Bayesian Learning in Linear Dynamical Systems

Bergamin E., Campagnola S., Capodaglio A., Nagaro Ortiz G., Valentini L.
September 30, 2023

Università degli Studi di Padova

Bibliography I

E. Fox, E. Sudderth, M. Jordan, and A. Willsky.
Nonparametric bayesian learning of switching linear dynamical systems.
volume 21, pages 457–464, 01 2008.

S. W. Linderman, A. C. Miller, R. P. Adams, D. M. Blei, L. Paninski, and M. J. Johnson.
Recurrent switching linear dynamical systems, 2016.

We are interested in a dynamical system, whose dynamics are influenced by a certain set of latent states.
The observable behaviour of the system depends on the underlying latent state.

We are interested in a dynamical system, whose dynamics are influenced by a certain set of latent states.
The observable behaviour of the system depends on the underlying latent state.
The latent states themselves change with markovian dynamics.
The generative model is ruled by the following laws [2]:

We are interested in a dynamical system, whose dynamics are influenced by a certain set of latent states.
The observable behaviour of the system depends on the underlying latent state.
The latent states themselves change with markovian dynamics.
The generative model is ruled by the following laws [2]:

$$z_{t+1} \sim \mathrm{Cat}(\mathbf{M}_{z_t})$$

$$\begin{aligned}\mathbf{y}_{t+1} &= A_{z_{t+1}}\mathbf{y}_t + \mathbf{b}_{z_{t+1}} + \mathbf{v}_{z_{t+1}} \\ &= \hat{A}_{z_{t+1}}\hat{\mathbf{y}}_t + \mathbf{v}_{z_{t+1}}\end{aligned}$$

$$\mathbf{v}_{z_{t+1}} \sim \mathcal{N}(\mathbf{0}, Q_{z_{t+1}})$$

- $z_t \in \{1, 2, ..., K\}$ are the discrete latent states
- $\mathbf{M}_k$ is the k'th row of the transition matrix M
- Q_k is the noise covariance matrix
- A_k is the linear transformation matrix
- $\mathbf{b}_k$ is the bias vector
- $\hat{A}_k$ is the concatenation $[\mathbf{b}_k, A_k]$
- $\hat{\mathbf{y}}_t$ is the concatenation $[1, \mathbf{y}_t]$

The likelihood function for the problem can be expressed in two equivalent formulations.

If $\mathbf{y}_{1:T}$ is a certain trajectory, and $z_{1:T}$ are the underlying latent states:

The likelihood function for the problem can be expressed in two equivalent formulations.
If $\mathbf{y}_{1:T}$ is a certain trajectory, and $z_{1:T}$ are the underlying latent states:

$$P(\mathbf{y}_{1:T}|z_{1:T},\{\hat{A}_k\},\{Q_k\},M) =$$

1

$$= \prod_{t=1}^{T-1}(2\pi)^{-\frac{N}{2}}|Q_{z_{t+1}}|^{-\frac{1}{2}} \cdot \exp\left\{\left[-\frac{1}{2}(\mathbf{y}_{t+1}-\hat{A}_{z_{t+1}}\hat{\mathbf{y}}_t)^T Q_{z_{t+1}}^{-1}(\mathbf{y}_{t+1}-\hat{A}_{z_{t+1}}\hat{\mathbf{y}}_t)\right]\right\}$$

2

$$= \prod_{k=1}^{K}(2\pi)^{-\frac{N \cdot N_k}{2}}|Q_k|^{-\frac{N_k}{2}} \cdot \exp\left\{\left[-\frac{1}{2}\sum_{t_i \in k}^{N_k}(\mathbf{y}_{t_i}-\hat{A}_k\hat{\mathbf{y}}_{t_i-1})^T Q_k^{-1}(\mathbf{y}_{t_i}-\hat{A}_k\hat{\mathbf{y}}_{t_i-1})\right]\right\}$$

AR-HMM Inference: Prior

The joint probability distribution for all the variables can be written as:

$$
\begin{aligned}
P(\mathbf{y}_{1:T}, z_{1:T}, \{\hat{A}_k\}, \{Q_k\}, M) =& P(\mathbf{y}_{1:T} | z_{1:T}, \{\hat{A}_k\}, \{Q_k\}, M) \cdot P(z_{1:T} | M) \cdot \\
& \cdot P(M) \cdot P(\{\hat{A}_k\}, \{Q_k\})
\end{aligned}
$$

AR-HMM Inference: Prior

The joint probability distribution for all the variables can be written as:

$$P(\mathbf{y}_{1:T}, z_{1:T}, \{\hat{A}_k\}, \{Q_k\}, M) = P(\mathbf{y}_{1:T} | z_{1:T}, \{\hat{A}_k\}, \{Q_k\}, M) \cdot P(z_{1:T} | M) \cdot \\ \cdot P(M) \cdot P(\{\hat{A}_k\}, \{Q_k\})$$

The probability of observing a given latent history is:

The joint probability distribution for all the variables can be written as:

$$P(\mathbf{y}_{1:T}, z_{1:T}, \{\hat{A}_k\}, \{Q_k\}, M) = P(\mathbf{y}_{1:T} | z_{1:T}, \{\hat{A}_k\}, \{Q_k\}, M) \cdot P(z_{1:T}|M) \cdot \\ \cdot P(M) \cdot P(\{\hat{A}_k\}, \{Q_k\})$$

The probability of observing a given latent history is:

$$P(z_{1:T}|M) = P(z_1) \cdot \prod_{t=1}^{T-1} M_{z_t, z_{t+1}} = \frac{1}{K} \cdot \prod_{t=1}^{T-1} M_{z_t, z_{t+1}}$$

where we have chosen an **uninformative prior** on the state z_1.

AR-HMM Inference: Prior (2)

Conjugate priors for the model parameters can be chosen, in order to simplify the inference task.
Indeed, the Dirichlet distribution is the conjugate prior of the categorical distribution, and the Inverse Wishart distribution is the conjugate prior for the covariance matrix of a multivariate normal distribution.

- $$P(M) = \prod_{k=1}^{K} P(\mathbf{M}_k|\boldsymbol{\alpha}_k) = \prod_{k=1}^{K} \frac{1}{B(\boldsymbol{\alpha}_k)} \prod_{l=1}^{K} M_{kl}^{\alpha_{kl}-1}$$

AR-HMM Inference: Prior (2)

Università degli Studi di Padova

Conjugate priors for the model parameters can be chosen, in order to simplify the inference task.
Indeed, the Dirichlet distribution is the conjugate prior of the categorical distribution, and the Inverse Wishart distribution is the conjugate prior for the covariance matrix of a multivariate normal distribution.

- $$P(M) = \prod_{k=1}^{K} P(\mathbf{M}_k | \boldsymbol{\alpha}_k) = \prod_{k=1}^{K} \frac{1}{B(\boldsymbol{\alpha}_k)} \prod_{l=1}^{K} M_{kl}^{\alpha_{kl}-1}$$

- $$P(\{\hat{A}_k\}, \{Q_k\}) = \prod_{k=1}^{K} P(\hat{A}_k, Q_k) = \prod_{k=1}^{K} \mathcal{MNIW}(\hat{A}_k, Q_k; C, V, S, \nu)$$
$$= \prod_{k=1}^{K} \mathcal{MN}(\hat{A}_k; C, Q_k, V) \cdot \mathcal{IW}(Q_k; S, \nu)$$

AR-HMM Inference: Hyperparameters

Università degli Studi di Padova

Hyperparameters choice has to reflect the possessed knowledge that, in this case, is very limited.
For this reason uninformative values are used:

AR-HMM Inference: Hyperparameters

Hyperparameters choice has to reflect the possessed knowledge that, in this case, is very limited.
For this reason uninformative values are used:

- $\boldsymbol{\alpha}_k = [1, ..., 1] \quad \forall k = 1, ..., K$
  concentration parameters for Dirichlet distribution
- $C = [\vec{\mathbf{0}}, \mathbb{1}_N]$
  mean matrix
- $V = \mathbb{1}_{N+1}$
  inverse of the covariance matrix along columns
- $S = \text{diag}(\lambda_1, ..., \lambda_N)$
  scale matrix associated to IW distribution, defines the scale of the expected data variability
- $\nu = N$
  degrees of freedom associated to IW distribution

- The objective is to perform Gibbs sampling for this kind of AR-HMM.
- We have to compute the conditional probabilities of each random variable, keeping fixed all the others.

AR-HMM: Message Passing

In order to sample the history of the latent states $z_{1:T}$ it's possible to develop a message passing algorithm. [1]

In order to sample the history of the latent states $z_{1:T}$ it's possible to develop a message passing algorithm. [1]
The starting step is the following:

$$P(z_{1:T}|\{\hat{A}_k\},\{Q_k\},M,\mathbf{y}_{1:T}) = P(z_1|\{\hat{A}_k\},\{Q_k\},M,\mathbf{y}_{1:T})$$

$$\cdot \prod_{t=2}^{T} P(z_t|z_{t-1},\{\hat{A}_k\},\{Q_k\},M,\mathbf{y}_{1:T}) \quad (1)$$

In order to sample the history of the latent states $z_{1:T}$ it's possible to develop a message passing algorithm. [1]
The starting step is the following:

$$P(z_{1:T}|\{\hat{A}_k\},\{Q_k\},M,\mathbf{y}_{1:T}) = P(z_1|\{\hat{A}_k\},\{Q_k\},M,\mathbf{y}_{1:T}) \cdot \prod_{t=2}^{T} P(z_t|z_{t-1},\{\hat{A}_k\},\{Q_k\},M,\mathbf{y}_{1:T}) \tag{1}$$

Sampling z_1, thanks to (1), one can proceed along the history of latent states and determine $(z_2|z_1)$, $(z_3|z_2)$, ...

AR-HMM: Message Passing (2)

Single conditional distributions can be iteratively computed from the message vector $\mathbf{m}_{t,t-1}$, whose k-th component is $m_{t,t-1}(z_{t-1} = k)$ defined as:

$$m_{t,t-1}(z_{t-1} = k) \propto \begin{cases} \sum_{j=1}^{K} M_{k,j} \cdot \mathcal{N}(\mathbf{y}_t - \hat{A}_j \hat{\mathbf{y}}_{t-1}, Q_j) \cdot m_{t+1,t}(z_t = j), & t \leq T \\ 1, \qquad t = T+1 \end{cases}$$

Each z_t can be sampled categorically from:

$$P(z_t = k | \{\hat{A}_k\}, \{Q_k\}, M, \mathbf{y}_{1:T}) \propto M_{z_{t-1},k} \cdot \mathcal{N}(\mathbf{y}_t - \hat{A}_k \hat{\mathbf{y}}_{t-1}, Q_k) \cdot m_{t+1,t}(z_t = k)$$

AR-HMM: Model parameters sampling

In order to sample the model parameters M, $\{\hat{A}_k\}$, $\{Q_k\}$, conditional distribution are found to follow the given proportionalities [1]:

In order to sample the model parameters M, $\{\hat{A}_k\}$, $\{Q_k\}$, conditional distribution are found to follow the given proportionalities [1]:

- $$P(\mathbf{M}_k | z_{1:T}, \{\hat{A}_k\}, \{Q_k\}, \mathbf{y}_{1:T}) \propto \prod_{t=1}^{T-1} \chi(z_t = k) \cdot M_{k,z_{t+1}} \cdot \prod_{l=1}^{K} M_{kl}^{\alpha_{kl}-1}$$
  $$\propto \prod_{l=1}^{K} M_{kl}^{\alpha_{kl} + \sum_t^{T-1} \chi(z_t = k)\chi(z_{t+1}=l) \ -1}$$
  $$\propto \mathrm{Dir}(\mathbf{M}_k | \boldsymbol{\alpha}_k + \boldsymbol{\chi}_k)$$

- $$P(Q_k | z_{1:T}, \hat{A}_k, M, \mathbf{y}_{1:T}) \propto \mathcal{IW}(Q_k; S + S_{y|\bar{y}}^{(k)}, \nu + N_k)$$

- $$P(\hat{A}_k | z_{1:T}, Q_k, M, \mathbf{y}_{1:T}) \propto \mathcal{MN}(\hat{A}_k | S_{y\bar{y}}^{(k)} S_{\bar{y}\bar{y}}^{(k)-1}, Q_k, S_{\bar{y}\bar{y}}^{(k)-1})$$

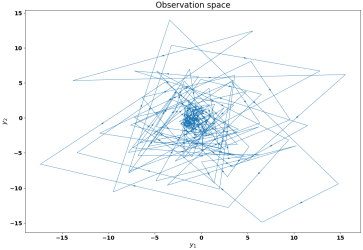
Università degli Studi di Padova

As a first climatization to this problem and framework we decided to generate some simple data:

```
# initial conditions
z[0] = int(np.random.randint(K)) # random number from 0 to K-1
prob_z[0, z[0]] = 1 # set the state probabilities to [0,0,...,1,..
y[0] = np.random.randn(N) # start with random point in observation

# time evolution of the dynamic system
for t in range(1,T):

    prob_z[t] = np.dot(prob_z[t-1].T, M).T
    z[t] = int(np.random.choice(K, p = M[z[t-1]]))
    nu = np.random.multivariate_normal(np.zeros(N), Q[z[t]])
    y[t] = np.dot(A[z[t]], y[t-1]) + b[z[t]] + nu
```



When performing inference on the described model we had to overcome one main problem:

- **Permutation Problem:** Inferred labels can be different from the true ones, because Gibbs sampler doesn't know which is the right 'name' to assign

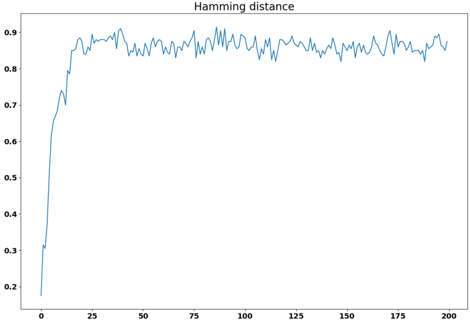
True Label:	**[0 1 1 1 4 4 3 1 1 2]**
Inferred Label:	**[4 2 2 2 3 3 1 2 2 0]**

Table: An example of the permutation problem.

- **Solution:** match inferred $\hat{A}_k$ with the true ones, depending on their Frobenius distance, and re-assign the labels

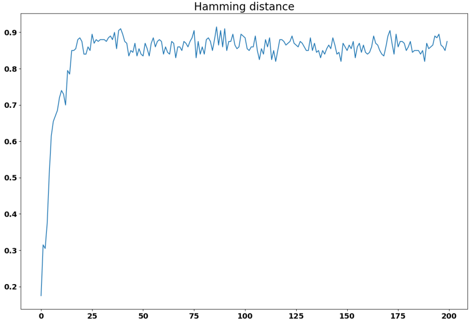
Toy Model : Inference (2)

Similarity between inferred latent history and the true one is measured in terms of normalized Hamming distance.



- Moving forward with Gibbs sampler iterations, we can see that similarity increases.

Similarity between inferred latent history and the true one is measured in terms of normalized Hamming distance.



- Moving forward with Gibbs sampler iterations, we can see that similarity increases.
- Expected values from the posterior are computed as the average of the last half of the Gibbs samples.

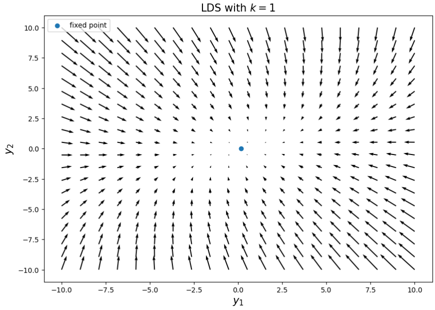


Figure: True dynamics. $K = 1$

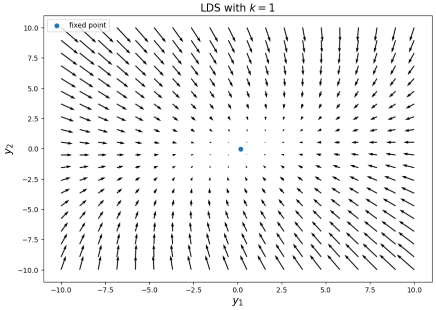


Figure: Inferred dynamics. $K = 1$

Toy model: True vs Inferred

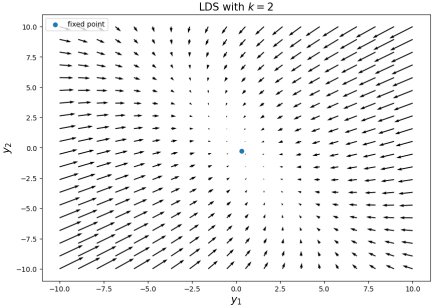


Figure: True dynamics. $K = 2$

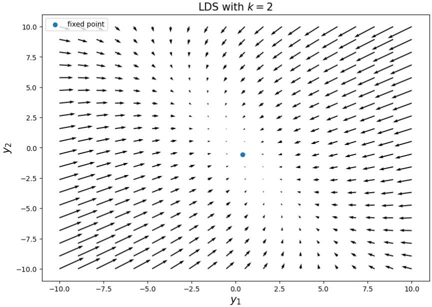


Figure: Inferred dynamics. $K = 2$

Toy model: True vs Inferred

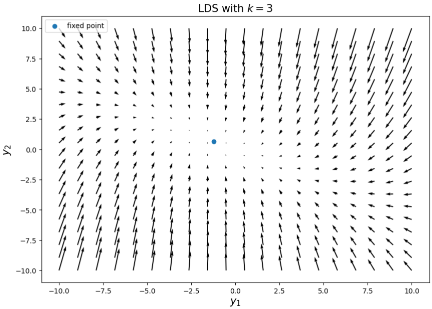


Figure: True dynamics. $K = 3$

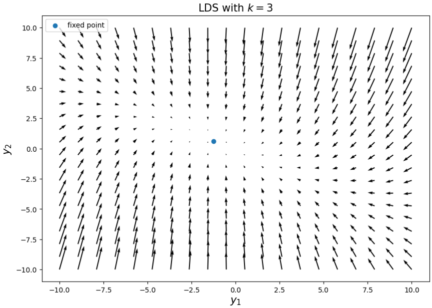


Figure: Inferred dynamics. $K = 3$

Toy model: True vs Inferred

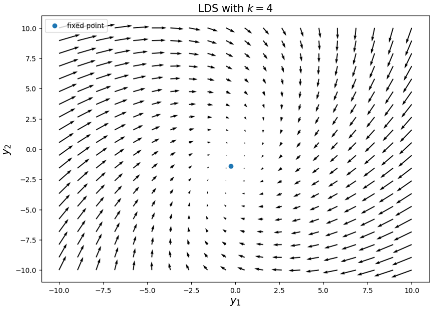


Figure: True dynamics. $K = 4$

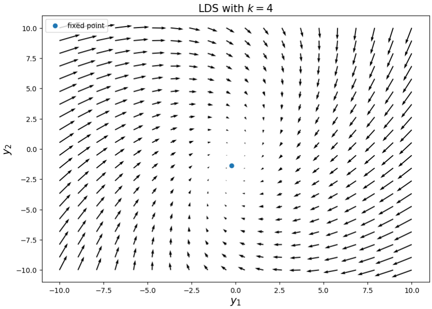


Figure: Inferred dynamics. $K = 4$

Toy model: True vs Inferred

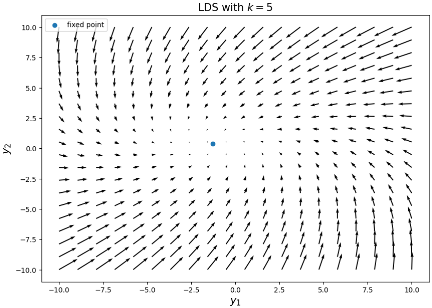


Figure: True dynamics. $K = 5$

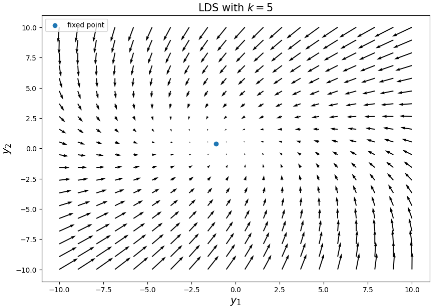


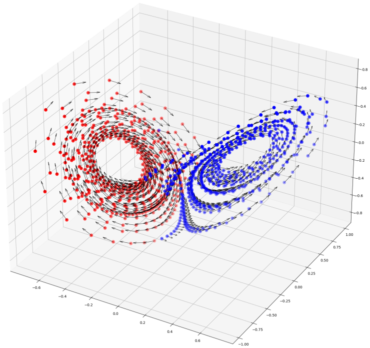
Figure: Inferred dynamics. $K = 5$

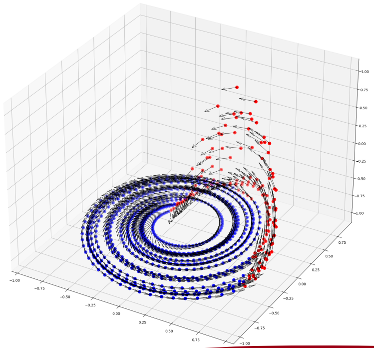
3D attractors

We also dipped our toes in the task with different sets of differential equations.

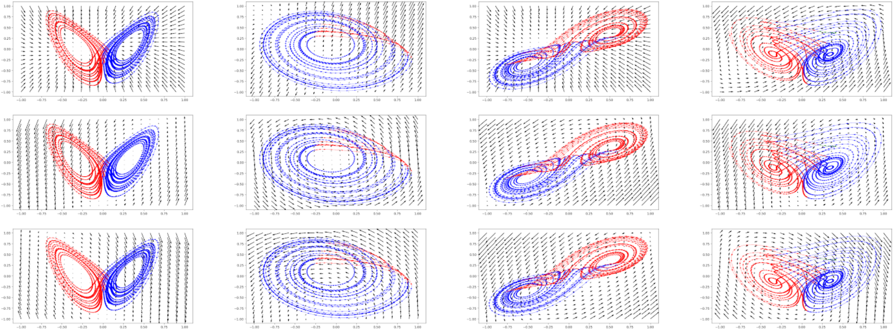
Lorenz Attractor: $$\begin{cases} \frac{dx}{dt} = \sigma(y - x) \\ \frac{dy}{dt} = x(\rho - z) - y \\ \frac{dz}{dt} = xy - \beta z \end{cases}$$

Rössler attractor: $$\begin{cases} \frac{dx}{dt} = -y - z \\ \frac{dy}{dt} = x + ay \\ \frac{dz}{dt} = b + z(x - c) \end{cases}$$

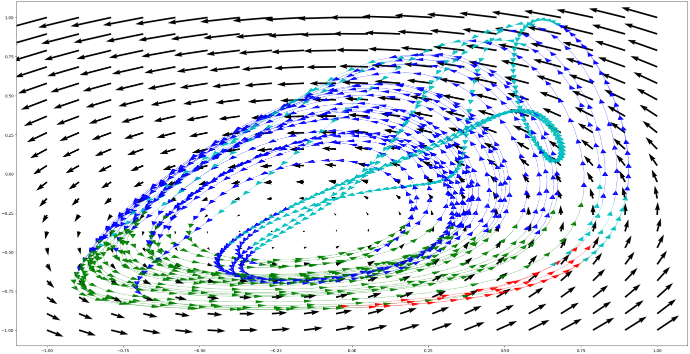


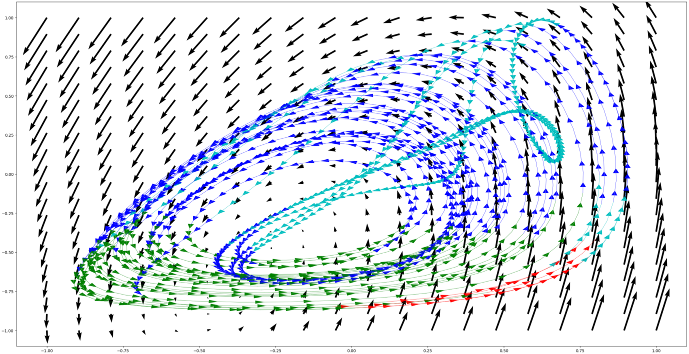


2D projections of 3D attractors

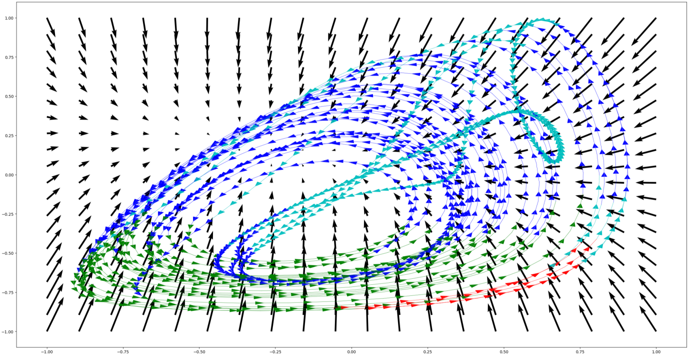


2D projections of 3D attractors - 4 States

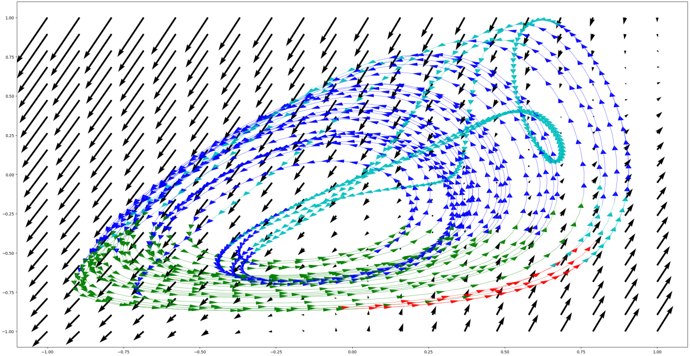



1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00

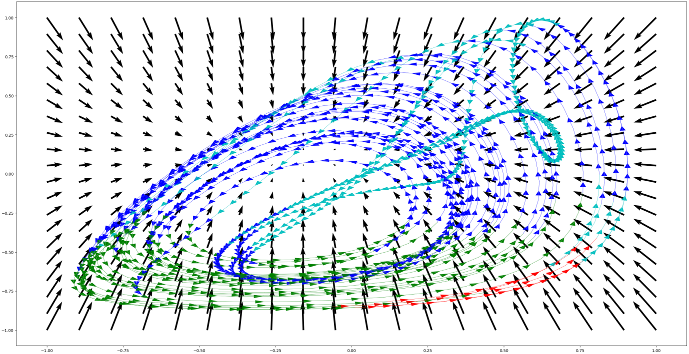
2D projections of 3D attractors - 4 States



2D projections of 3D attractors - 4 States



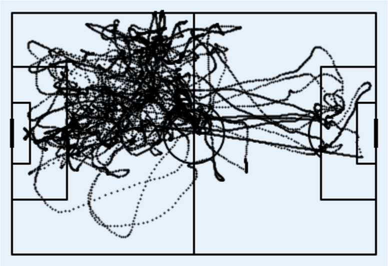
2D projections of 3D attractors - 4 States



Real case scenario: football data

data source: https://kloppy.pysport.org/getting-started/metrica/

Team	Type	Subtype	Period	Start Frame	Start Time [s]	End Frame	End Time [s]	From	To	Start X	Start Y	End X	End Y
Away	SET PIECE	KICK OFF	1	1	0.04	0	0	Player19		NaN	NaN	NaN	NaN
Away	PASS		1	1	0.04	3	0.12	Player19	Player21	0.45	0.39	0.55	0.43
Away	PASS		1	3	0.12	17	0.68	Player21	Player15	0.55	0.43	0.58	0.21
Away	PASS		1	45	1.8	61	2.44	Player15	Player19	0.55	0.19	0.45	0.31
Away	PASS		1	77	3.08	96	3.84	Player19	Player21	0.45	0.32	0.49	0.47
Away	PASS		1	191	7.64	217	8.68	Player21	Player22	0.4	0.73	0.32	0.98
Away	PASS		1	279	11.16	303	12.12	Player22	Player17	0.39	0.96	0.49	0.98
Away	BALL LOST	INTERCEPTION	1	346	13.84	380	15.2	Player17		0.51	0.97	0.27	0.75
Home	RECOVERY	INTERCEPTION	1	378	15.12	378	15.12	Player2		0.27	0.78	NaN	NaN
Home	BALL LOST	INTERCEPTION	1	378	15.12	452	18.08	Player2		0.27	0.78	0.59	0.64



Plot of players' trajectories on the pitch. The table above represents the data linked to the trajectories of the players.

Single player trajectory - Data

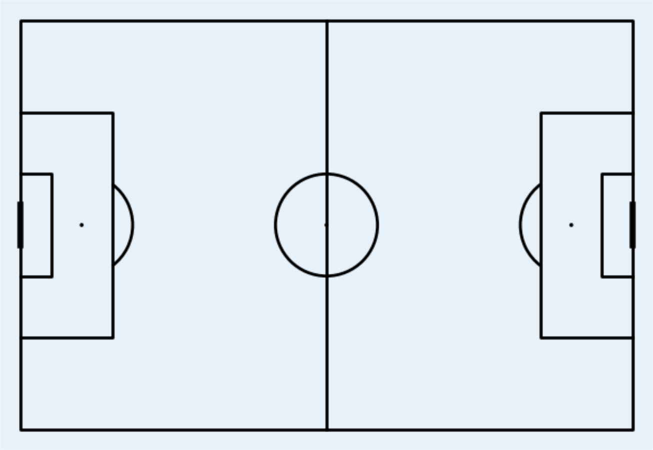


Figure: Trajectory of player 0.

Single player trajectory - Inferred data

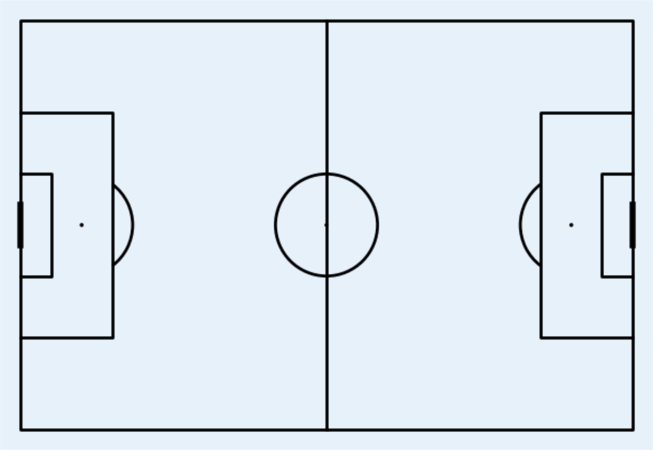


Figure: Inferred states of player 0.

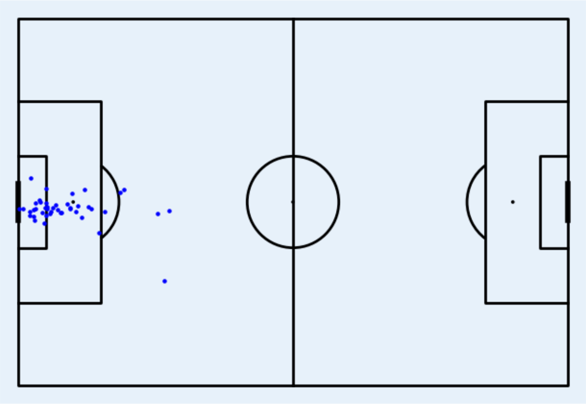


Figure: All the instances of player 10 in state 0.

GIFs - Goaly (state 1)

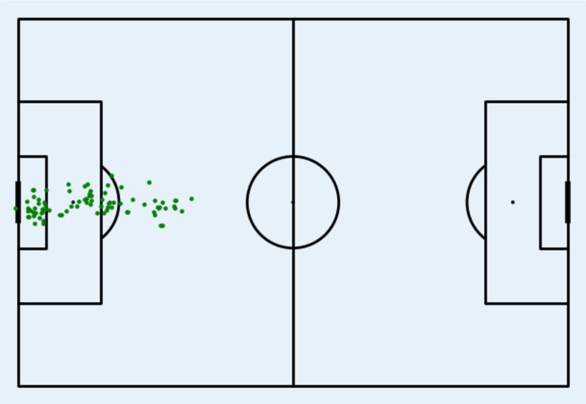


Figure: All the instances of player 10 in state 1.

GIFs - Goaly (state 2)

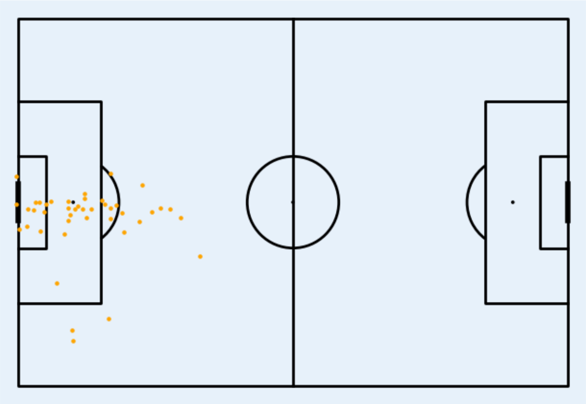


Figure: All the instances of player 10 in state 2.

GIFs - Goaly (state 3)

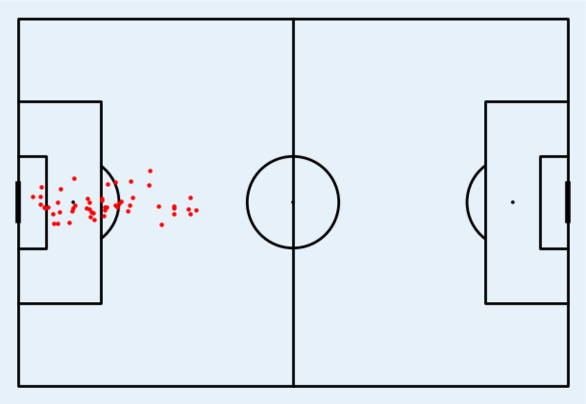


Figure: All the instances of player 10 in state 3.

GIFs - Goaly (state 4)

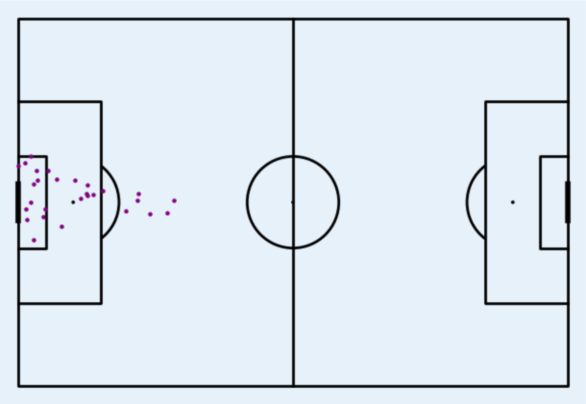


Figure: All the instances of player 10 in state 4.

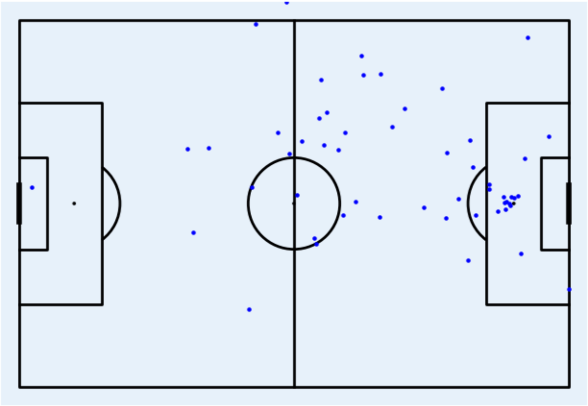


Figure: All the instances of player 8 in state 0.

GIFs - Player 8 (state 1)

Università degli Studi di Padova

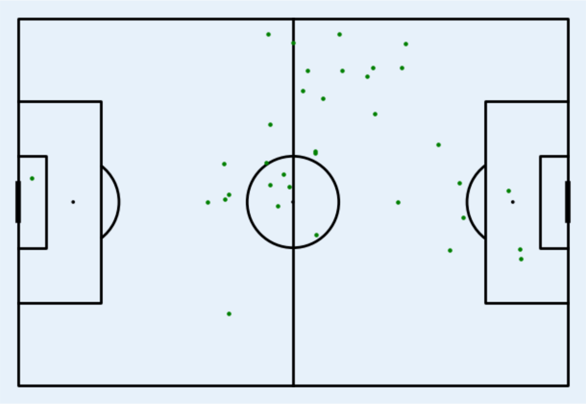


Figure: All the instances of player 8 in state 1.

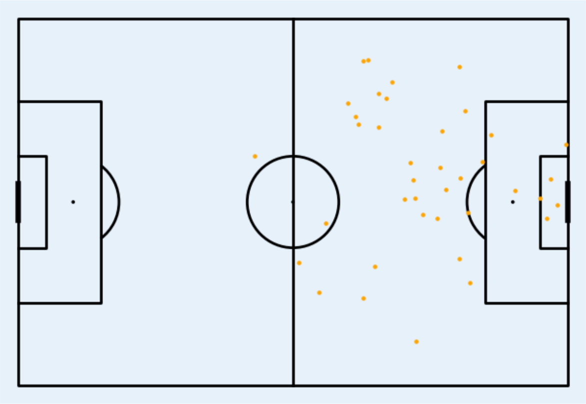


Figure: All the instances of player 8 in state 2.

GIFs - Player 8 (state 3)

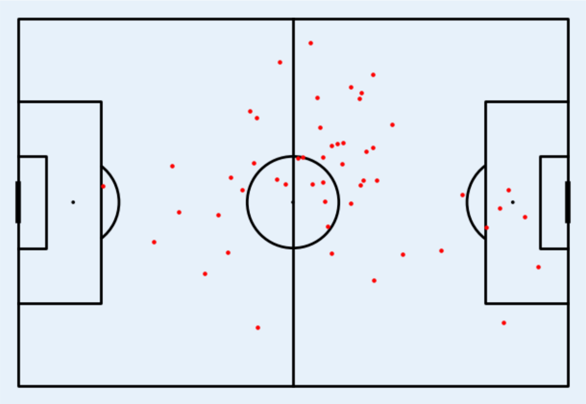


Figure: All the instances of player 8 in state 3.

GIFs - Player 8 (state 4)

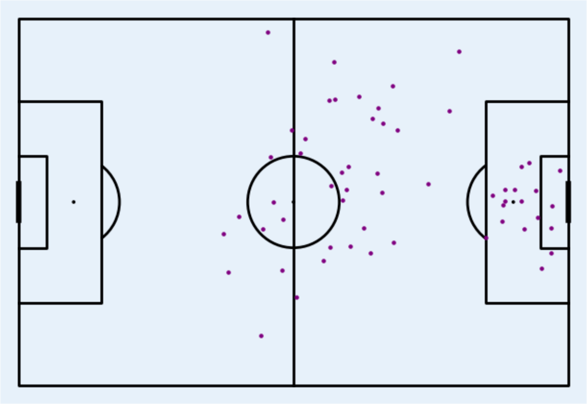


Figure: All the instances of player 8 in state 4.

GIFs - Defenders Inference

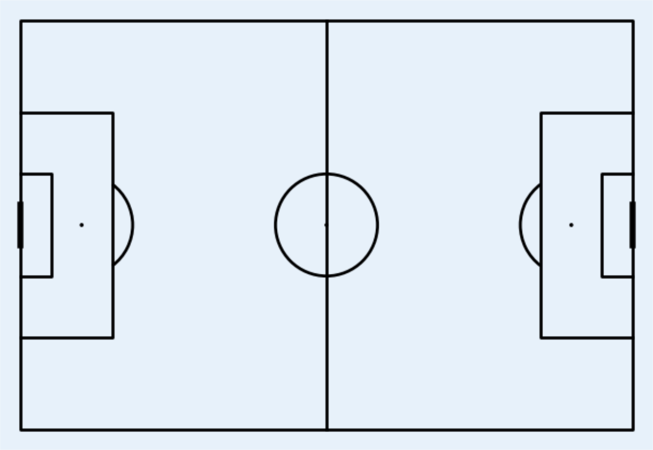


Figure: Inference performed on defenders.

GIFs - Team Inference

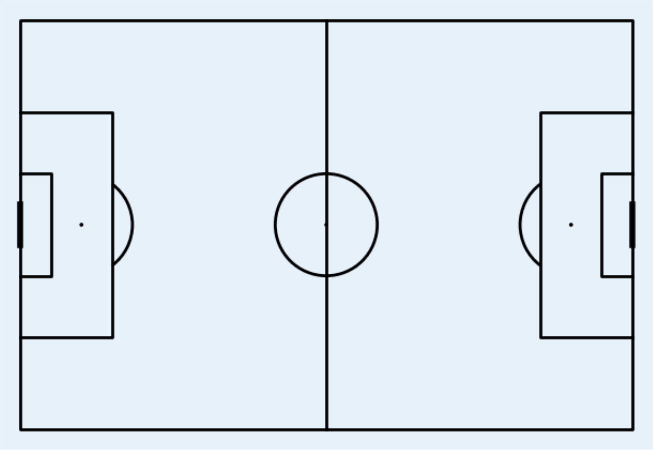


Figure: Inference performed on the whole team

Events Analysis


```python
from kloppy import metrica
events = pd.read_csv("./Datasets/Sample_Game_1_RawEventsData.csv")

events = events[events["Period"]==1]
events = events[events["Team"]=="Home"].reset_index(drop=True)
events['state'] = 0
events['nature'] = "_"
Attack = ['BALL LOST' , 'SHOT', 'FAULT RECEIVED']
Defense = ['RECOVERY','CARD']
Neutral = ["PASS",'CHALLENGE','SET PIECE','BALL OUT']
for event in events.iterrows():
    if event[1].Type in Attack:
        events.at[event[0],'nature'] =  "Attack"
    elif event[1].Type in Defense:
        events.at[event[0],'nature'] =  "Defense"
    elif event[1].Type in Neutral:
        if event[1]["Start X"] > 0.66:
            events.at[event[0],'nature'] =  "Attack"
        elif event[1]["Start X"] < 0.33:
            events.at[event[0],'nature'] =  "Defense"
        else: events.at[event[0],'nature'] =  "Neutral"
```

```
individual_dfs = []
state_duration = []
for k in range(len(best_z)):
    events_player = events[events["From"]=="Player"+(str(k+1)) ].reset_index(drop=True)
    for event in range(len(events_player)):
        frame = events_player.iloc[(event)]["Start Time [s]"]
        pos = round(frame*5)
        events_player.at[event,'state'] =  int(best_z[k][pos])
    individual_dfs.append(events_player)

for z in best_z:
    _,c =np.unique(z, return_counts=True)
    state_duration.append(c/300)
```

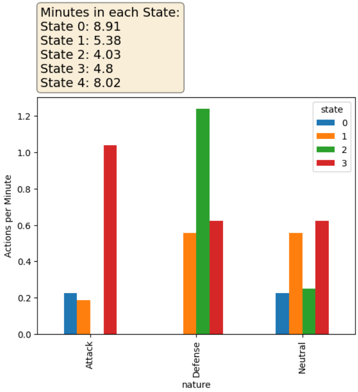


Figure: Player 0.

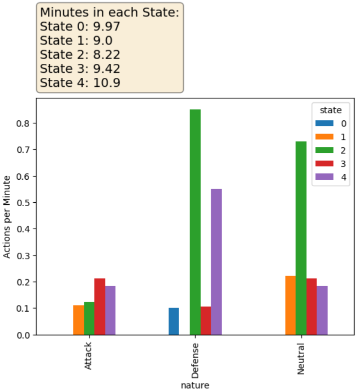


Figure: Player 1.

Events Bar Plots

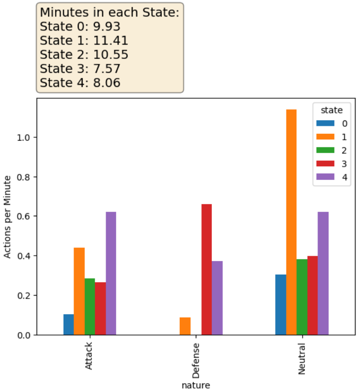


Figure: Player 6.

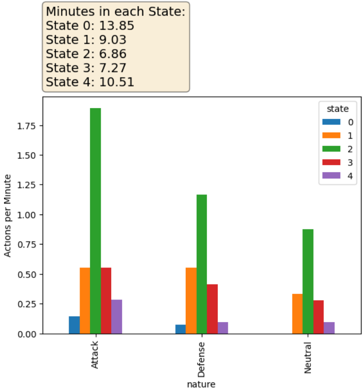


Figure: Player 7.

Events Bar Plots

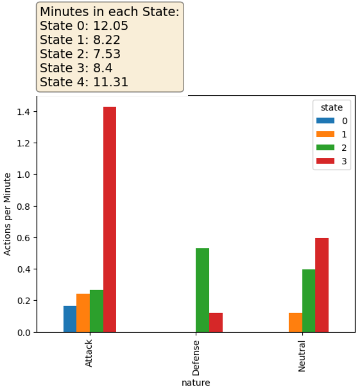


Figure: Player 8.

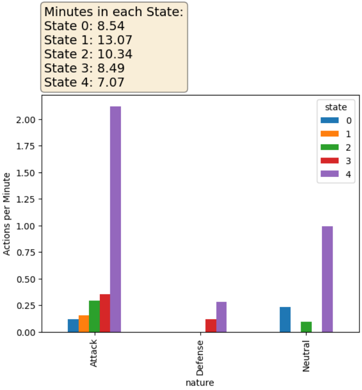
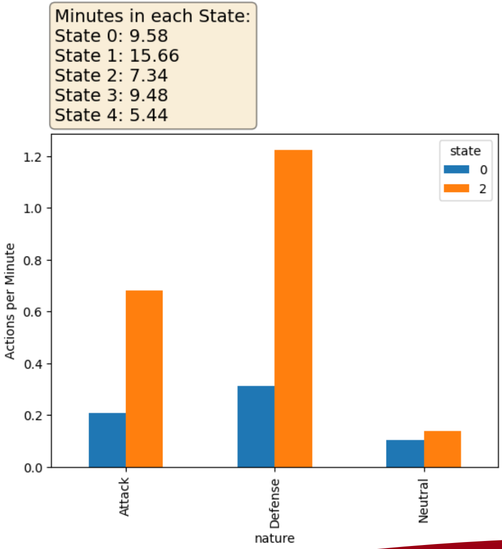


Figure: Player 9.

Events Bar Plots



Thank you for your attention!

Backup and additional slides

The joint probability of all the variables can be written as the product of likelihood and priors, counting only the terms in which z_t appears.
This yields:

- $$P(z_1 | z_{2:T}, \{\hat{A}_k\}, \{Q_k\}, M, \mathbf{y}_{1:T}) \propto M_{z_1, z_2}$$

AR HMM: Gibbs Sampling (2)

Università degli Studi di Padova

The joint probability of all the variables can be written as the product of likelihood and priors, counting only the terms in which z_t appears.
This yields:

- $$P(z_1 | z_{2:T}, \{\hat{A}_k\}, \{Q_k\}, M, \mathbf{y}_{1:T}) \propto M_{z_1, z_2}$$

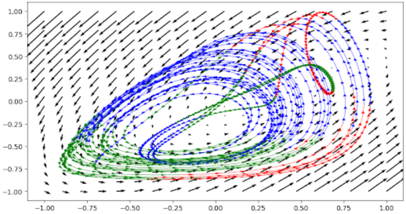
- $$P(z_t | z_{1:T}^{-t}, \{\hat{A}_k\}, \{Q_k\}, M, \mathbf{y}_{1:T}) \propto M_{z_{t-1}, z_t} \cdot M_{z_t, z_{t+1}} \cdot \mathcal{N}(\mathbf{y}_t - \hat{A}_{z_t} \hat{\mathbf{y}}_{t-1}, Q_{z_t})$$

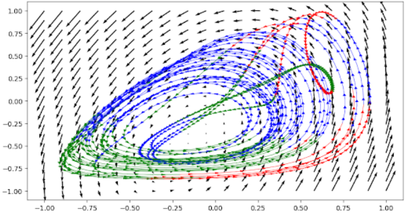
The joint probability of all the variables can be written as the product of likelihood and priors, counting only the terms in which z_t appears.
This yields:

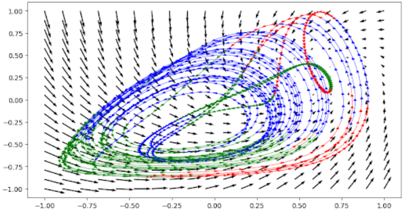
- $$P(z_1 | z_{2:T}, \{\hat{A}_k\}, \{Q_k\}, M, \mathbf{y}_{1:T}) \propto M_{z_1, z_2}$$

- $$P(z_t | z_{1:T}^{-t}, \{\hat{A}_k\}, \{Q_k\}, M, \mathbf{y}_{1:T}) \propto M_{z_{t-1}, z_t} \cdot M_{z_t, z_{t+1}} \cdot \mathcal{N}(\mathbf{y}_t - \hat{A}_{z_t} \hat{\mathbf{y}}_{t-1}, Q_{z_t})$$

- $$P(z_T | z_{1:T-1}, \{\hat{A}_k\}, \{Q_k\}, M, \mathbf{y}_{1:T}) \propto M_{z_{T-1}, z_T} \cdot \mathcal{N}(\mathbf{y}_T - \hat{A}_{z_T} \hat{\mathbf{y}}_{T-1}, Q_{z_T})$$


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00