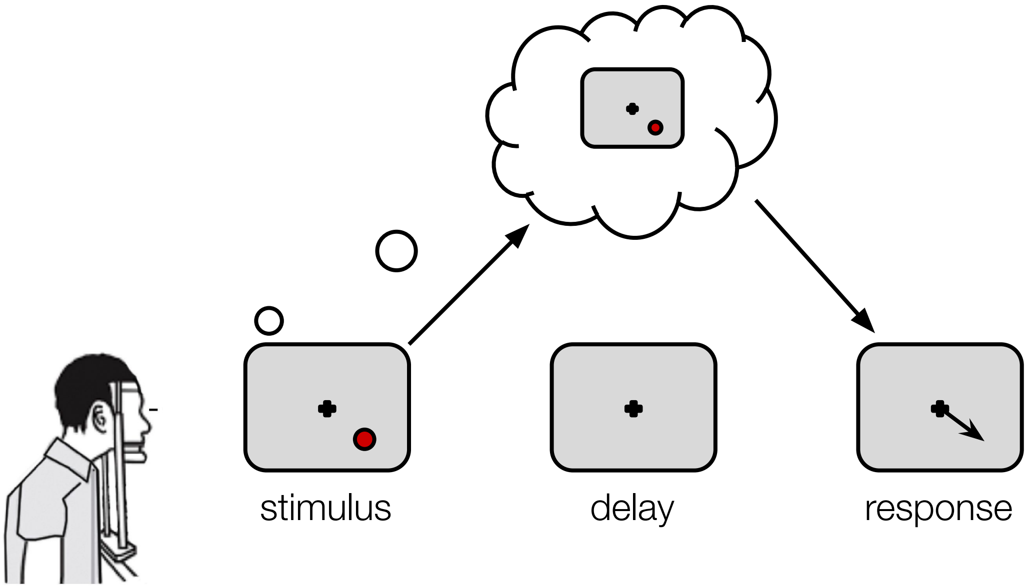
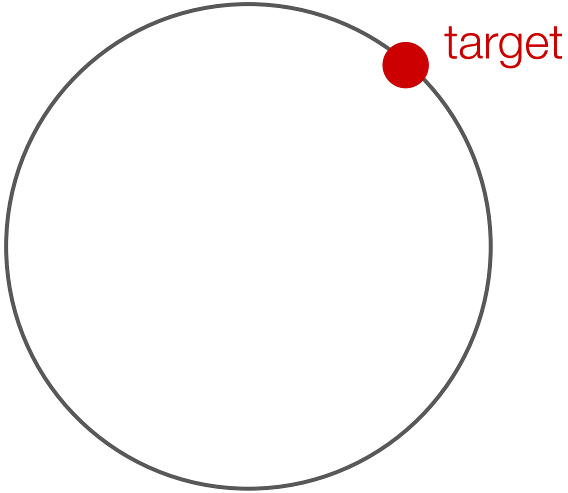
Mixture Models

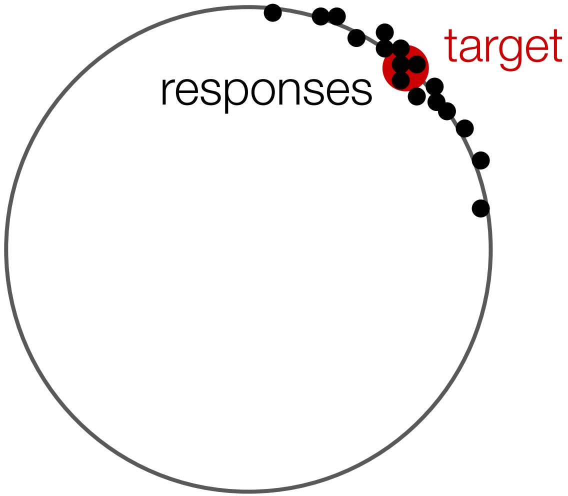
Classic WM task



Response distributions

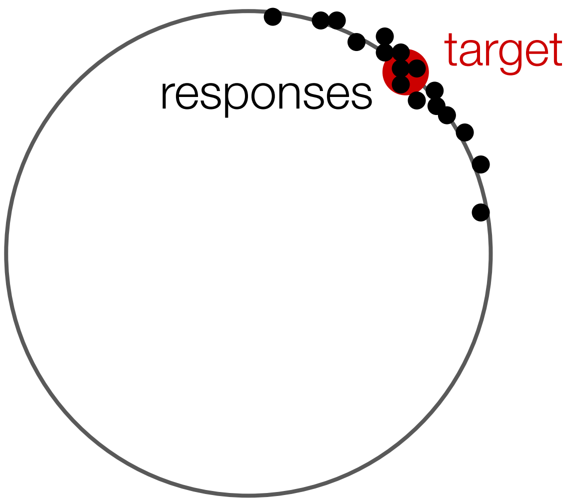


Response distributions

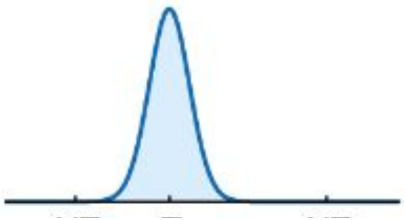


How precise is the working memory representation?

Response distributions

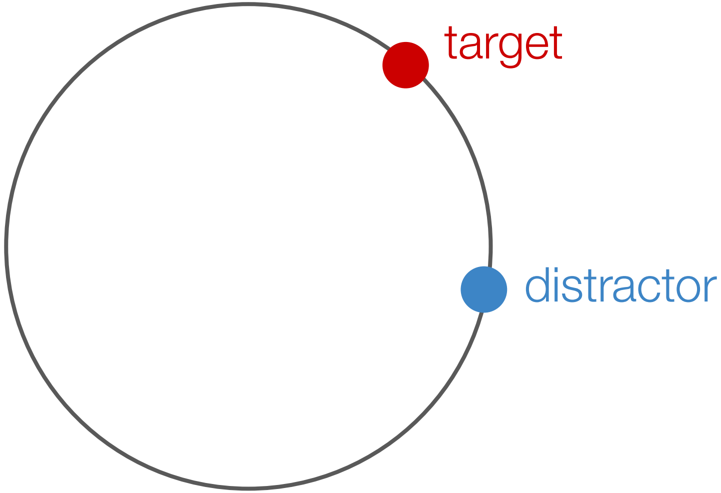


How precise is the working memory representation?

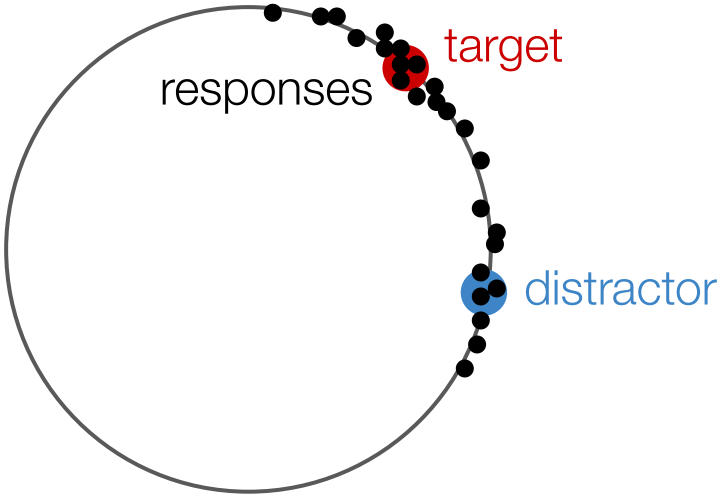


$$\theta = \{\mu, \sigma^2\}$$

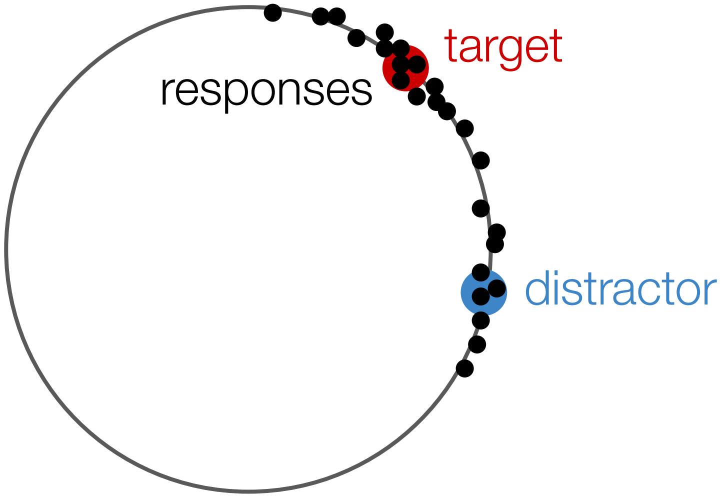
Response distributions



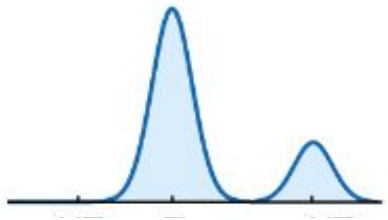
Response distributions



Response distributions

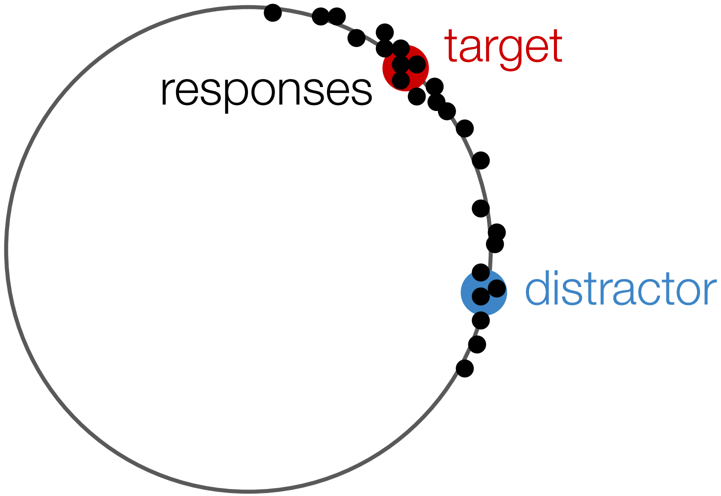


How precise is the working memory representation?

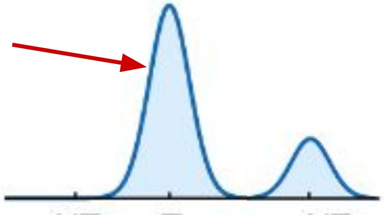


$$\theta = \{\mu_1, \mu_2, \sigma_1^2, \sigma_2^2\}$$

Response distributions

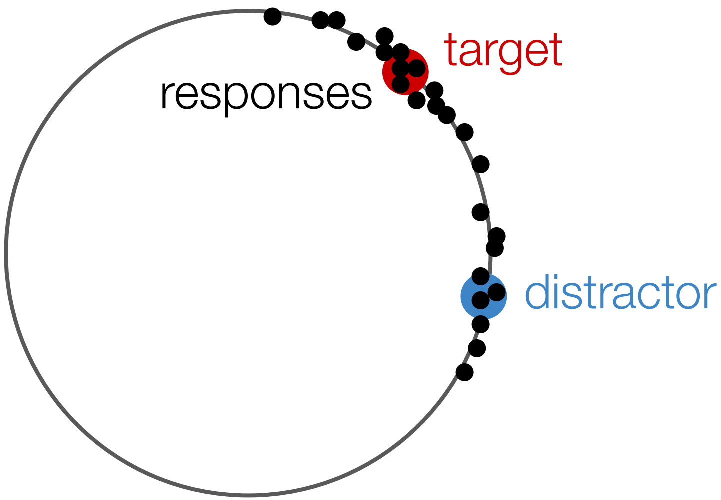


How precise is the working memory representation?

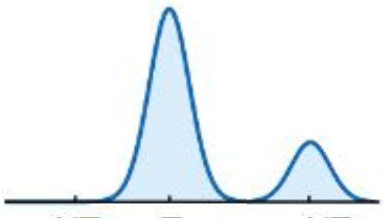


$$\theta = \{\boxed{\mu_1}, \mu_2, \boxed{\sigma_1^2}, \sigma_2^2\}$$

Response distributions

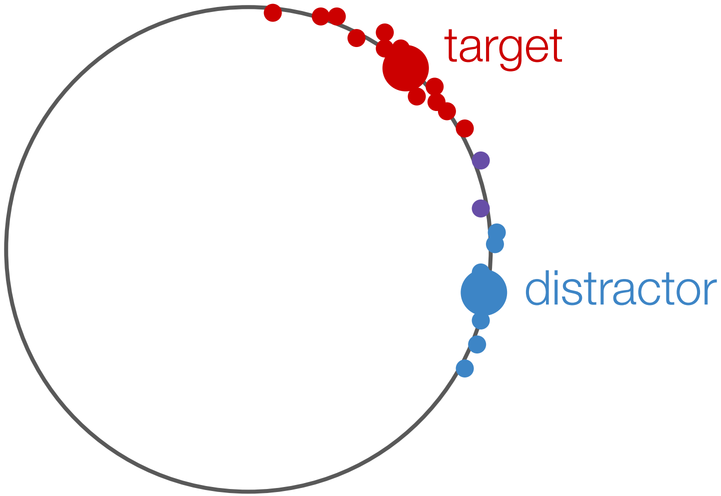


How precise is the working memory representation?

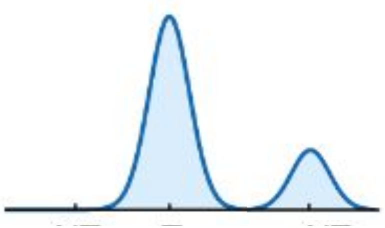


Which stimulus did the subject hold in mind in each trial?

Response distributions

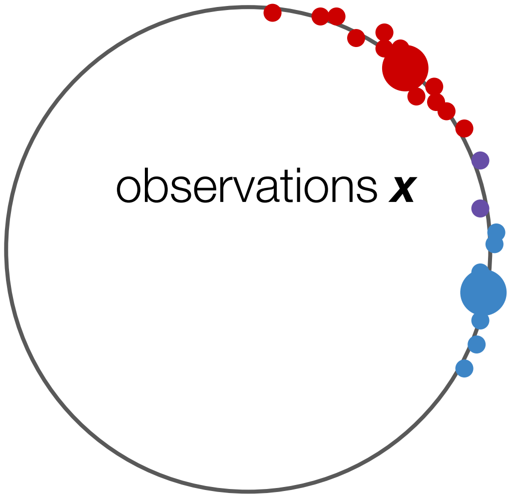


How precise is the working memory representation?



Which stimulus did the subject hold in mind in each trial?

Mixture models

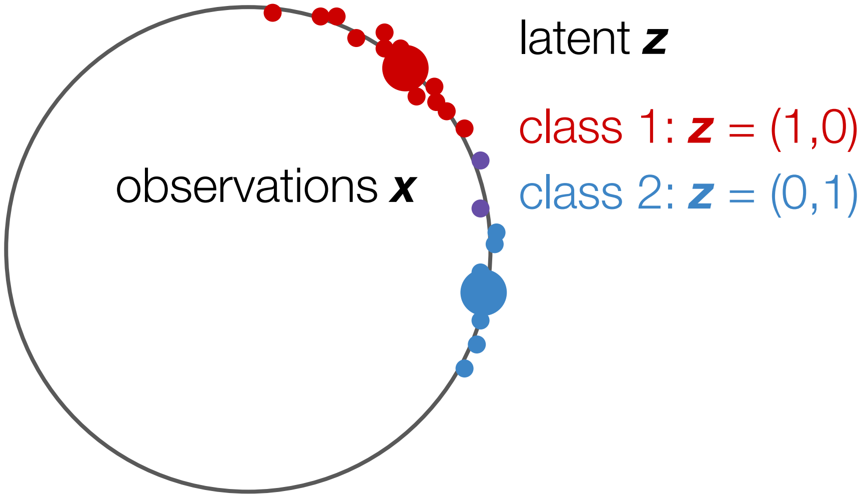


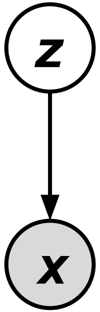
latent $\boldsymbol{z}$

class 1: $\boldsymbol{z} = (1,0)$

class 2: $\boldsymbol{z} = (0,1)$

Mixture models



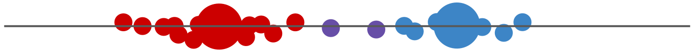


Mixture models

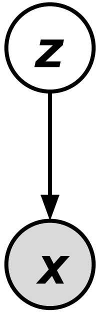
latent $\boldsymbol{z}$

class 1: $\boldsymbol{z}$ = (1,0)

class 2: $\boldsymbol{z}$ = (0,1)



observations $\boldsymbol{x}$



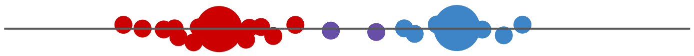
$$p(\boldsymbol{x}|\boldsymbol{z}_k = 1) = \mathcal{N}(\mu_k, \sigma_k^2)$$

Mixture models

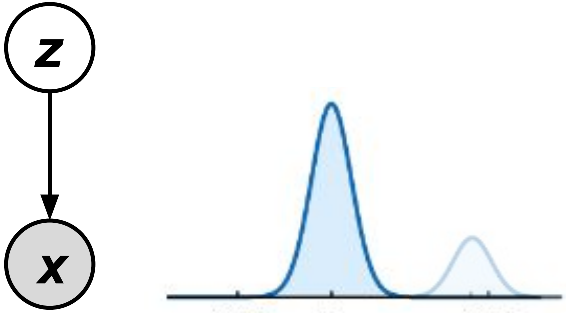
latent ***z***

class 1: ***z*** = (1,0)

class 2: ***z*** = (0,1)



observations ***x***



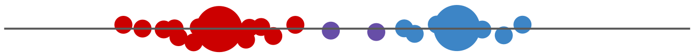
$$p(\boldsymbol{x}|\boldsymbol{z}_k = 1) = \mathcal{N}(\mu_k, \sigma_k^2)$$

Mixture models

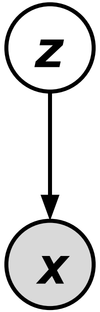
latent ***z***

class 1: ***z*** = (1,0)

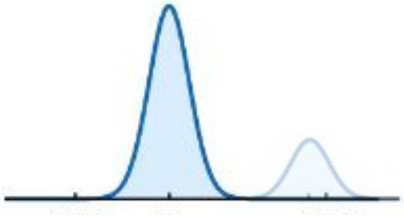
class 2: ***z*** = (0,1)



observations ***x***



But: We don't know the value of ***z***!



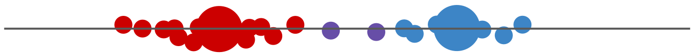
$$p(\boldsymbol{x}|\boldsymbol{z}_k = 1) = \mathcal{N}(\mu_k, \sigma_k^2)$$

Mixture models

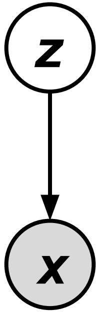
latent $\boldsymbol{z}$

class 1: $\boldsymbol{z}$ = (1,0)

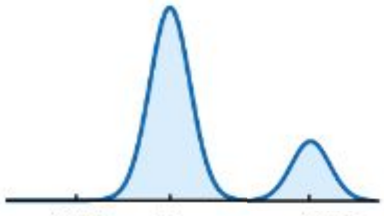
class 2: $\boldsymbol{z}$ = (0,1)



observations $\boldsymbol{x}$



But: We don’t know the value of $\boldsymbol{z}$!



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

marginalization

$$= \sum_k \pi_k \mathcal{N}(\mu_k, \sigma_k^2)$$

Expectation maximization (EM)

EM tackles two problems:

1) Determine which data point belongs to which class

2) Fit class-specific parameters

Expectation maximization (EM)

EM tackles two problems:

1) Determine which data point belongs to which class
   → Inference of the **posterior** over the latent indicator variable ***z***

$$
\begin{aligned}
p(\boldsymbol{z}|\boldsymbol{x}) &= \frac{p(\boldsymbol{z})p(\boldsymbol{x}|\boldsymbol{z})}{p(\boldsymbol{x})} \\
&= \frac{\pi_k \mathcal{N}(\mu_k, \sigma_k)}{\sum_k \pi_k \mathcal{N}(\mu_k, \sigma_k)}
\end{aligned}
$$

Expectation maximization (EM)

EM tackles two problems:

1) Determine which data point belongs to which class
   → Inference of the **posterior** $p(\boldsymbol{z}|\boldsymbol{x}, \theta)$ over the latent indicator variable ***z***

2) Fit class-specific parameters
   ML solution: Optimize $\ln p(\boldsymbol{x}) = \ln \sum_{z} p(\boldsymbol{x}, \boldsymbol{z}|\theta)$, but high computational cost

Expectation maximization (EM)

EM tackles two problems:

1) Determine which data point belongs to which class
   → Inference of the **posterior** $p(\boldsymbol{z}|\boldsymbol{x},\theta)$ over the latent indicator variable ***z***

2) Fit class-specific parameters
   ML solution: Optimize $\ln p(\boldsymbol{x}) = \ln \sum_{z} p(\boldsymbol{x}, \boldsymbol{z}|\theta)$, but high computational cost

   Alternatively, complete-data log LL $\ln p(\boldsymbol{x}, \boldsymbol{z}|\theta)$, but we don't know values of ***z***

Expectation maximization (EM)

EM tackles two problems:

1) Determine which data point belongs to which class
   $\longrightarrow$ Inference of the **posterior** $p(\boldsymbol{z}|\boldsymbol{x},\theta)$ over the latent indicator variable ***z***

2) Fit class-specific parameters
   ML solution: Optimize $\ln p(\boldsymbol{x}) = \ln \sum_z p(\boldsymbol{x},\boldsymbol{z}|\theta)$, but high computational cost

   Alternatively, complete-data log LL $\ln p(\boldsymbol{x},\boldsymbol{z}|\theta)$, but we don't know values of ***z***

   Solution: Optimize expected value $\mathbb{E}_{\mathbf{z}}\left[\ln p(\boldsymbol{x},\boldsymbol{z}|\theta')\right] = \sum_z p(\boldsymbol{z}|\boldsymbol{x},\theta) \ln p(\boldsymbol{x},\boldsymbol{z}|\theta')$

Expectation maximization (EM)

EM tackles two problems:

1) Determine which data point belongs to which class
   → Inference of the **posterior** $p(\boldsymbol{z}|\boldsymbol{x},\theta)$ over the latent indicator variable ***z***

2) Fit class-specific parameters
   → Find parameters that optimize expected complete-data log likelihood

$$\mathbb{E}_{\mathbf{z}}\left[\ln p(\boldsymbol{x},\boldsymbol{z}|\theta')\right] = \sum_{z} p(\boldsymbol{z}|\boldsymbol{x},\theta) \ln p(\boldsymbol{x},\boldsymbol{z}|\theta')$$

Iterate

Tutorial

- Simulate observations from a generative model

Data is two-dimensional!

$$\mu = (\mu_x, \mu_y)$$

$$\Sigma = \begin{pmatrix} \sigma_x^2 & \sigma_x\sigma_y \\ \sigma_y\sigma_x & \sigma_y^2 \end{pmatrix}$$

for each class

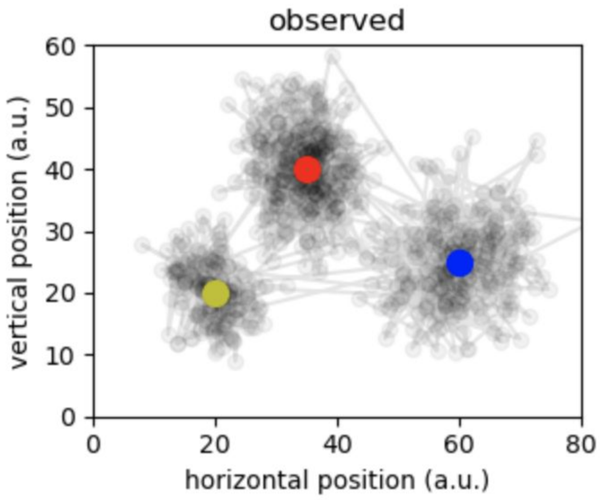
- Use EM to fit a Gaussian mixture model to simulated data

Mixture models

Hidden Markov Models

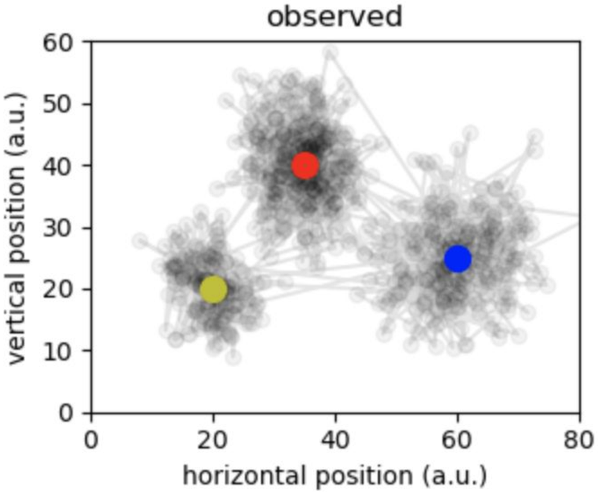
Hidden Markov Models

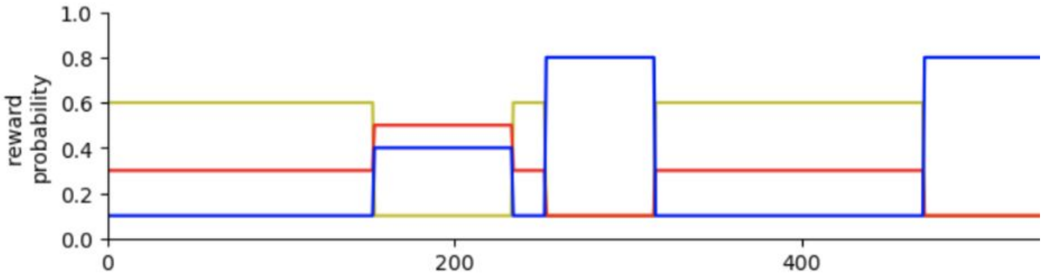
Same setup as before, but with sequential dependencies between data points



Hidden Markov Models

Same setup as before, but with sequential dependencies between data points



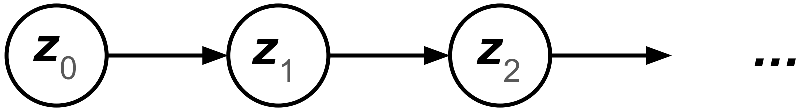


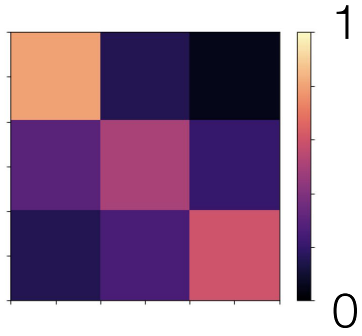
(e.g. because reward probabilities fluctuate in a block structure)

Hidden Markov Models

To capture sequential dependencies in choice behavior, we need an explicit model of transition structure of a latent variable, $p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1})$

→ Transition matrix $\boldsymbol{A}$

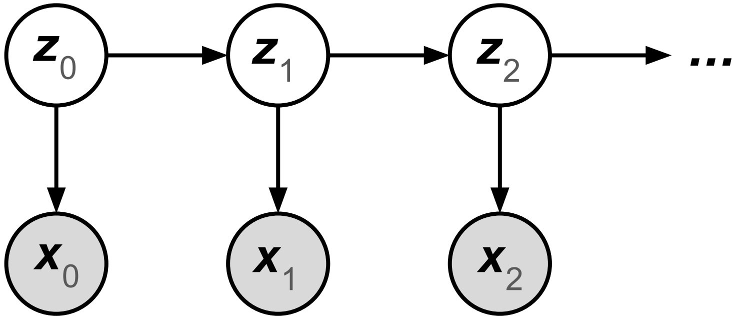




Hidden Markov Models

To capture sequential dependencies in choice behavior, we need an explicit model of transition structure of a latent variable, $p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1})$

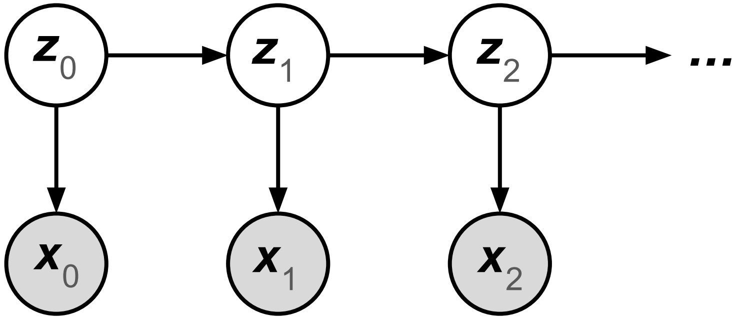
Saccades ***x*** are an "emission" of the latent choice variable



Hidden Markov Models

To capture sequential dependencies in choice behavior, we need an explicit model of transition structure of a latent variable, $p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1})$

Saccades ***x*** are an “emission” of the latent choice variable



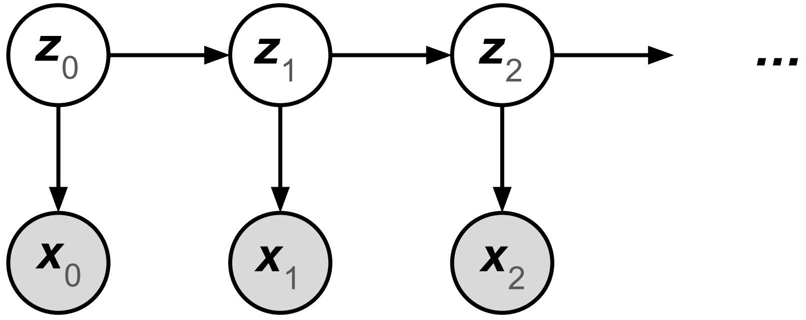
Joint distribution:

$$p(\boldsymbol{x}, \boldsymbol{z}|\theta)$$
$$= p(\boldsymbol{z}_0) \prod_{n=1}^{N} p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \prod_{n=0}^{N} p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta)$$

Hidden Markov Models

To capture sequential dependencies in choice behavior, we need an explicit model of transition structure of a latent variable, $p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1})$

Saccades ***x*** are an "emission" of the latent choice variable

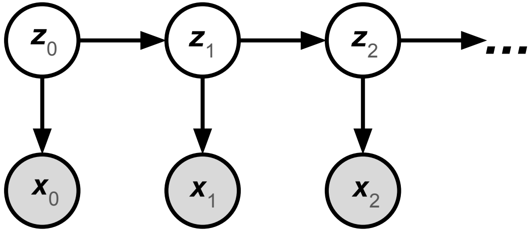


Joint distribution:

$$p(\boldsymbol{x}, \boldsymbol{z} | \theta) = \underbrace{p(\boldsymbol{z}_0)}_{\text{initial state } \pi} \underbrace{\prod_{n=1}^{N} p(\boldsymbol{z}_n | \boldsymbol{z}_{n-1})}_{\text{state transitions } A} \underbrace{\prod_{n=0}^{N} p(\boldsymbol{x}_n | \boldsymbol{z}_n, \theta)}_{\text{likelihoods}}$$

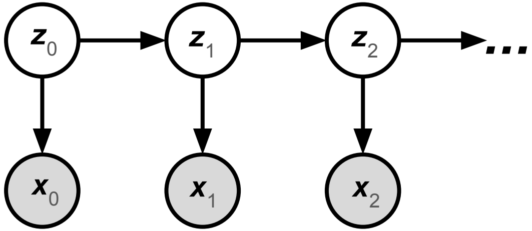
$$\mathcal{N}(\boldsymbol{x}_n; \mu_k, \Sigma_k)$$

EM for Hidden Markov Models



$$p(\boldsymbol{x}, \boldsymbol{z}|\theta) = p(\boldsymbol{z}_0) \prod_{n=1}^{N} p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \prod_{n=0}^{N} p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta)$$

EM for Hidden Markov Models

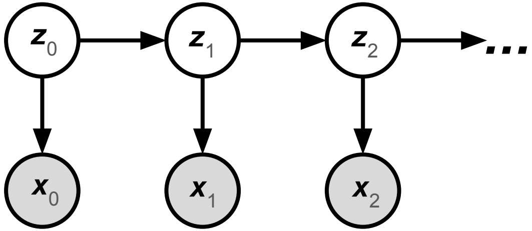


$$p(\boldsymbol{x}, \boldsymbol{z}|\theta) = p(\boldsymbol{z}_0) \prod_{n=1}^{N} p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \prod_{n=0}^{N} p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta)$$

We want to infer the latent states ***z***, and estimate parameters $\theta = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$

E-Step: infer posteriors and calculate initial state and state transition probabilities $\boldsymbol{\pi}, \boldsymbol{A}$

EM for Hidden Markov Models



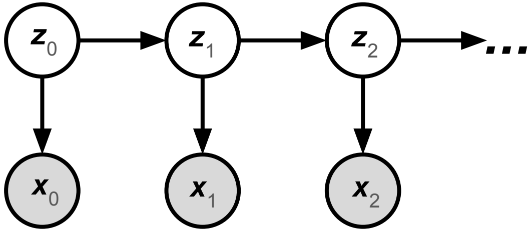
$$p(\boldsymbol{x}, \boldsymbol{z}|\theta) = p(\boldsymbol{z}_0) \prod_{n=1}^{N} p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \prod_{n=0}^{N} p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta)$$

We want to infer the latent states ***z***, and estimate parameters $\theta = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$

E-Step: infer posteriors and calculate initial state and state transition probabilities $\boldsymbol{\pi}, \boldsymbol{A}$

M-Step: update Gaussian parameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ based on complete-data likelihood

EM for Hidden Markov Models



$$p(\boldsymbol{x}, \boldsymbol{z}|\theta) = p(\boldsymbol{z}_0) \prod_{n=1}^{N} p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \prod_{n=0}^{N} p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta)$$

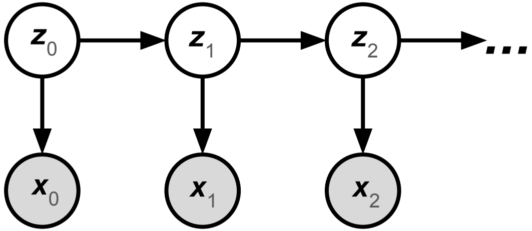
We want to infer the latent states ***z***, and estimate parameters $\theta = \{\boldsymbol{\pi}, \boldsymbol{A}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$

E-Step: infer posteriors and calculate initial state and state transition probabilities $\boldsymbol{\pi}, \boldsymbol{A}$

M-Step: update Gaussian parameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ based on complete-data likelihood

$\rightarrow$ **easy!**

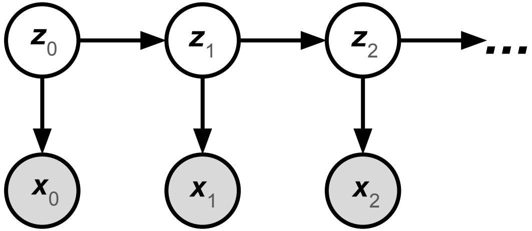
EM for Hidden Markov Models



$$p(\boldsymbol{x}, \boldsymbol{z}|\theta) = p(\boldsymbol{z}_0) \prod_{n=1}^{N} p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \prod_{n=0}^{N} p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta)$$

E-Step: infer posteriors

EM for Hidden Markov Models

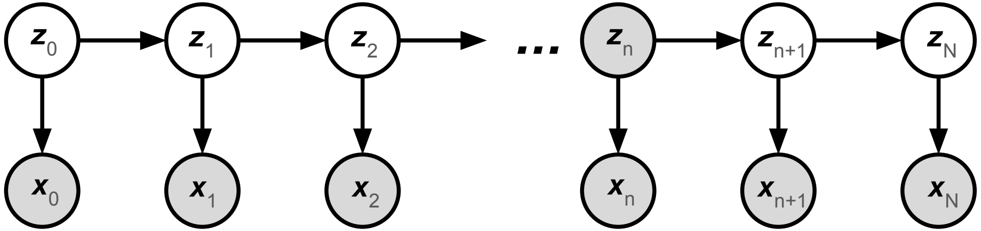


$$p(\boldsymbol{x}, \boldsymbol{z}|\theta) = p(\boldsymbol{z}_0) \prod_{n=1}^{N} p(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) \prod_{n=0}^{N} p(\boldsymbol{x}_n|\boldsymbol{z}_n, \theta)$$

E-Step: infer posteriors
→ In HMMs, there are two posteriors over $\boldsymbol{z}$:

$p(\boldsymbol{z}_n|\boldsymbol{x}, \theta)$	probability of each latent state value
$p(\boldsymbol{z}_n, \boldsymbol{z}_{n-1}|\boldsymbol{x}, \theta)$	probability of observing a pair of subsequent states

EM for Hidden Markov Models

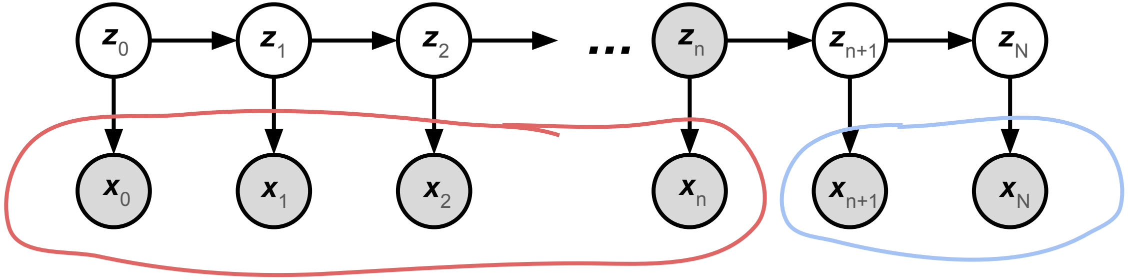


E-Step: infer posteriors

For this, again we use Bayes theorem $p(\boldsymbol{z}_n|\boldsymbol{x}_{0:N},\theta) = \frac{p(\boldsymbol{x}_{0:N}|\boldsymbol{z}_n,\theta)p(\boldsymbol{z}_n)}{p(\boldsymbol{x}_{0:N})}$

(and equivalent for $p(\boldsymbol{z}_n, \boldsymbol{z}_{n-1}|\boldsymbol{x},\theta)$)

EM for Hidden Markov Models



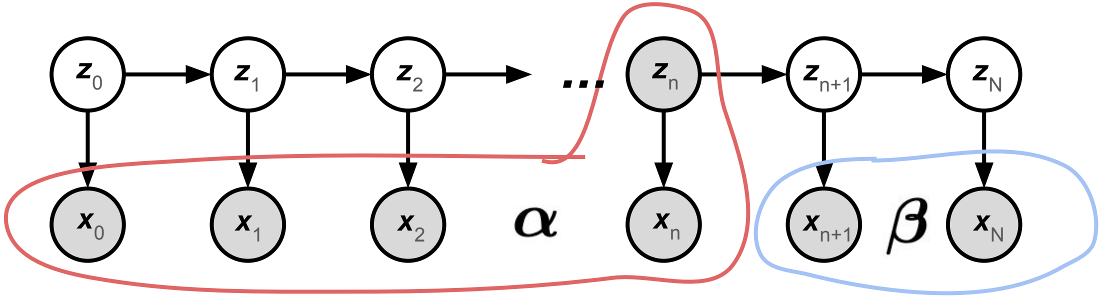
For a given state $\boldsymbol{z}_n$ in trial n, we can split the likelihood in two terms:

$p(\boldsymbol{x}_{0:n}|\boldsymbol{z}_n, \theta)$, $p(\boldsymbol{x}_{n+1:N}|\boldsymbol{z}_n, \theta)$

E-Step: infer posteriors

For this, again we use Bayes theorem $\quad p(\boldsymbol{z}_n|\boldsymbol{x}_{0:N}, \theta) = \dfrac{p(\boldsymbol{x}_{0:N}|\boldsymbol{z}_n, \theta)p(\boldsymbol{z}_n)}{p(\boldsymbol{x}_{0:N})}$

EM for Hidden Markov Models



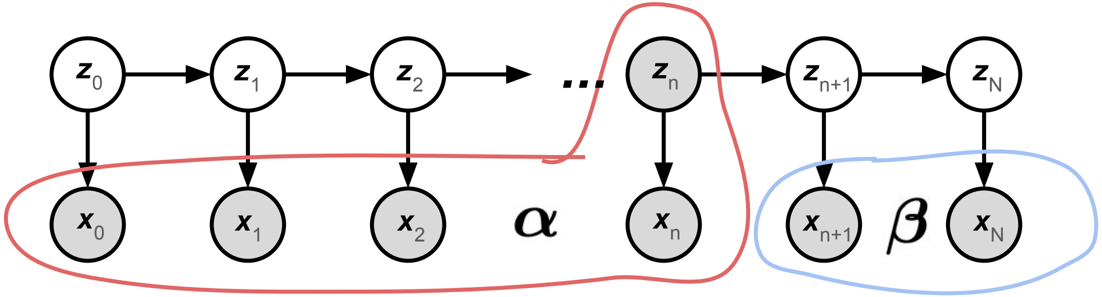
We include $p(\boldsymbol{z}_n)$:

$$\underbrace{p(\boldsymbol{x}_{0:n}, \boldsymbol{z}_n|\theta)}_{\boldsymbol{\alpha}}, \quad \underbrace{p(\boldsymbol{x}_{n+1:N}|\boldsymbol{z}_n, \theta)}_{\boldsymbol{\beta}}$$

E-Step: infer posteriors

For this, again we use Bayes theorem $\quad p(\boldsymbol{z}_n|\boldsymbol{x}_{0:N}, \theta) = \dfrac{p(\boldsymbol{x}_{0:N}|\boldsymbol{z}_n, \theta)p(\boldsymbol{z}_n)}{p(\boldsymbol{x}_{0:N})}$

EM for Hidden Markov Models



We include $p(\boldsymbol{z}_n)$:

$p(\boldsymbol{x}_{0:n}, \boldsymbol{z}_n|\theta)$, $p(\boldsymbol{x}_{n+1:N}|\boldsymbol{z}_n, \theta)$

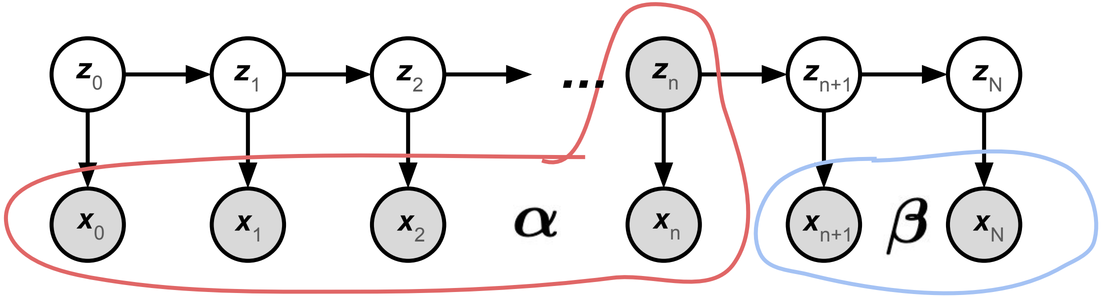
$\boldsymbol{\alpha}$ $\boldsymbol{\beta}$

E-Step: infer posteriors

For this, again we use Bayes theorem $$p(\boldsymbol{z}_n|\boldsymbol{x}_{0:N}, \theta) = \frac{p(\boldsymbol{x}_{0:N}|\boldsymbol{z}_n, \theta)p(\boldsymbol{z}_n)}{p(\boldsymbol{x}_{0:N})}$$

Then the posterior becomes $$p(\boldsymbol{z}_n|\boldsymbol{x}_{0:N}, \theta) = \frac{\boldsymbol{\alpha}(\boldsymbol{z}_n)\boldsymbol{\beta}(\boldsymbol{z}_n)}{p(\boldsymbol{x}_{0:N})}$$

EM for Hidden Markov Models



We include $p(\boldsymbol{z}_n)$:

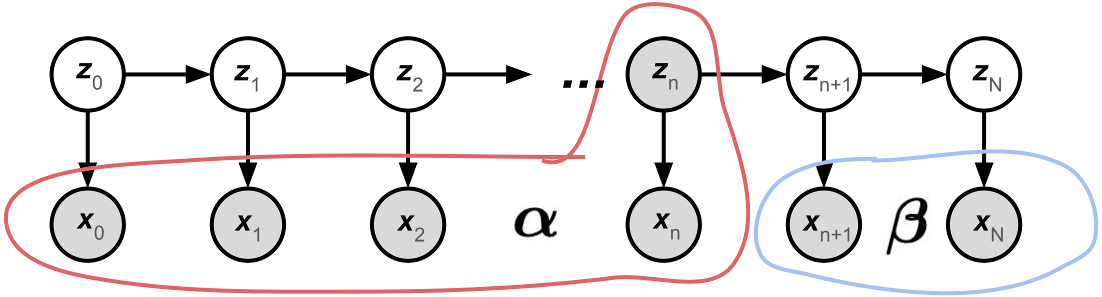
$p(\boldsymbol{x}_{0:n}, \boldsymbol{z}_n|\theta)$, $p(\boldsymbol{x}_{n+1:N}|\boldsymbol{z}_n, \theta)$

$\boldsymbol{\alpha}$ $\boldsymbol{\beta}$

→ **Good news:**

1. There is an efficient algorithm for calculating $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ (the Baum-Welch / forward-backward algorithm)

EM for Hidden Markov Models



We include $p(\boldsymbol{z}_n)$:

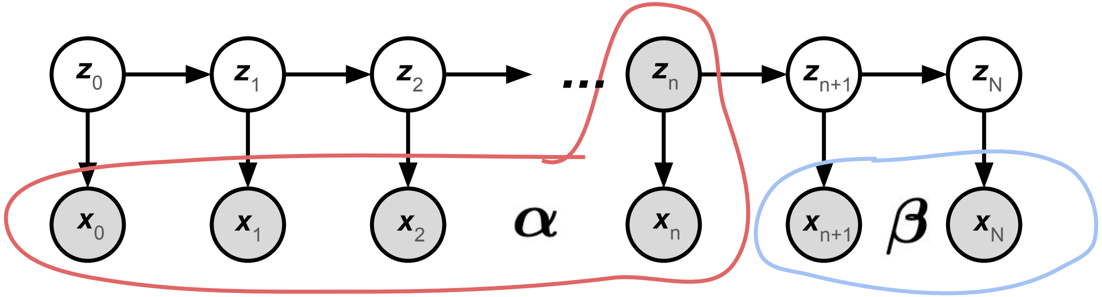
$p(\boldsymbol{x}_{0:n}, \boldsymbol{z}_n | \theta)$, $p(\boldsymbol{x}_{n+1:N} | \boldsymbol{z}_n, \theta)$

$\boldsymbol{\alpha}$ $\boldsymbol{\beta}$

→ **Good news:**

1. There is an efficient algorithm for calculating $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ (the Baum-Welch / forward-backward algorithm)
2. Both posteriors can be calculated from $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

EM for Hidden Markov Models



We include $p(\boldsymbol{z}_n)$:

$p(\boldsymbol{x}_{0:n}, \boldsymbol{z}_n|\theta)$, $p(\boldsymbol{x}_{n+1:N}|\boldsymbol{z}_n, \theta)$

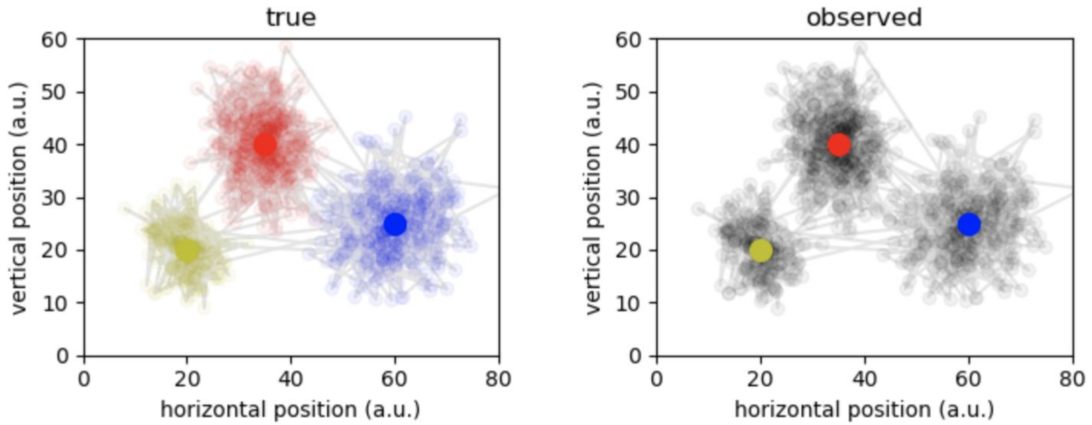
$\boldsymbol{\alpha}$ $\boldsymbol{\beta}$

→ **Good news:**

1. There is an efficient algorithm for calculating $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ (the Baum-Welch / forward-backward algorithm)
2. Both posteriors can be calculated from $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$
3. M-Step is straightforward

Tutorial

- Simulate observations from a generative model



- Use EM with forward-backward algorithm to fit an HMM to simulated data

Closing remarks

Mixture Models

More flexible models are possible! e.g. when target positions change from trial to trial

$$x^{(1)} = x_{saccade} - x_{target} \qquad x^{(2)} = x_{saccade} - x_{distractor}$$

Mixture Models

More flexible models are possible! e.g. when target positions change from trial to trial

$$x^{(1)} = x_{saccade} - x_{target} \qquad x^{(2)} = x_{saccade} - x_{distractor}$$

Mixtures of different distributions (e.g. Gaussian, uniform, Student t… → last exercise)

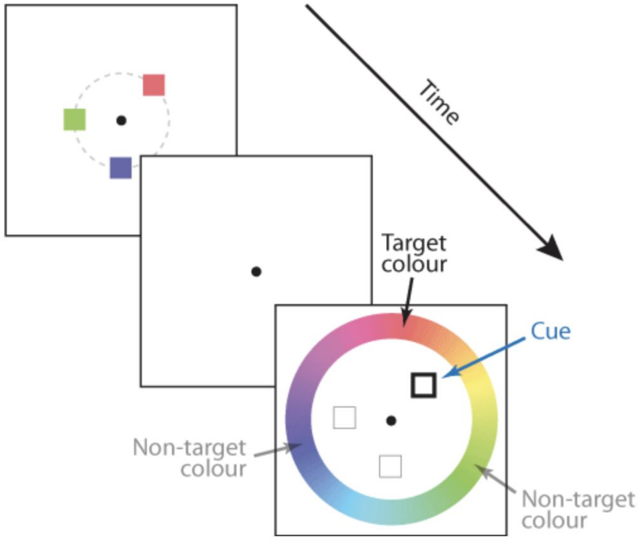
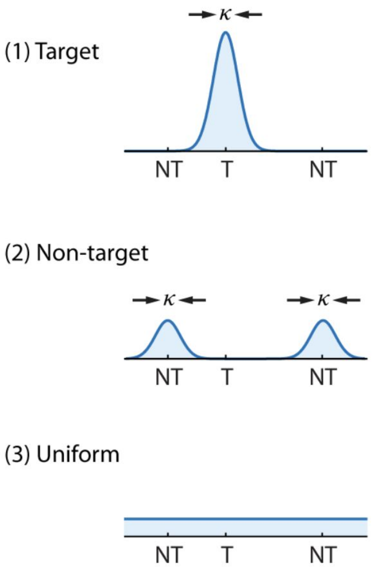


Figure 1 | The colour report task.



Bays, Catalao & Husain, *J Vis* (2009); Schneegans & Bays, *Cortex* (2016)

Remarks on emission models

Gaussian emission models: $p(\boldsymbol{x}|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}|\mu_k, \Sigma_k)$

Other (including more complex) emission models are possible:

- Student t (see tutorial), Poisson, etc… (continuous observations ***x***)
- Categorical emissions (discrete observations ***x***)

Remarks on emission models

Gaussian emission models: $p(\boldsymbol{x}|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}|\mu_k, \Sigma_k)$

Other (including more complex) emission models are possible:

- Student t (see tutorial), Poisson, etc… (continuous observations ***x***)
- Categorical emissions (discrete observations ***x***)

- Linear models of input ***u***: $p(\boldsymbol{x}|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{W}_k\boldsymbol{u} + c_k, \Sigma_k)$

- Autoregressive models: $p(\boldsymbol{x}_n|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{A}_k\boldsymbol{x}_{n-1} + d_k, \Sigma_k)$

Remarks on emission models

Gaussian emission models: $p(\boldsymbol{x}|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}|\mu_k, \Sigma_k)$

Other (including more complex) emission models are possible:

- Student t (see tutorial), Poisson, etc… (continuous observations ***x***)
- Categorical emissions (discrete observations ***x***)

- Linear models of input ***u***: $p(\boldsymbol{x}|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{W}_k\boldsymbol{u} + c_k, \Sigma_k)$

- Autoregressive models: $p(\boldsymbol{x}_n|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{A}_k\boldsymbol{x}_{n-1} + d_k, \Sigma_k)$

- Combine them, you get switching DDMs

-

Remarks on emission models

Gaussian emission models: $$p(\boldsymbol{x}|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}|\mu_k, \Sigma_k)$$

Other (including more complex) emission models are possible:

- Student t (see tutorial), Poisson, etc… (continuous observations $\boldsymbol{x}$)
- Categorical emissions (discrete observations $\boldsymbol{x}$)

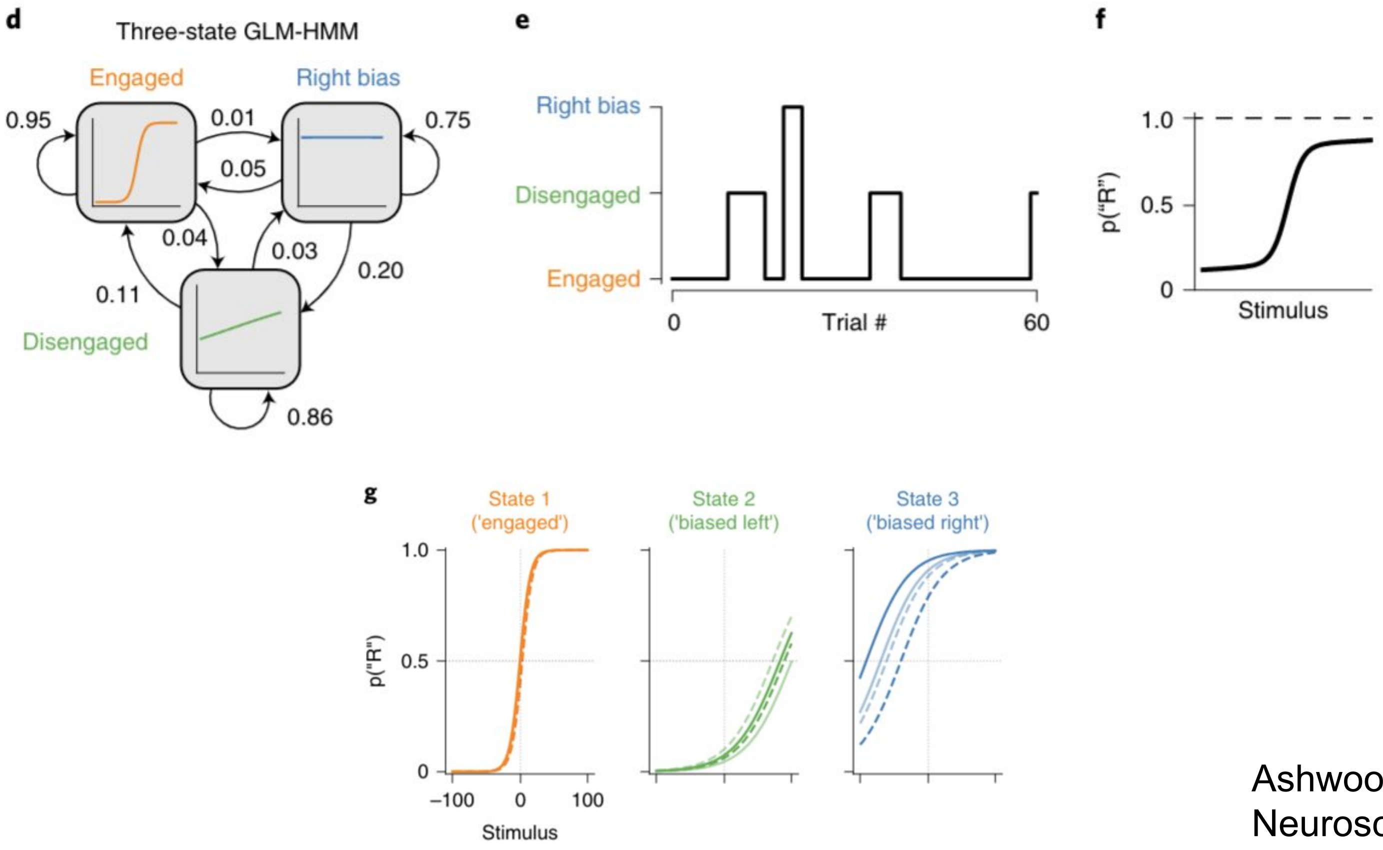
- Linear models of input $\boldsymbol{u}$: $$p(\boldsymbol{x}|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{W}_k\boldsymbol{u} + c_k, \Sigma_k)$$

- Autoregressive models: $$p(\boldsymbol{x}_n|z_k = 1, \theta_k) = \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{A}_k\boldsymbol{x}_{n-1} + d_k, \Sigma_k)$$

- Combine them, you get switching DDMs

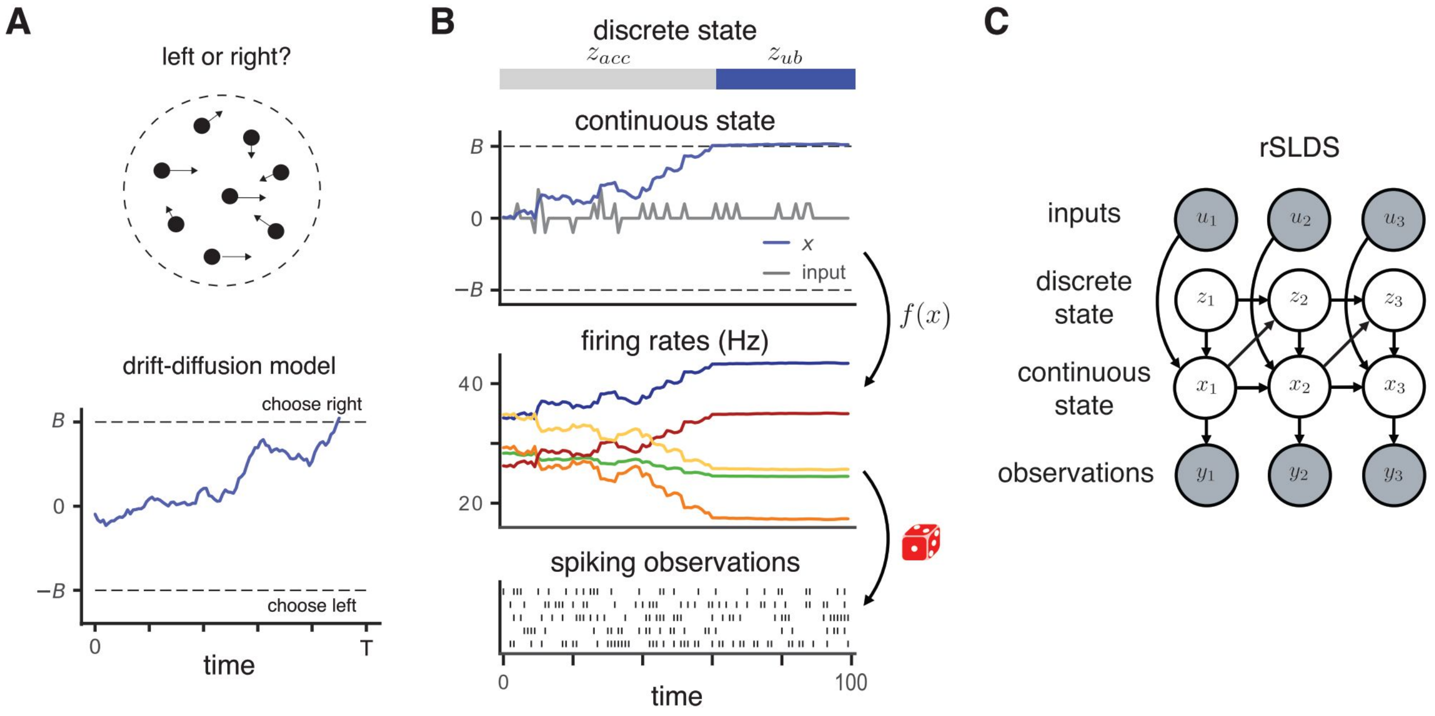
- Switching factor analysis (if you include an extra set of continuous latents that depend on the state)

Examples of HMMs in neuroscience



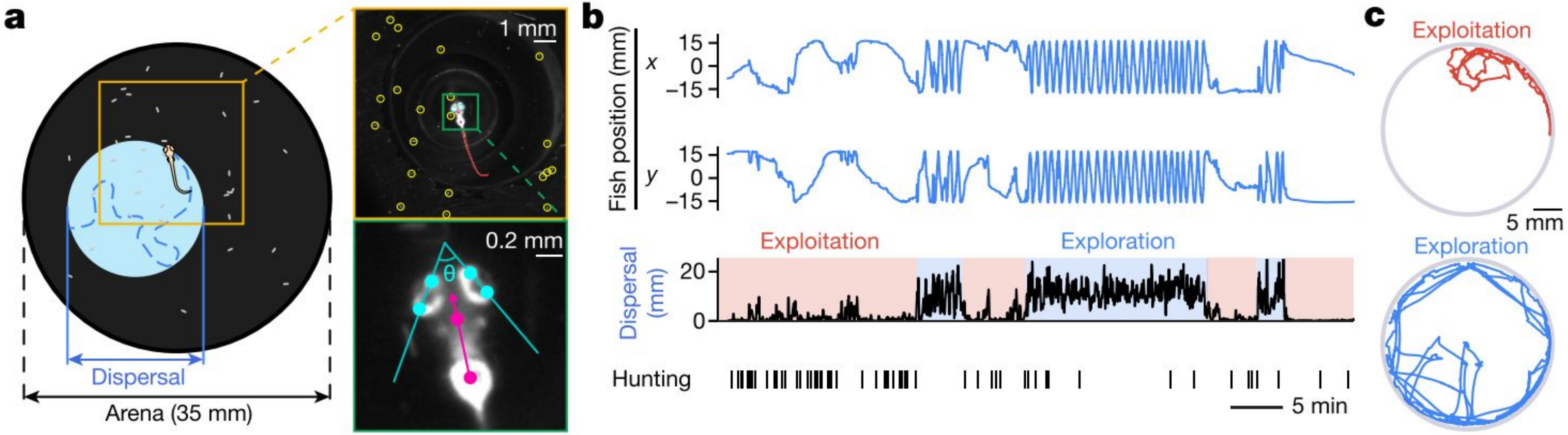
Ashwood et al., Nat Neurosci, 2022

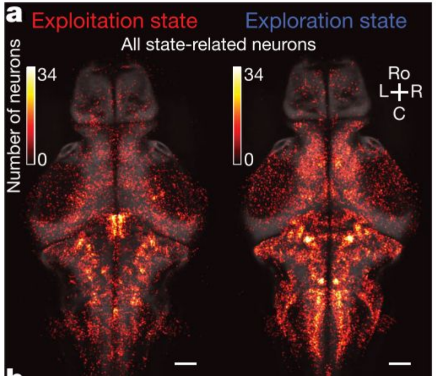
Examples of HMMs in neuroscience

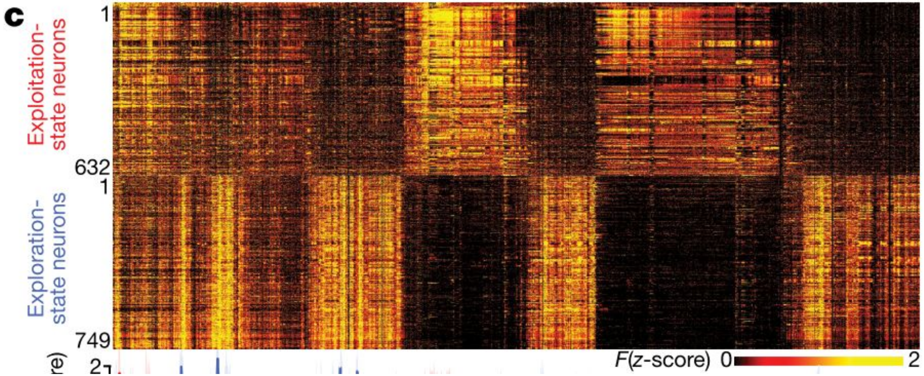


Zoltowski et al., ICML, 2020

Examples of HMMs in neuroscience







Marques et al., Nature, 2019

Examples of HMMs in neuroscience

Wiltschko et al., Neuron, 2015