IFT 4030/7030, Machine Learning for Signal Processing

Week 9: Time Series Modeling

Cem Subakan

Admin

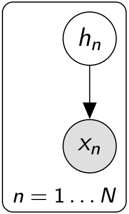
- Homework 2 is out. You can do it all on the notebook.
  - Devoir 2 est publié. Vous pouvez tout faire sur le notebook.
- If you have doubts on your projects contact me. Thanks for those of you who already did. If you do a good project it will help you in your job search / grad school search.
  - Si vous avez des doutes sur vos projets, c'est très important que vous me contactez. Merci pour ceux qui a déjà fait. Si vous faites un bon projet ça va vous aider à trouver un job, ou un grad student position...

Admin

- Homework 2 is out. You can do it all on the notebook.
  - Devoir 2 est publié. Vous pouvez tout faire sur le notebook.
- If you have doubts on your projects contact me. Thanks for those of you who already did. If you do a good project it will help you in your job search / grad school search.
  - Si vous avez des doutes sur vos projets, c'est très important que vous me contactez. Merci pour ceux qui a déjà fait. Si vous faites un bon projet ça va vous aider à trouver un job, ou un grad student position...
- I will share a sign-up sheet soon for project presentations.
  - Je partager un sign-up sheet bientot pour les presentations.
- This week: / Cette semaine: Time series!

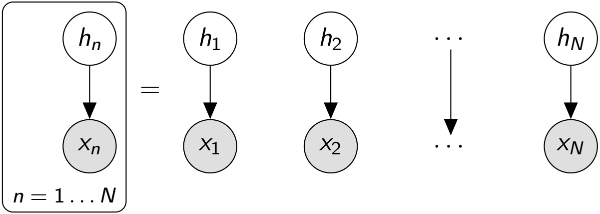
Our view of data so far (well, mostly)

- IID: (not the institute) Independent and Identically Distributed
- Remember last week? / Souvenez-vous de ça de la semaine dernière?



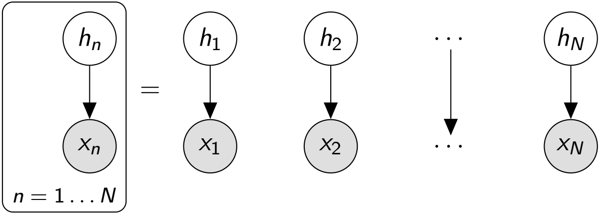
Our view of data so far (well, mostly)

- IID: (not the institute) Independent and Identically Distributed
- Remember last week? / Souvenez-vous de ça de la semaine dernière?

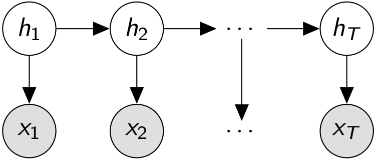


Our view of data so far (well, mostly)

- IID: (not the institute) Independent and Identically Distributed
- Remember last week? / Souvenez-vous de ça de la semaine dernière?

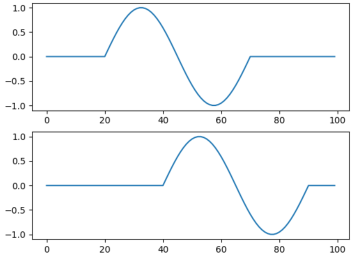


- Well, this week we will have this instead:
  - Bon, cette semain on aura ça:



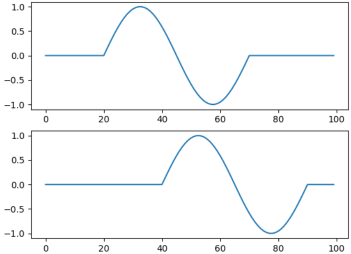
Motivation

- Are these signals similar? / Ces deux signals, sont-ils similaires?



Motivation

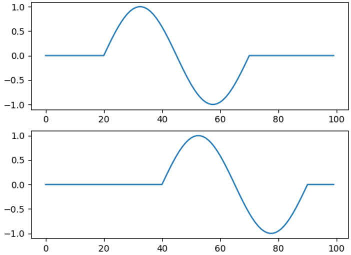
- Are these signals similar? / Ces deux signals, sont-ils similaires?



- If we calculate a distance in the 100d space they are distant.
  - Si on calcule une distance dans une espace de 100d, ils sont distants.

Motivation

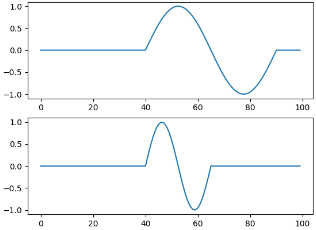
- Are these signals similar? / Ces deux signals, sont-ils similaires?



- If we calculate a distance in the 100d space they are distant.
  - Si on calcule une distance dans une espace de 100d, ils sont distants.
- What can we do?
  - Que peut-on faire?

How about these?

- Are these similar? / Sont-ils similaires?



- How about these? / Et ces deux?

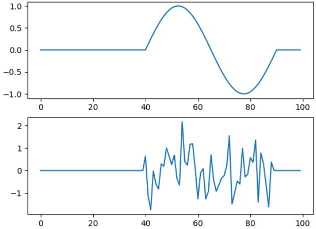


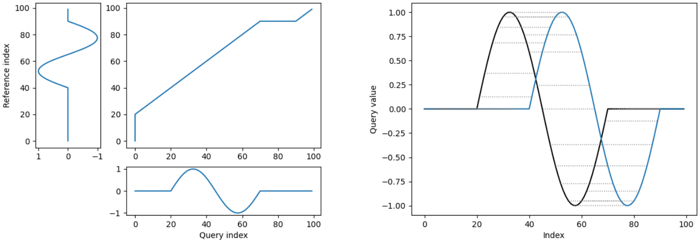
Table of Contents

Dynamic Time Warping

Hidden Markov Models
- Inference
- Learning
- Decoding
- HMM Applications
- HMM Variants

Dynamic Time Warping

- We can to find a 'warping function' between two signals to match them.
  - On peut trouver une fonction de 'warping' entre les deux signaux pour les matcher.



Dynamic Time Warping

- The problem formulation / La formulation du problème

$$\text{DTW}(x, y) = \min_{\pi} \sqrt{\sum_{(i,j)\in\pi} d(x_i, y_i)}$$

  - where, π is a path defined over the indices. / où π est un path défini sur les indices.
- More formally/Plus formellement $\pi = [(i_0, j_0), (i_1, j_1), \ldots, (i_K, j_K)]$.
- Constraints: / Contraints:

$$\begin{aligned} \pi_0 &= (0, 0) \\ \pi_K &= (n-1, m-1) \\ i_{k-1} \leq i_k &\leq \ i_{k-1} + 1 \\ j_{k-1} \leq j_k &\leq \ j_{k-1} + 1 \end{aligned}$$

The dynamic programming solution

- The recursion / La recursion

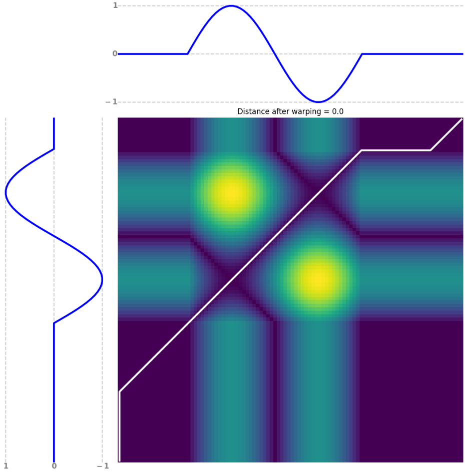
$$C_{i,j} = d(x_i, y_j) + \min(C_{i-1,j}, C_{i,j-1}, C_{i-1,j-1})$$

The dynamic programming solution

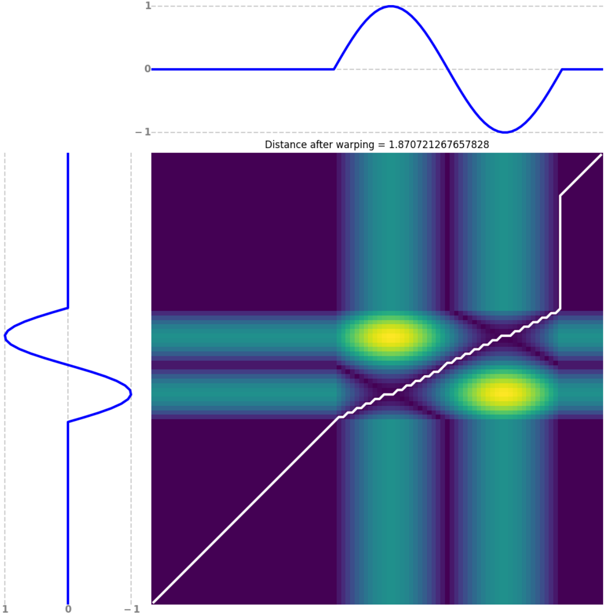
- The recursion / La recursion

$$C_{i,j} = d(x_i, y_j) + \min(C_{i-1,j}, C_{i,j-1}, C_{i-1,j-1})$$

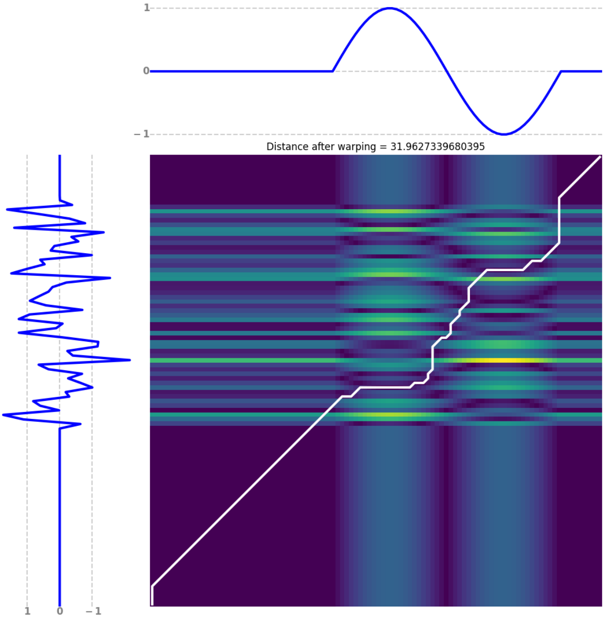
- Optimal path after backtracking / La trace optimale après backtracking



DTW examples

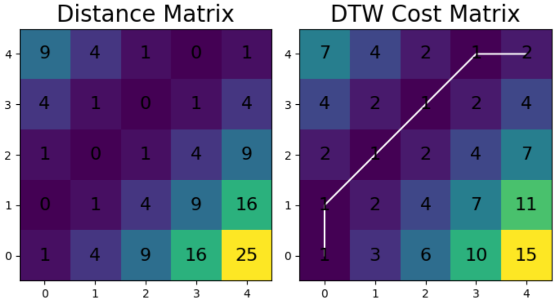


DTW examples



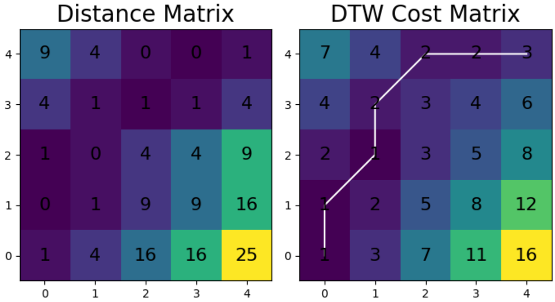
DTW on two arrays

- $x = [0, 1, 2, 3, 4]$, $y = [1, 2, 3, 4, 5]$.



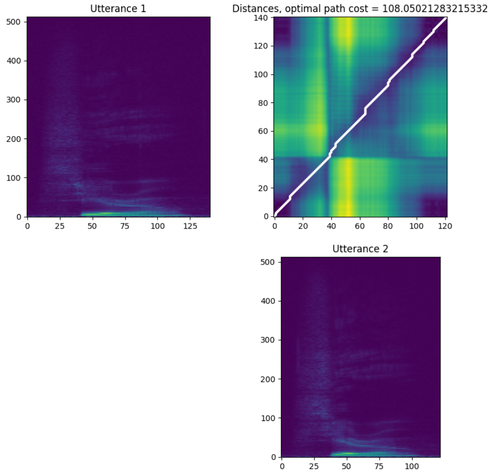
DTW on two arrays

- $x = [0, 1, 2, 3, 4]$, $y = [1, 2, 4, 4, 5]$.



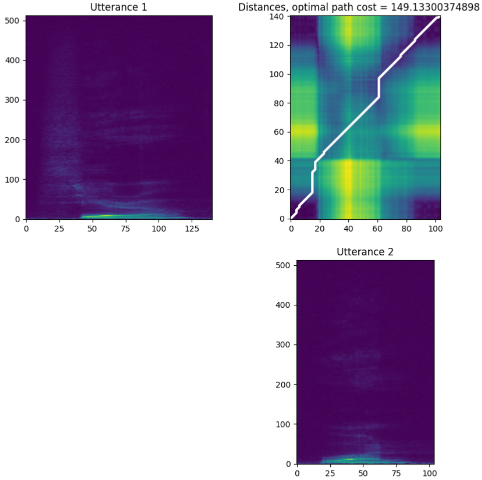
DTW on Audio

- DTW on same person saying 'zero' two different times.
  - DTW sur la meme personne qui dit 'zero' deux differentes fois.



DTW on Audio

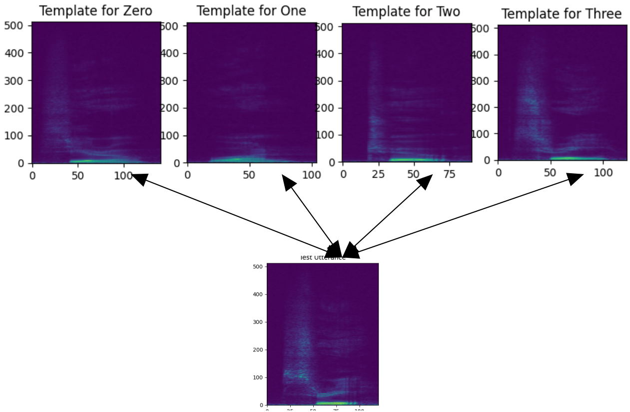
- DTW on same person saying 'zero' and 'one'
  - DTW sur la meme personne qui dit 'zero' et 'one'.



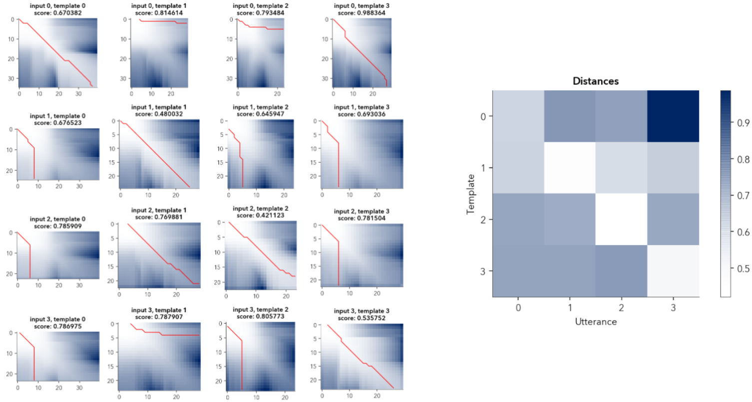
  - Do you notice something? / Vous remarquez quelque chose qui peut etre utile?

Using DTW for sequence classification

- An extremely basic speech classifier / Un classificateur vocale extremement simple.
- We can store templates for each class and then assign to the one with smallest DTW cost.
  - On peut garder des templates pour chaque classe, et assigner le séquence test à la classe avec le cout DTW le plus petit.



DTW classification in action



Taken from UIUC MLSP class slides

To summarize DTW

- It's a distance over time-series, and having that is awesome.
  - C'est une distance sur des time-series, et c'est magnique d'en avoir un.
- You can train a neural net with it, do clustering (maybe? can you? what would be a problem?).
  - Vous pouvez entrainer un réseau neural avec ca, ou clustering (peut-etre vous pouvez?)

To summarize DTW

- It's a distance over time-series, and having that is awesome.
  - C'est une distance sur des time-series, et c'est magnique d'en avoir un.
- You can train a neural net with it, do clustering (maybe? can you? what would be a problem?).
  - Vous pouvez entrainer un réseau neural avec ca, ou clustering (peut-etre vous pouvez?)
- You can even dub movies! / Vous pouvez meme faire du doublage avec DTW.

Table of Contents

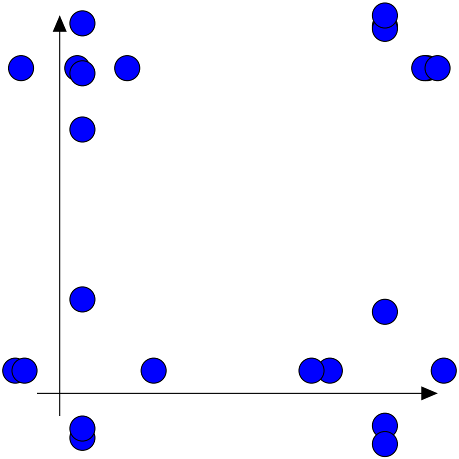
Dynamic Time Warping

Hidden Markov Models
- Inference
- Learning
- Decoding
- HMM Applications
- HMM Variants

A model over time series

- DTW is nice and all, but it's just a distance, it's not a model.
  - DTW est bon mais ce n'est qu'une distance, n'est pas une modèle.
- Remember the duality between likelihood / and distances (e.g. Gaussian and Euclidean distance)

Let's remember GMMs



GMMs with time

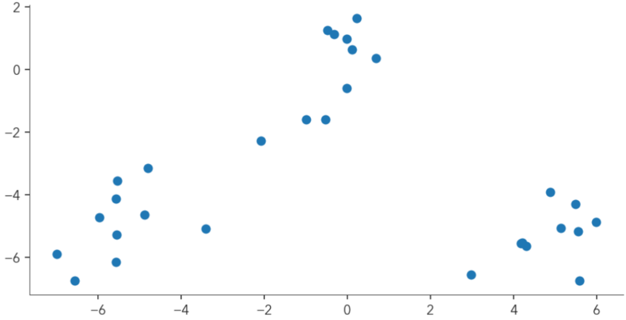


Image taken from UIUC MLSP class

GMMs with time

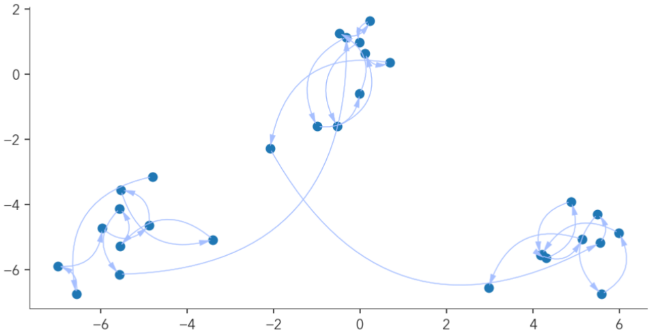
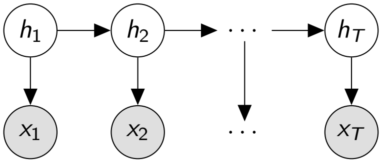


Image taken from UIUC MLSP class

Tired of IID models? HMMs

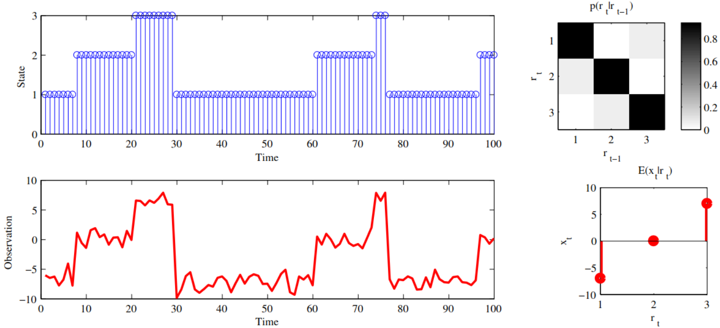
- Model / Modèle:



$$h_n | h_{n-1} \sim \text{Discrete}(A(:, h_{n-1}))$$
$$x_n | h_n \sim p(x_n | h_n, O)$$

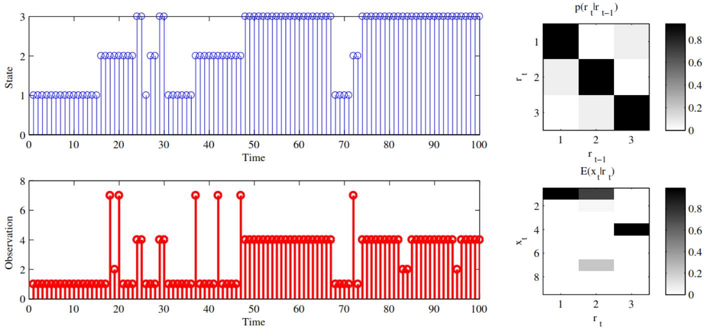
- $h_n \in \{1, \ldots, K\}$, latent variables (embeddings). Difference from before is h_n and h_{n+1} are connected!
  - Différence avec avant est que les variables latent sont connéctés.
- $x_n \in \mathbb{R}^L$, observed data items / les données observées.
- The parameters $\theta = \{O, A\}$ / Les paramètres.
- $O \in \mathbb{R}^{L \times K}$, the emission matrix, $A \in \mathbb{R}^{K \times K}$, the transition matrix.
- Learning is conceptually all the same. Just that E-step is little different.
  - L'apprentissage est meme qu'avant. C'est juste que E-step est un peu différent.

Continuous data generated from an HMM



In this case $p(x_t|r_t, O) = \mathcal{N}(x_n; \mu_{r_t}, \sigma^2 I)$.

Discrete data generated from an HMM



In this case $p(x_t | r_t, O) = \text{Discrete}(x_t; O[:, r_t])$.

Statistical Problems to solve with HMMs

- **Inference / Evaluation**
  - How do we calculate $p(x_{1:T}|\theta)$. / Comment est-ce qu'on calcule ce marginale?
- **Decoding**
  - What are the optimal state values (not probabilities) given $x_{1:T}$ and a learnt model. / Comment obtiens-t-on les valeurs des états optimales?
- **Learning**
  - Given a sequence $x_{1:T}$, how do we learn the optimal model parameters? / Étant donné une séquence $x_{1:T}$ comment est-ce qu'on peut apprendre les paramètres optimales du modèle?

Table of Contents

Dynamic Time Warping

Hidden Markov Models
- Inference
- Learning
- Decoding
- HMM Applications
- HMM Variants

Inference in HMMs

- The precise inference question / La question précise pour l'inférence.

$$p(x_{1:T}|\theta) = \sum_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta)$$

Inference in HMMs

- The precise inference question / La question précise pour l'inférence.

$$p(x_{1:T}|\theta) = \sum_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta)$$
$$= \sum_{h_{1:T}} \prod_{t=1}^{T} p(x_t|h_t)p(h_t|h_{t-1})$$

Inference in HMMs

- The precise inference question / La question précise pour l'inférence.

$$
\begin{aligned}
p(x_{1:T}|\theta) &= \sum_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta) \\
&= \sum_{h_{1:T}} \prod_{t=1}^{T} p(x_t|h_t) p(h_t|h_{t-1}) \\
&= \sum_{h_T} \cdots \sum_{h_2} \sum_{h_1} p(x_T|h_T) p(h_T|h_{T-1}) \ldots p(x_2|h_2) p(h_2|h_1) p(h_1)
\end{aligned}
$$

Inference in HMMs

- The precise inference question / La question précise pour l'inférence.

$$
\begin{aligned}
p(x_{1:T}|\theta) &= \sum_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta) \\
&= \sum_{h_{1:T}} \prod_{t=1}^{T} p(x_t|h_t) p(h_t|h_{t-1}) \\
&= \sum_{h_T} \cdots \sum_{h_2} \sum_{h_1} p(x_T|h_T) p(h_T|h_{T-1}) \ldots p(x_2|h_2) p(h_2|h_1) p(h_1)
\end{aligned}
$$

- This is a huge sum! / C'est une operation immense.

Inference in HMMs

- The precise inference question / La question précise pour l'inférence.

$$\begin{aligned} p(x_{1:T}|\theta) &= \sum_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta) \\ &= \sum_{h_{1:T}} \prod_{t=1}^{T} p(x_t|h_t)p(h_t|h_{t-1}) \\ &= \sum_{h_T} \cdots \sum_{h_2} \sum_{h_1} p(x_T|h_T)p(h_T|h_{T-1}) \ldots p(x_2|h_2)p(h_2|h_1)p(h_1) \end{aligned}$$

- This is a huge sum! / C'est une operation immense.
- What can we do? / Qu'est-ce qu'on peut faire?

Inference in HMMs

- The forward inference: (The filtering density)
  - Inférence en avançant

$$\alpha(h_t) := p(x_{1:t}, h_t)$$

- The backward inference:
  - Inférence en réculant

$$\beta(h_t) := p(x_{t+1:T} | h_t)$$

The Dynamic Programming Solution (Again)

$$\alpha(h_t) = p(x_t|h_t) \sum_{h_{t-1}} p(h_t|h_{t-1}) \underbrace{p(x_{t-1}|h_{t-1}) \dots \underbrace{p(x_2|h_2) \sum_{h_1} p(h_2|h_1) p(x_1|h_1) \underbrace{p(h_1)}_{\alpha(h_1)}}_{\alpha(h_2)}}_{\alpha(h_{t-1})}$$

$$\beta(h_t) = \sum_{h_{t+1}} p(h_{t+1}|h_t) p(x_{t+1}|h_{t+1}) \dots \underbrace{\underbrace{\sum_{h_T} p(h_T|h_{T-1}) p(x_T|h_T) \underbrace{1}_{\beta(h_T)}}_{\beta(h_{T-1})}}_{\beta(h_{t+1})}$$

The forward and backward recursions

- The forward recursion / La recurrence en avançant:

$$\alpha(h_t) = p(x_t|h_t) \sum_{h_{t-1}} p(h_t|h_{t-1})\, \alpha(h_{t-1})$$

- The backward recursion / La recurrence en reculant:

$$\beta(h_t) = \sum_{h_{t+1}} p(h_{t+1}|h_t) p(x_{t+1}|h_{t+1})\, \beta(h_{t+1})$$

Ok but what happened to the likelihood question?

- Note:

$$\alpha(h_T) = p(x_1, x_2, \ldots, x_T, h_T)$$

- So how do we get $p(x_{1:T})$?

Ok but what happened to the likelihood question?

- Note:

$$\alpha(h_T) = p(x_1, x_2, \ldots, x_T, h_T)$$

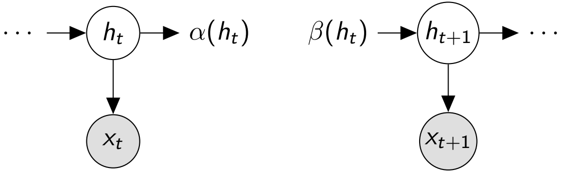
- So how do we get $p(x_{1:T})$?
- $p(x_{1:T}) = \sum_{h_T} p(x_{1:T}, h_T) = \sum_{h_T} \alpha(h_T)$.

Why do we need the backward?

- $\alpha(h_t)$ are "forward messages". $\beta(h_t)$ are "backward messages". One forward pass and one backward pass is sufficient since, / Une passe en avançant et une passe en réculant suffissent parce que,

$$
\begin{aligned}
p(h_t|x_{1:T}) \propto & \, p(h_t, x_{1:T}) \\
= & \, p(h_t, x_{1:t})p(x_{t+1:T}|h_t) \\
= & \, \alpha(h_t)\beta(h_t)
\end{aligned}
$$

- Traditionally (EE traditions), $\alpha_{1:T}$ is known as the filtering density. $\gamma_{1:T} := \alpha_{1:T}.*\beta_{1:T}$ is the smoothing density (la densité smoothing).



Forward Pass in Action

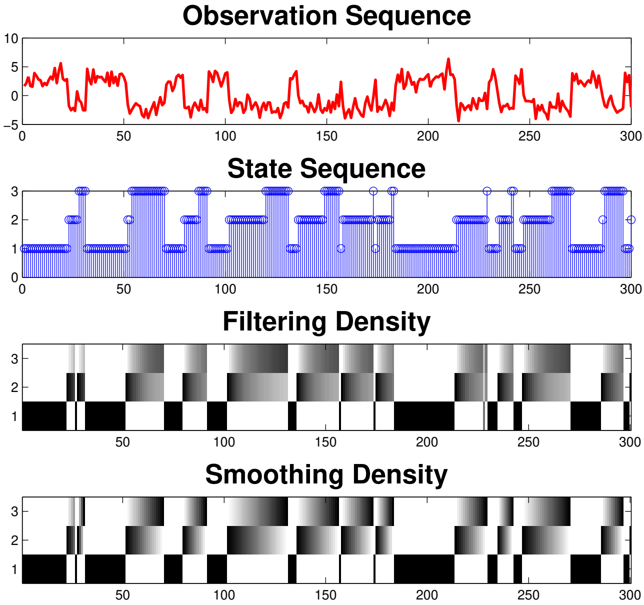


Image taken from Boun CMPE58K slides

Table of Contents

Dynamic Time Warping

Hidden Markov Models
- Inference
- Learning
- Decoding
- HMM Applications
- HMM Variants

The learning question is the same as GMMs

- The learning question / La problématique d'apprentissage

$$\begin{aligned}\widehat{\theta} &= \arg\max_{\theta} p(x_{1:N}|\theta) \\ &= \arg\max_{\theta} \sum_{h_{1:N}} p(x_{1:N}, h_{1:N}|\theta)\end{aligned}$$

The learning question is the same as GMMs

- The learning question / La problématique d'apprentissage

$$\widehat{\theta} = \arg\max_{\theta} p(x_{1:N}|\theta) = \arg\max_{\theta} \sum_{h_{1:N}} p(x_{1:N}, h_{1:N}|\theta)$$

- Write down/Écrit log-likelihood:

$$\log p(x_{1:N}|\theta) = \log \sum_{h_{1:N}} \frac{p(x_{1:N}, h_{1:N}|\theta)}{q(h_{1:N})} q(h_{1:N}) = \log \mathbb{E}_q \left[\frac{p(x_{1:N}, h_{1:N}|\theta)}{q(h_{1:N})} \right]$$

The learning question is the same as GMMs

- The learning question / La problématique d'apprentissage

$$\begin{aligned}\widehat{\theta} &= \arg\max_{\theta} p(x_{1:N}|\theta) \\ &= \arg\max_{\theta} \sum_{h_{1:N}} p(x_{1:N}, h_{1:N}|\theta)\end{aligned}$$

- Write down/Écrit log-likelihood:

$$\begin{aligned}\log p(x_{1:N}|\theta) &= \log \sum_{h_{1:N}} \frac{p(x_{1:N}, h_{1:N}|\theta)}{q(h_{1:N})} q(h_{1:N}) = \log \mathbb{E}_q \left[\frac{p(x_{1:N}, h_{1:N}|\theta)}{q(h_{1:N})} \right] \\ &\geq VLB := \mathbb{E}_q \left[\log \frac{p(x_{1:N}, h_{1:N}|\theta)}{q(h_{1:N})} \right] =^{+} \mathbb{E}_q \left[\log p(x_{1:N}, h_{1:N}|\theta) \right]\end{aligned}$$

The learning question is the same as GMMs

- The learning question / La problématique d'apprentissage

$$\widehat{\theta} = \arg\max_{\theta} p(x_{1:N}|\theta) \\ = \arg\max_{\theta} \sum_{h_{1:N}} p(x_{1:N}, h_{1:N}|\theta)$$

- Write down/Écrit log-likelihood:

$$\log p(x_{1:N}|\theta) = \log \sum_{h_{1:N}} \frac{p(x_{1:N}, h_{1:N}|\theta)}{q(h_{1:N})} q(h_{1:N}) = \log \mathbb{E}_q \left[\frac{p(x_{1:N}, h_{1:N}|\theta)}{q(h_{1:N})} \right] \\ \geq VLB := \mathbb{E}_q \left[\log \frac{p(x_{1:N}, h_{1:N}|\theta)}{q(h_{1:N})} \right] =^{+} \mathbb{E}_q \left[\log p(x_{1:N}, h_{1:N}|\theta) \right]$$

- Except the fact that the posterior distribution is $q(h_t|x_{1:T}) = p(h_t|x_{1:T})$. Not $p(h_t|x_t)$ (unlike the GMM case. / La différence du cas des GMMs est que maintenant le posterior ne se factorise pas sur temps.

EM Algorithm for HMMs

Randomly initialize θ.
while Not converged **do**
  E-step:
    Do a Forward and backward pass. Get all $\alpha(h_t)$ and $\beta(h_t)$.
  M-step:
    $\widehat{\mu}_k = \frac{\sum_{t=1}^{T} \mathbb{E}_q[h_t=k]x_t}{\sum_{t=1}^{T} \mathbb{E}_q[h_t=k]}$
    $\widehat{A}_{ij} = \frac{\sum_{t=1}^{T-1} \mathbb{E}_q[h_t=j, h_{t+1}=i]}{\sum_{t=1}^{T-1} \mathbb{E}_q[h_t=j]}$
end while

- $\mathbb{E}_q[h_t] = \alpha(h_t)\beta(h_{t+1})/Z$
- $\mathbb{E}_q[h_t, h+1] = p(h_t, h_{t+1}|x_{1:T}) \propto p(x_{1:t}, h_t)p(h_{t+1}|h_t)p(x_{t+1}|h_{t+1})p(x_{t+2:T}|h_{t+1}) = \alpha(h_t)p(h_{t+1}|h_t)p(x_{t+1}|h_{t+1})\beta(h_{t+1}) = \alpha(h_t)AO[:, h_{t+1}]\beta(h_{t+1})$

Table of Contents

Dynamic Time Warping

Hidden Markov Models
- Inference
- Learning
- Decoding
- HMM Applications
- HMM Variants

Decoding in HMMs

- The precise inference question / La question précise pour l'inférence.

$$p(x_{1:T}, h^*_{1:T}|\theta) = \max_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta)$$

Decoding in HMMs

- The precise inference question / La question précise pour l'inférence.

$$
\begin{aligned}
p(x_{1:T}, h^*_{1:T}|\theta) &= \max_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta) \\
&= \max_{h_{1:T}} \prod_{t=1}^{T} p(x_t|h_t)p(h_t|h_{t-1})
\end{aligned}
$$

Decoding in HMMs

- The precise inference question / La question précise pour l'inférence.

$$
\begin{aligned}
p(x_{1:T}, h^*_{1:T}|\theta) &= \max_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta) \\
&= \max_{h_{1:T}} \prod_{t=1}^{T} p(x_t|h_t)p(h_t|h_{t-1}) \\
&= \max_{h_T} \ldots \max_{h_2} \max_{h_1} p(x_T|h_T)p(h_T|h_{T-1}) \ldots p(x_2|h_2)p(h_2|h_1)p(h_1)
\end{aligned}
$$

Decoding in HMMs

- The precise inference question / La question précise pour l'inférence.

$$
\begin{aligned}
p(x_{1:T}, h^*_{1:T}|\theta) &= \max_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta) \\
&= \max_{h_{1:T}} \prod_{t=1}^{T} p(x_t|h_t)p(h_t|h_{t-1}) \\
&= \max_{h_T} \ldots \max_{h_2} \max_{h_1} p(x_T|h_T)p(h_T|h_{T-1}) \ldots p(x_2|h_2)p(h_2|h_1)p(h_1)
\end{aligned}
$$

- This is a huge max! / C'est une operation immense.

Decoding in HMMs

- The precise inference question / La question précise pour l'inférence.

$$
\begin{aligned}
p(x_{1:T}, h^*_{1:T}|\theta) &= \max_{h_{1:T}} p(x_{1:T}, h_{1:T}|\theta) \\
&= \max_{h_{1:T}} \prod_{t=1}^{T} p(x_t|h_t)p(h_t|h_{t-1}) \\
&= \max_{h_T} \ldots \max_{h_2} \max_{h_1} p(x_T|h_T)p(h_T|h_{T-1}) \ldots p(x_2|h_2)p(h_2|h_1)p(h_1)
\end{aligned}
$$

- This is a huge max! / C'est une operation immense.
- What can we do? / Qu'est-ce qu'on peut faire?

The Dynamic Programming Solution (Again)

$$V(h_t) = p(x_t|h_t) \max_{h_{t-1}} p(h_t|h_{t-1}) \underbrace{p(x_{t-1}|h_{t-1}) \ldots \underbrace{p(x_2|h_2) \max_{h_1} p(h_2|h_1)p(x_1|h_1) \underbrace{p(h_1)}_{V(h_1)}}_{V(h_2)}}_{V(h_{t-1})}$$

We run this recursion, and then backtrack to find the optimal path $h^*_{1:T}$. (The Viterbi algorithm) / On roule la recurrence et puis backtrack pour trouver la trace optimale $h^*_{1:T}$.

Table of Contents

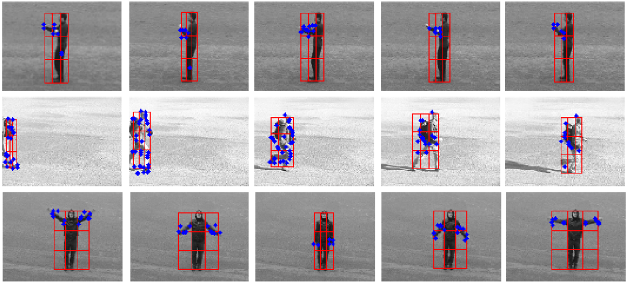
Dynamic Time Warping

Hidden Markov Models

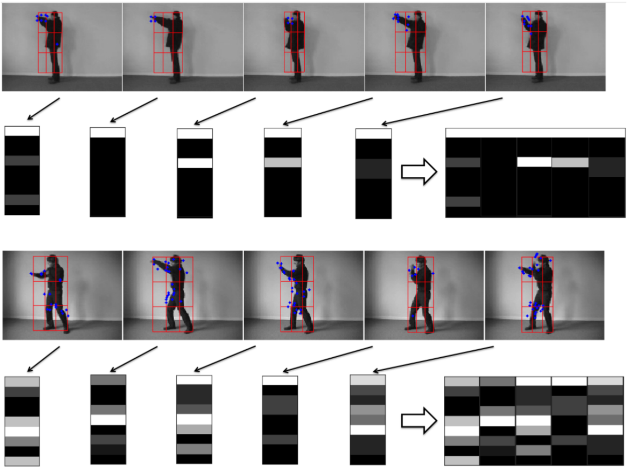
- Inference
- Learning
- Decoding
- HMM Applications
- HMM Variants

An HMM Learning Application

- Human Action Recognition



Getting the sequences



HMMs for classification

- Train an HMM for each class / On entraine un HMM pour chaque classe.
- In test time we assign to the HMM that yields the max likelihood.

$$\widehat{c}_n = \arg\max_k p(x_n|\theta_k)$$

HMMs for classification

- Train an HMM for each class / On entraine un HMM pour chaque classe.
- In test time we assign to the HMM that yields the max likelihood.

$$\widehat{c}_n = \arg\max_k p(x_n|\theta_k)$$

- We saw this type of thing before, remember? / Vous vous souvenez de ça?

HMMs for classification

- Train an HMM for each class / On entraine un HMM pour chaque classe.
- In test time we assign to the HMM that yields the max likelihood.

$$\widehat{c}_n = \arg\max_k p(x_n|\theta_k)$$

- We saw this type of thing before, remember? / Vous vous souvenez de ça?
- Generative classification.. B=boxing, HC=Hand Clapping, HW=Hand Waving ...

EM, 70.1 %

	B	HC	HW	J	R	W
B	**32**	4	1	0	1	0
HC	1	**31**	6	0	1	0
HW	0	1	**29**	0	0	0
J	1	0	0	**17**	20	3
R	0	0	0	7	**10**	0
W	2	0	0	12	4	**33**

HMMs for speech recognition

- Each state should correspond to a semantically meaningful thing. (e.g. a phoneme) / Chaque état HMM devrait correspondre à quelque chose qui a une sense semantique.
- Someone saying 'one'. / Quelqu'un qui dit 'one'.

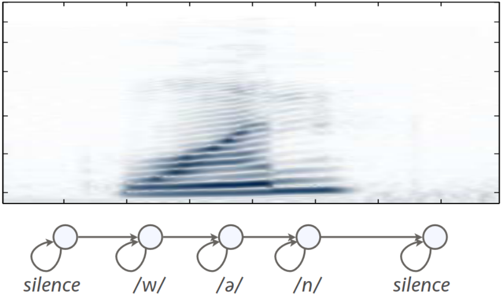


Image taken from UIUC MLSP course.

HMM learning on speech

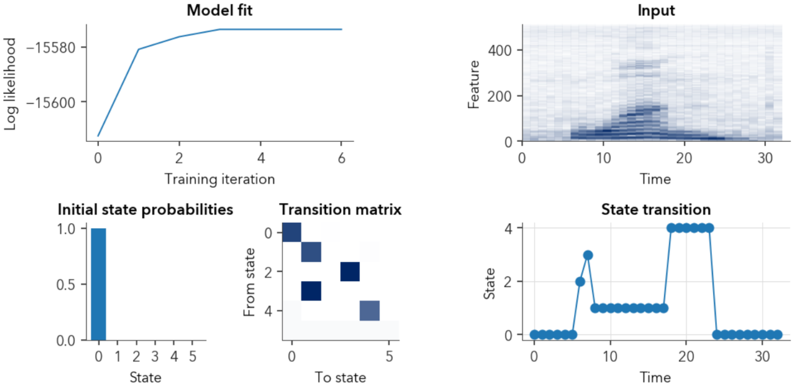


Image taken from UIUC MLSP course.

HMM learning on speech

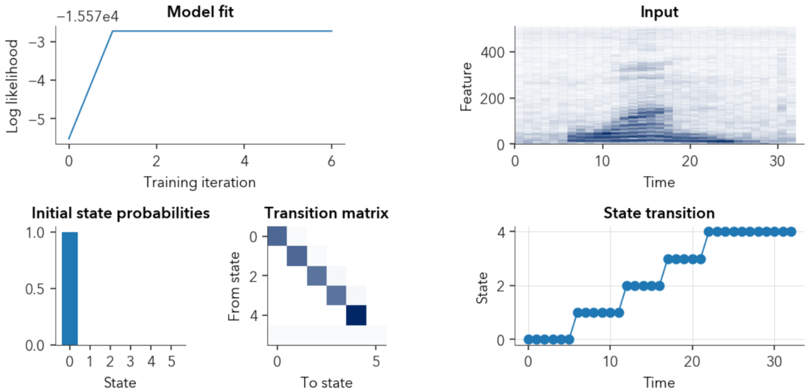


Image taken from UIUC MLSP course.

A big ASR system with HMMs

- A real life HMM speech recognizer in a nut shell / La sommaire d'un système ASR avec HMMs.

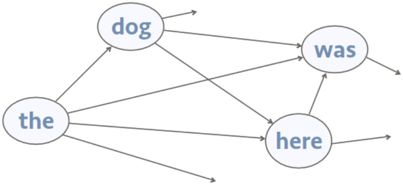


Image taken from UIUC MLSP course.

- You have an HMM for each word, then you connect the HMMs..

And don’t say HMMs are outdated!

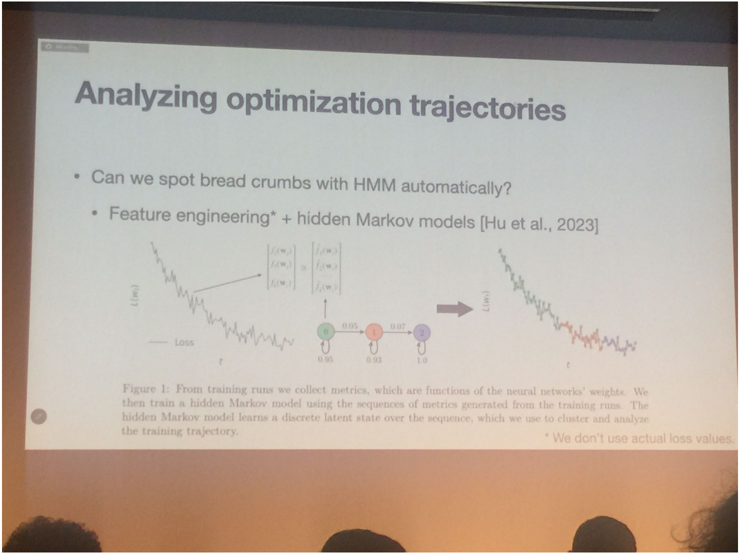


Image taken when I was at SANE 2023 last year

Using HMMs to analyze the training behavior of LLMs from 2023.

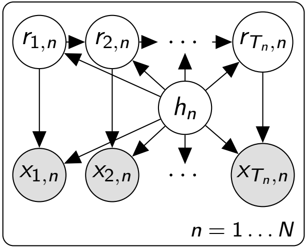
Table of Contents

Dynamic Time Warping

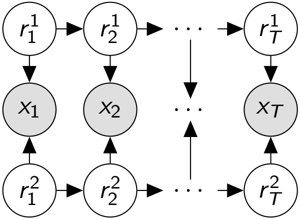
Hidden Markov Models
- Inference
- Learning
- Decoding
- HMM Applications
- HMM Variants

More Advanced HMM Variants

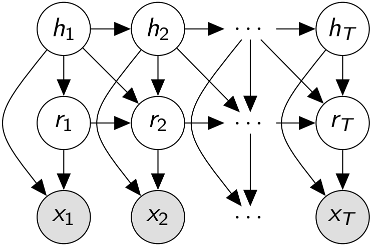
- Mixture of HMMs



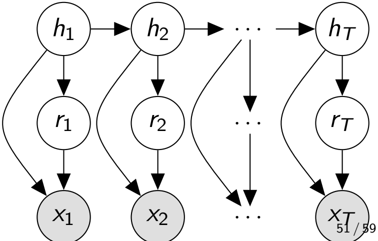
- Factorial HMM



- Switching HMMs

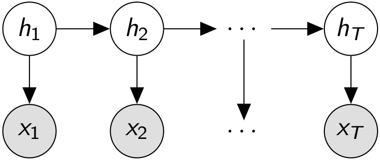


- HMM with Mixture observations



Super Similar but different: Linear Dynamical System

- Model:



$$
\begin{aligned}
h_n | h_{n-1} &\sim \mathcal{N}(h_n; A h_{n-1}, \Sigma_1) \\
x_n | h_n &\sim \mathcal{N}(x_n; O h_n, \Sigma_2)
\end{aligned}
$$

- $h_n \in \mathbb{R}^K$, latent variables/variables latents.
- $x_n \in \mathbb{R}^L$, observed data items / données observées.
- $O \in \mathbb{R}^{L \times K}$, the emission matrix / la matrice d'emission, $A \in \mathbb{R}^{K \times K}$, the transition matrix / la matrice de transition.
- $\theta = \{O, A\}$ parameters / paramètres.
- The inference is done with a Kalman filter. / Le fameux filtre de Kalman est utilisé pour l'inférence.

Tired of directed graphs? MRFs

- The joint distribution is defined with clique "potentials".

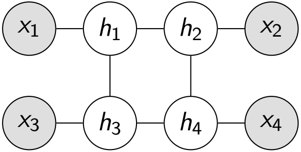
$$p(h_{1:K}, x_{1:J}|\theta) = \frac{1}{Z(\theta)} \prod_{C \in \mathcal{G}} \exp(\theta^T \phi(x_C, h_C))$$

Tired of directed graphs? MRFs

- The joint distribution is defined with clique "potentials".

$$p(h_{1:K}, x_{1:J}|\theta) = \frac{1}{Z(\theta)} \prod_{C \in \mathcal{G}} \exp(\theta^T \phi(x_C, h_C))$$

- Example: (An image segmentation model)



$$\begin{aligned}\phi(x_C, h_C) &= \phi_1(h_i, h_{\mathcal{N}(i)}) + \phi_2(x_i, h_i) \\ &= \theta_1 1_{[h_i = h_{\mathcal{N}(i)}]} + \theta_2 1_{[h_i \neq h_{\mathcal{N}(i)}]} \\ &\quad + \sum_{l,k} \theta_{3,i,k} 1_{[x_i = l][h_i = k]}\end{aligned}$$

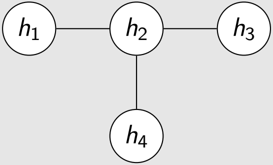
$$Z(\theta) = \int \prod_{C \in \mathcal{G}} \exp(\theta^T \phi(x_C, h_C)) dx_{1:J} dh_{1:K}$$

The notorious partition function!

How to do inference in general graphs?

- Forward-Backward algorithm is an instance of "Belief Propagation".

Example

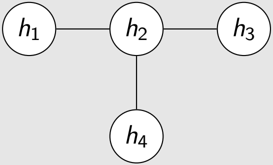


$$p(h_{1:4}) = \frac{1}{Z}\psi(h_1, h_2)\psi(h_2, h_4)\psi(h_2, h_3)$$

$$p(h_2) \propto \sum_{h_1,h_3,h_4} \psi(h_1, h_2)\psi(h_2, h_4)\psi(h_2, h_3)$$
$$= \underbrace{\left(\sum_{h_1} \psi(h_1, h_2)\right)}_{\mathrm{m}_{1\to 2}} \underbrace{\left(\sum_{h_4} \psi(h_2, h_4)\right)}_{\mathrm{m}_{4\to 2}} \underbrace{\left(\sum_{h_3} \psi(h_2, h_3)\right)}_{\mathrm{m}_{3\to 2}}$$

Example continued

Example



$$p(h_{1:4}) = \frac{1}{Z}\psi(h_1, h_2)\psi(h_2, h_4)\psi(h_2, h_3)$$

$$\begin{aligned}
p(h_1) &\propto \sum_{h_2, h_3, h_4} \psi(h_1, h_2)\psi(h_2, h_4)\psi(h_2, h_3) \\
&= \sum_{h_2} \psi(h_1, h_2) \left(\sum_{h_4} \psi(h_2, h_4) \right) \left(\sum_{h_3} \psi(h_2, h_3) \right) \\
&= \sum_{h_2} \psi(h_1, h_2) \mathrm{m}_{4\to 2}(h_2) \mathrm{m}_{3\to 2}(h_2)
\end{aligned}$$

BP, summarized

- Compute all messages for all possible (i, j) pairs with,

$$\mathrm{m}_{i\to j}(h_j) = \sum_{h_i} \psi(h_i, h_j) \overbrace{\prod_{l\in\mathcal{N}(i)\setminus j} \mathrm{m}_{l\to i}(h_i)}^{\text{Incoming Messages to node } i}$$

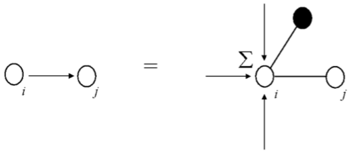


Figure is taken from Yedidia et al. 2001.

BP, summarized

- Compute all messages for all possible (i, j) pairs with,

$$\mathrm{m}_{i\to j}(h_j) = \sum_{h_i} \psi(h_i, h_j) \overbrace{\prod_{l\in\mathcal{N}(i)\setminus j} \mathrm{m}_{l\to i}(h_i)}^{\text{Incoming Messages to node } i}$$

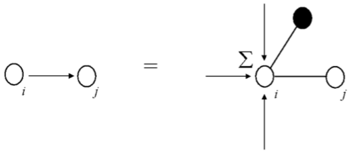


Figure is taken from Yedidia et al. 2001.

- The Belief for node i is $B(h_i) = p(h_i) = \prod_{j\in\mathcal{N}(i)} \mathrm{m}_{j\to i}(h_i)$.

BP, summarized

- Compute all messages for all possible (i, j) pairs with,

$$m_{i\to j}(h_j) = \sum_{h_i} \psi(h_i, h_j) \overbrace{\prod_{l\in\mathcal{N}(i)\backslash j} m_{l\to i}(h_i)}^{\text{Incoming Messages to node } i}$$

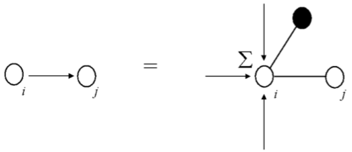


Figure is taken from Yedidia et al. 2001.

- The Belief for node i is $B(h_i) = p(h_i) = \prod_{j\in\mathcal{N}(i)} m_{j\to i}(h_i)$.
- One pass from leaves to root and one pass from leaves to root, and we are done.

BP, summarized

- Compute all messages for all possible (i, j) pairs with,

$$m_{i \to j}(h_j) = \sum_{h_i} \psi(h_i, h_j) \overbrace{\prod_{l \in \mathcal{N}(i) \setminus j} m_{l \to i}(h_i)}^{\text{Incoming Messages to node } i}$$

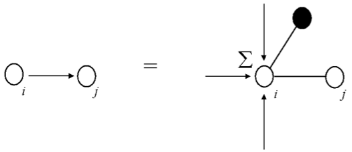


Figure is taken from Yedidia et al. 2001.

- The Belief for node i is $B(h_i) = p(h_i) = \prod_{j \in \mathcal{N}(i)} m_{j \to i}(h_i)$.
- One pass from leaves to root and one pass from leaves to root, and we are done.
- BP converges to true beliefs in trees. What about general graphs?

Recap

- Dynamic Time Warping
  - It's a way to measure distances between sequences. / Way to measure distances between sequences
- HMMs
  - Probability distribution over sequences. Can be thought of generalization of DTW. / HMMs définissent une distribution de probabilité sur les séquences. Vous pouvez le voir comme étant une géneralization de DTW.
- HMM Applications
  - Speech Recognition, Human Action Recognition, Sequence Clustering, LLM State Transition Understanding, ...
- More Advanced HMMs
  - Mixture of HMMs, Swiching HMMs, HMMs with Mixture observations, Linear Dynamical Systems, MRFs, ...

Suggested reading

- The classic HMM Tutorial:
  `http://www.cs.ubc.ca/~murphyk/Bayes/rabiner.pdf`
- Bishop chapter 13.

Next week

- Graph Signal Processing / Machine Learning