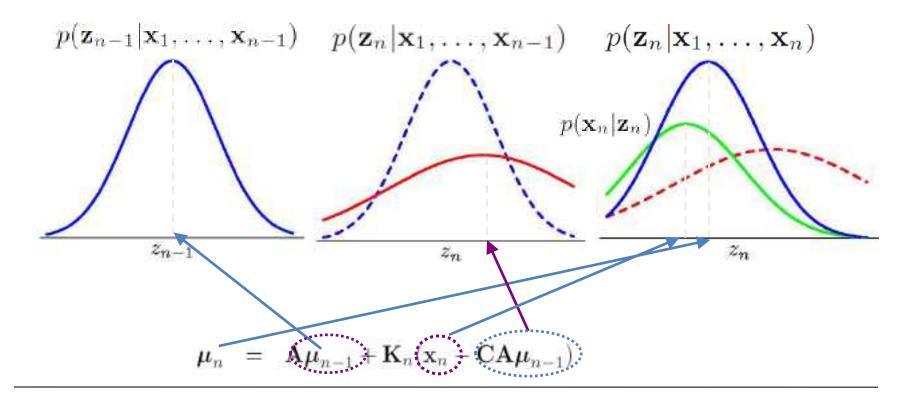
$$\mu_n = \mathbf{A}\mu_{n-1} + \mathbf{K}_n(\mathbf{x}_n - \mathbf{C}\mathbf{A}\mu_{n-1})$$
 $\mathbf{V}_n = (\mathbf{I} - \mathbf{K}_n\mathbf{C})\mathbf{P}_{n-1}$
 $c_n = \mathcal{N}(\mathbf{x}_n|\mathbf{C}\mathbf{A}\mu_{n-1}, \mathbf{C}\mathbf{P}_{n-1}\mathbf{C}^T + \mathbf{\Sigma})$

$$\mathbf{P}_{n-1} = \mathbf{A}\mathbf{V}_{n-1}\mathbf{A}^{\mathrm{T}} + \mathbf{\Gamma}$$
 $\mathbf{K}_n = \mathbf{P}_{n-1}\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\mathbf{P}_{n-1}\mathbf{C}^{\mathrm{T}} + \mathbf{\Sigma}\right)^{-1}$

$$\widehat{\alpha}(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{x}_1, \dots, \mathbf{x}_n) = \mathcal{N}(\mathbf{z}_n | \boldsymbol{\mu}_n, \mathbf{V}_n)$$
$$c_n = p(\mathbf{x}_n | \mathbf{x}_1, \dots, \mathbf{x}_{n-1})$$

Interpretation of Kalman Filters

- ☐ Kalman filter as a process of
 - ✓ Making successive predictions
 - ✓ Correcting the predictions using new observations.



Mean and Variance of Full Kalman Filter

☐ Full Kalman filter equations

$$\widehat{\boldsymbol{\mu}}_{n} = \boldsymbol{\mu}_{n} + \boldsymbol{J}_{n} \left(\widehat{\boldsymbol{\mu}}_{n+1} - \boldsymbol{A} \boldsymbol{\mu}_{N} \right)
\widehat{\boldsymbol{V}}_{n} = \boldsymbol{V}_{n} + \boldsymbol{J}_{n} \left(\widehat{\boldsymbol{V}}_{n+1} - \boldsymbol{P}_{n} \right) \boldsymbol{J}_{n}^{T}$$

$$\mathbf{J}_n = \mathbf{V}_n \mathbf{A}^{\mathrm{T}} \left(\mathbf{P}_n \right)^{-1}$$

 $\mathbf{A} \mathbf{V}_n = \mathbf{P}_n \mathbf{J}_n^{\mathrm{T}}$

$$\gamma(\mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{x}_1, \dots, \mathbf{x}_N) = \mathcal{N}(\mathbf{z}_n | \widehat{\boldsymbol{\mu}}_n, \widehat{\mathbf{V}}_n)$$

Covariance of Sequential States

■ Joint posterior of sequential states is Gaussian

$$\xi(\mathbf{z}_{n-1}, \mathbf{z}_n) = p(\mathbf{z}_{n-1}, \mathbf{z}_n | \mathbf{X}) = \frac{p(\mathbf{X} | \mathbf{z}_{n-1}, \mathbf{z}_n) p(\mathbf{z}_{n-1}, \mathbf{z}_n)}{p(\mathbf{X})}$$

$$= \frac{\mathcal{N}(\mathbf{z}_{n-1} | \boldsymbol{\mu}_{n-1}, \mathbf{V}_{n-1}) \mathcal{N}(\mathbf{z}_n | \mathbf{A} \mathbf{z}_{n-1}, \boldsymbol{\Gamma}) \mathcal{N}(\mathbf{x}_n | \mathbf{C} \mathbf{z}_n, \boldsymbol{\Sigma}) \mathcal{N}(\mathbf{z}_n | \hat{\boldsymbol{\mu}}_n, \hat{\mathbf{V}}_n)}{c_n \hat{\alpha}(\mathbf{z}_n)}$$

$$cov[\mathbf{z}_n, \mathbf{z}_{n-1}] = \mathbf{J}_{n-1}\widehat{\mathbf{V}}_n$$

Learning Problem

- Determining the parameters $\vartheta = \{A, \Gamma, C, \Sigma, \mu_0, V_0\}$ using the *EM algorithm*.
- \square The complete data $\{X, Z\}$ log likelihood function

$$\ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) = \ln p(\mathbf{z}_1|\boldsymbol{\mu}_0, \mathbf{V}_0) + \sum_{n=2}^{N} \ln p(\mathbf{z}_n|\mathbf{z}_{n-1}, \mathbf{A}, \boldsymbol{\Gamma})$$
$$+ \sum_{n=1}^{N} \ln p(\mathbf{x}_n|\mathbf{z}_n, \mathbf{C}, \boldsymbol{\Sigma})$$

Expectation of Latent Variables

■ The expectation of latent variables

$$\mathbb{E} \left[\mathbf{z}_{n} \right] = \widehat{\boldsymbol{\mu}}_{n}
\mathbb{E} \left[\mathbf{z}_{n} \mathbf{z}_{n-1}^{\mathrm{T}} \right] = \mathbf{J}_{n-1} \widehat{\mathbf{V}}_{n} + \widehat{\boldsymbol{\mu}}_{n} \widehat{\boldsymbol{\mu}}_{n-1}^{\mathrm{T}}
\mathbb{E} \left[\mathbf{z}_{n} \mathbf{z}_{n}^{\mathrm{T}} \right] = \widehat{\mathbf{V}}_{n} + \widehat{\boldsymbol{\mu}}_{n} \widehat{\boldsymbol{\mu}}_{n}^{\mathrm{T}}$$

$$cov[\mathbf{z}_n, \mathbf{z}_{n-1}] = \mathbf{J}_{n-1}\widehat{\mathbf{V}}_n$$

Expectation of Log Likelihood Function

■ The expectation of the log likelihood function with respect to $p(\mathbf{Z} \mid \mathbf{X}, \theta^{\text{old}})$

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

$$= -\frac{1}{2} \ln |\mathbf{V}_0| - \mathbb{E}_{\mathbf{Z}|\boldsymbol{\theta}^{\text{old}}} \left[\frac{1}{2} (\mathbf{z}_1 - \boldsymbol{\mu}_0)^{\text{T}} \mathbf{V}_0^{-1} (\mathbf{z}_1 - \boldsymbol{\mu}_0) \right] + \text{const}$$

$$= -\frac{N-1}{2} \ln |\mathbf{\Gamma}| - \mathbb{E}_{\mathbf{Z}|\boldsymbol{\theta}^{\text{old}}} \left[\frac{1}{2} \sum_{n=2}^{N} (\mathbf{z}_n - \mathbf{A} \mathbf{z}_{n-1})^{\text{T}} \mathbf{\Gamma}^{-1} (\mathbf{z}_n - \mathbf{A} \mathbf{z}_{n-1}) \right] + \text{const}$$

$$= -\frac{N}{2} \ln |\mathbf{\Sigma}| - \mathbb{E}_{\mathbf{Z}|\boldsymbol{\theta}^{\text{old}}} \left[\frac{1}{2} \sum_{n=1}^{N} (\mathbf{x}_n - \mathbf{C} \mathbf{z}_n)^{\text{T}} \mathbf{\Sigma}^{-1} (\mathbf{x}_n - \mathbf{C} \mathbf{z}_n) \right] + \text{const}.$$

Maximization of LSD Parameters

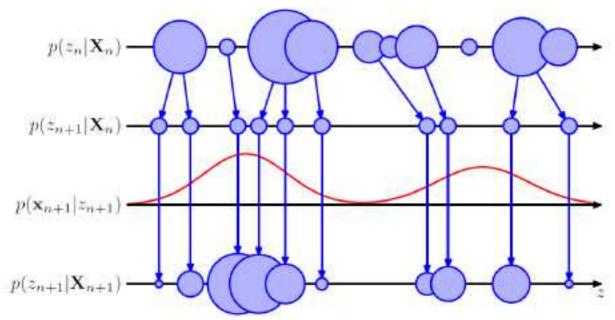
$$\mu_{0}^{\text{new}} = \mathbb{E}[\mathbf{z}_{1}] \\
\mathbf{V}_{0}^{\text{new}} = \mathbb{E}[\mathbf{z}_{1}\mathbf{z}_{1}^{\text{T}}] - \mathbb{E}[\mathbf{z}_{1}]\mathbb{E}[\mathbf{z}_{1}^{\text{T}}] \\
\mathbf{A}^{\text{new}} = \left(\sum_{n=2}^{N} \mathbb{E}\left[\mathbf{z}_{n}\mathbf{z}_{n-1}^{\text{T}}\right]\right) \left(\sum_{n=2}^{N} \mathbb{E}\left[\mathbf{z}_{n-1}\mathbf{z}_{n-1}^{\text{T}}\right]\right)^{-1} \\
\mathbf{\Gamma}^{\text{new}} = \frac{1}{N-1} \sum_{n=2}^{N} \left\{ \mathbb{E}\left[\mathbf{z}_{n}\mathbf{z}_{n}^{\text{T}}\right] - \mathbf{A}^{\text{new}}\mathbb{E}\left[\mathbf{z}_{n-1}\mathbf{z}_{n}^{\text{T}}\right] - \mathbb{E}\left[\mathbf{z}_{n}\mathbf{z}_{n-1}^{\text{T}}\right] \mathbf{A}^{\text{new}} + \mathbf{A}^{\text{new}}\mathbb{E}\left[\mathbf{z}_{n-1}\mathbf{z}_{n-1}^{\text{T}}\right] (\mathbf{A}^{\text{new}})^{\text{T}} \right\} \\
\mathbf{C}^{\text{new}} = \left(\sum_{n=1}^{N} \mathbf{x}_{n}\mathbb{E}\left[\mathbf{z}_{n}^{\text{T}}\right]\right) \left(\sum_{n=1}^{N} \mathbb{E}\left[\mathbf{z}_{n}\mathbf{z}_{n}^{\text{T}}\right]\right)^{-1} \\
\mathbf{\Sigma}^{\text{new}} = \frac{1}{N} \sum_{n=1}^{N} \left\{\mathbf{x}_{n}\mathbf{x}_{n}^{\text{T}} - \mathbf{C}^{\text{new}}\mathbb{E}\left[\mathbf{z}_{n}\right]\mathbf{x}_{n}^{\text{T}} - \mathbf{x}_{n}\mathbb{E}\left[\mathbf{z}_{n}^{\text{T}}\right] \mathbf{C}^{\text{new}} + \mathbf{C}^{\text{new}}\mathbb{E}\left[\mathbf{z}_{n}\mathbf{z}_{n}^{\text{T}}\right] \mathbf{C}^{\text{new}} \right\}$$

Extensions of LDS

- ☐ Problem: Beyond the linear-Gaussian assumption.
 - ✓ Considerable interest in extending the basic linear dynamical system in order to increase its capabilities.
 - ✓ Gaussian $p(\mathbf{z}_n | \mathbf{x}_n)$ A significant limitation.
- Some extensions
 - ✓ Gaussian mixture $p(\mathbf{z}_n)$.
 - ✓ The extended Kalman filter.
 - ✓ The switching state space model
 - ✓ The switching hidden Markov model

Particle filters

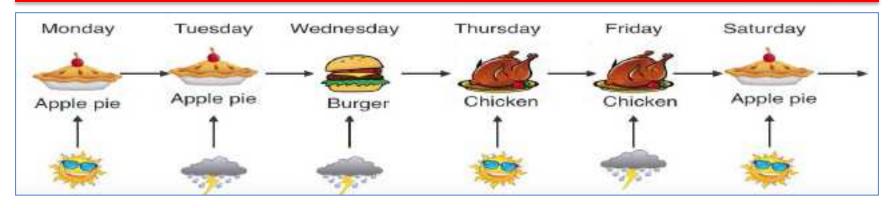
- \square Non-Gaussian emission density $p(\mathbf{x}_n | \mathbf{z}_n)$
 - ✓ non-Gaussian $p(z_n | x_1, ..., x_n)$
 - ✓ mathematically intractable integral
- Sampling-importance-resampling

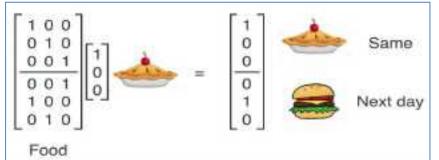


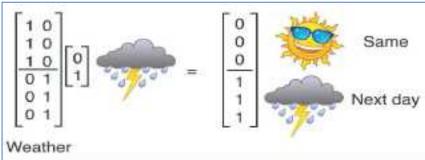
Outlines

- Hidden Markov Models
- Maximum Likelihood and EM for HMM
- Forward-Backward and Sum-Product Algorithms
- Viterbi and Max-Product Algorithms
- Linear Dynamics Systems
- Kalman Filters and LDS Learning
- RNN and LSTM

Complicated Sequential Data



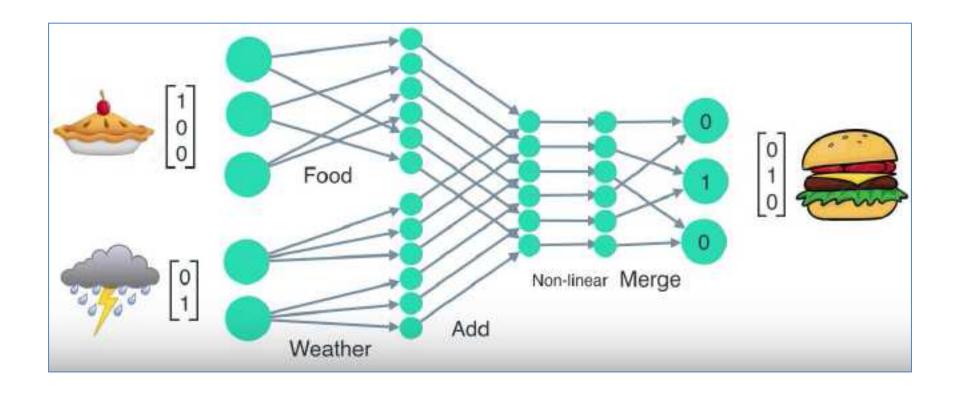




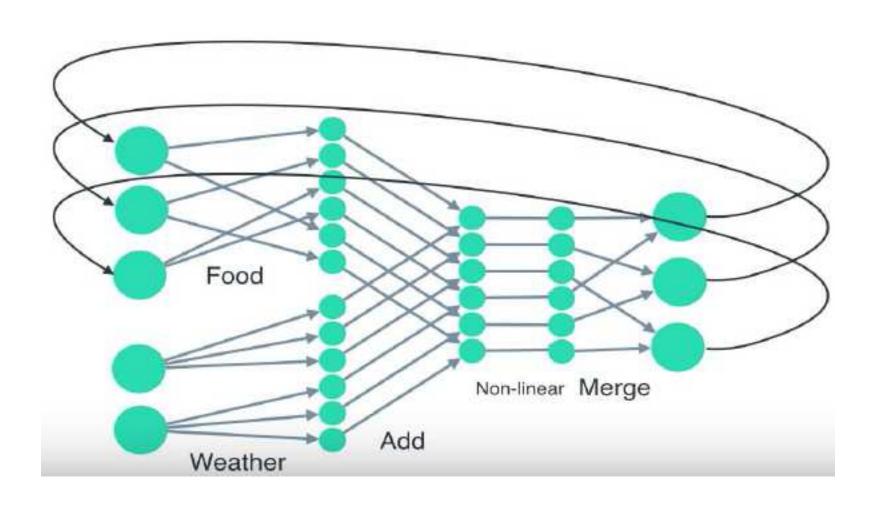


$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$
Same
$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$
Same
$$= \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 2 \\ 1 \end{bmatrix}$$
Next day

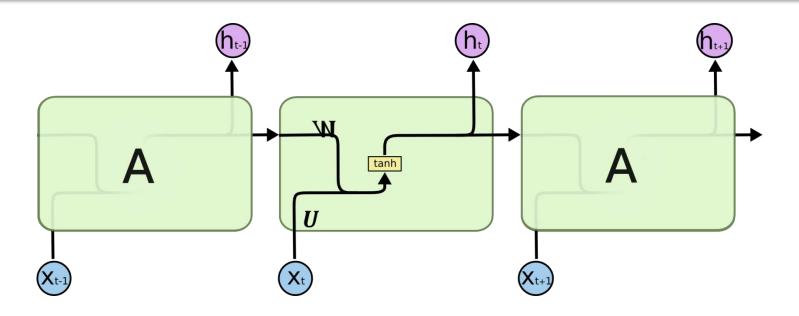
Neural Network



Recurrent Neural Network



Standard RNN Modules

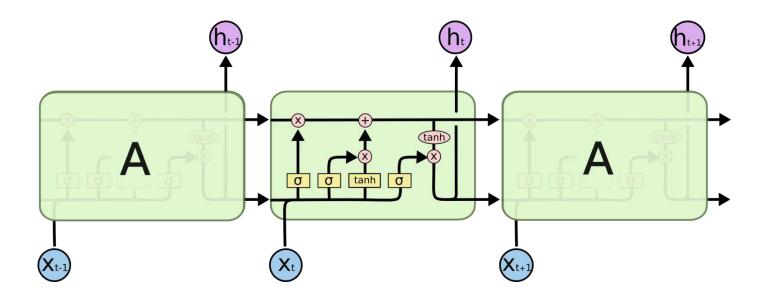


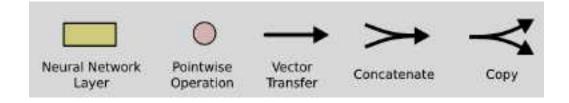
$$h_t = f(Ux_t + Wh_{t-1} + b)$$

 \mathbf{h}_t : output

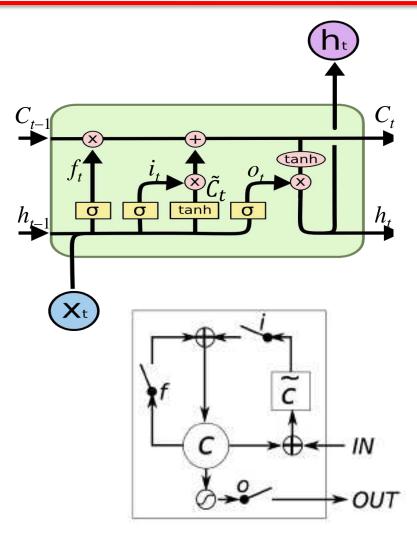
 x_t : input

Long Short Term Memory





Long Short Term Memory



$$f_t = \sigma \left(W_f \cdot [h_{t-1}, x_t] + b_f \right)$$

$$i_t = \sigma \left(W_i \cdot [h_{t-1}, x_t] + b_i \right)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$o_t = \sigma \left(W_o \left[h_{t-1}, x_t \right] + b_o \right)$$

$$h_t = o_t * \tanh(C_t)$$

 C_t : cell state

 \tilde{C}_t : cell state prediction

 f_t : forget gate

 i_t : input gate

 o_t : output gate

 h_t : output

 x_t : input

Summary

- Hidden Markov Models
- Maximum Likelihood and EM for HMM
- Forward-Backward and Sum-Product Algorithms
- Viterbi and Max-Product Algorithms
- Linear Dynamics Systems
- Kalman Filters and LDS Learning
- RNN and LSTM