Methods for static and sequential clustering

Matt Whiteway

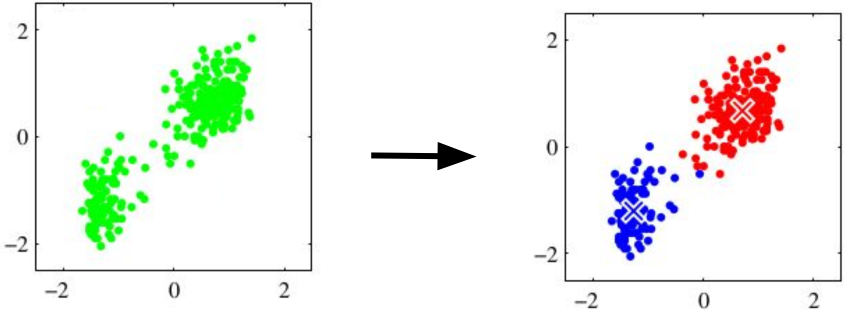
Advanced Theory Seminar

March 2020

Outline

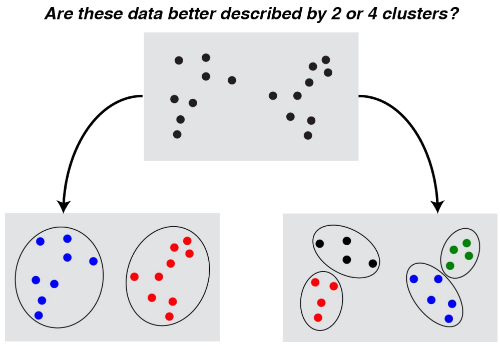
- Motivation: clustering in neuroscience
- Static clustering
    - k-means algorithm
    - Gaussian mixture models (GMMs)
- Sequential clustering - hidden Markov models (HMMs)
    - EM for HMMs
    - Extensions

The clustering problem



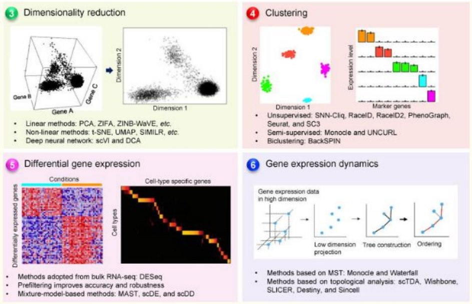
The problems with clustering:

- Unsupervised problem – no ground truth!
- Lots of ways clustering can fail (which is why so many different algorithms exist)
- Choosing the number of clusters

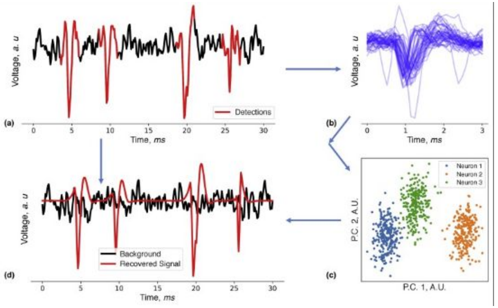


http://alexhwilliams.info/itsneuronalblog/2015/09/11/clustering1/

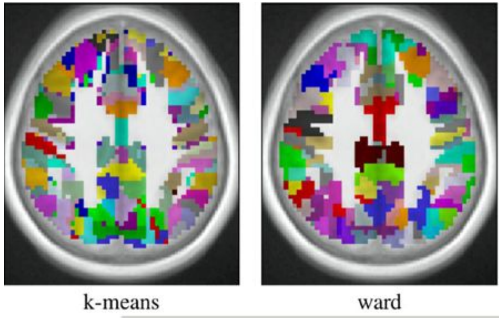
Clustering in neuroscience



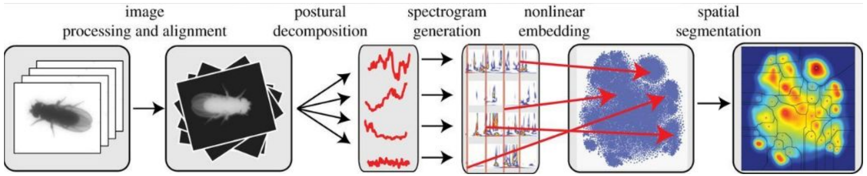
cell type identification from gene expression data



spike sorting from voltage signal data



parcelling brain regions from fMRI data



behavioral clustering from video data

Outline

- Motivation: clustering in neuroscience
- Static clustering
    - k-means algorithm
    - Gaussian mixture models (GMMs)
- Sequential clustering - hidden Markov models (HMMs)
    - EM for HMMs
    - Extensions

k-means: setup

Data:

$$\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, \quad \mathbf{x}_n \in \mathbb{R}^D$$

Goal: partition data into *K* clusters

cluster prototypes
(parameters)

$$\{\boldsymbol{\mu}_k\}_{k=1}^K$$

cluster indicators
(latent variables)

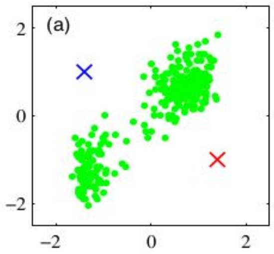
$$\begin{aligned} r_{nk} &= 1 \text{ if } \mathbf{x}_n \text{ is in cluster } k \\ r_{nk} &= 0 \text{ otherwise} \end{aligned}$$

k-means: algorithm

Cost function:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

k-means: algorithm



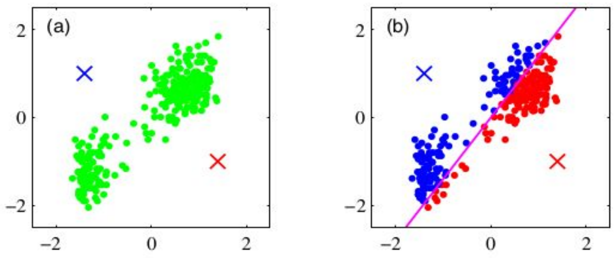
Cost function:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Optimization:

1) Initialize parameters (e.g. random values)

k-means: algorithm



Cost function:

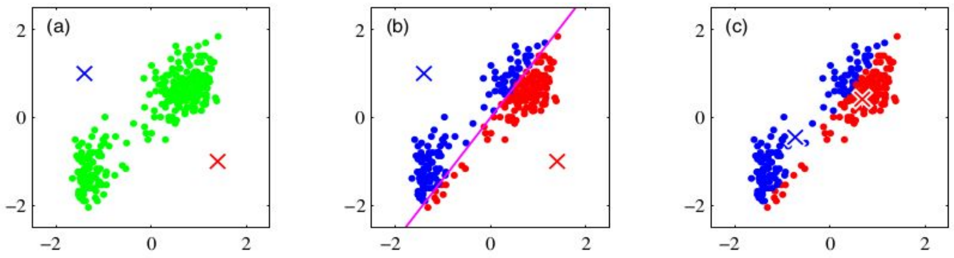
$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Optimization:

1) Initialize parameters (e.g. random values)
2) Fix parameters, optimize latents - assign each data point to the closest cluster center

$$r_{nk} = \begin{cases} 1 & \text{if } k = \operatorname{argmin}_j \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2 \\ 0 & \text{otherwise} \end{cases}$$

k-means: algorithm



Cost function:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

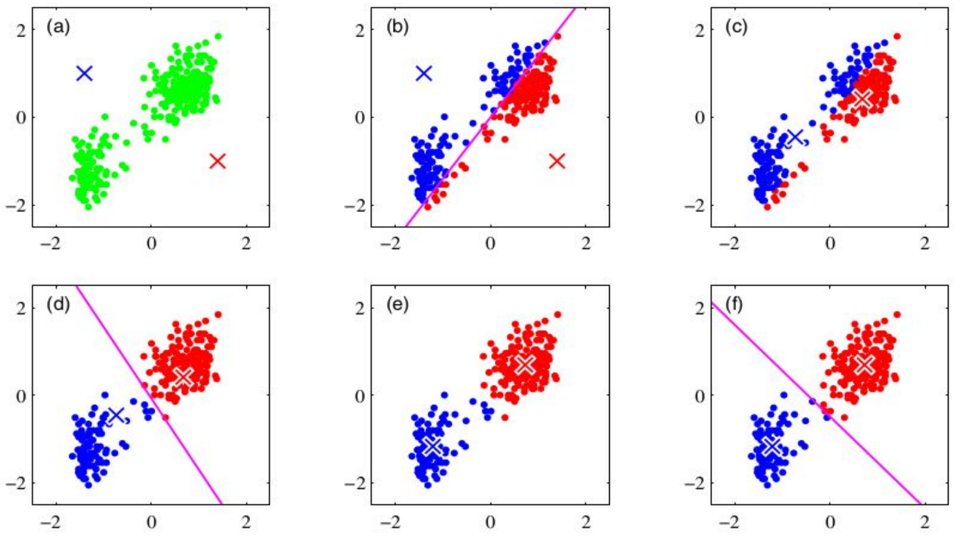
Optimization:

1) Initialize parameters (e.g. random values)
2) Fix parameters, optimize latents – assign each data point to the closest cluster center
3) Fix latents, optimize parameters – update each cluster center as mean of all data points assigned to cluster

$$r_{nk} = \begin{cases} 1 & \text{if } k = \operatorname{argmin}_j \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2 \\ 0 & \text{otherwise} \end{cases}$$

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}}$$

k-means: algorithm



Cost function:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

Optimization:

1) Initialize parameters (e.g. random values)
2) Fix parameters, optimize latents - assign each data point to the closest cluster center
3) Fix latents, optimize parameters - update each cluster center as mean of all data points assigned to cluster
4) Repeat (2) and (3) until convergence

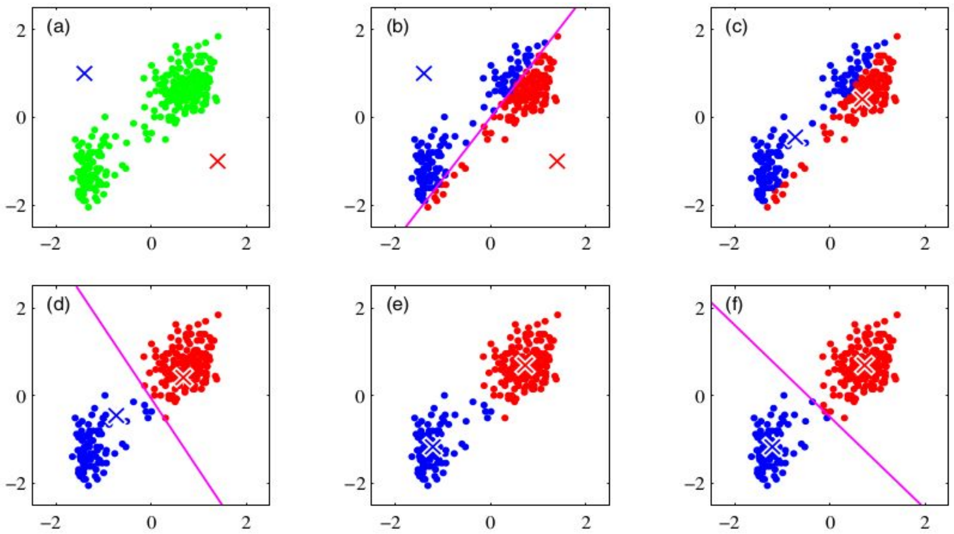
$$r_{nk} = \begin{cases} 1 & \text{if } k = \operatorname{argmin}_j \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2 \\ 0 & \text{otherwise} \end{cases}$$

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}$$

k-means: algorithm

Cost function:

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2$$

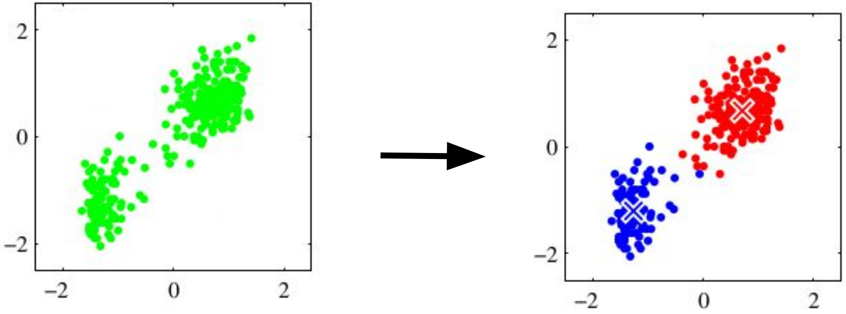


Notes:

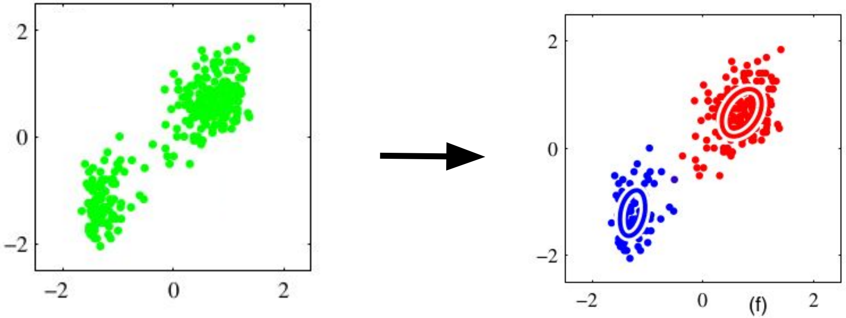
- Convergence criterion: max iterations, cluster assignments don't change, etc.
- Convergence: each iteration of (2) and (3) will further minimize J; local but not global minimum guaranteed
- Alternation between (2) and (3) correspond to (E) and (M) steps of the EM algorithm
- k-means is a *hard assignment* algorithm - each datapoint is assigned to a single cluster

Gaussian mixture models (GMMs)

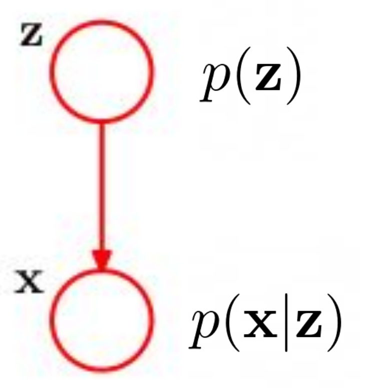
k-means



GMM

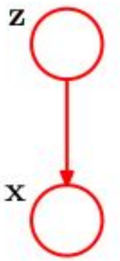


GMM: graphical model



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

GMM: model



Cluster indicators:

$$\begin{aligned} z_k &= 1 \text{ if } \mathbf{x} \text{ is in cluster } k \\ z_k &= 0 \text{ otherwise} \end{aligned}$$

Mixing coefficients:

$$p(z_k = 1) = \pi_k$$

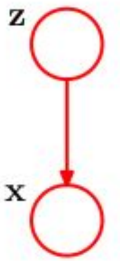
$$p(\mathbf{x}|z_k = 1) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

- $0 \leq \pi_k \leq 1$
- $\sum_{k=1}^{K} \pi_k = 1$

$$\boxed{p(\mathbf{z}) = \prod_{k=1}^{K} \pi_k^{z_k}}$$

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

GMM: model



Joint distribution (distribution over *complete data*):

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

Marginal distribution (distribution over *incomplete data*):

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

GMM: setup

Data:

$$\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, \quad \mathbf{x}_n \in \mathbb{R}^D$$

Goal: partition data into *K* clusters

Gaussian means/covs,
mixing coefficients
(parameters)

$$\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}_{k=1}^K$$

cluster indicators
(latent variables)

$$\begin{aligned} z_{nk} &= 1 \text{ if } \mathbf{x}_n \text{ is in cluster } k \\ z_{nk} &= 0 \text{ otherwise} \end{aligned}$$

GMM: algorithm

$\{\mathbf{x}_n\}$ data

$\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}$ parameters

$\{z_{nk}\}$ latents

Cost function 1 (*incomplete data log-likelihood*):

$$p(\mathbf{X}|\{\boldsymbol{\mu}_k\},\{\boldsymbol{\Sigma}_k\},\boldsymbol{\pi}) = \prod_{n=1}^{N}\sum_{k=1}^{K}\pi_k\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)$$

independence assumption!

$$\ln p(\mathbf{X}|\{\boldsymbol{\mu}_k\},\{\boldsymbol{\Sigma}_k\},\boldsymbol{\pi}) = \sum_{n=1}^{N}\ln\left[\sum_{k=1}^{K}\pi_k\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\right]$$

sum inside log :(

GMM: algorithm

$\{\mathbf{x}_n\}$ data
$\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}$ parameters
$\{z_{nk}\}$ latents

Cost function 2 (*complete data log-likelihood*):

$$p(\mathbf{X}, \mathbf{Z} | \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}, \boldsymbol{\pi}) = \prod_{n=1}^{N} \prod_{k=1}^{K} [\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)]^{z_k}$$

$$\ln p(\mathbf{X}, \mathbf{Z} | \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}, \boldsymbol{\pi}) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \left[\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]$$

But wait!
We don't know **Z** → Use our "best guess", $p(\mathbf{Z}|\mathbf{X})$

logs inside sum :)

GMM: algorithm

$$\{\mathbf{x}_n\}$$ data

$$\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}$$ parameters

$$\{z_{nk}\}$$ latents

Cost function 3 (*expected complete data log-likelihood*):

$$\ln p(\mathbf{X}, \mathbf{Z} | \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}, \boldsymbol{\pi}) = \sum_{n=1}^{N} \sum_{k=1}^{K} z_{nk} \left[\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]$$

$$\mathbb{E}_{p(\mathbf{Z}|\mathbf{X})} \ln p(\mathbf{X}, \mathbf{Z} | \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}, \boldsymbol{\pi}) = \sum_{n=1}^{N} \sum_{k=1}^{K} \mathbb{E}_{p(\mathbf{Z}|\mathbf{X})}[z_{nk}] \left[\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]$$

$$= \sum_{n=1}^{N} \sum_{k=1}^{K} \underline{\gamma(z_{nk})} \left[\ln \pi_k + \ln \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right]$$

"Responsibility" of component k for $\mathbf{x}_n$

$$\gamma(z_{nk}) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

GMM: algorithm

$\{\mathbf{x}_n\}$ data

$\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}$ parameters

$\{z_{nk}\}$ latents

$$\operatorname{argmax}_{\{\{\boldsymbol{\mu}_k\},\{\boldsymbol{\Sigma}_k\},\boldsymbol{\pi}\}} \mathbb{E}_{p(\mathbf{Z}|\mathbf{X})} \ln p(\mathbf{X}, \mathbf{Z} | \{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}, \boldsymbol{\pi})$$

$$\downarrow$$

$$\begin{aligned}
N_k &= \sum_n \gamma(z_{nk}) \\
\boldsymbol{\mu}_k^{\text{new}} &= \frac{1}{N_k} \sum_n \gamma(z_{nk}) \mathbf{x}_n \\
\boldsymbol{\Sigma}_k^{\text{new}} &= \frac{1}{N_k} \sum_n \gamma(z_{nk}) (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^{\intercal} \\
\pi_k^{\text{new}} &= \frac{N_k}{N}
\end{aligned}$$

This is not a closed form solution!
Remember,

$$\gamma(z_{nk}) = \frac{\boxed{\pi_k} \mathcal{N}(\mathbf{x}_n | \boxed{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k})}{\sum_{j=1}^{K} \boxed{\pi_j} \mathcal{N}(\mathbf{x}_n | \boxed{\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j})}$$

old parameter values

GMM: optimization

$\{\mathbf{x}_n\}$ data

$\boldsymbol{\theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}$ parameters

$\{z_{nk}\}$ latents

1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\text{old}}$

GMM: optimization

$\{\mathbf{x}_n\}$ data

$\boldsymbol{\theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}$ parameters

$\{z_{nk}\}$ latents

1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\text{old}}$
2) Compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$

GMM: optimization

$$\{\mathbf{x}_n\}$$ data

$$\boldsymbol{\theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}$$ parameters

$$\{z_{nk}\}$$ latents

1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\mathrm{old}}$
2) Compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\mathrm{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$
3) Maximize $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\mathrm{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ to get $\boldsymbol{\mu}_k^{\mathrm{new}}, \boldsymbol{\Sigma}_k^{\mathrm{new}}, \pi_k^{\mathrm{new}}$

GMM: optimization

$\{\mathbf{x}_n\}$ data

$\boldsymbol{\theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\}$ parameters

$\{z_{nk}\}$ latents

1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\text{old}}$
2) Compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$
3) Maximize $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ to get $\boldsymbol{\mu}_k^{\text{new}}, \boldsymbol{\Sigma}_k^{\text{new}}, \pi_k^{\text{new}}$
4) Repeat (2) and (3) until convergence (and set $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$)

GMM: optimization

$$\{\mathbf{x}_n\} \quad \text{data}$$
$$\boldsymbol{\theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\} \quad \text{parameters}$$
$$\{z_{nk}\} \quad \text{latents}$$

1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\text{old}}$
2) Compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$
3) Maximize $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ to get $\boldsymbol{\mu}_k^{\text{new}}, \boldsymbol{\Sigma}_k^{\text{new}}, \pi_k^{\text{new}}$
4) Repeat (2) and (3) until convergence (and set $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$)

GMM

E step: $$\mathbb{E}[z_{nk}] = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$$

k-means

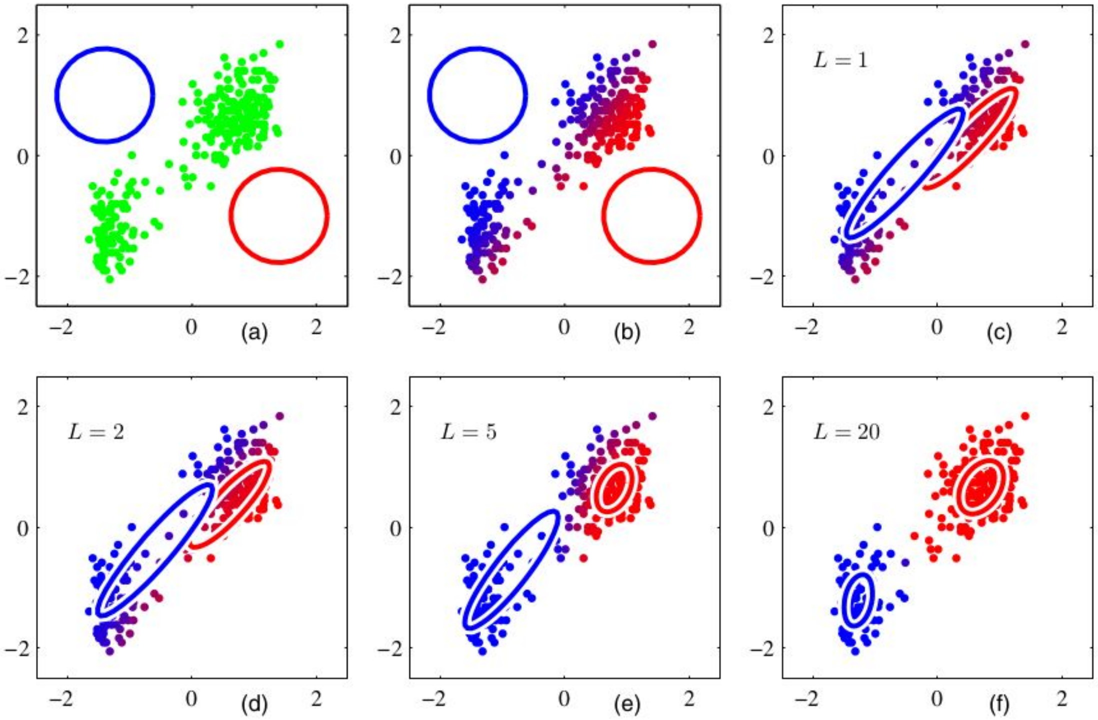
$$r_{nk} = \begin{cases} 1 & \text{if } k = \operatorname{argmin}_j \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2 \\ 0 & \text{otherwise} \end{cases}$$

GMM: optimization

$$\{\mathbf{x}_n\} \quad \text{data}$$
$$\boldsymbol{\theta} = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k\} \quad \text{parameters}$$
$$\{z_{nk}\} \quad \text{latents}$$

1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\text{old}}$
2) Compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$
3) Maximize $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ to get $\boldsymbol{\mu}_k^{\text{new}}, \boldsymbol{\Sigma}_k^{\text{new}}, \pi_k^{\text{new}}$
4) Repeat (2) and (3) until convergence (and set $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$)

	GMM	**k-means**
E step:	$\mathbb{E}[z_{nk}] = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}$	$r_{nk} = \begin{cases} 1 & \text{if } k = \text{argmin}_j \|\mathbf{x}_n - \boldsymbol{\mu}_k\|^2 \\ 0 & \text{otherwise} \end{cases}$
M step:	$\boldsymbol{\mu}_k^{\text{new}} = \frac{\sum_n \gamma(z_{nk}) \mathbf{x}_n}{\sum_n \gamma(z_{nk})}$	$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}}$



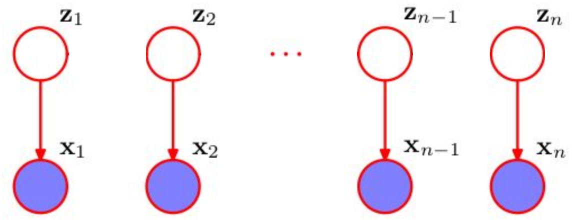
2
0
−2
−2
0
2
(a)
(b)
(c)
(d)
(e)
(f)
$L = 1$
$L = 2$
$L = 5$
$L = 20$

Outline

- Motivation: clustering in neuroscience
- Static clustering
    - k-means algorithm
    - Gaussian mixture models (GMMs)
- Sequential clustering - hidden Markov models (HMMs)
    - EM for HMMs
    - Extensions

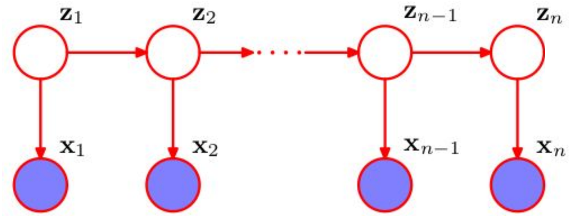
Sequential models

Before: i.i.d. assumption (independent and identically distributed)



$$P(\mathbf{X}, \mathbf{Z}) = \prod_{n=1}^{N} p(\mathbf{x}_n | \mathbf{z}_n) p(\mathbf{z}_n)$$

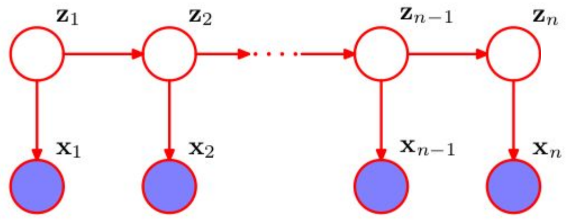
Now: Markov assumption



$$P(\mathbf{X}, \mathbf{Z}) = [p(\mathbf{x}_1 | \mathbf{z}_1) p(\mathbf{z}_1)] \prod_{n=2}^{N} p(\mathbf{x}_n | \mathbf{z}_n) p(\mathbf{z}_n | \mathbf{z}_{n-1})$$

Sequential models

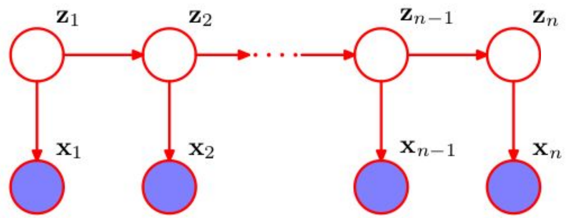
Hidden Markov Model (HMM)



← discrete

← discrete or continuous

Linear (or Latent) Dynamical System (LDS)

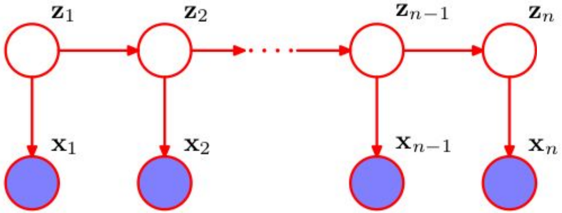


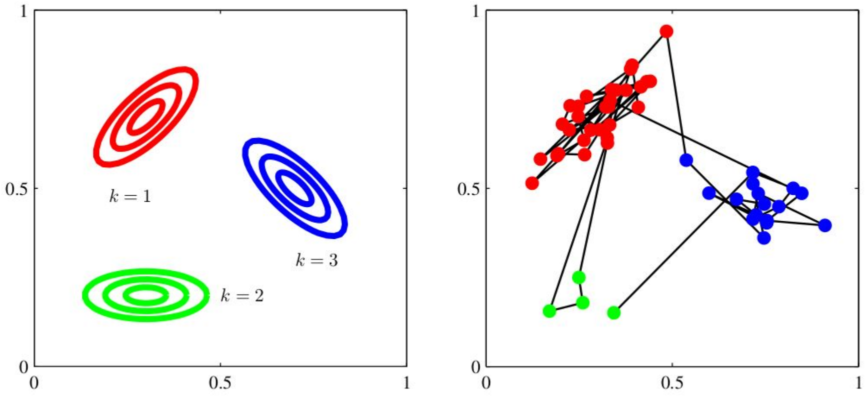
← Gaussian

← linear dependence

← Gaussian

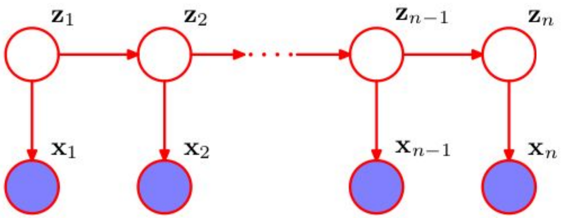
HMM as a generative model





- GMM a special case of the HMM (if each row in **A** is identical)
- If diagonal elements of **A** are close to 1, long runs of observations from the same state

Hidden Markov models (HMMs)



$$P(\mathbf{X}, \mathbf{Z}) = [p(\mathbf{x}_1|\mathbf{z}_1)p(\mathbf{z}_1)] \prod_{n=2}^{N} p(\mathbf{x}_n|\mathbf{z}_n)p(\mathbf{z}_n|\mathbf{z}_{n-1})$$

Initial state dist

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

Transitions

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

Emissions

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

- Discrete tables
- Gaussians
- Mixtures of Gaussians
- Neural networks
- ...

HMM: setup

Data:

$$\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, \quad \mathbf{x}_n \in \mathbb{R}^D$$

Goal: partition data into *K* clusters

Gaussian means/covs,
initial state distribution,
transition matrix
(parameters)

$$\{\{\boldsymbol{\mu}_k\}, \{\boldsymbol{\Sigma}_k\}, \{\pi_k\}, \mathbf{A}\}$$

cluster indicators
(latent variables)

$$\begin{aligned} z_{nk} &= 1 \text{ if } \mathbf{x}_n \text{ is in cluster } k \\ z_{nk} &= 0 \text{ otherwise} \end{aligned}$$

EM for GMMs

1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\text{old}}$
2) Compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$
3) Maximize $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ to get $\boldsymbol{\mu}_k^{\text{new}}, \boldsymbol{\Sigma}_k^{\text{new}}, \pi_k^{\text{new}}$
4) Repeat (2) and (3) until convergence (and set $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$)

EM for HMMs

1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\text{old}}$
2) Compute $\mathbb{E}_{\boxed{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})}} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$
3) Maximize $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ to get $\boldsymbol{\mu}_k^{\text{new}}, \boldsymbol{\Sigma}_k^{\text{new}}, \pi_k^{\text{new}}, \boxed{\mathbf{A}^{\text{new}}}$
4) Repeat (2) and (3) until convergence (and set $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$)

E step: compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X},\mathbf{Z}|\boldsymbol{\theta})$

This corresponds to computing the following quantities:

$$\begin{aligned}
\gamma(\mathbf{z}_n) &= p(\mathbf{z}_n|\mathbf{X},\boldsymbol{\theta}^{\text{old}}) \\
\gamma(z_{nk}) &= \mathbb{E}[z_{nk}]
\end{aligned}$$

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

E step: compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X},\mathbf{Z}|\boldsymbol{\theta})$

This corresponds to computing the following quantities:

$$\begin{aligned}
\gamma(\mathbf{z}_n) &= p(\mathbf{z}_n|\mathbf{X},\boldsymbol{\theta}^{\text{old}}) \\
\gamma(z_{nk}) &= \mathbb{E}[z_{nk}]
\end{aligned}$$

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

- In the GMM,

  $$\gamma(\mathbf{z}_n) = p(\mathbf{z}_n|\mathbf{x}_n,\boldsymbol{\theta}^{\text{old}})$$

  and can hence be calculated for each datapoint independently

E step: compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$

This corresponds to computing the following quantities:

$$
\begin{aligned}
\gamma(\mathbf{z}_n) &= p(\mathbf{z}_n|\mathbf{X}, \boldsymbol{\theta}^{\text{old}}) \\
\gamma(z_{nk}) &= \mathbb{E}[z_{nk}]
\end{aligned}
$$

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

- In the GMM,

  $$\gamma(\mathbf{z}_n) = p(\mathbf{z}_n|\mathbf{x}_n, \boldsymbol{\theta}^{\text{old}})$$

  and can hence be calculated for each datapoint independently
- In the HMM, $\gamma(\mathbf{z}_n)$ and $\xi(\mathbf{z}_{n-1}, \mathbf{z}_n)$ are computed recursively using the *forward-backward* algorithm

M step: maximize $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$

$$
\begin{aligned}
N_k &= \sum_n \gamma(z_{nk}) \\
\boldsymbol{\mu}_k^{\text{new}} &= \frac{1}{N_k} \sum_n \gamma(z_{nk}) \mathbf{x}_n \\
\boldsymbol{\Sigma}_k^{\text{new}} &= \frac{1}{N_k} \sum_n \gamma(z_{nk})(\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^\intercal \\
\pi_k^{\text{new}} &= \frac{\gamma(z_{1k})}{\sum_j \gamma(z_{1j})} \\
A_{jk}^{\text{new}} &= \frac{\sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nk})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j}, z_{nl})}
\end{aligned}
$$

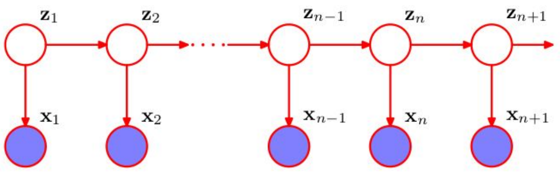
Same form as GMM!

EM for HMMs

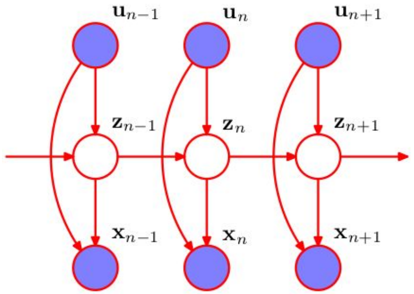
1) Initialize parameters (e.g. using k-means) as $\boldsymbol{\theta}^{\text{old}}$
2) Compute $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$
3) Maximize $\mathbb{E}_{p(\mathbf{Z}|\mathbf{X},\boldsymbol{\theta}^{\text{old}})} \ln p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$ to get $\boldsymbol{\mu}_k^{\text{new}}, \boldsymbol{\Sigma}_k^{\text{new}}, \pi_k^{\text{new}}, \mathbf{A}^{\text{new}}$
4) Repeat (2) and (3) until convergence (and set $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$)

HMM extensions

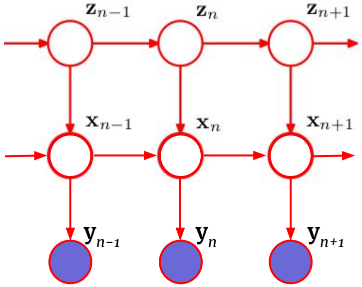
HMM



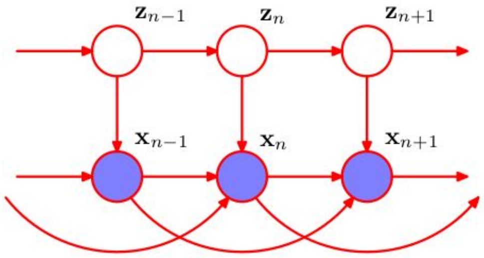
GLM-HMM



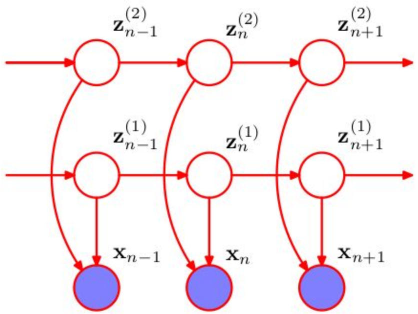
Switching LDS



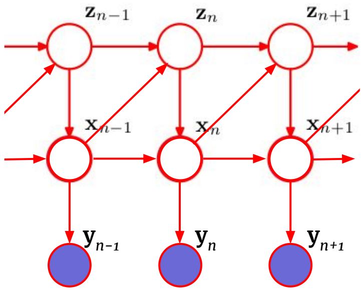
Autoregressive HMM



Factorial HMM



Recurrent switching LDS



HMMs in neuroscience

Modeling neural spike trains

- Petreska et al (2011) Dynamical segmentation of single trials from population neural data [sLDS]
- Escola et al (2011) Hidden Markov models for the stimulus-response relationships of multistate neural systems [GLM-HMM]
- Mazzucato et al (2015) Dynamics of multistable states during ongoing and evoked cortical activity [HMM]
- Maboudi et al (2018) Uncovering temporal structure in hippocampal output patterns [HMM]
- Linderman et al (2019) Hierarchical recurrent state space models reveal discrete and continuous dynamics of neural activity in C. elegans [rsLDS]
- Zoltowski et al (2020) Unifying and generalizing models of neural dynamics during decision-making [rsLDS]
- Recanatesi et al (2020) Metastable attractors explain the variable timing of stable behavioral action sequences [HMM]

Modeling behavior

- McFarland et al (2014) High-resolution eye tracking using V1 neuron activity [~GLM-HMM]
- Wiltschko et al (2015) Mapping sub-second structure in mouse behavior [ARHMM]
- Johnson et al (2016) Composing graphical models with neural networks for structured representations and fast inference [sLDS]
- Markowitz et al (2018) The striatum organizes 3D behavior via moment-to-moment action selection [ARHMM]
- Batty et al (2019) BehaveNet: nonlinear embedding and Bayesian neural decoding of behavioral videos [ARHMM]
- Calhoun et al (2019) Unsupervised identification of the internal states that shape natural behavior [GLM-HMM]

References

- General clustering
  - Alex Williams blog post: http://alexhwilliams.info/itsneuronalblog/2015/09/11/clustering1/
  - Estivill-Castro (2002). "Why so many clustering algorithms: a position paper." SIGKDD Explorations.
- k-means algorithm
  - Bishop Ch. 9
  - Murphy Ch. 11
- Gaussian mixture models
  - Bishop Ch. 9
  - Murphy Ch. 11
- HMMs
  - Bishop Chs. 8, 13
  - Murphy Ch. 17
  - ssm package: https://github.com/slinderman/ssm