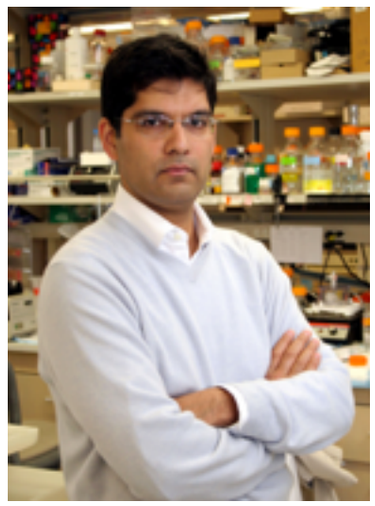
Composing graphical models with neural networks for structured representations and fast inference

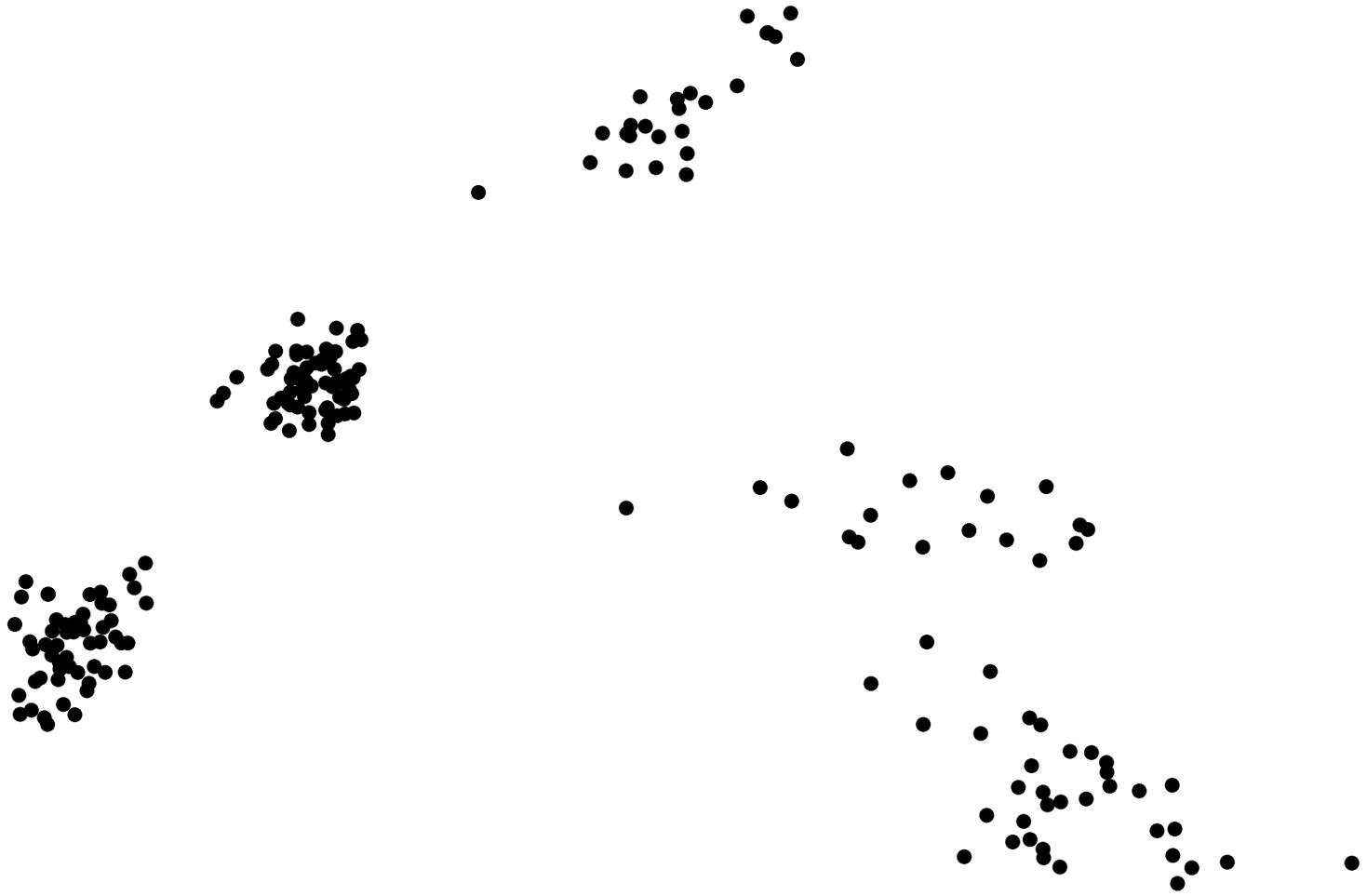
Matt Johnson, David Duvenaud, Alex Wiltschko, Bob Datta, Ryan Adams

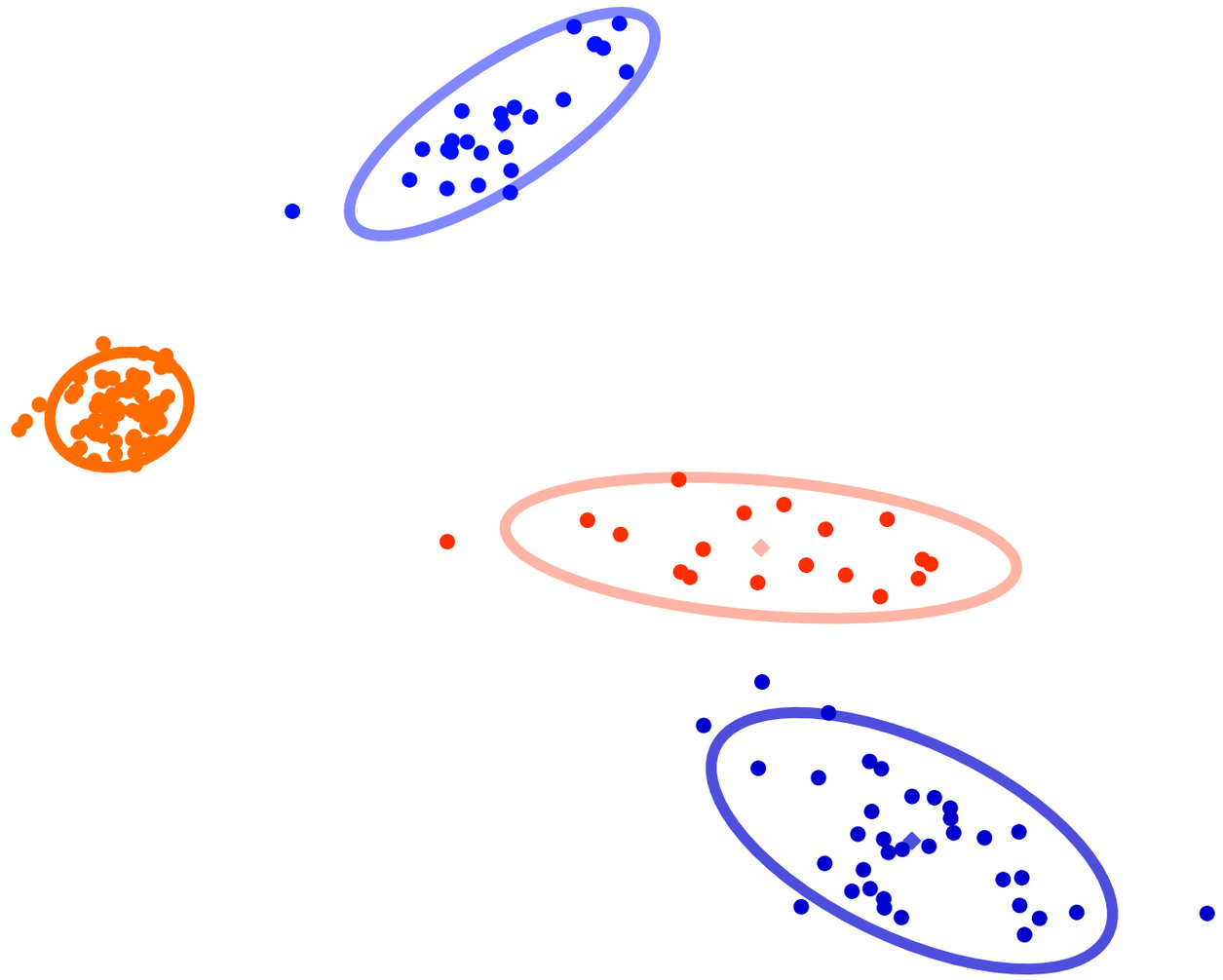


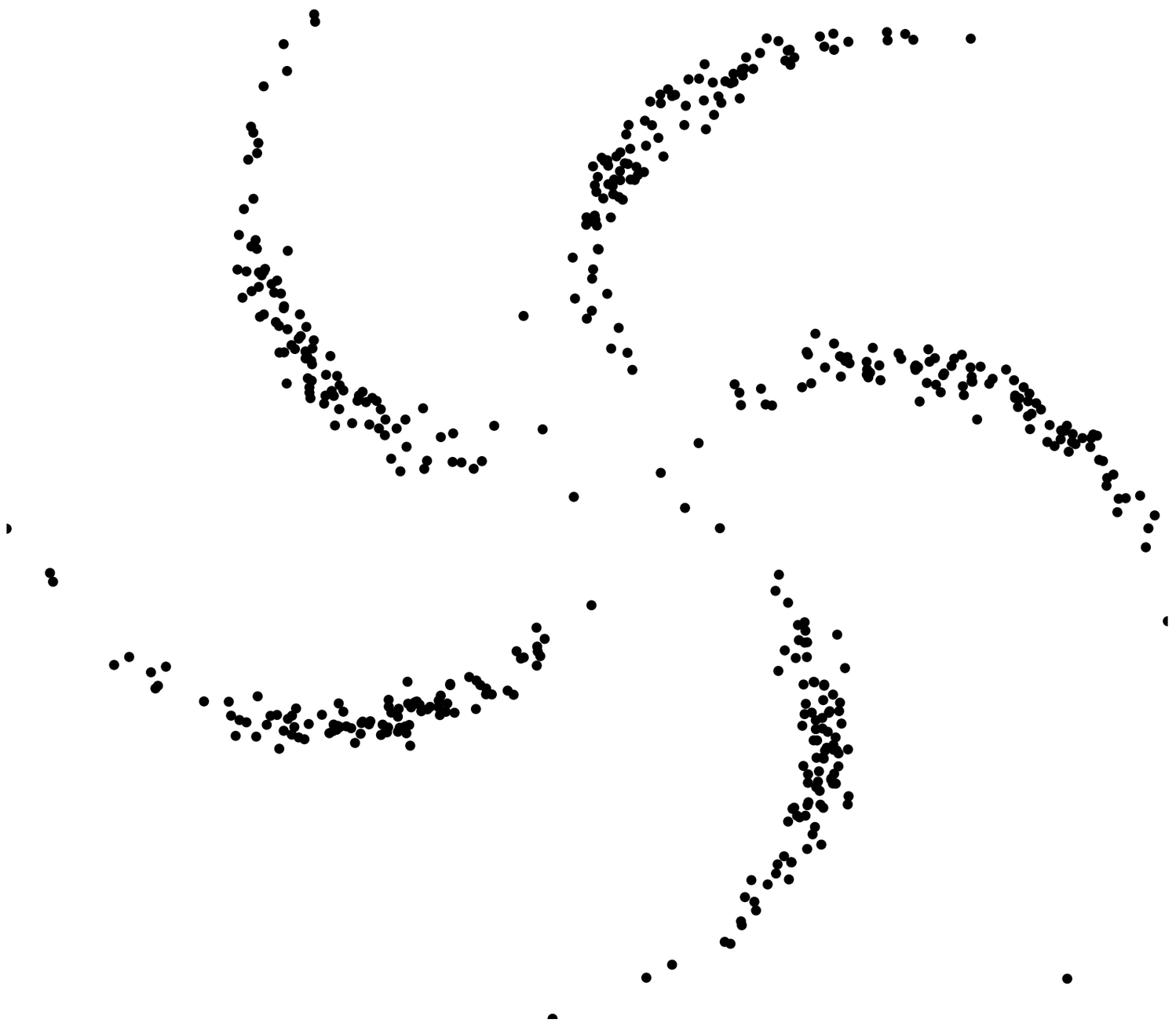



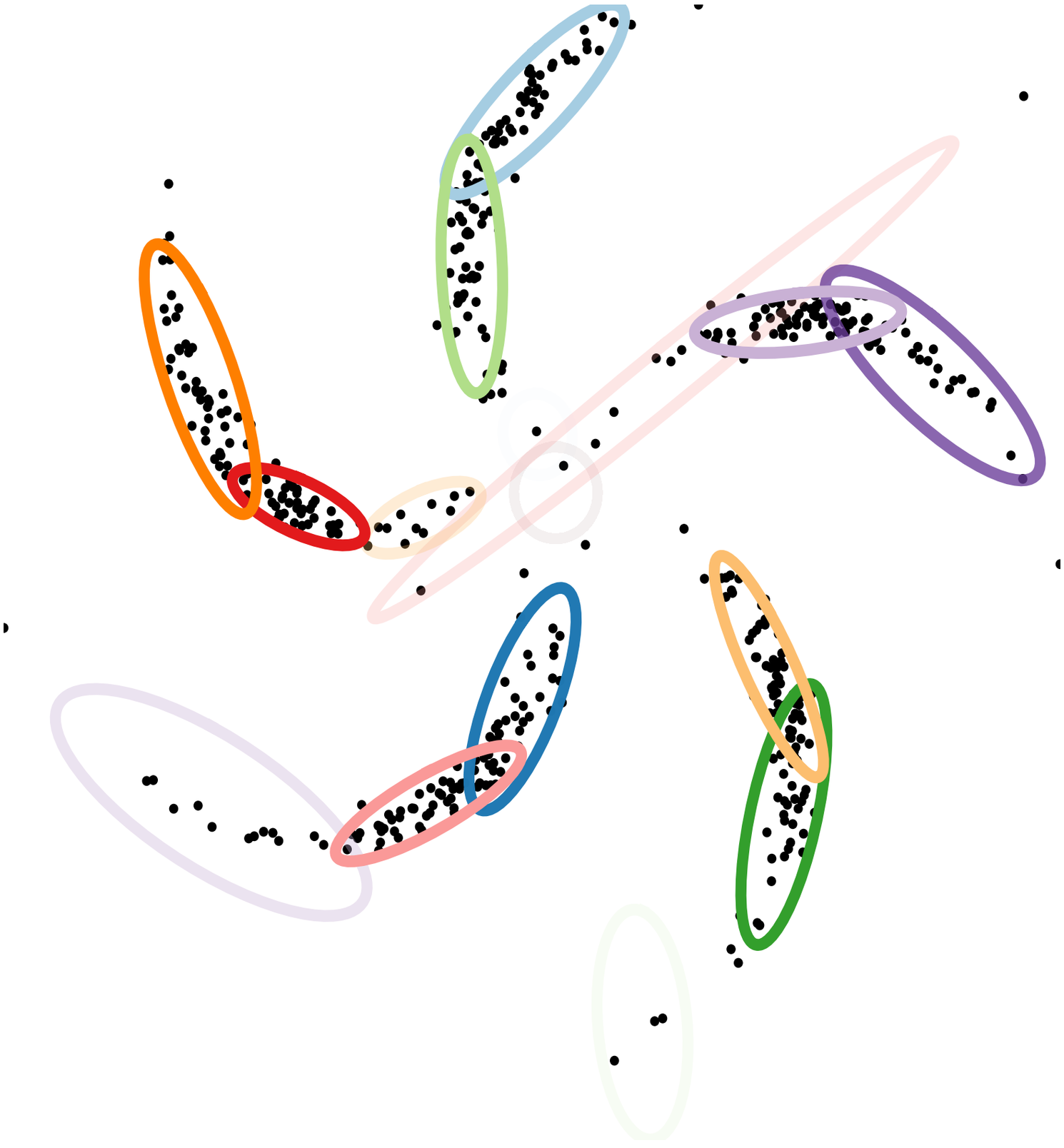

HARVARD
UNIVERSITY

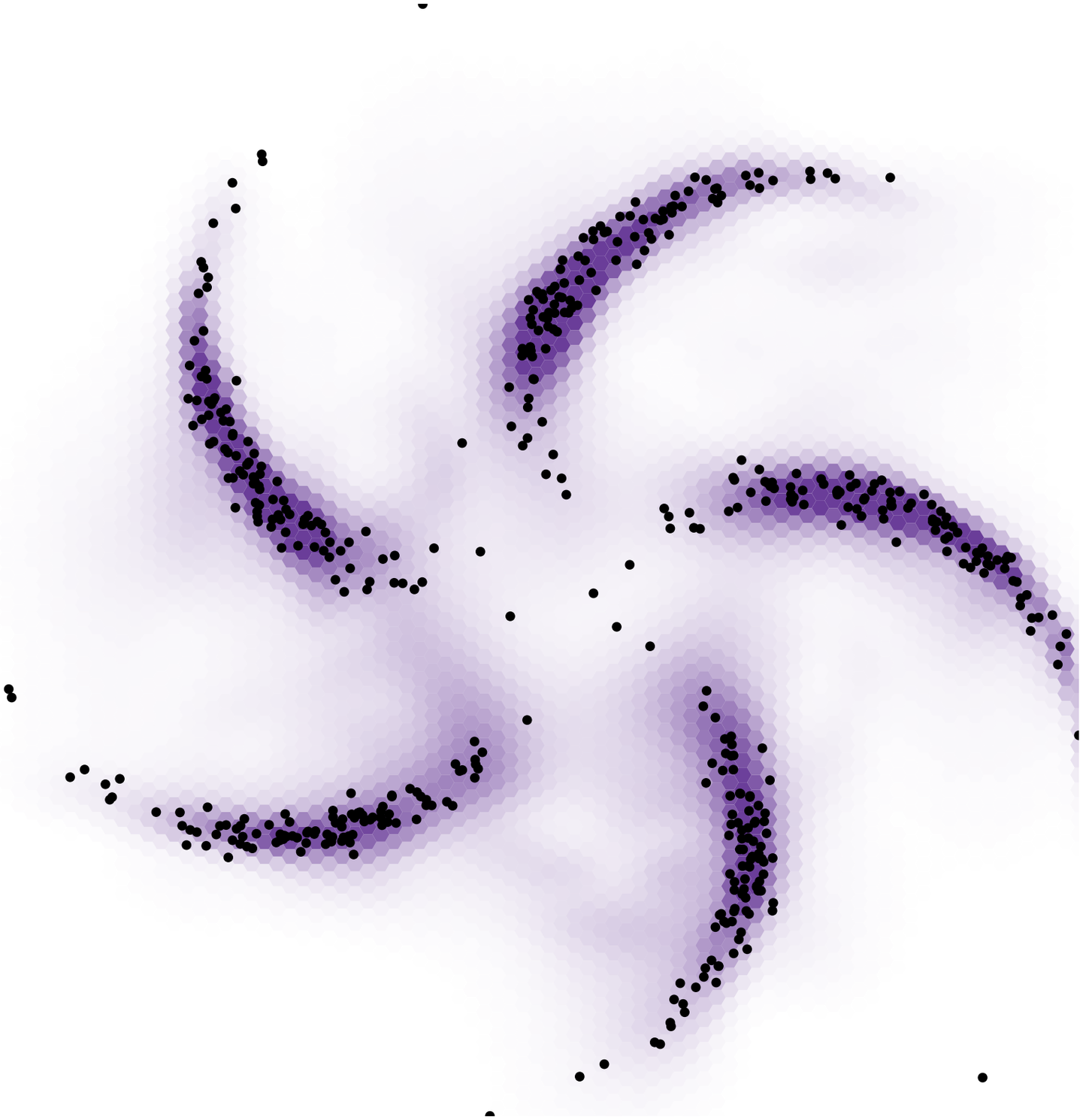
UNIVERSITY OF
TORONTO

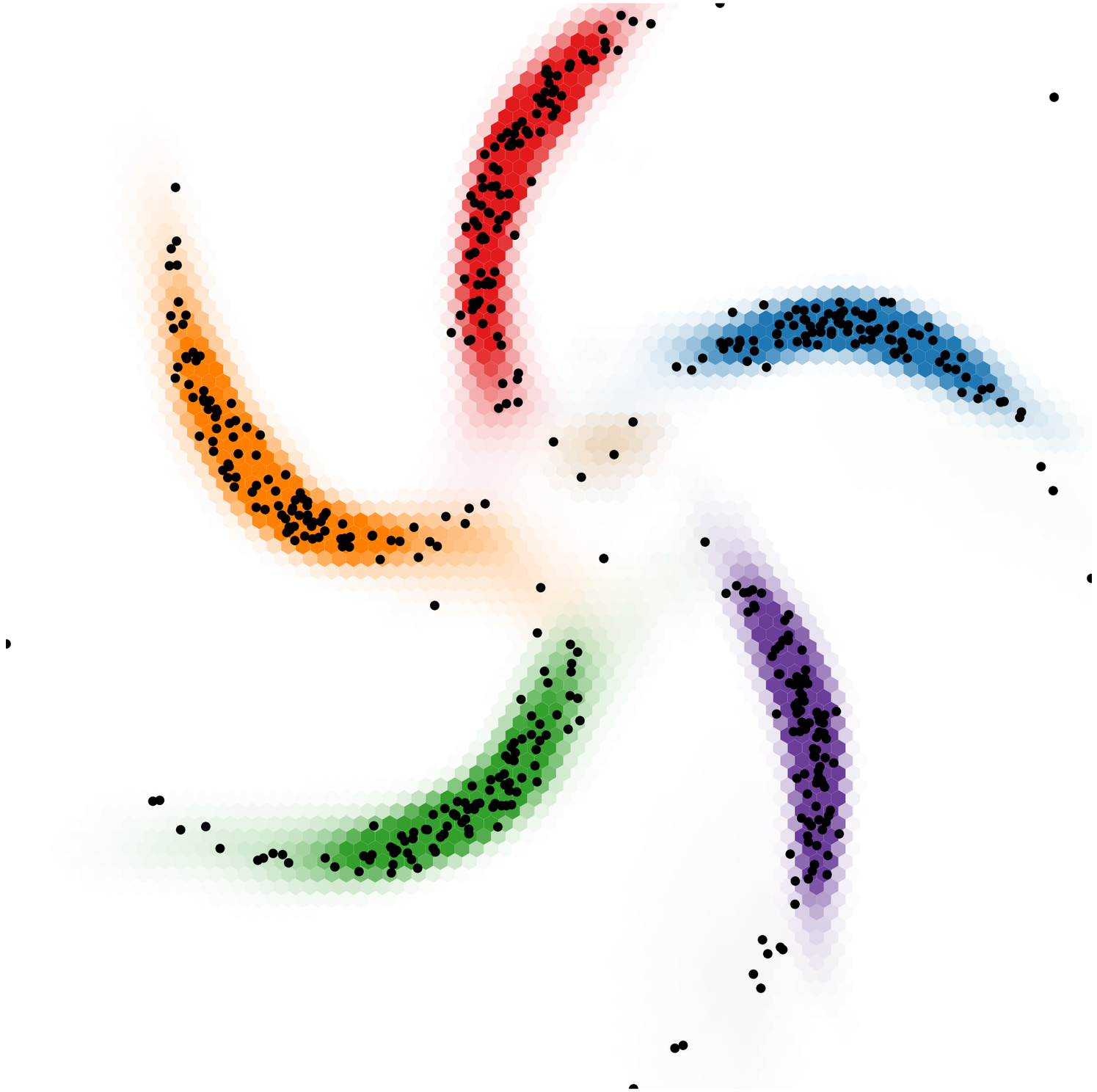


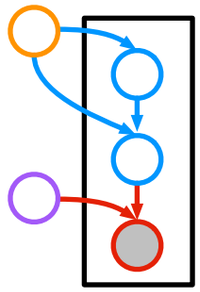






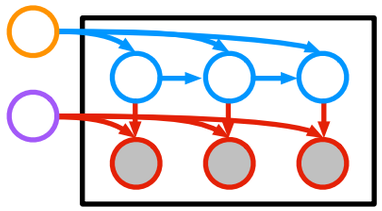






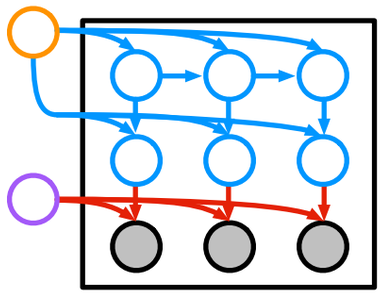
[1]

Gaussian mixture model



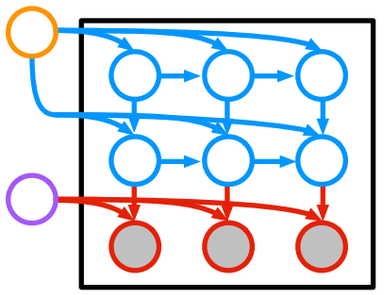
[2]

Linear dynamical system



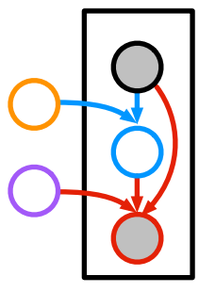
[3]

Hidden Markov model



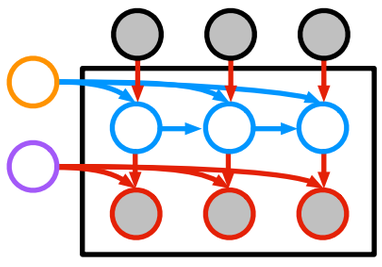
[4]

Switching LDS



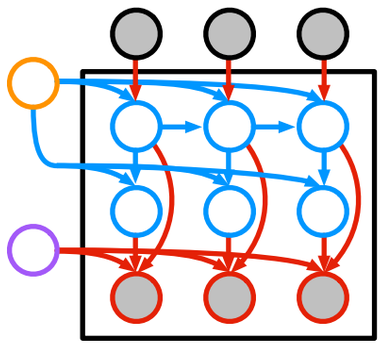
[5]

Mixture of Experts



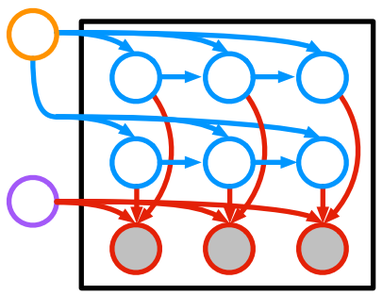
[2]

Driven LDS



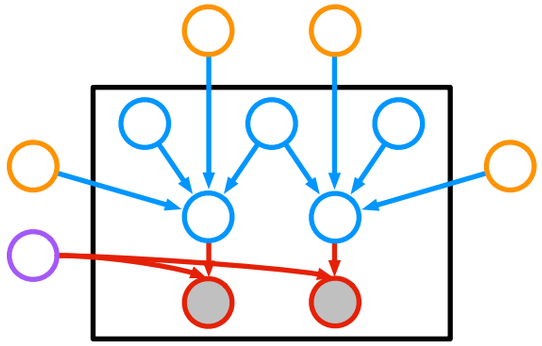
[6]

IO-HMM



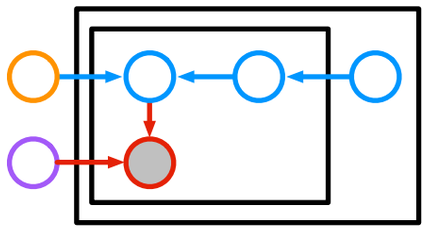
[7]

Factorial HMM



[8,9]

Canonical correlations analysis



[10]

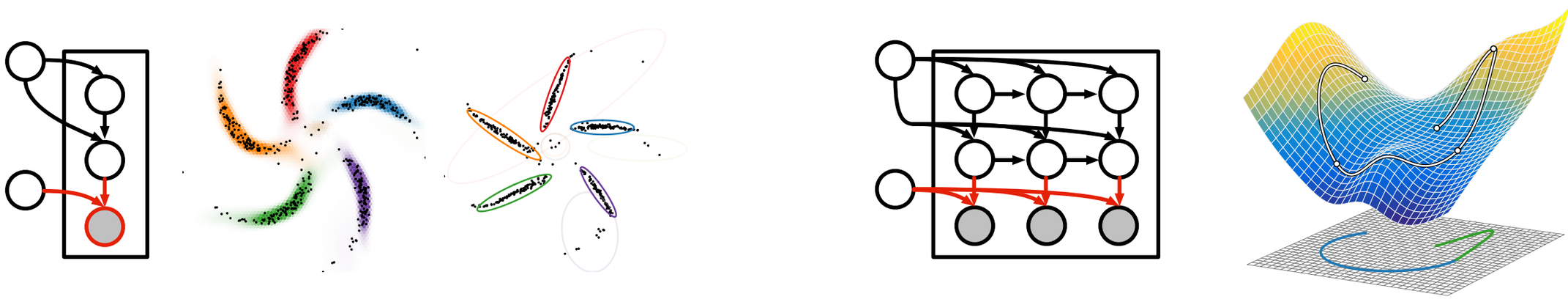
admixture / LDA / NMF

[1] Palmer, Wipf, Kreutz-Delgado, and Rao. Variational EM algorithms for non-Gaussian latent variable models. NIPS 2005.
[2] Ghahramani and Beal. Propagation algorithms for variational Bayesian learning. NIPS 2001.
[3] Beal. Variational algorithms for approximate Bayesian inference, Ch. 3. U of London Ph.D. Thesis 2003.
[4] Ghahramani and Hinton. Variational learning for switching state-space models. Neural Computation 2000.
[5] Jordan and Jacobs. Hierarchical Mixtures of Experts and the EM algorithm. Neural Computation 1994.
[6] Bengio and Frasconi. An Input Output HMM Architecture. NIPS 1995.
[7] Ghahramani and Jordan. Factorial Hidden Markov Models. Machine Learning 1997.
[8] Bach and Jordan. A probabilistic interpretation of Canonical Correlation Analysis. Tech. Report 2005.
[9] Archambeau and Bach. Sparse probabilistic projections. NIPS 2008.
[10] Hoffman, Bach, Blei. Online learning for Latent Dirichlet Allocation. NIPS 2010.

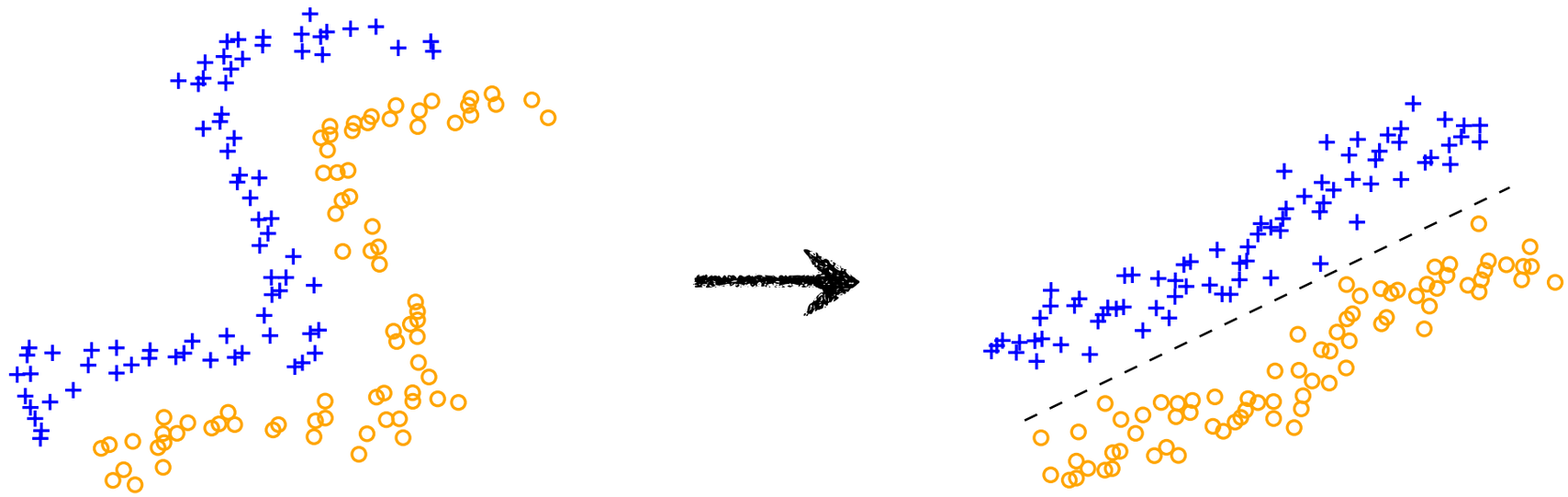
Probabilistic graphical models	Deep learning
+ structured representations	– neural net “goo”
+ priors and uncertainty	– difficult parameterization
+ data and computational efficiency	– can require lots of data
– rigid assumptions may not fit	+ flexible
– feature engineering	+ feature learning
– top-down inference	+ recognition networks


MAKE PGMS
GREAT AGAIN

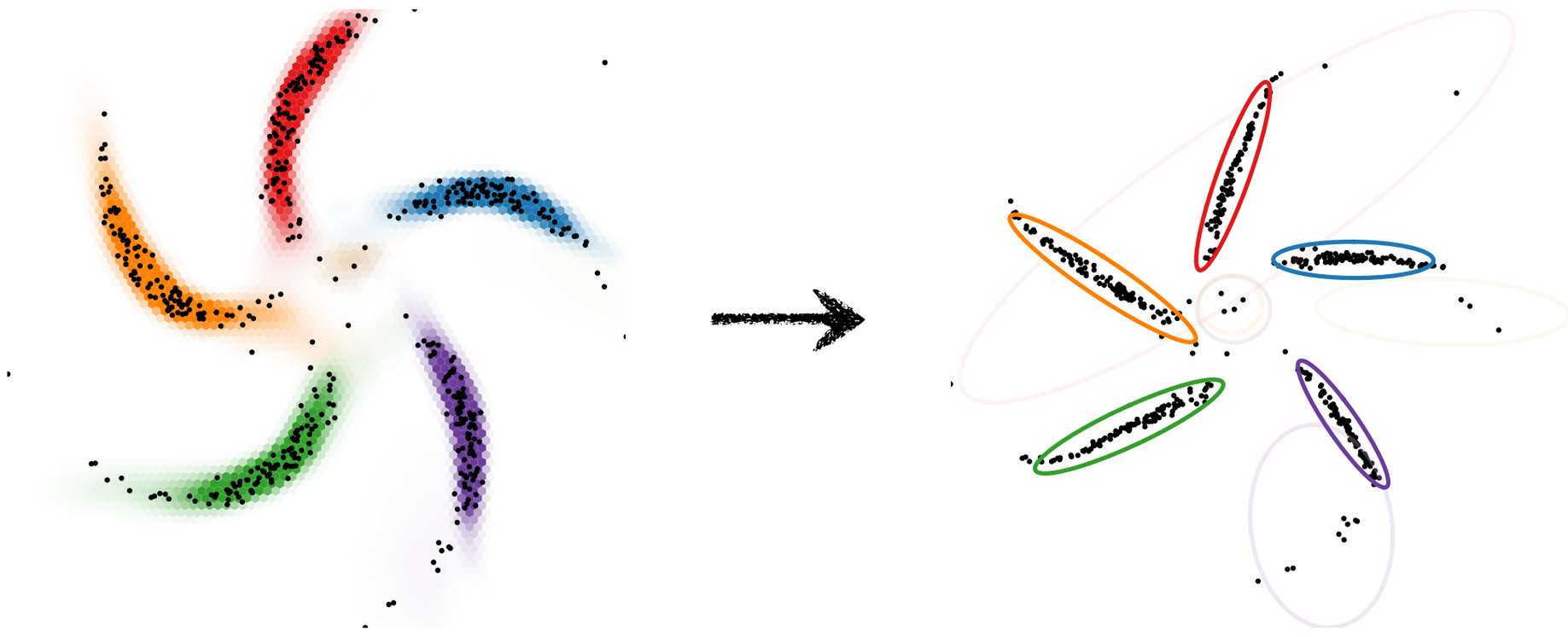
Modeling idea: graphical models on latent variables,
neural network models for observations

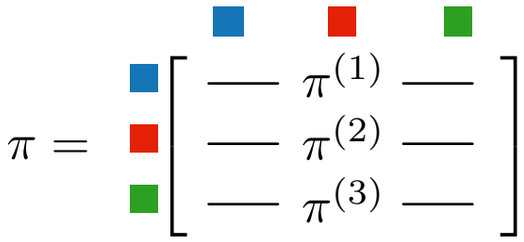


supervised learning

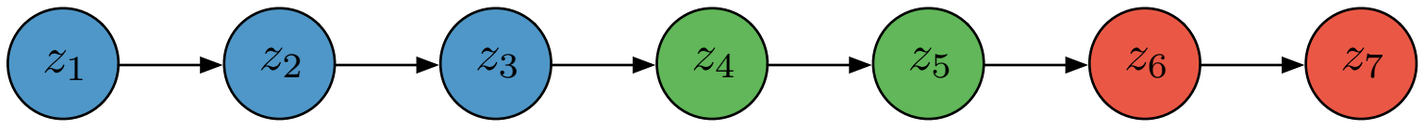


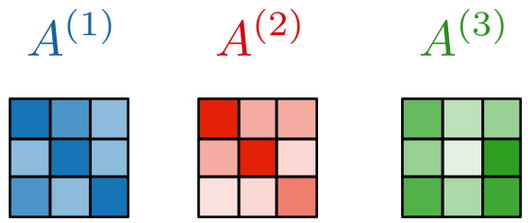
unsupervised learning



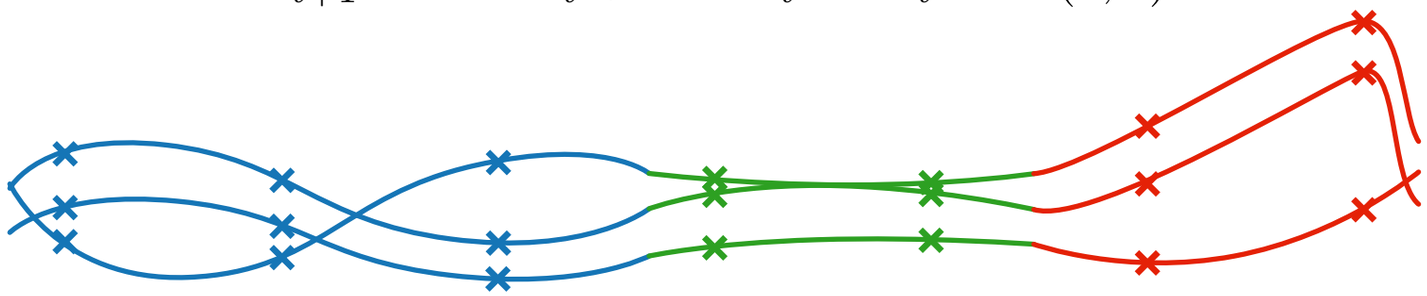


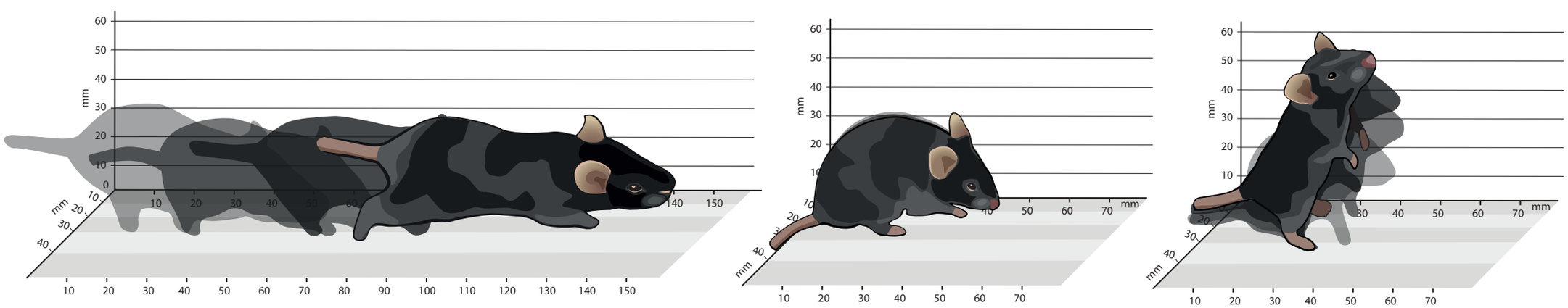
$$z_{t+1} \sim \pi^{(z_t)}$$

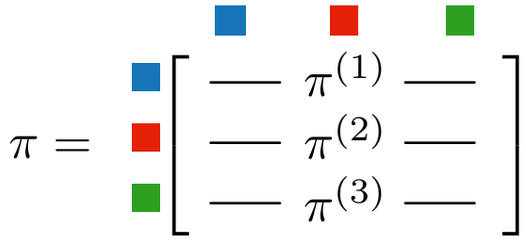


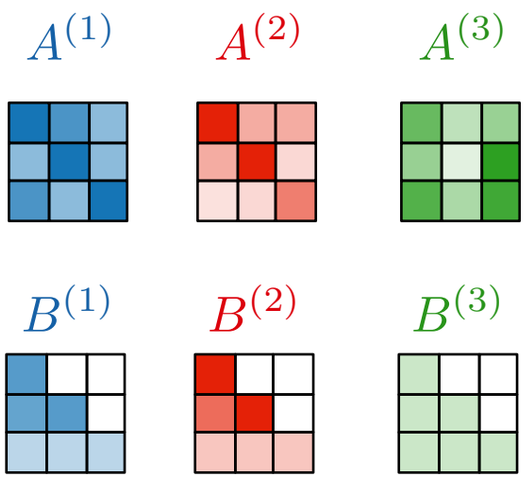


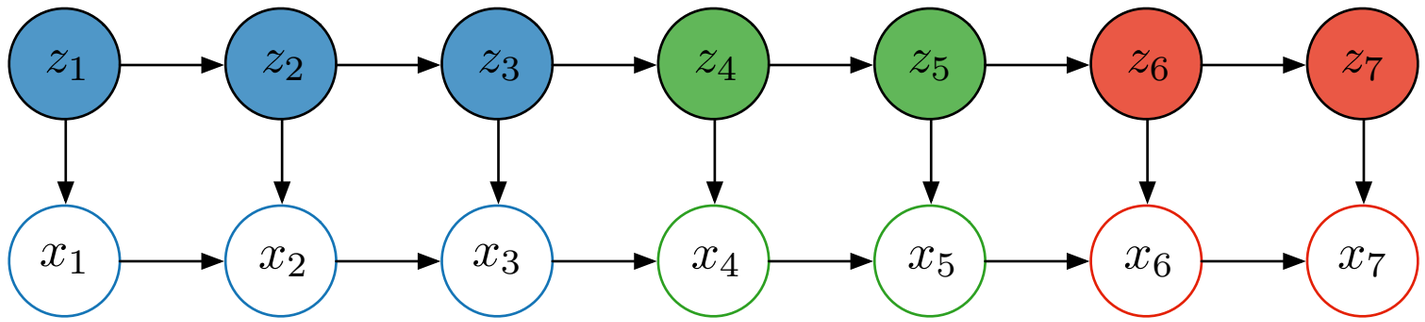
$$x_{t+1} = A^{(z_t)} x_t + B^{(z_t)} u_t \qquad u_t \overset{\text{iid}}{\sim} \mathcal{N}(0, I)$$

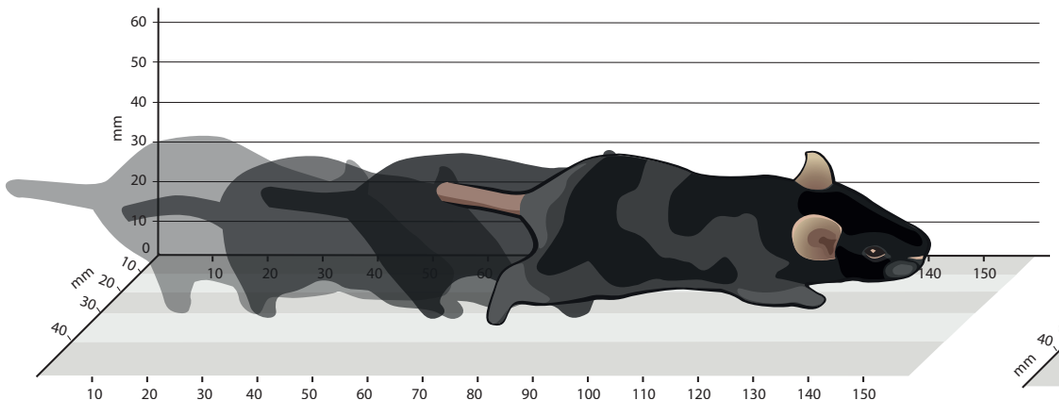


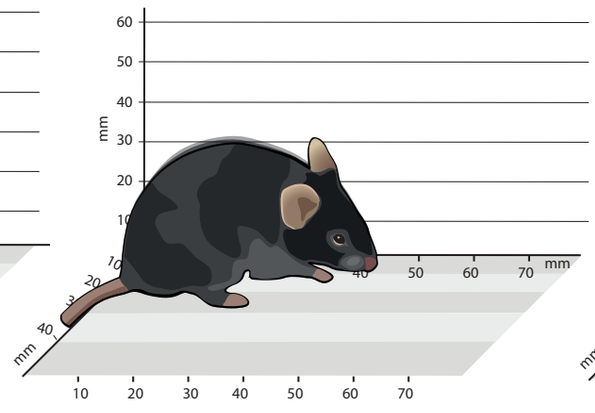


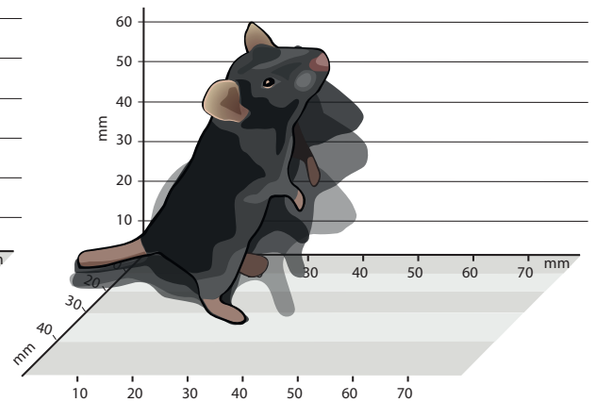

$\pi =$
$\pi^{(1)}$
$\pi^{(2)}$
$\pi^{(3)}$

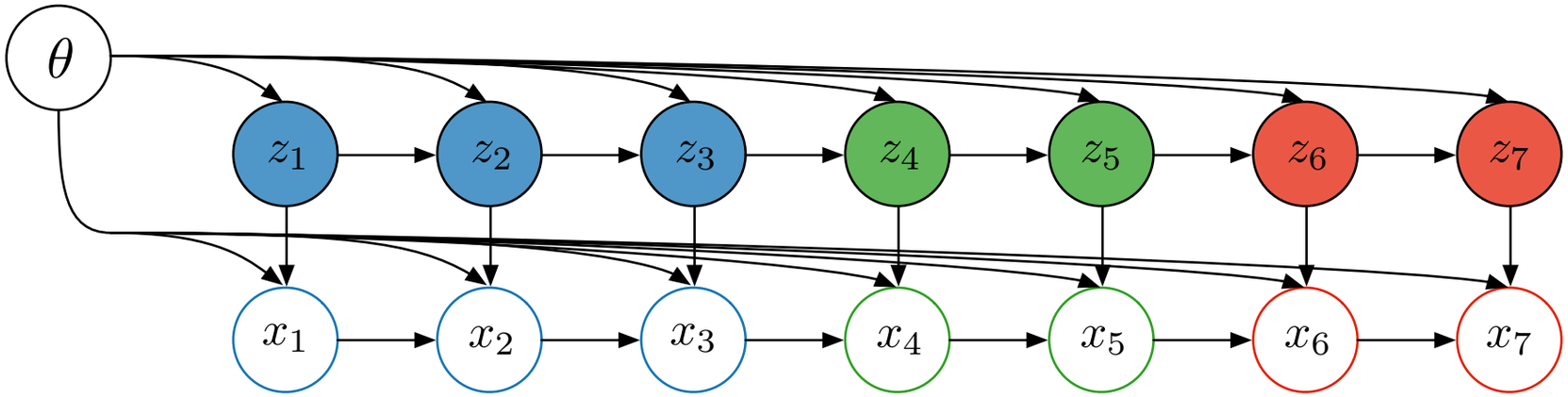

$A^{(1)}$
$A^{(2)}$
$A^{(3)}$
$B^{(1)}$
$B^{(2)}$
$B^{(3)}$

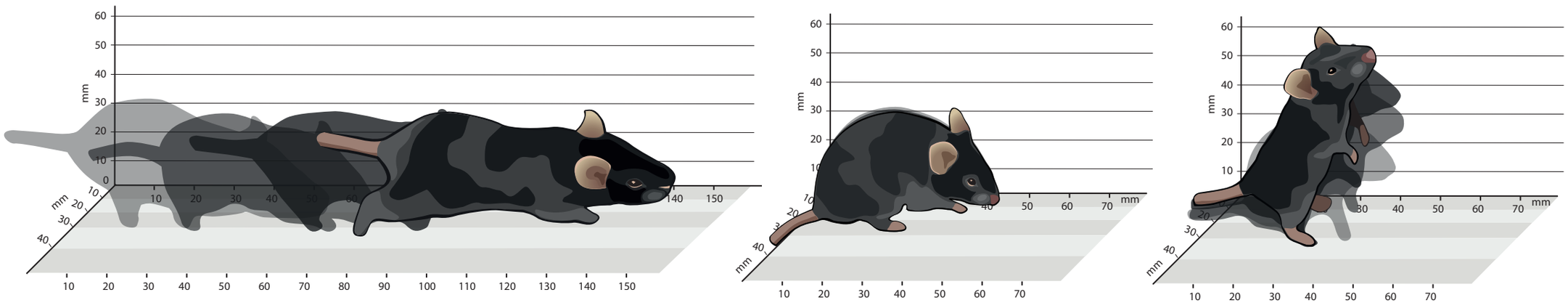

z_1
z_2
z_3
z_4
z_5
z_6
z_7
x_1
x_2
x_3
x_4
x_5
x_6
x_7

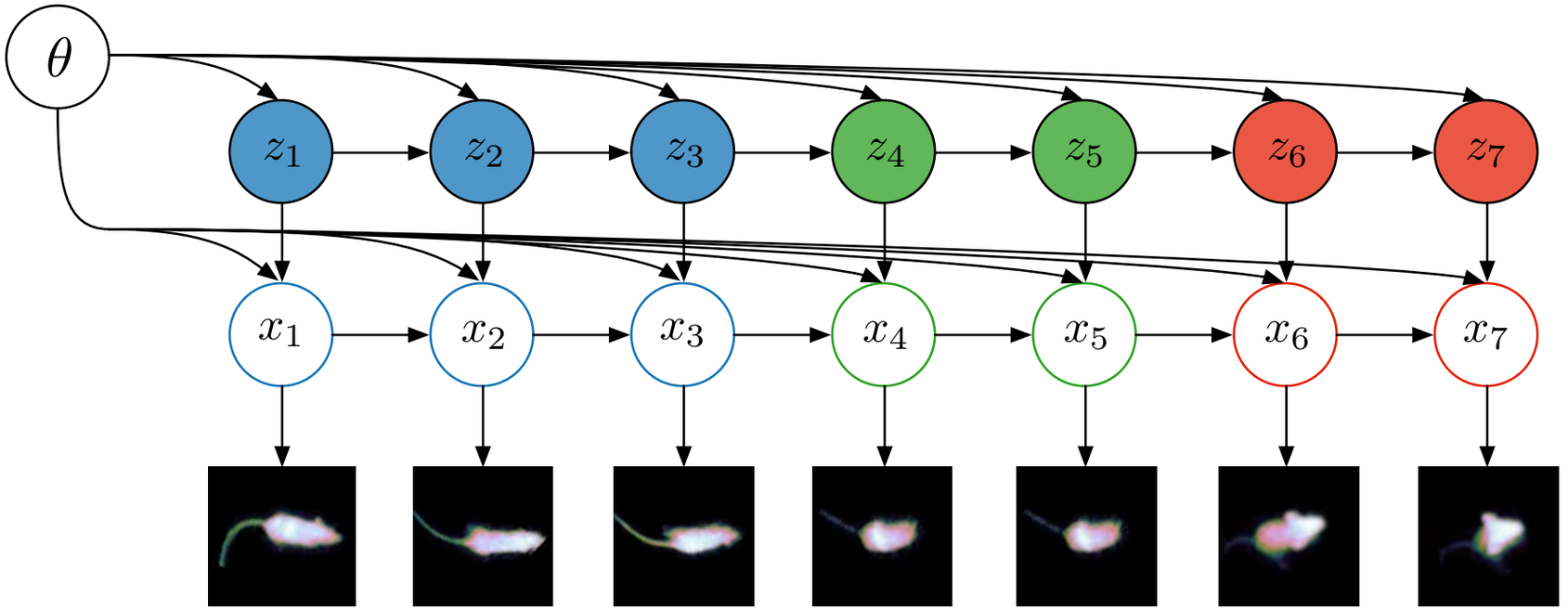

mm

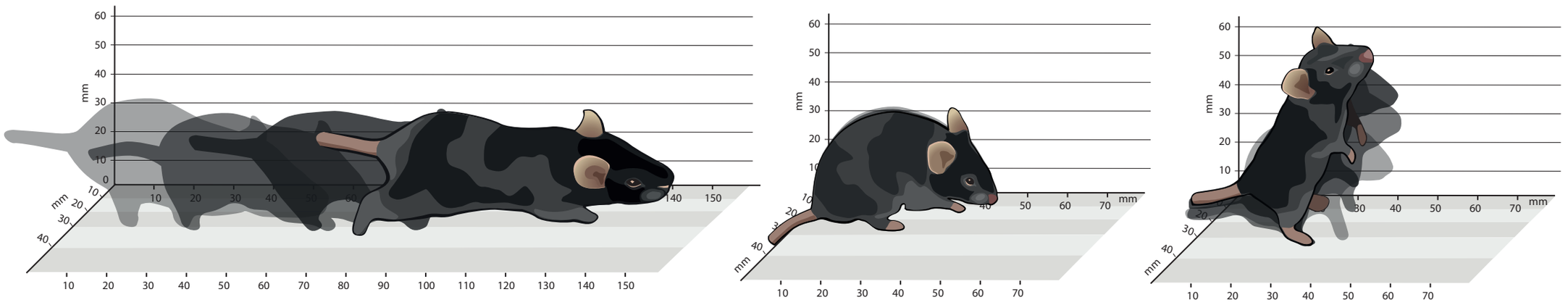

mm

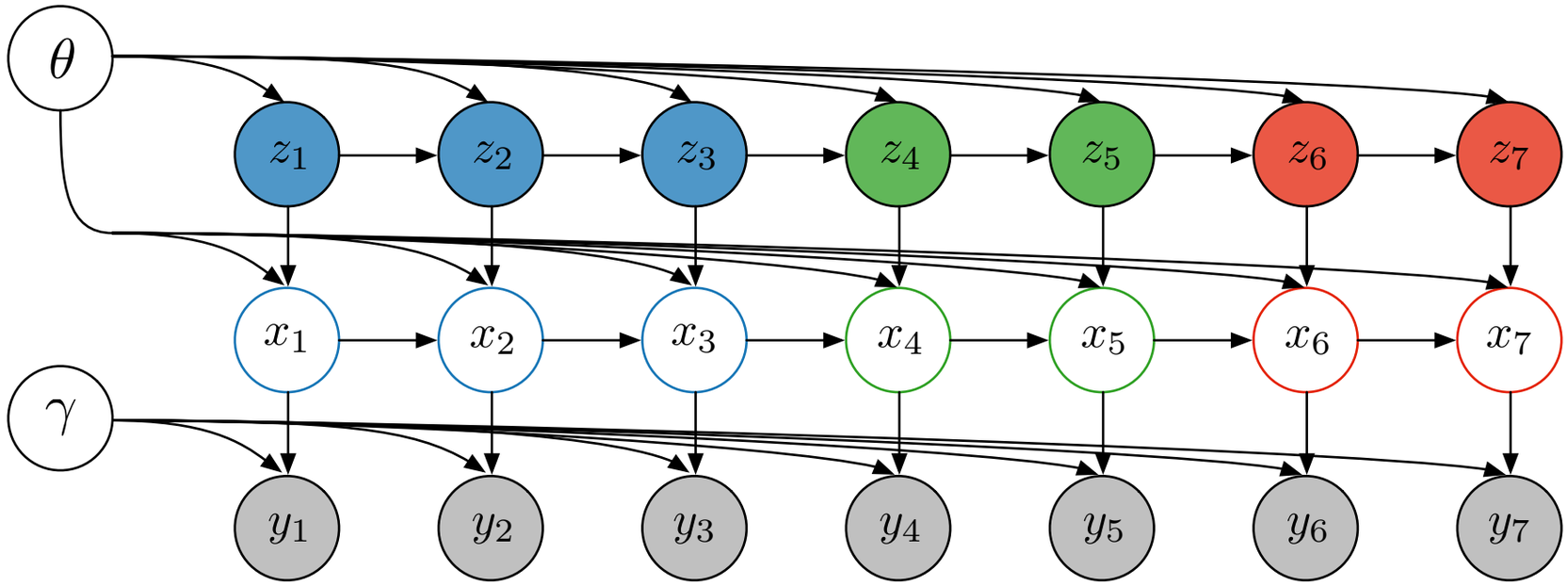

mm


θ
z_1
z_2
z_3
z_4
z_5
z_6
z_7
x_1
x_2
x_3
x_4
x_5
x_6
x_7

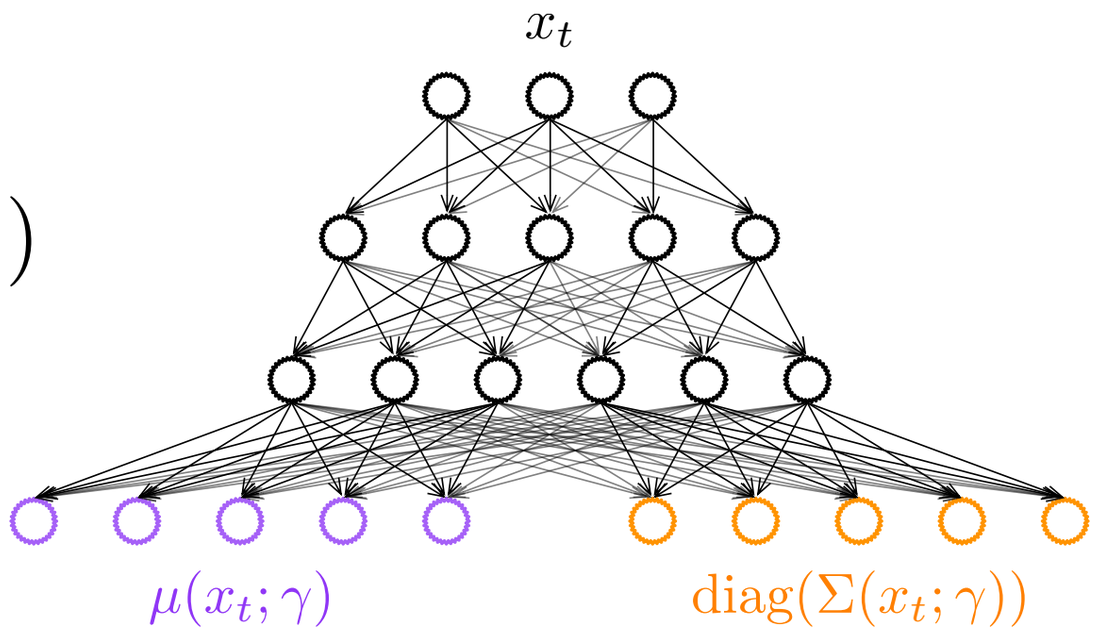

mm
0
10
20
30
40
50
60
70
80
90
100
110
120
130
140
150

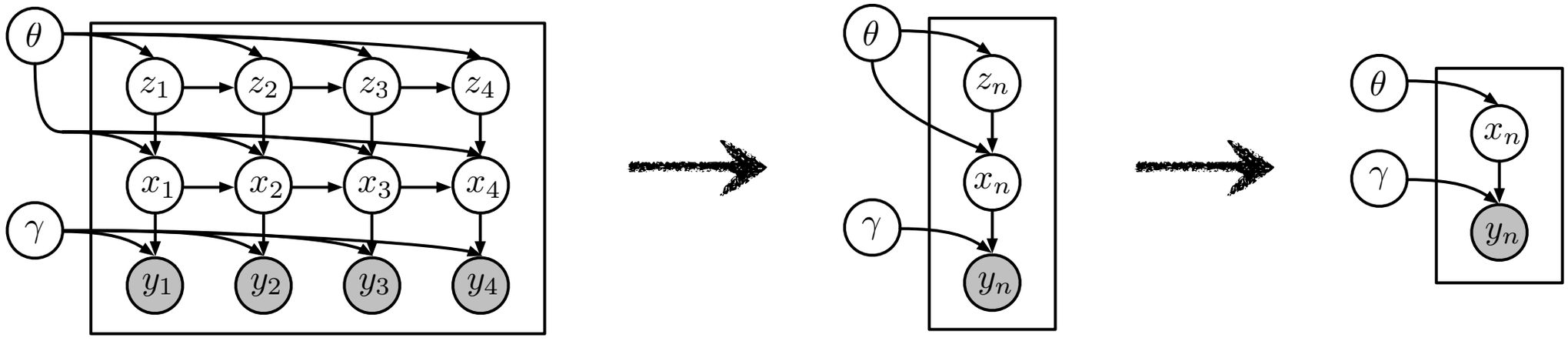

θ
z_1
z_2
z_3
z_4
z_5
z_6
z_7
x_1
x_2
x_3
x_4
x_5
x_6
x_7

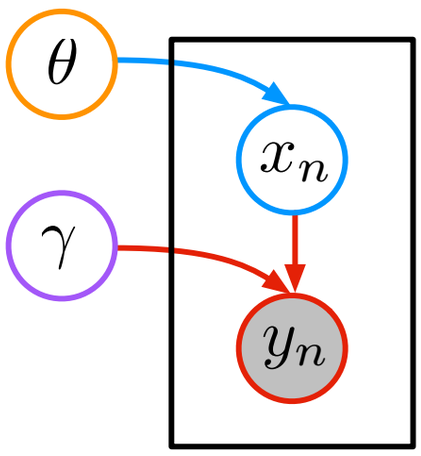

mm
mm
mm



$$y_t \,|\, x_t, \gamma \;\sim\; \mathcal{N}(\mu(x_t;\gamma),\, \Sigma(x_t;\gamma))$$





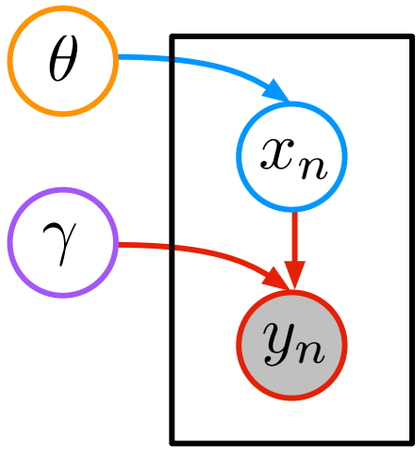


$p(\theta)$ conjugate prior on global variables

$p(x \mid \theta)$ exponential family on local variables

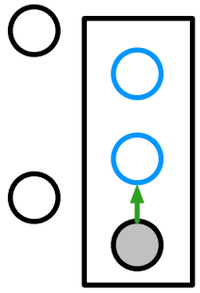
$p(\gamma)$ any prior on observation parameters

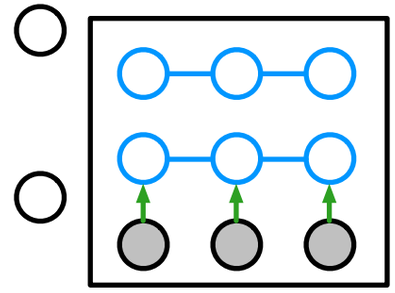
$p(y \mid x, \gamma)$ neural network observation model



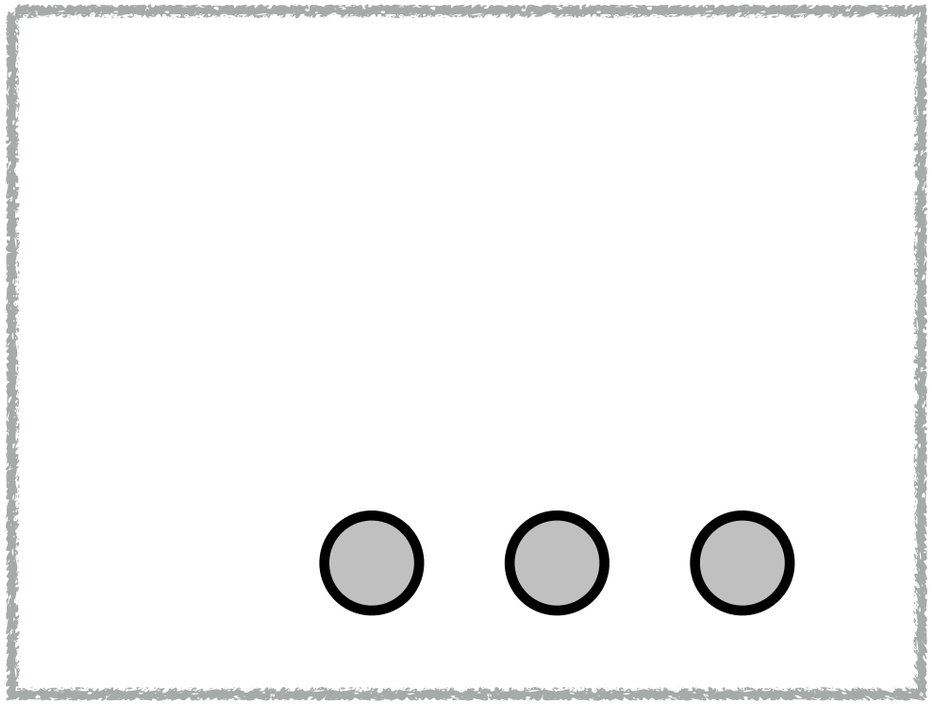
Inference?

SVAEs: recognition networks output conjugate potentials, then apply fast graphical model algorithms

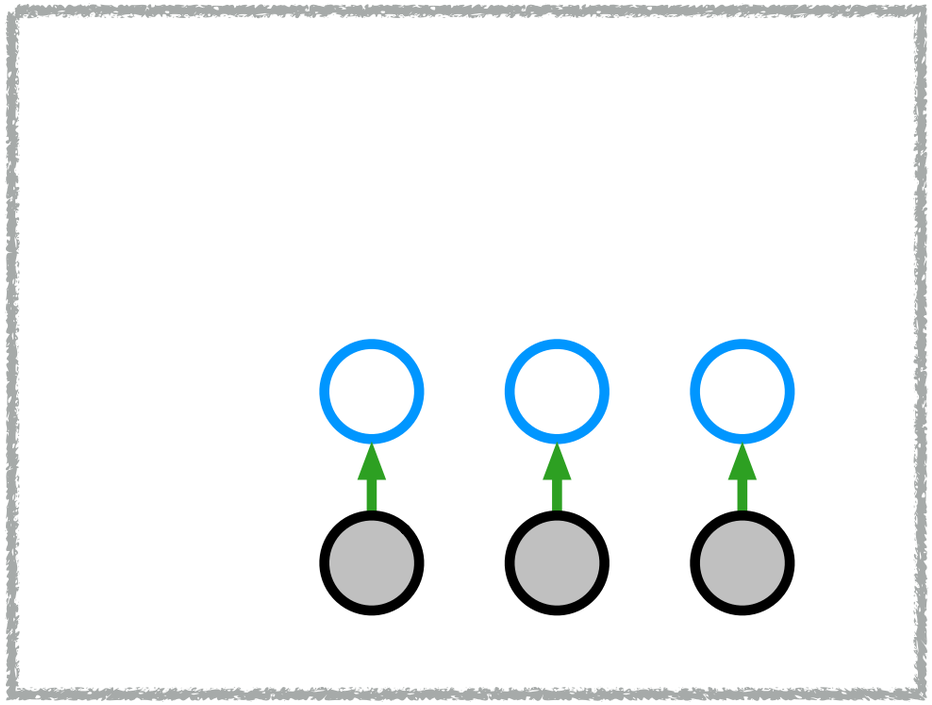




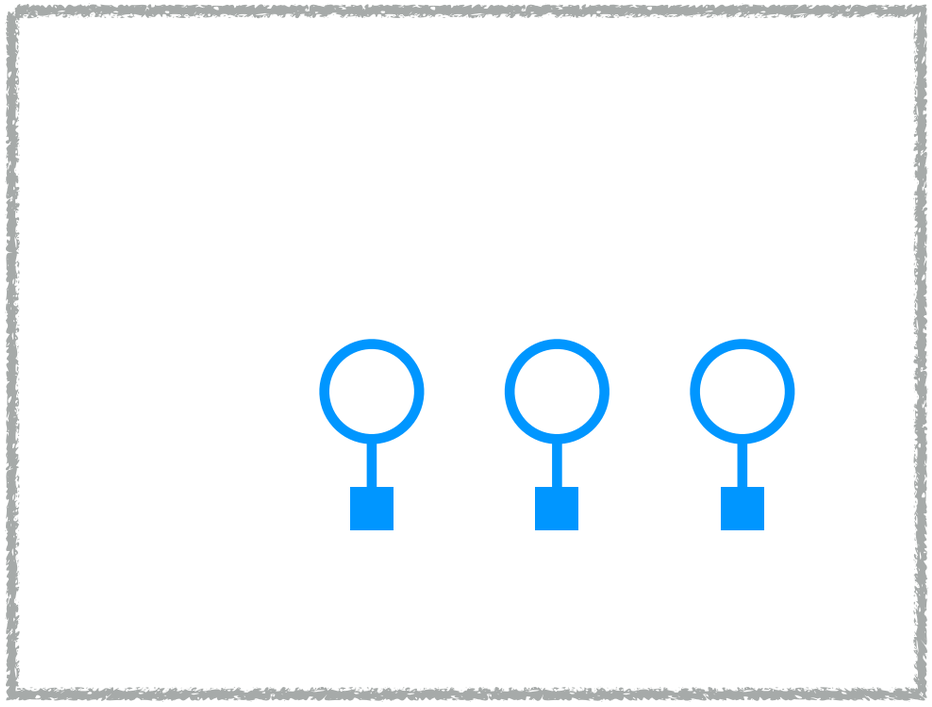
Step 1: apply recognition network



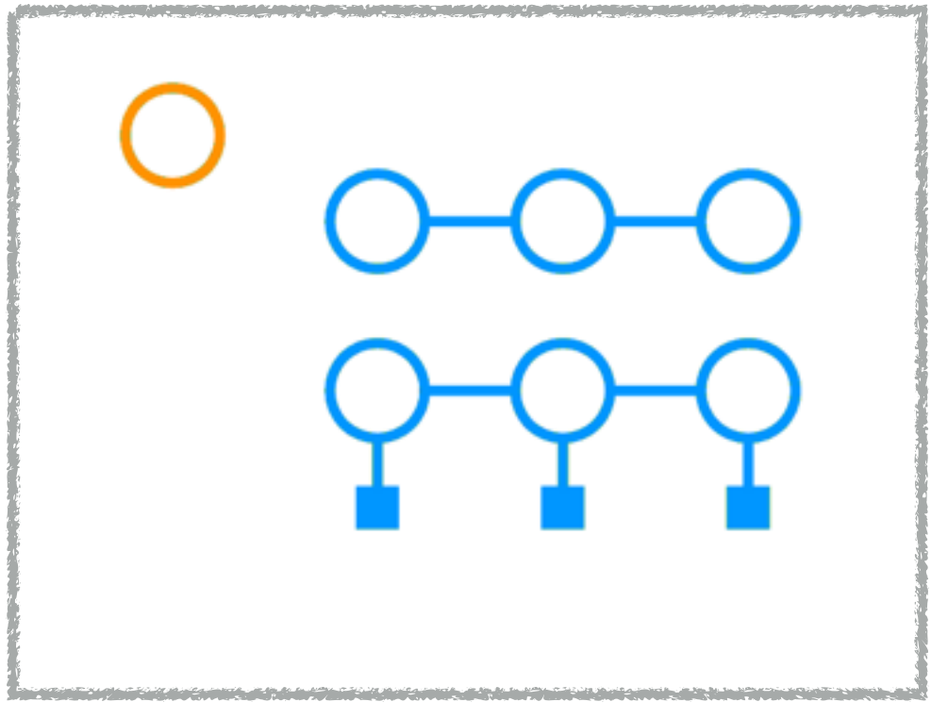
Step 1: apply recognition network



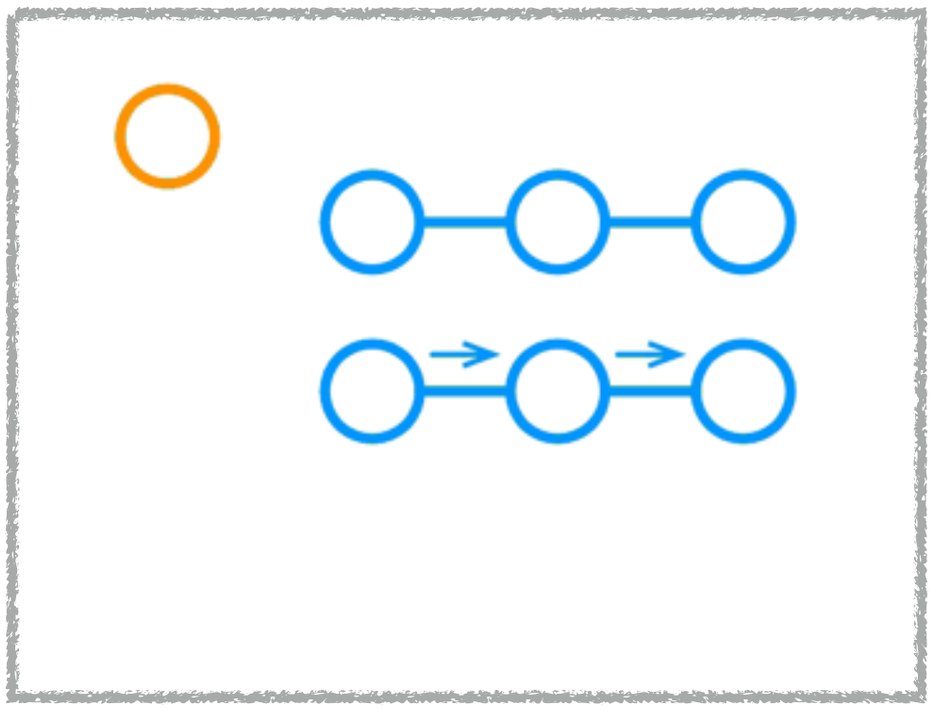
Step 1: apply recognition network



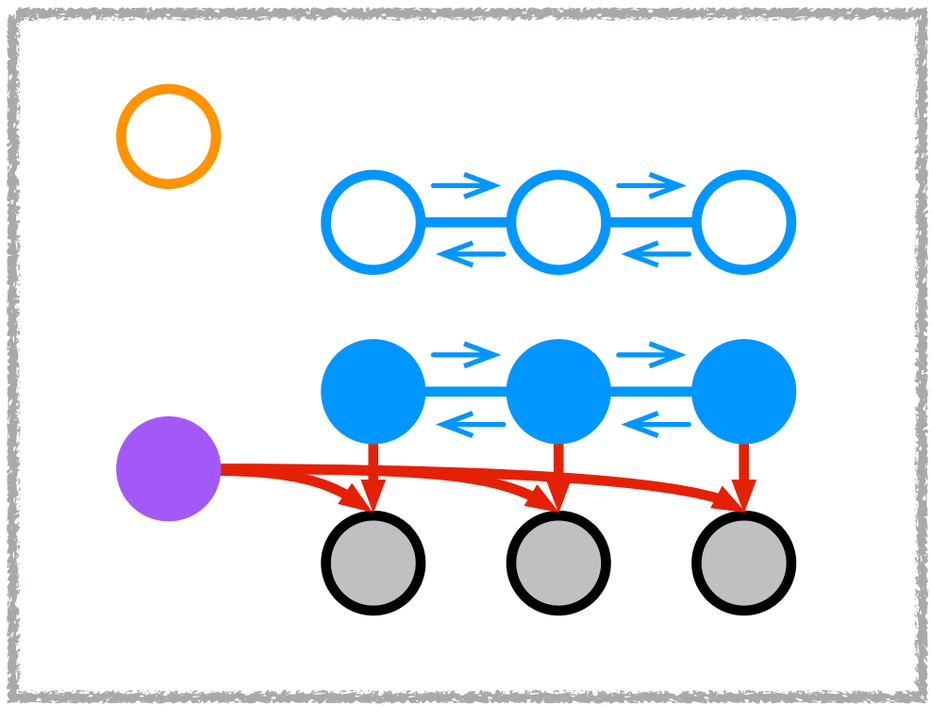
Step 2: run fast PGM algorithms

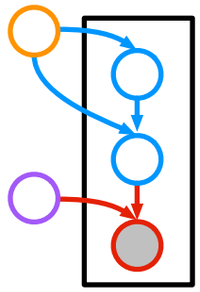


Step 3: sample, compute flat grads



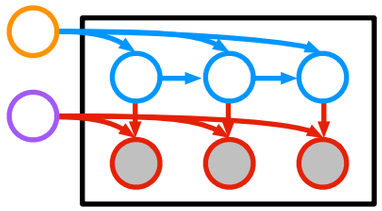
Step 4: compute natural gradient





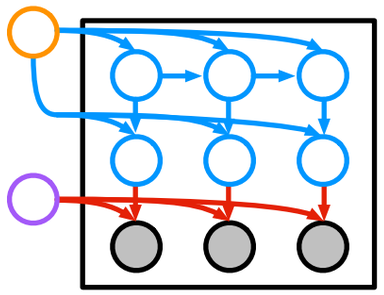
[1]

Gaussian mixture model



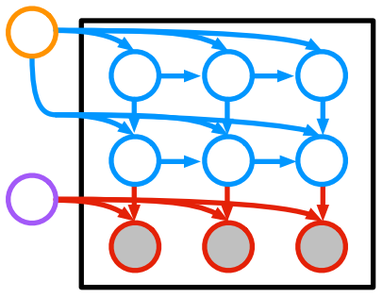
[2]

Linear dynamical system



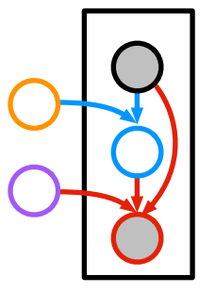
[3]

Hidden Markov model



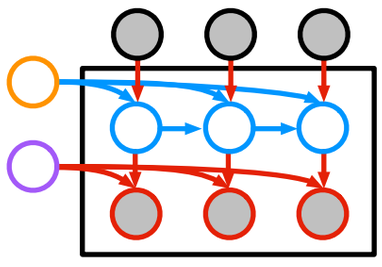
[4]

Switching LDS



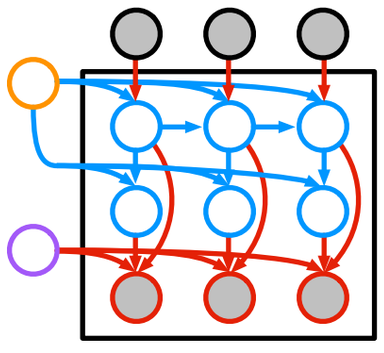
[5]

Mixture of Experts



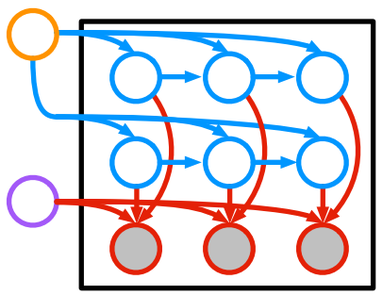
[2]

Driven LDS



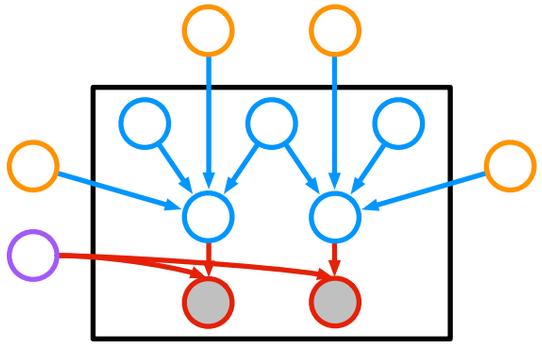
[6]

IO-HMM



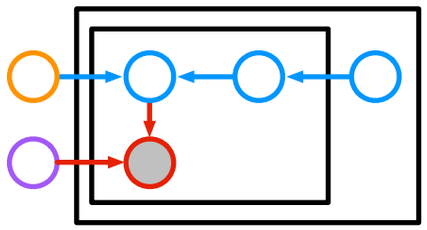
[7]

Factorial HMM



[8,9]

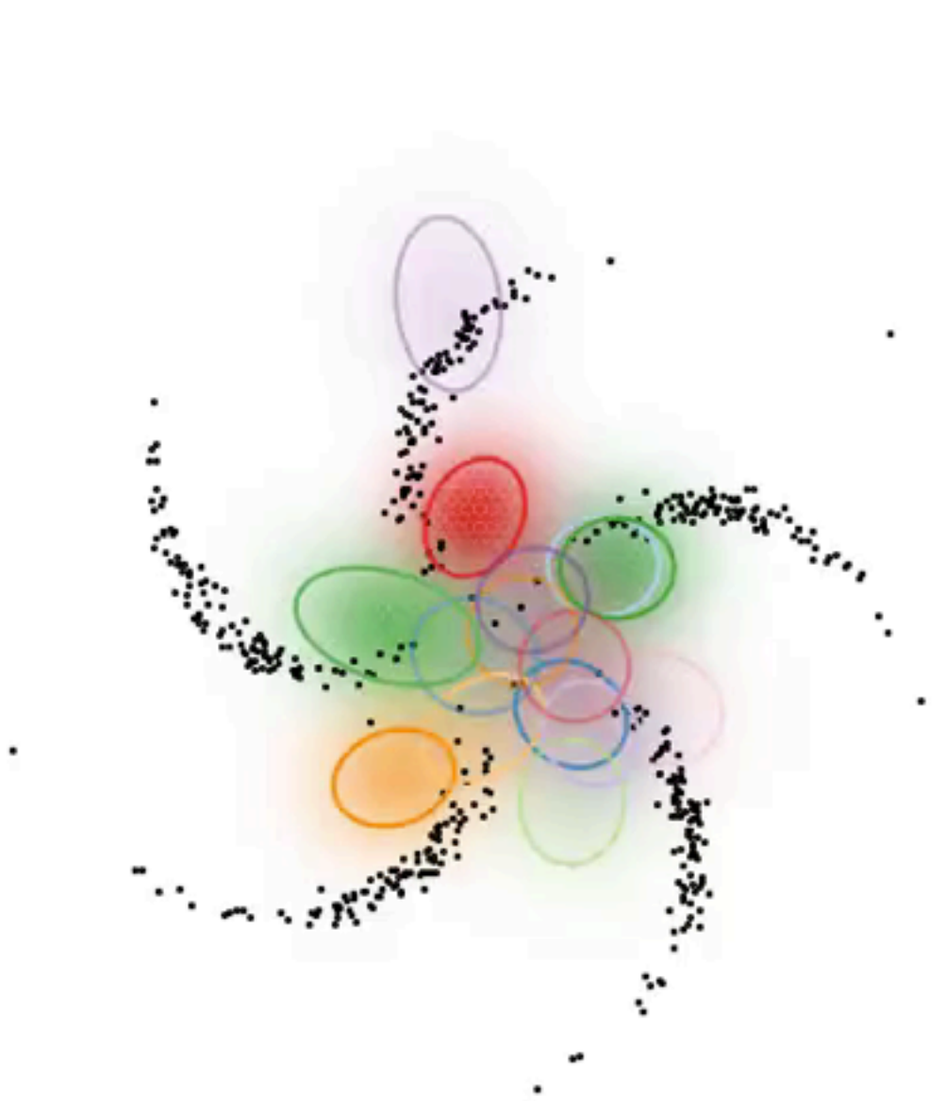
Canonical correlations analysis



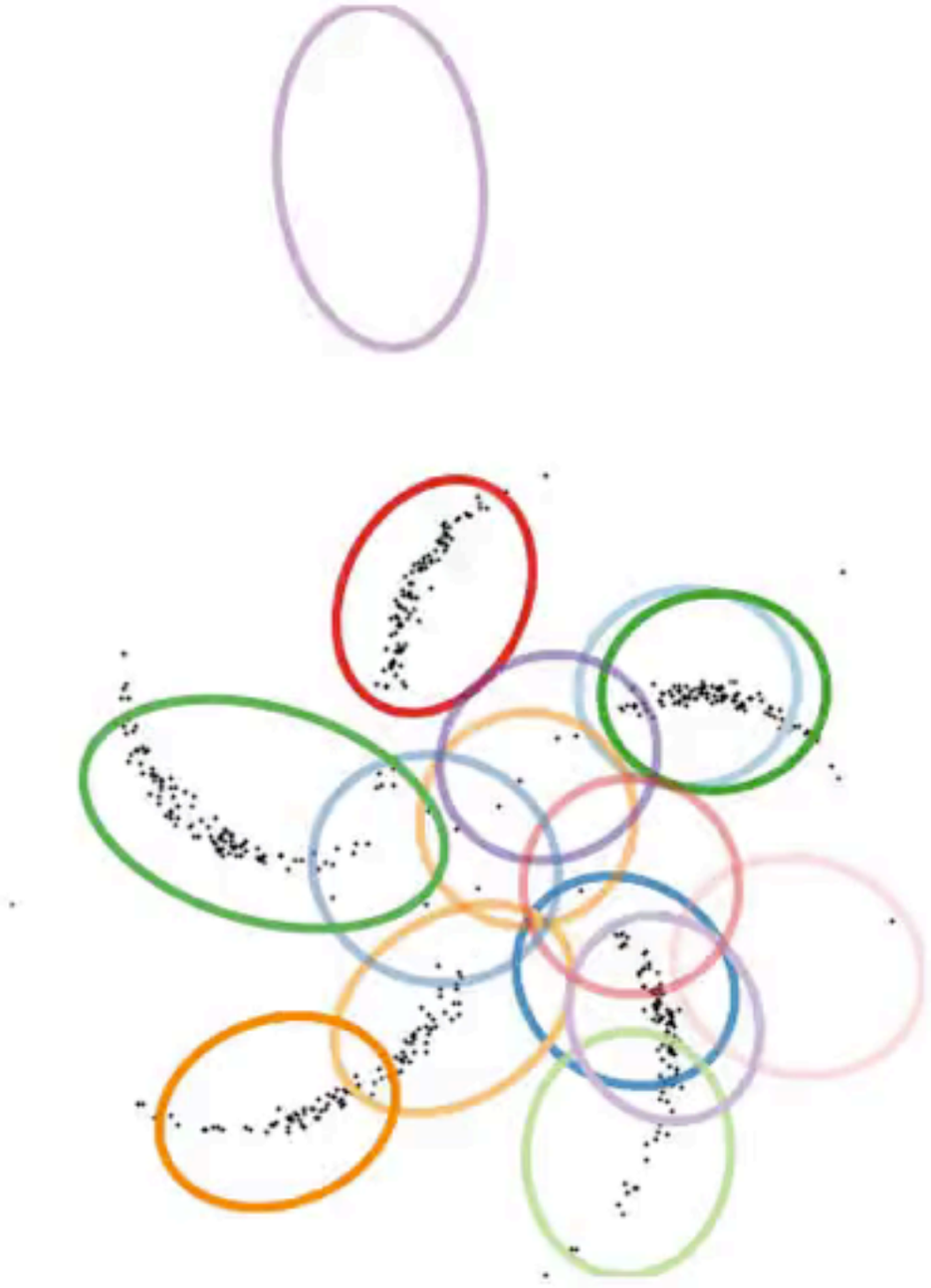
[10]

admixture / LDA / NMF

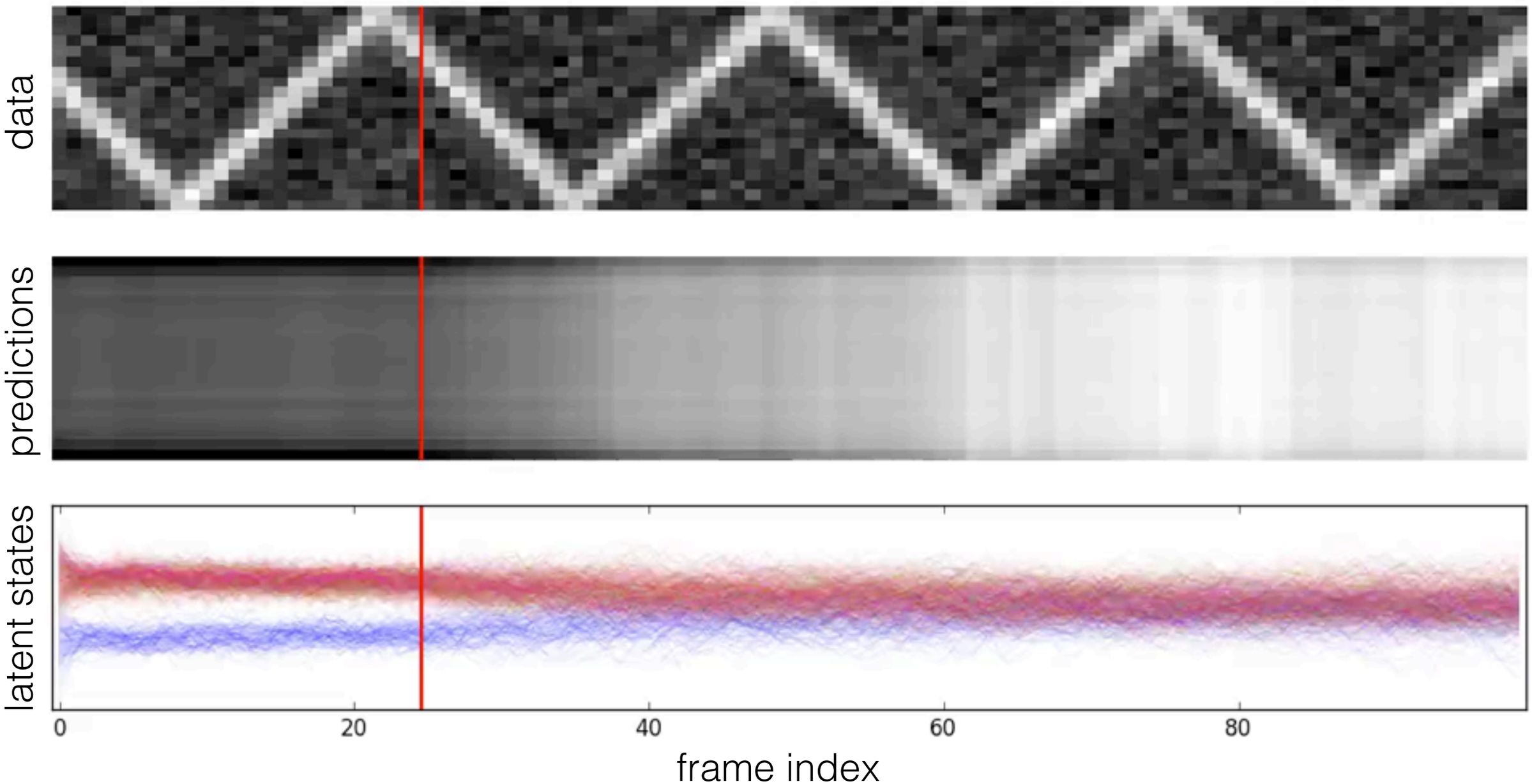
[1] Palmer, Wipf, Kreutz-Delgado, and Rao. Variational EM algorithms for non-Gaussian latent variable models. NIPS 2005.
[2] Ghahramani and Beal. Propagation algorithms for variational Bayesian learning. NIPS 2001.
[3] Beal. Variational algorithms for approximate Bayesian inference, Ch. 3. U of London Ph.D. Thesis 2003.
[4] Ghahramani and Hinton. Variational learning for switching state-space models. Neural Computation 2000.
[5] Jordan and Jacobs. Hierarchical Mixtures of Experts and the EM algorithm. Neural Computation 1994.
[6] Bengio and Frasconi. An Input Output HMM Architecture. NIPS 1995.
[7] Ghahramani and Jordan. Factorial Hidden Markov Models. Machine Learning 1997.
[8] Bach and Jordan. A probabilistic interpretation of Canonical Correlation Analysis. Tech. Report 2005.
[9] Archambeau and Bach. Sparse probabilistic projections. NIPS 2008.
[10] Hoffman, Bach, Blei. Online learning for Latent Dirichlet Allocation. NIPS 2010.

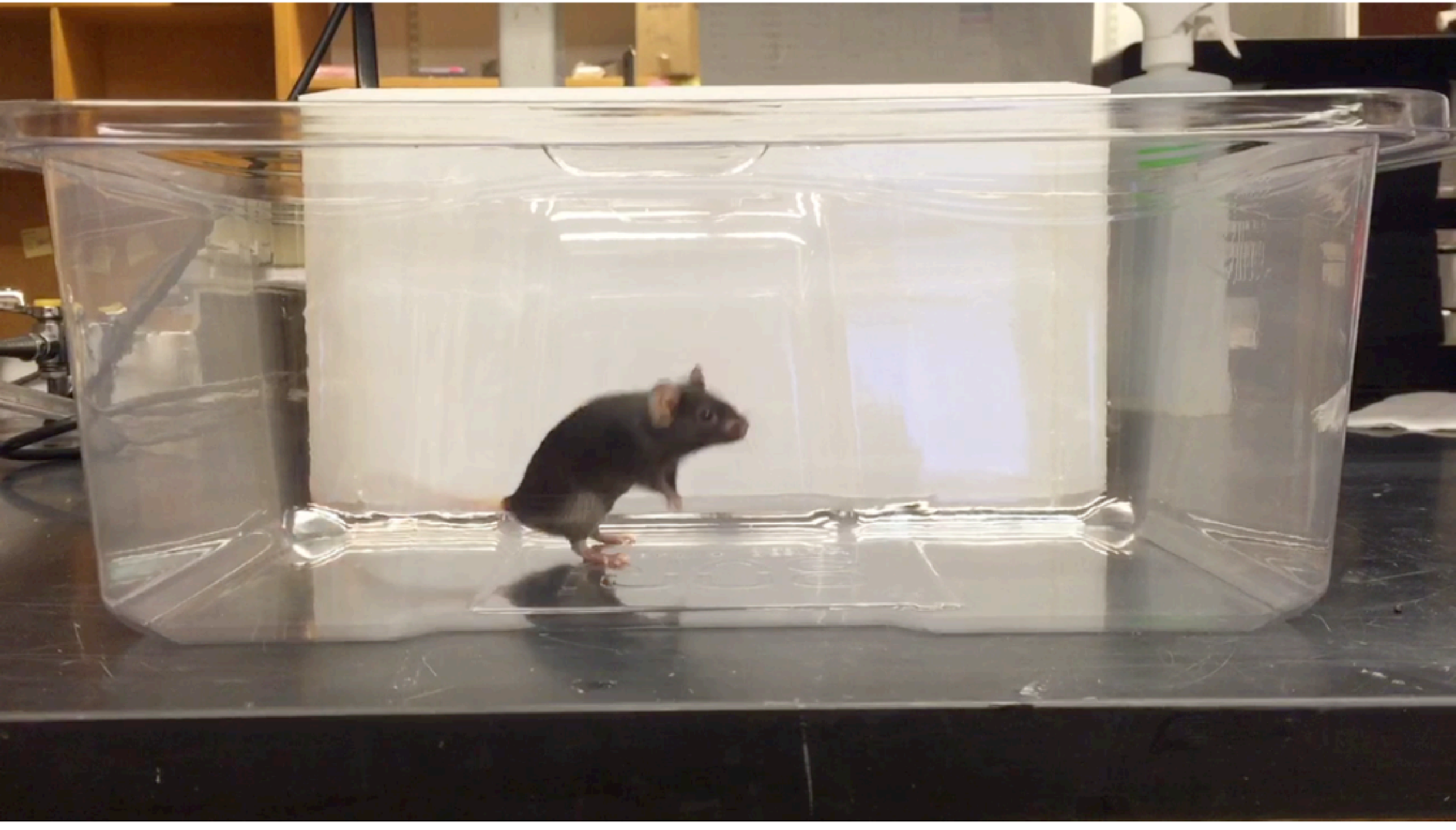


data space

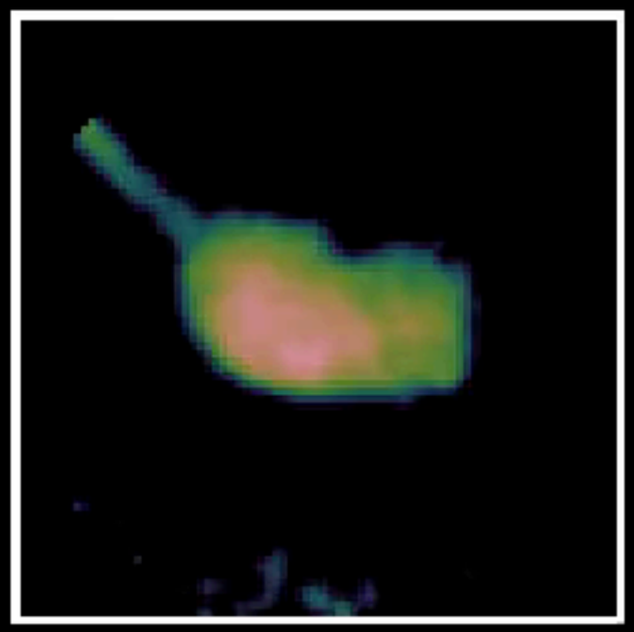


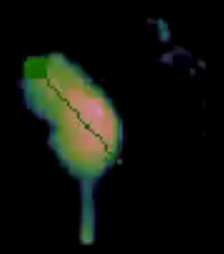
latent space


data
predictions
latent states
0
20
40
60
80
frame index

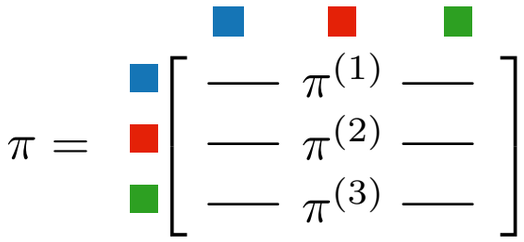


Frame 0

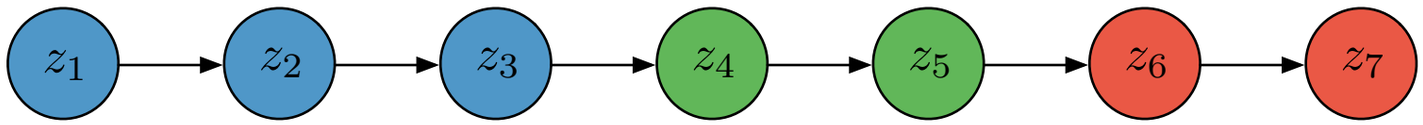


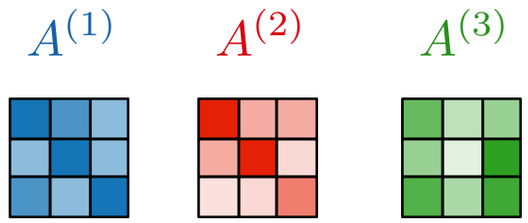


Alexander Wiltschko, Matthew Johnson, et al., Neuron 2015.

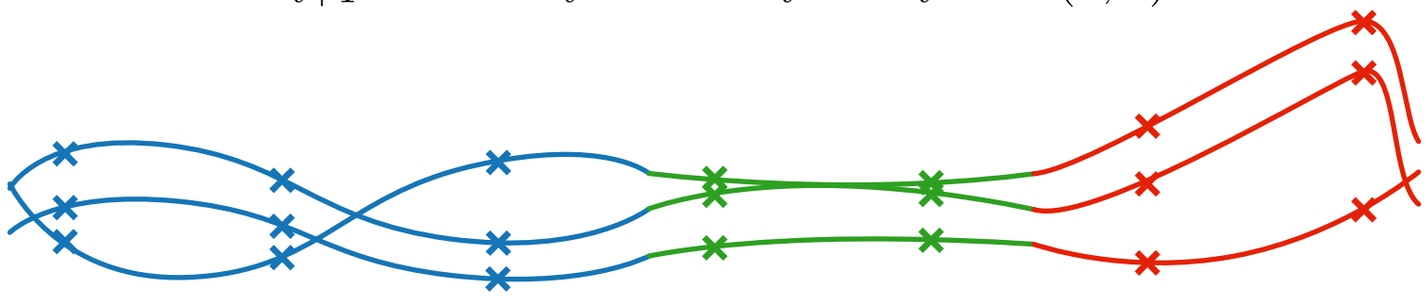


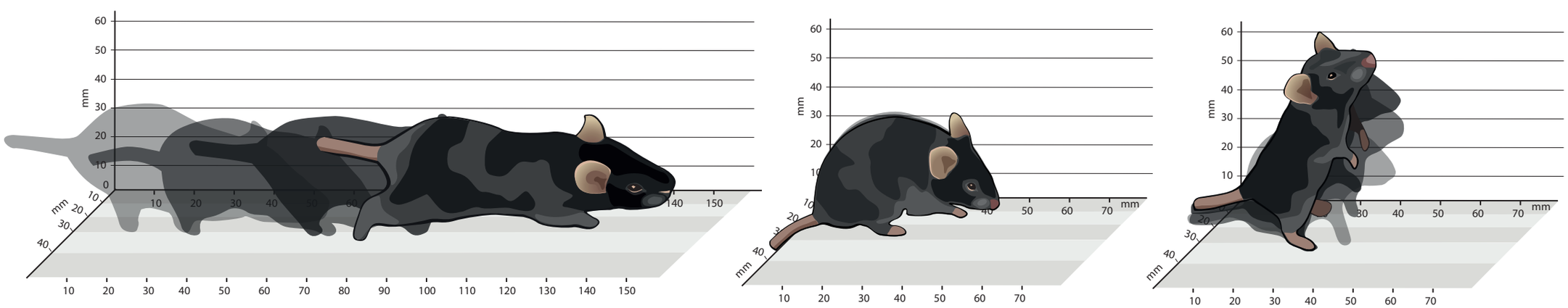
$$z_{t+1} \sim \pi^{(z_t)}$$

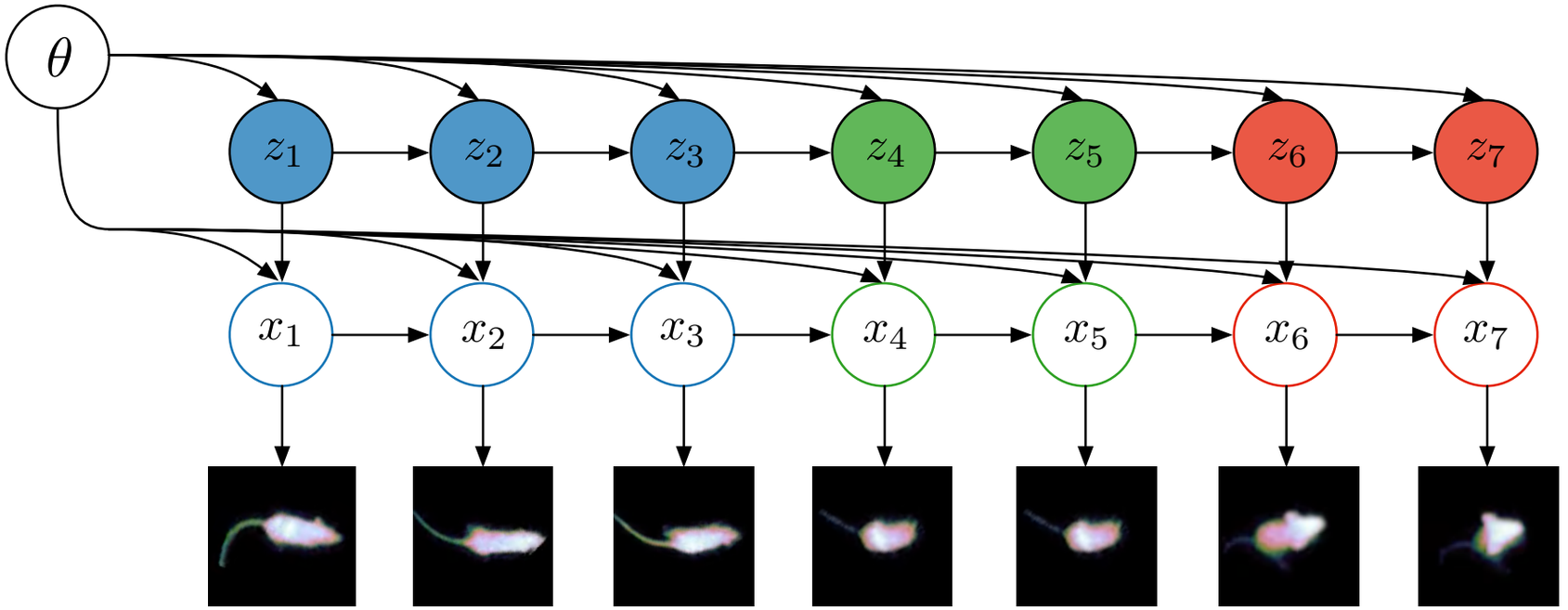


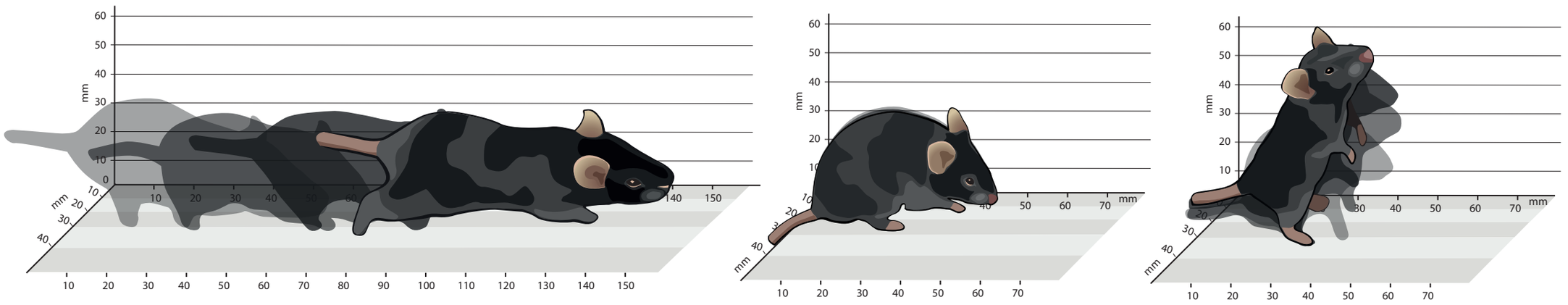


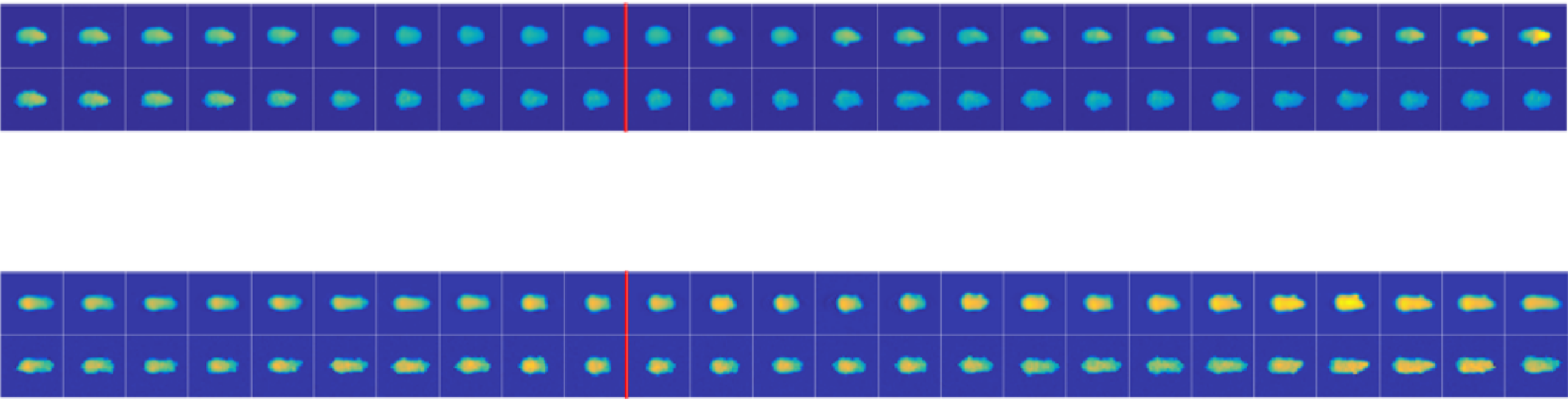
$$x_{t+1} = A^{(z_t)} x_t + B^{(z_t)} u_t \qquad u_t \overset{\text{iid}}{\sim} \mathcal{N}(0, I)$$

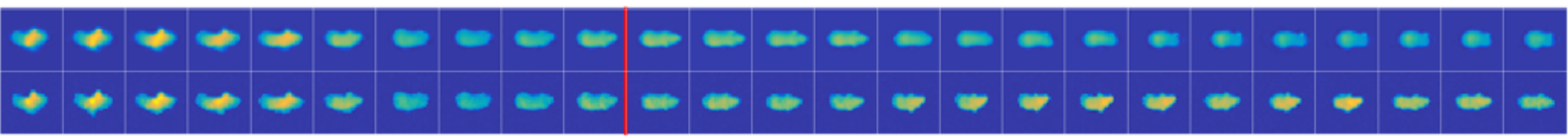


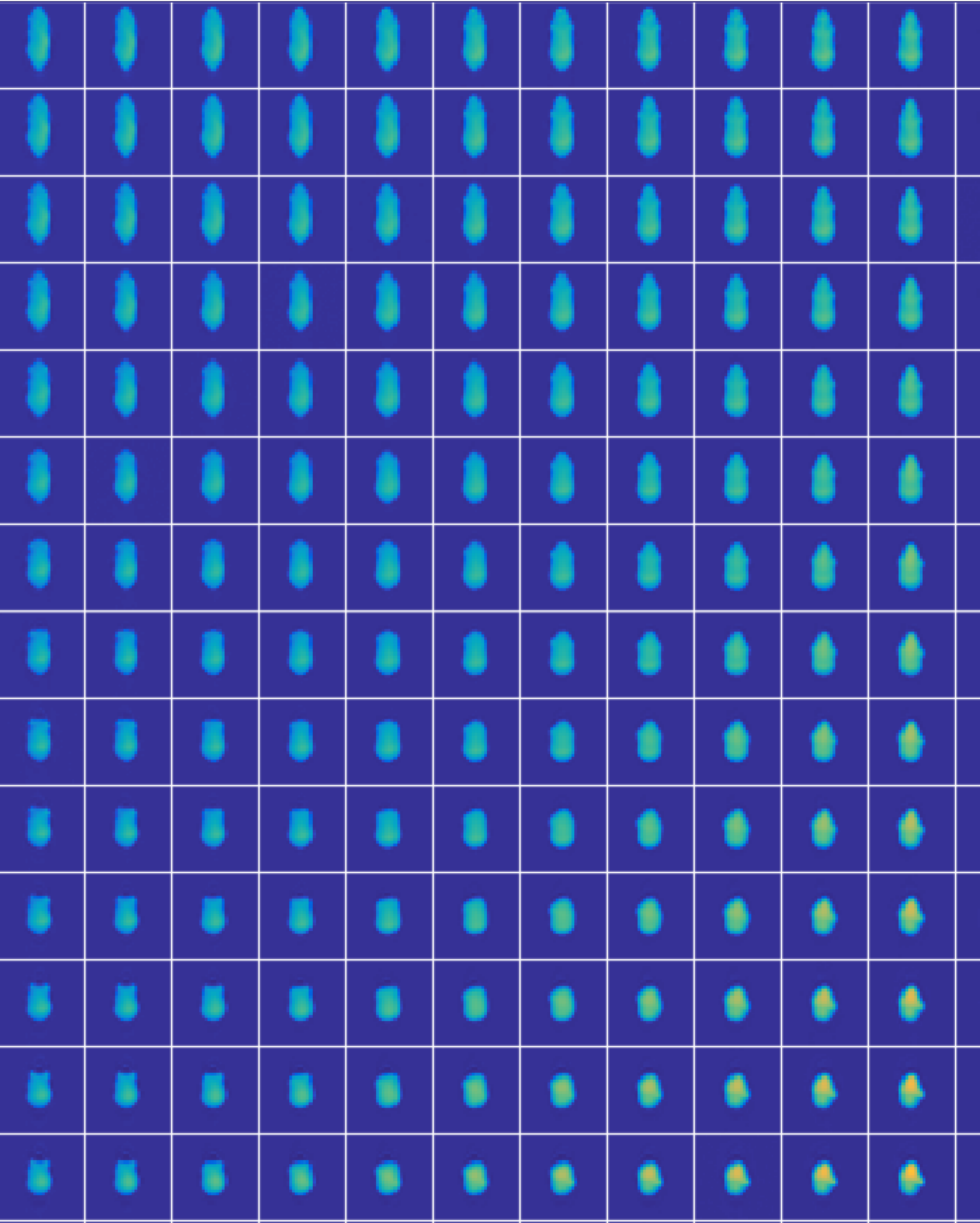


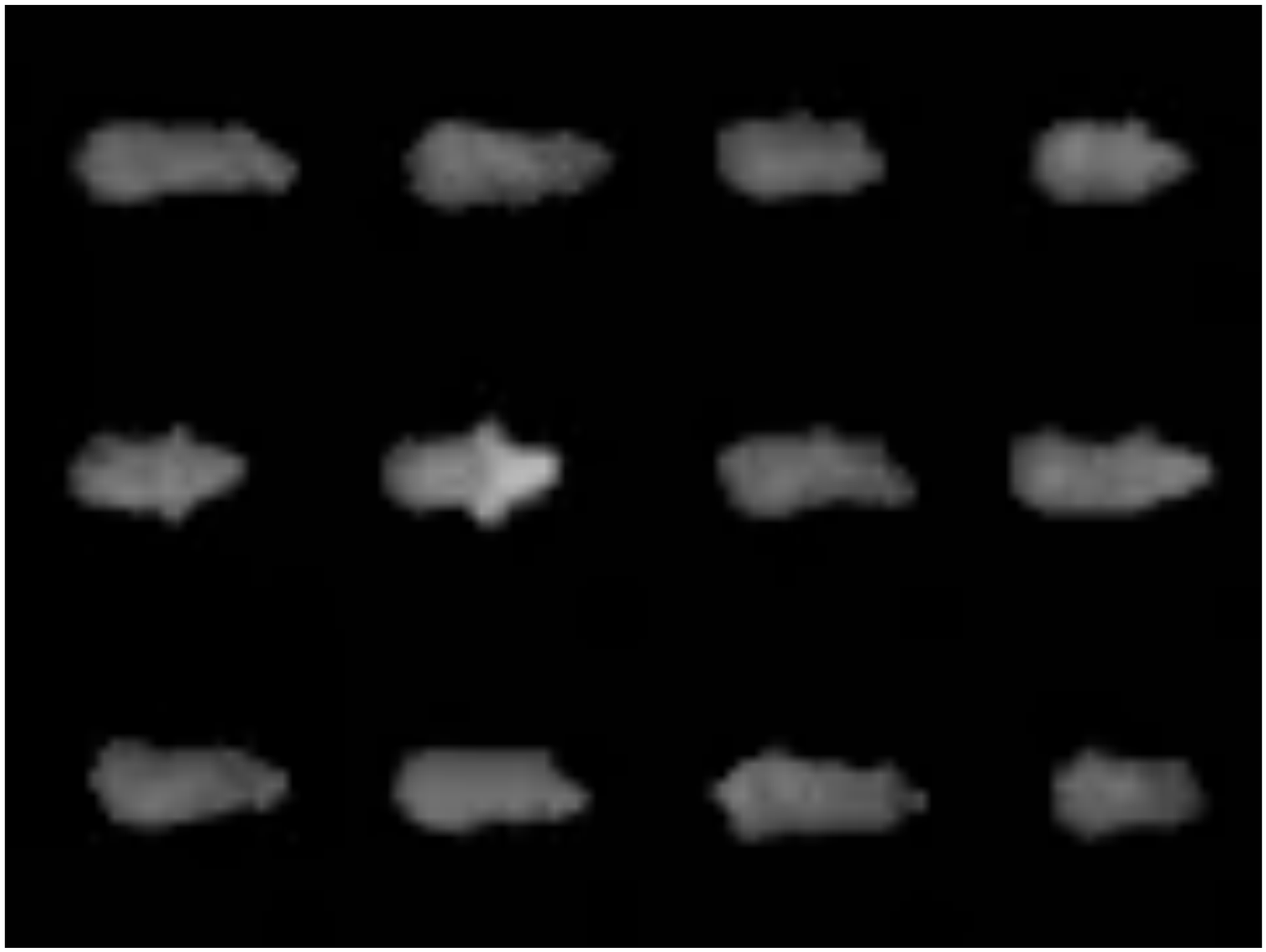

θ
z_1
z_2
z_3
z_4
z_5
z_6
z_7
x_1
x_2
x_3
x_4
x_5
x_6
x_7


mm
mm
mm

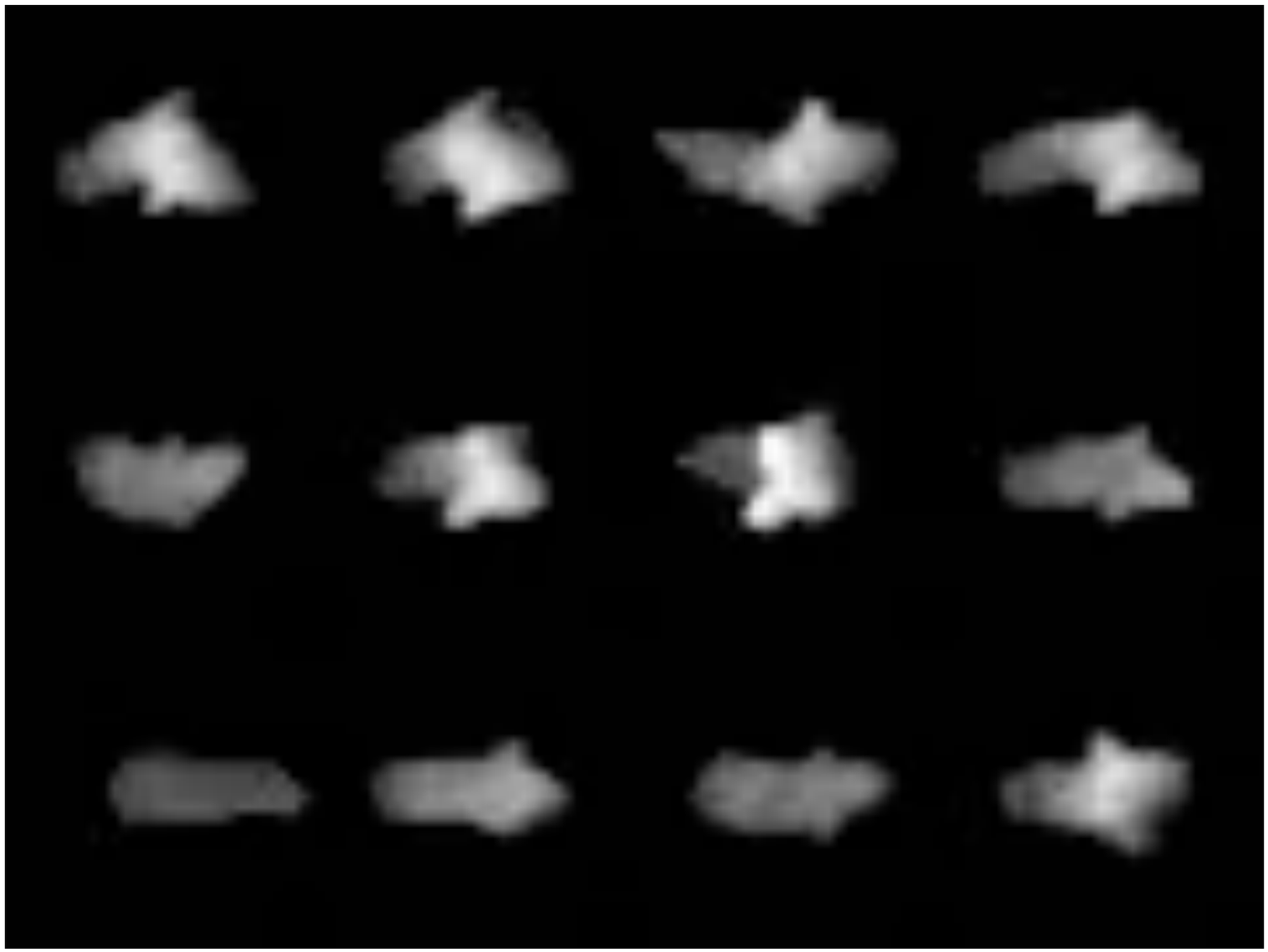




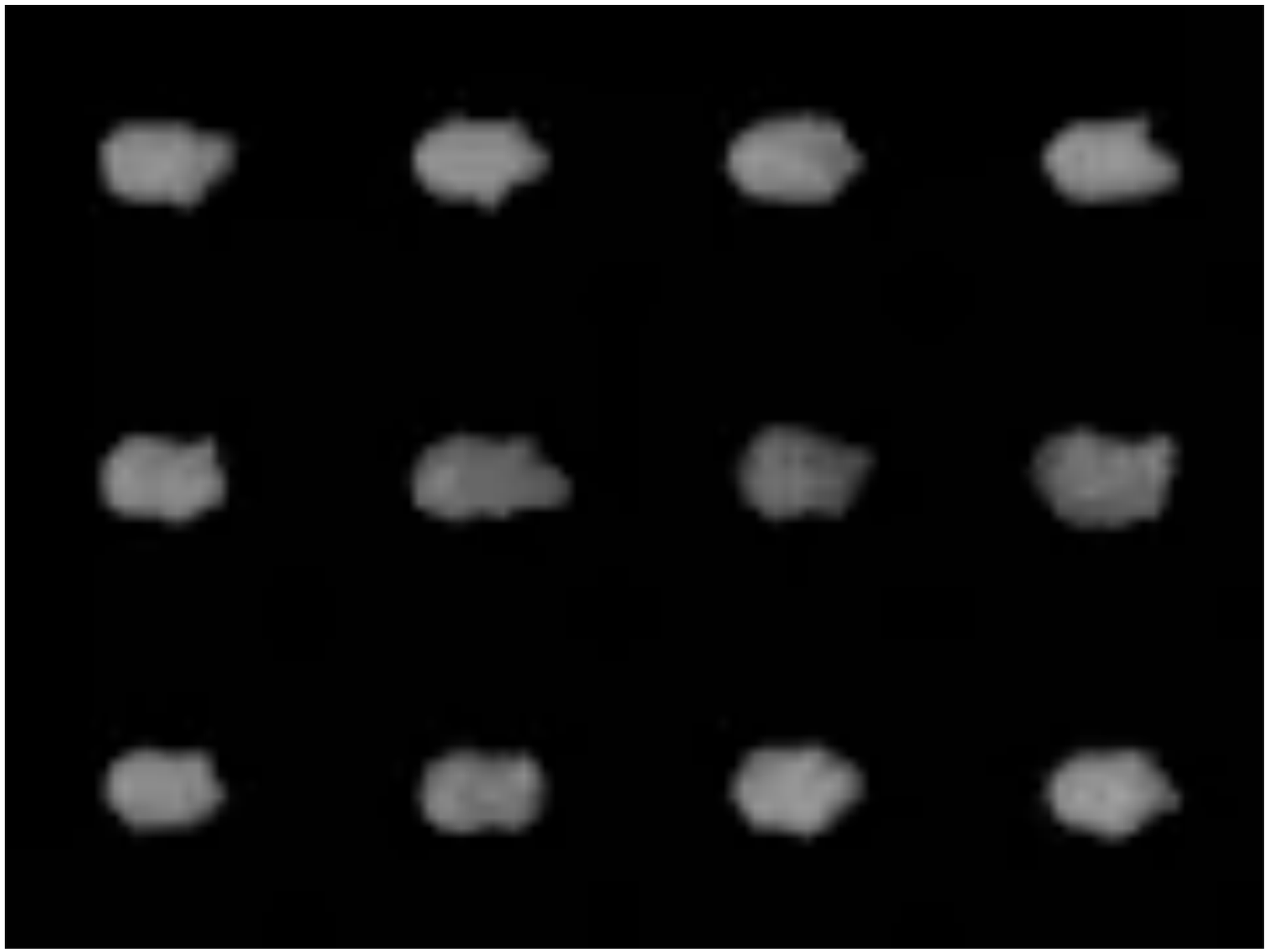




rearing up



fall from rear



grooming

Limitations and future work

capacity

- How expressive is latent linear structure?
  - word embeddings [1], analogical reasoning in image models
  - SVAE can use nonlinear latent structure

complexity

- PGMs get complicated
  - SVAE keeps complexity modular

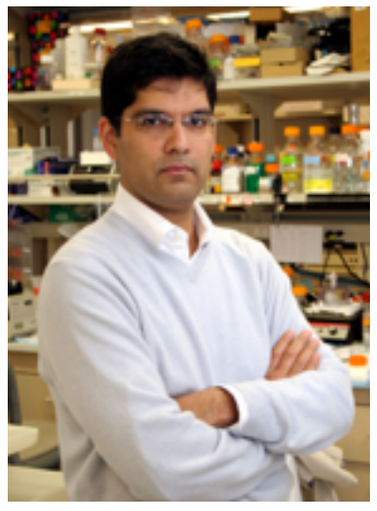
future work

- model-based reinforcement learning
- automatic structure search [2,3]
- semi-supervised applications

[1] Hashimoto, Alvarez-Melis, and Jaakkola, Word, graph and manifold embedding from Markov processes, Preprint 2015.
[2] Grosse et al., Exploiting compositionality to explore a large space of model structures, UAI 2012.
[3] Duvenaud et al., Structure discovery in nonparametric regression through compositional kernel search, ICML 2013.

Matt Johnson, David Duvenaud, Alex Wiltschko, Bob Datta, Ryan Adams




Thanks!

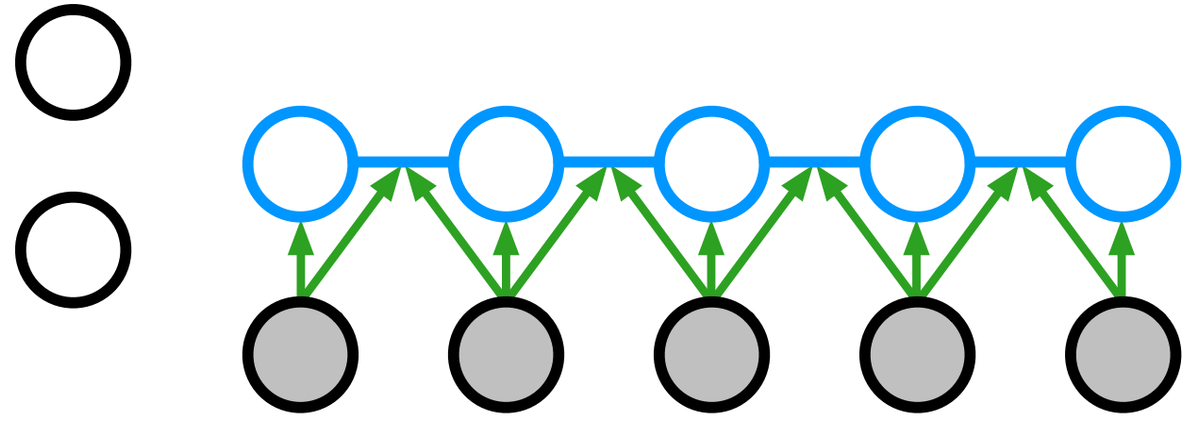
github.com/mattjj/svae

HARVARD
UNIVERSITY

UNIVERSITY OF
TORONTO

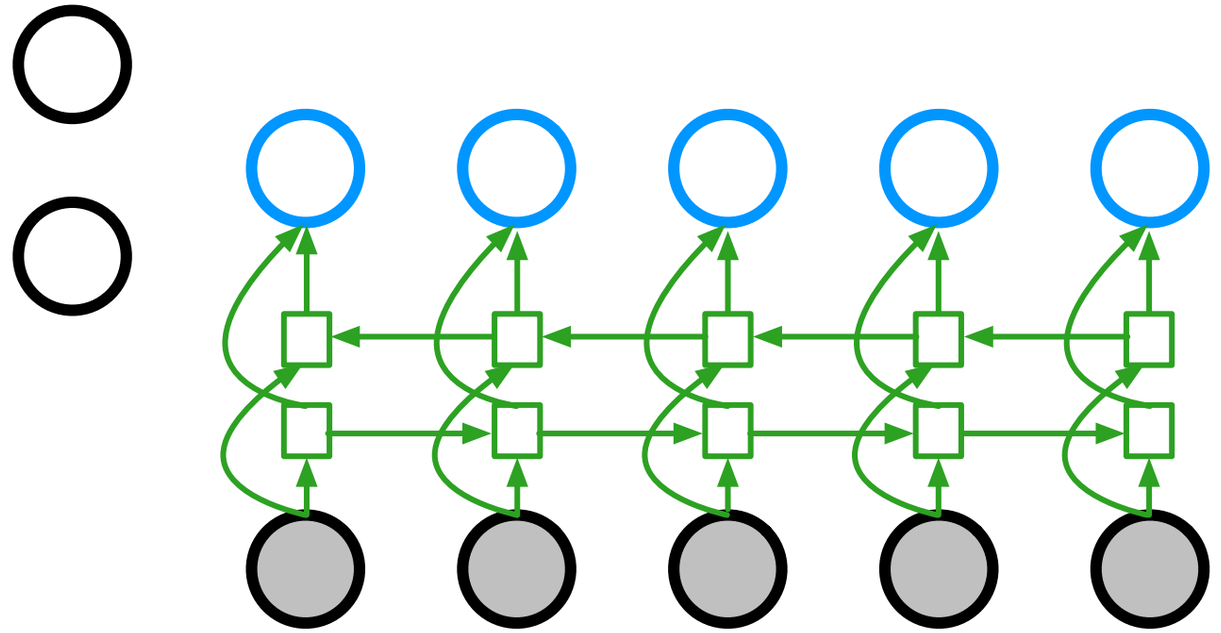
[1,2]

$$\mu_t(y_t; \phi_\mu)$$
$$J_{t,t}(y_t; \phi_D)$$
$$J_{t,t+1}(y_t, y_{t+1}; \phi_B)$$



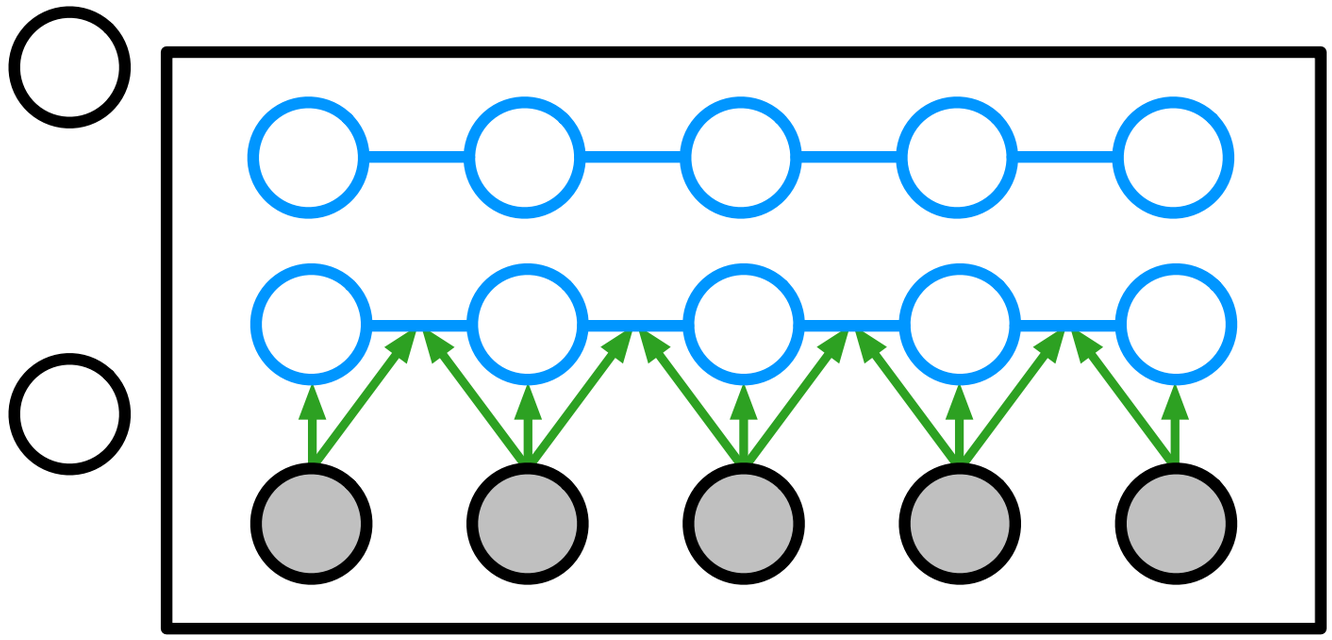
[3]

$$\mu_t(y_{1:T}, \hat{x}_{t-1}; \phi)$$
$$\Sigma_t(y_{1:T}, \hat{x}_{t-1}; \phi)$$



[1] Archer, Park, Buesing, Cunningham, Paninski. Black box variational inference for state space models. ICLR 2016 Workshops.
[2] Gao*, Archer*, Paninski, Cunningham. Linear dynamical neural population models through nonlinear embeddings. NIPS 2016.
[3] Krishnan, Shalit, Sontag. Structured inference networks for nonlinear state space models. AISTATS 2017.

SVAEs can use any inference network architecture



[1] Archer, Park, Buesing, Cunningham, Paninski. Black box variational inference for state space models. ICLR 2016 Workshops.
[2] Gao*, Archer*, Paninski, Cunningham. Linear dynamical neural population models through nonlinear embeddings. NIPS 2016.

Per-variable recognition nets allow arbitrary inference queries