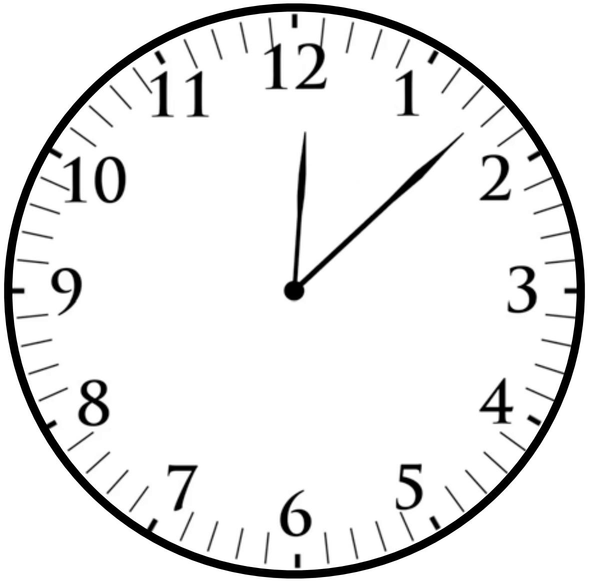
DEEP SEQUENTIAL LATENT VARIABLE MODELS

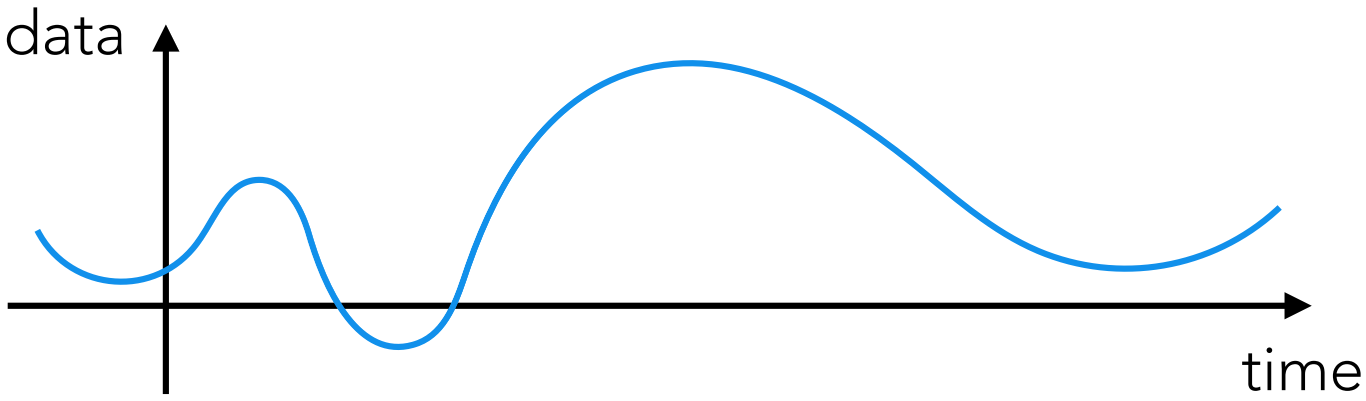
JOSEPH MARINO

CALTECH

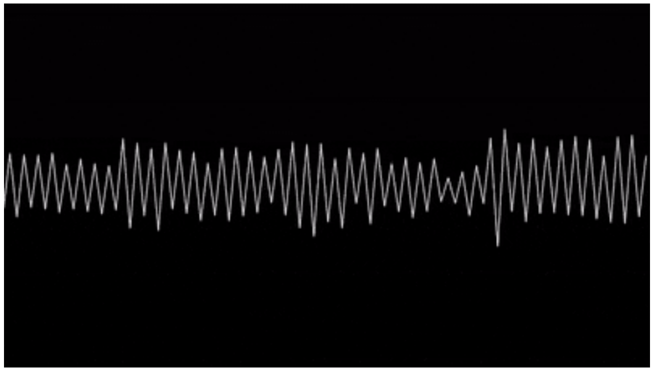


time is a fundamental aspect of the universe

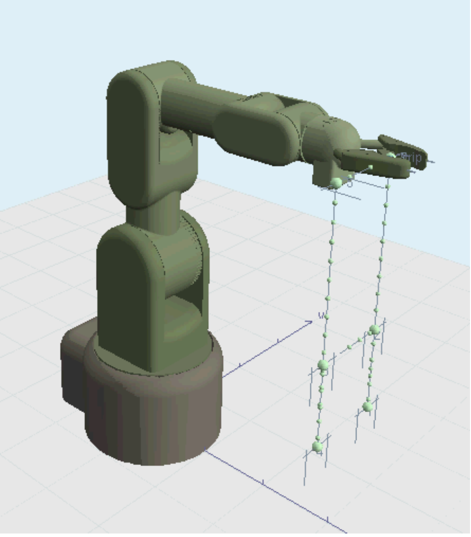
observed data are sequential



audio

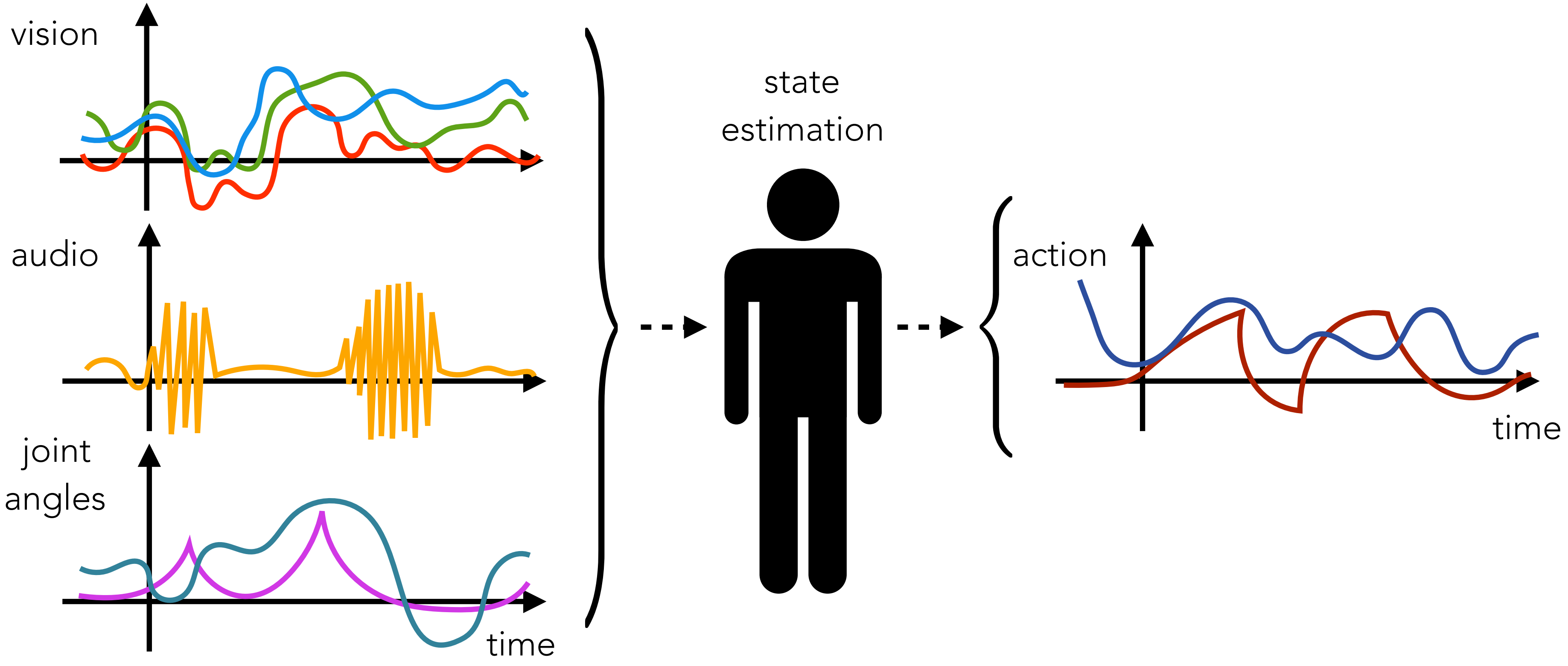


vision



joint angles

interacting in the world involves processing sequences of data



vision

audio

joint angles

time

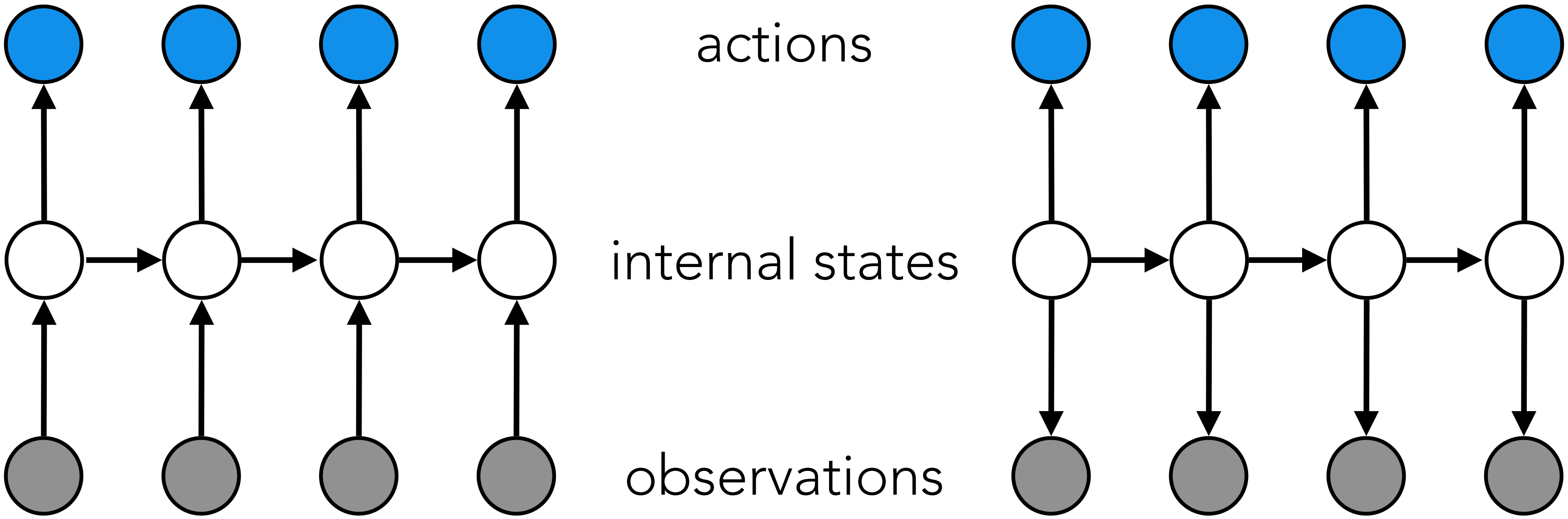
state estimation

action

time

COMPUTATIONAL APPROACHES TO STATE ESTIMATION

discriminative **generative**



actions

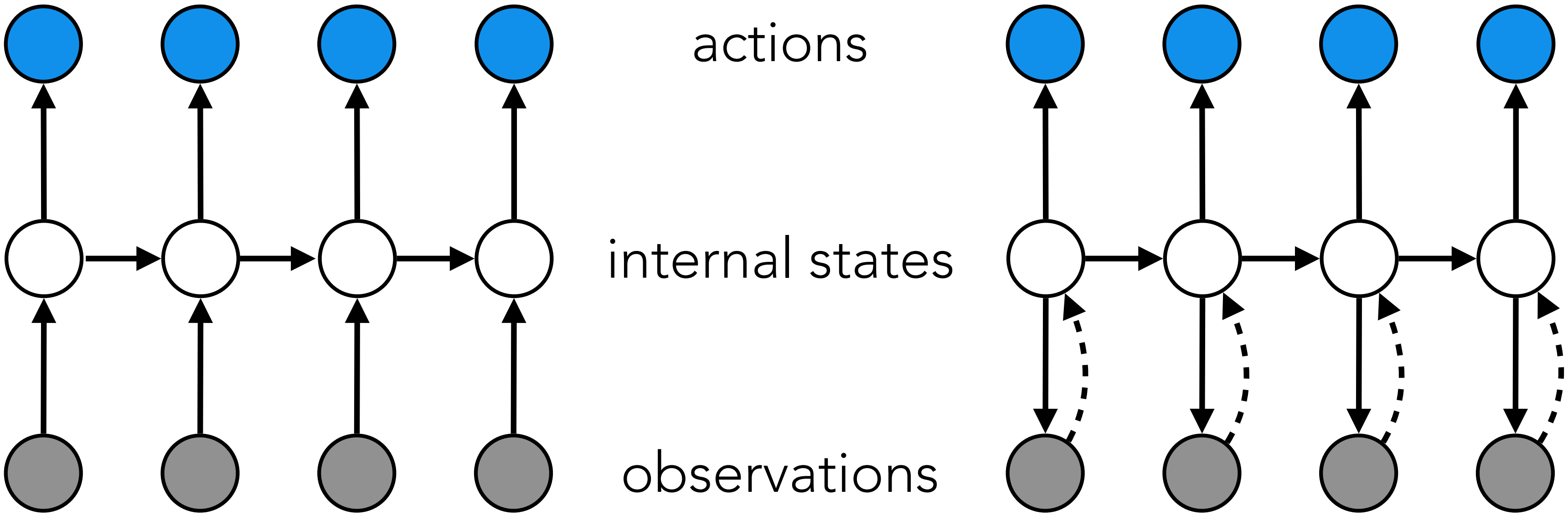
internal states

observations

COMPUTATIONAL APPROACHES TO STATE ESTIMATION

discriminative

generative



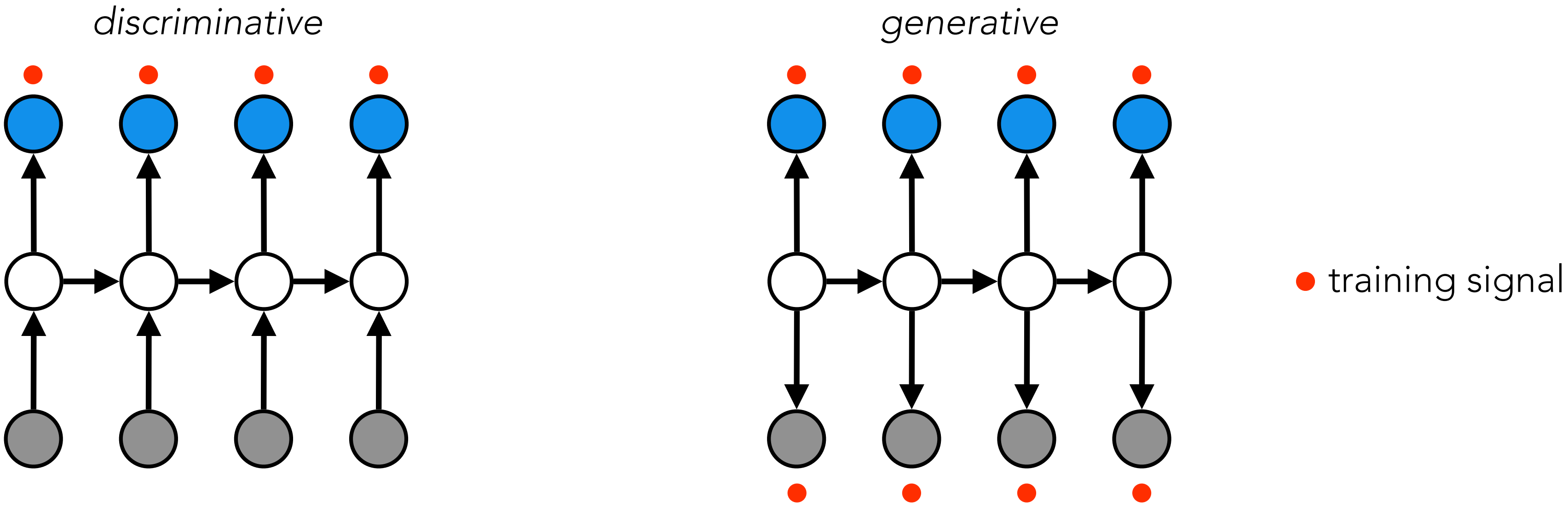
actions

internal states

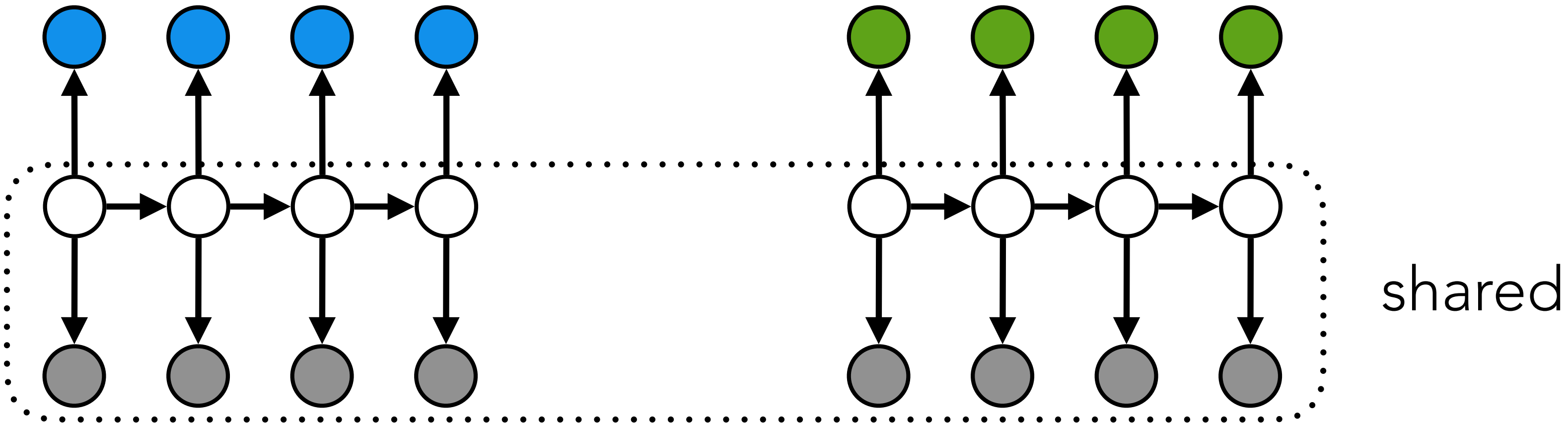
observations

ADVANTAGES OF GENERATIVE MODELING

unsupervised learning: *learn from the data*

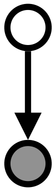


generalization: *learn a task-agnostic representation*

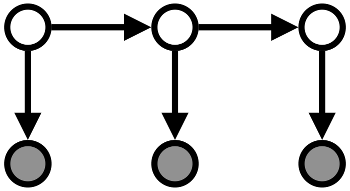


OUTLINE

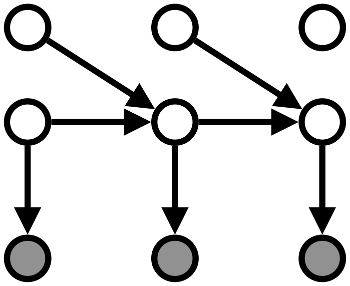
BACKGROUND

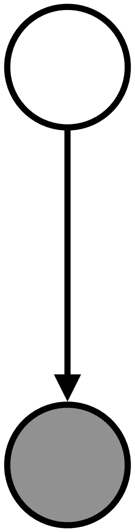


DEEP SEQUENTIAL LATENT VARIABLE MODELS



MODEL-BASED RL

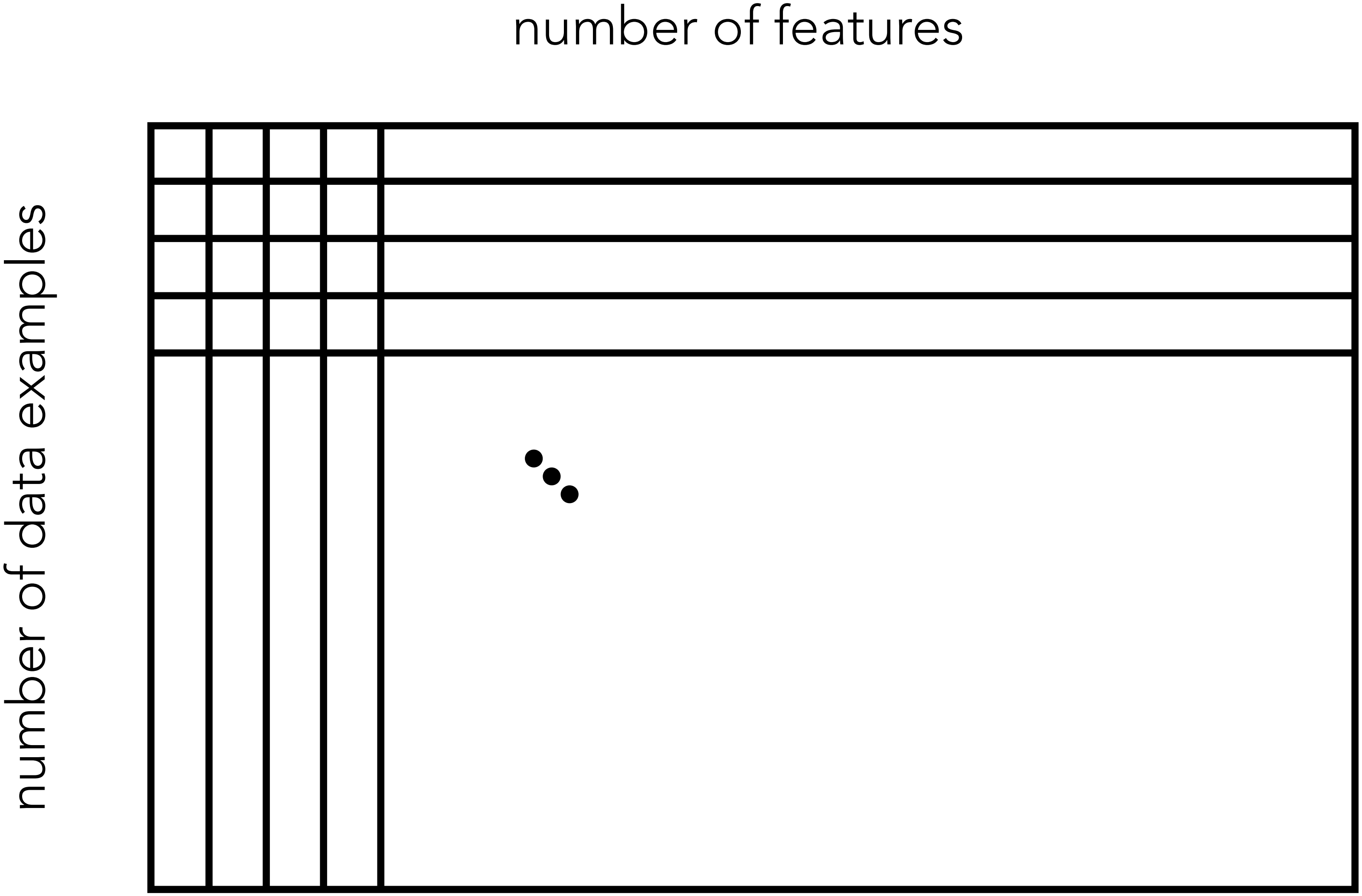




BACKGROUND

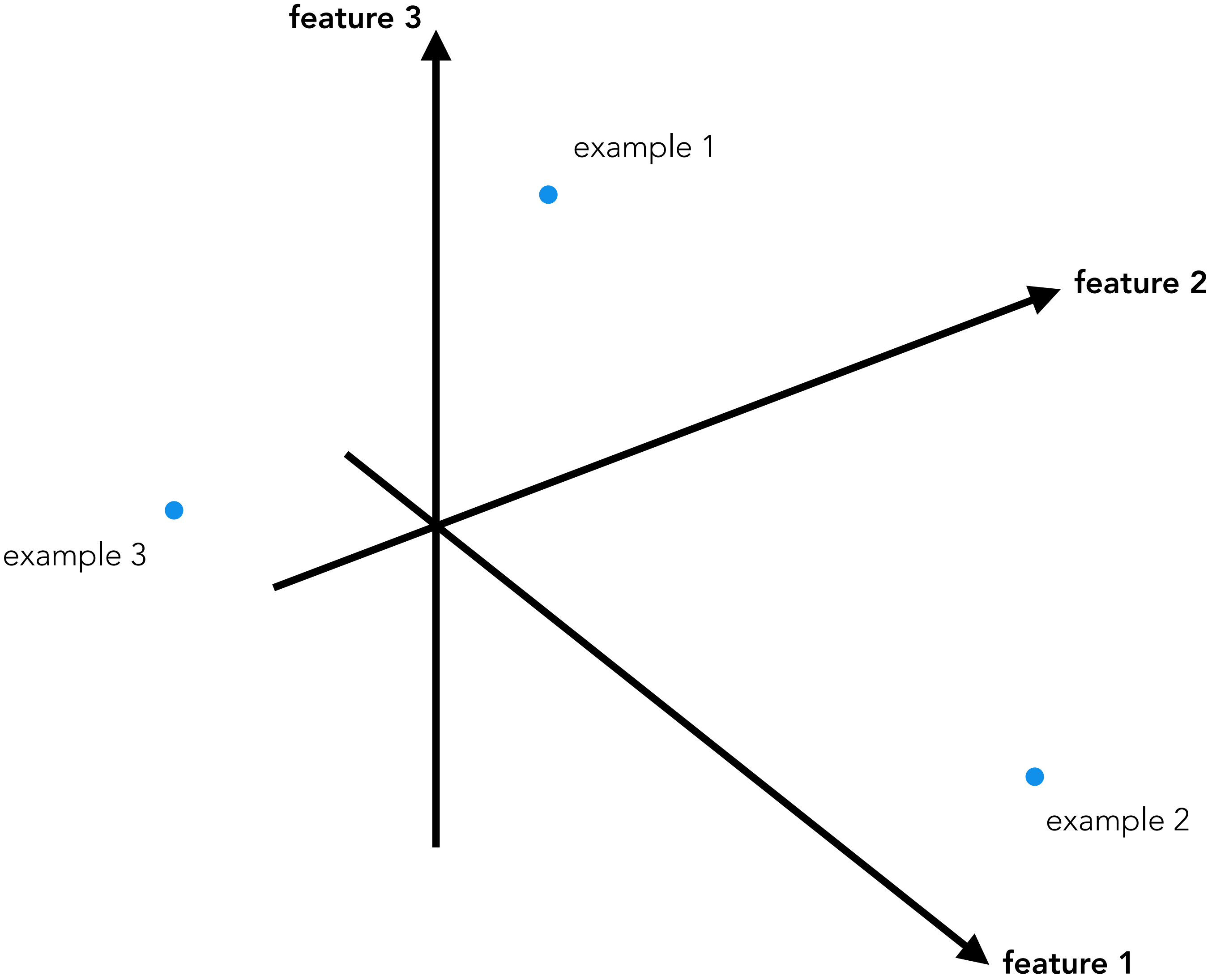
GENERATIVE MODEL

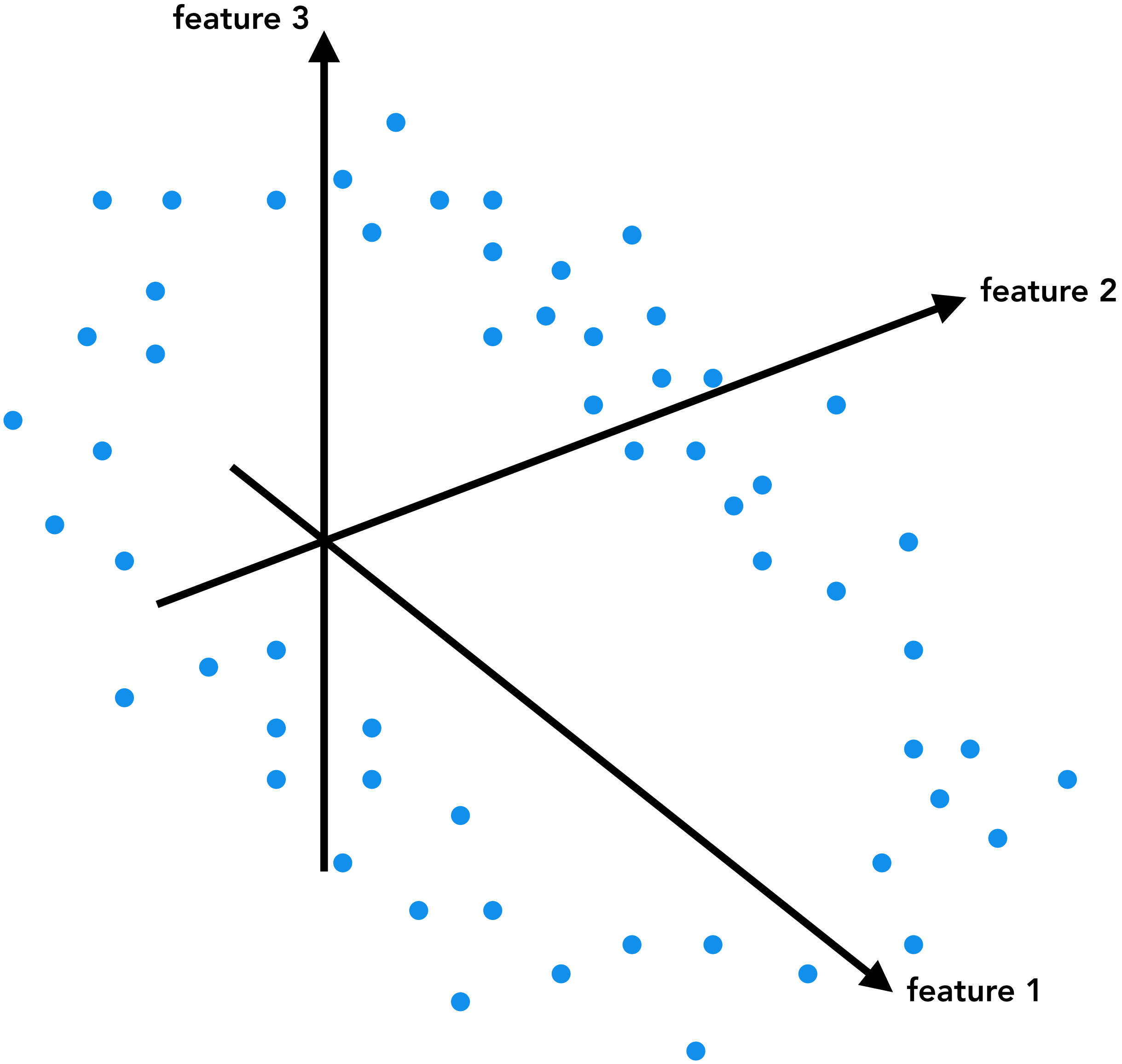
a model of the density of observed data number of features



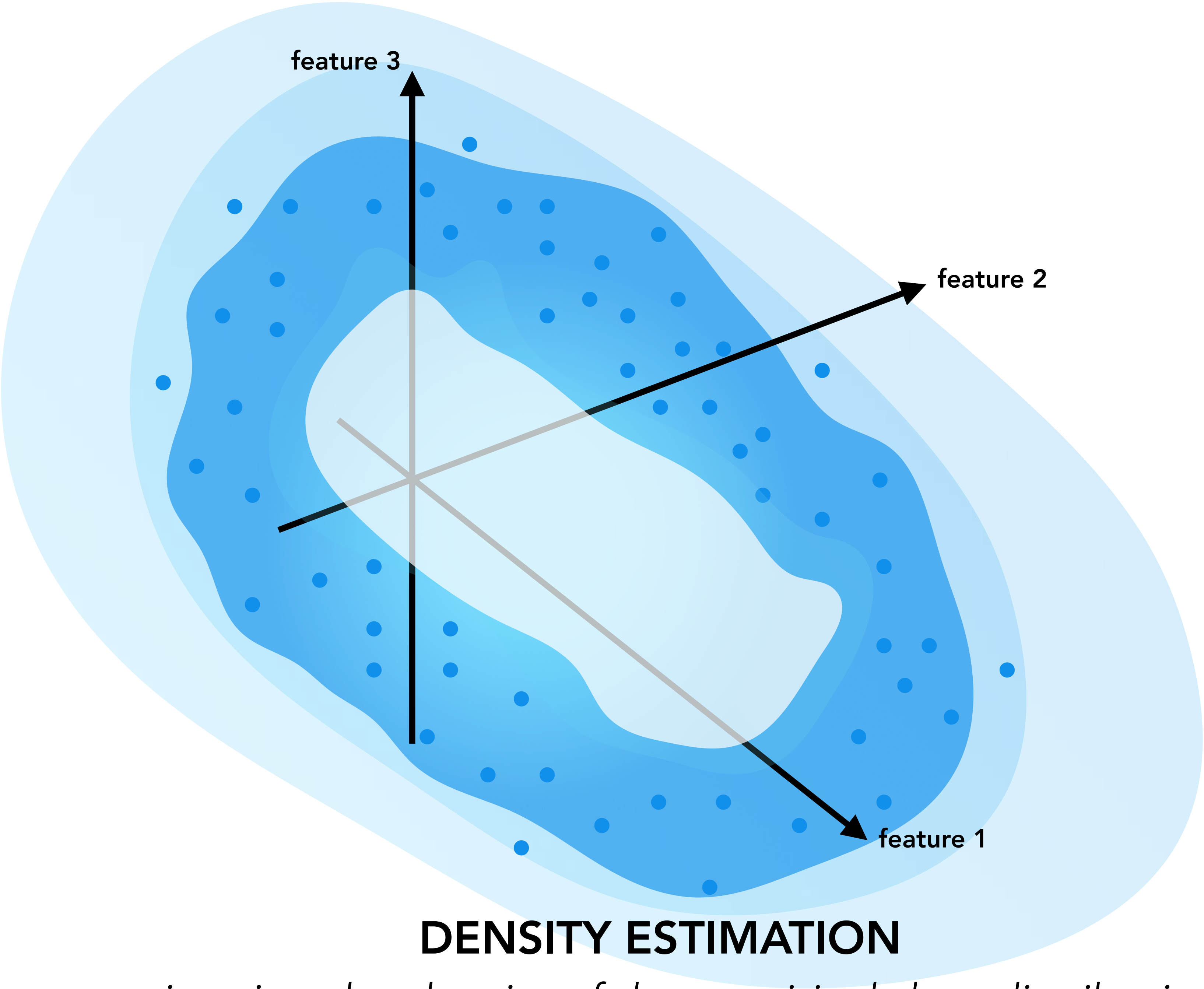
number of data examples

DATA


feature 3
example 1
feature 2
example 3
example 2
feature 1



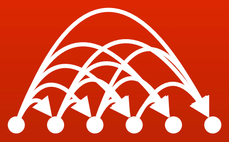
EMPIRICAL DATA DISTRIBUTION



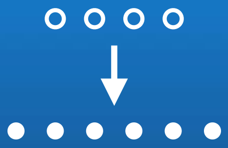
DENSITY ESTIMATION

estimating the density of the empirical data distribution

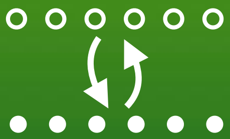
FAMILIES OF GENERATIVE MODELS



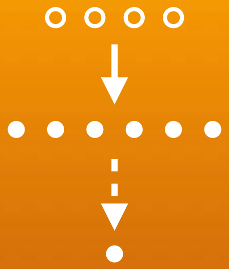
autoregressive models



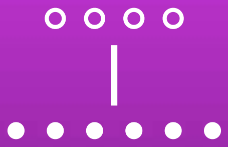
explicit
latent variable models



invertible
latent variable models



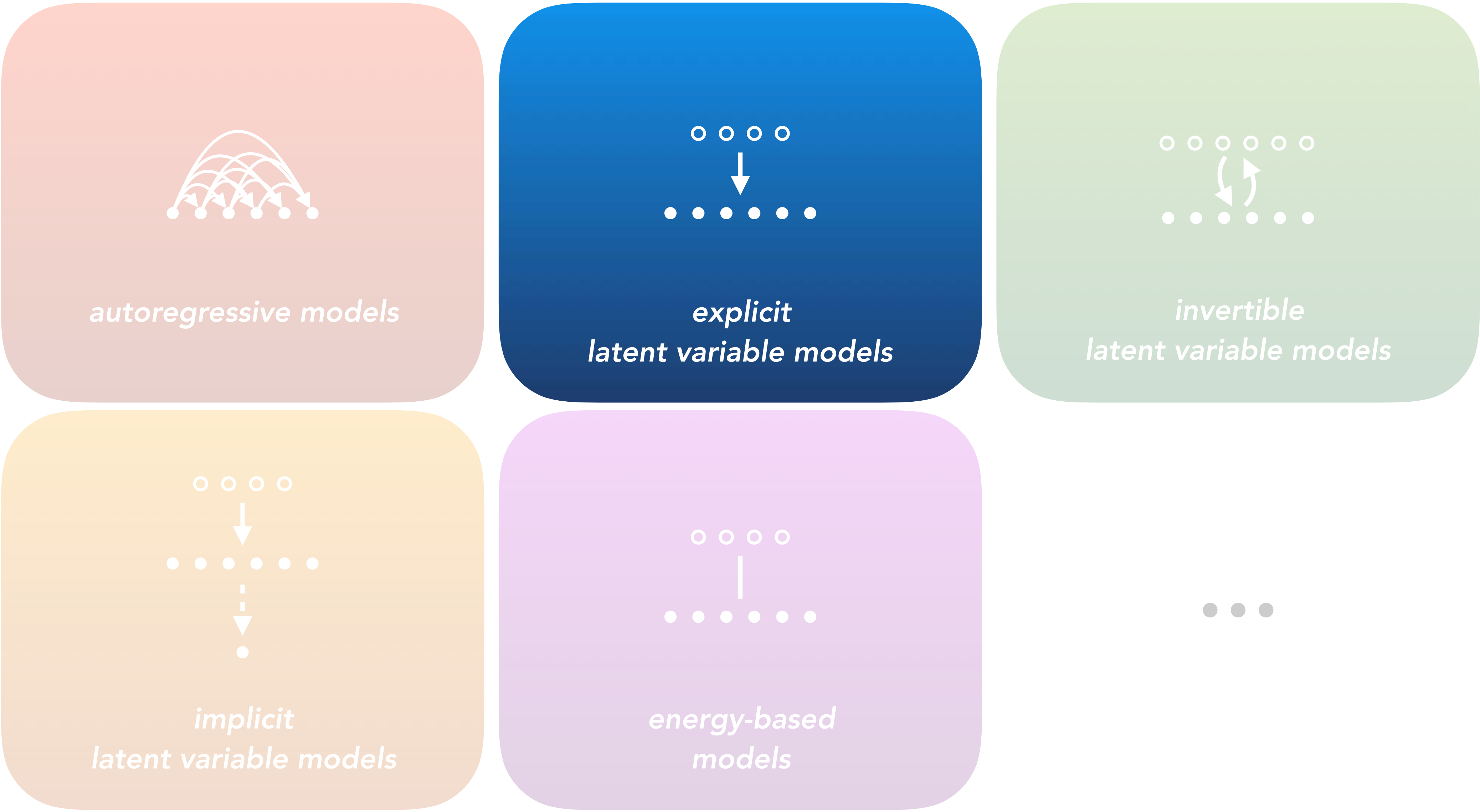
implicit
latent variable models



energy-based
models

• • •

FAMILIES OF GENERATIVE MODELS



autoregressive models

explicit
latent variable models

invertible
latent variable models

implicit
latent variable models

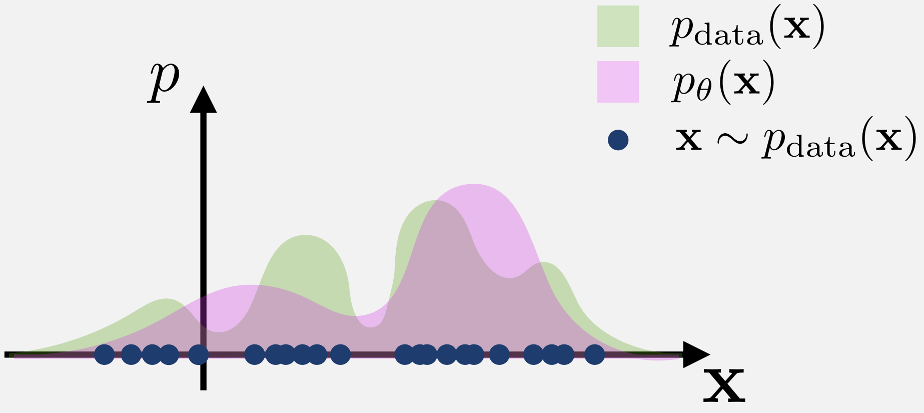
energy-based
models

MAXIMUM LIKELIHOOD

data: $p_{\text{data}}(\mathbf{x})$

model: $p_\theta(\mathbf{x})$

parameters: θ

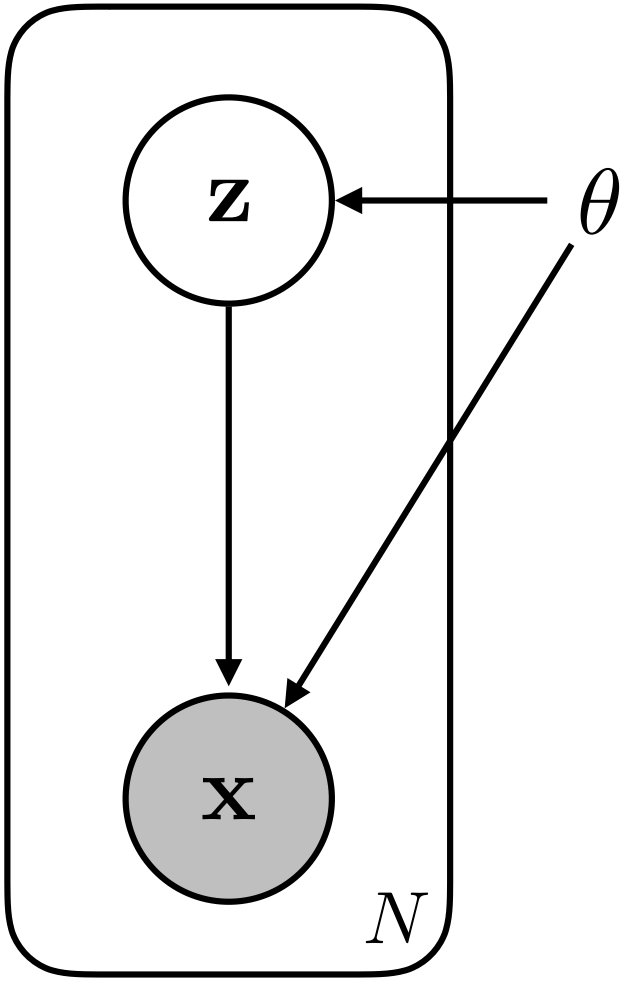


maximum likelihood estimation

find the model that assigns the *maximum likelihood* to the data

$$
\begin{aligned}
\theta^* &= \arg\min_\theta \; D_{KL}(p_{\text{data}}(\mathbf{x}) || p_\theta(\mathbf{x})) \\
&= \arg\min_\theta \; \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \left[\log p_{\text{data}}(\mathbf{x}) - \log p_\theta(\mathbf{x})\right] \\
&= \arg\max_\theta \; \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \left[\log p_\theta(\mathbf{x})\right] \approx \frac{1}{N} \sum_{i=1}^{N} \log p_\theta(\mathbf{x}^{(i)})
\end{aligned}
$$

LATENT VARIABLE MODELS



model:

$$\underbrace{p_\theta(\mathbf{x}, \mathbf{z})}_{\textit{joint}} = \underbrace{p_\theta(\mathbf{x}|\mathbf{z})}_{\substack{\textit{conditional} \\ \textit{likelihood}}} \underbrace{p_\theta(\mathbf{z})}_{\textit{prior}}$$

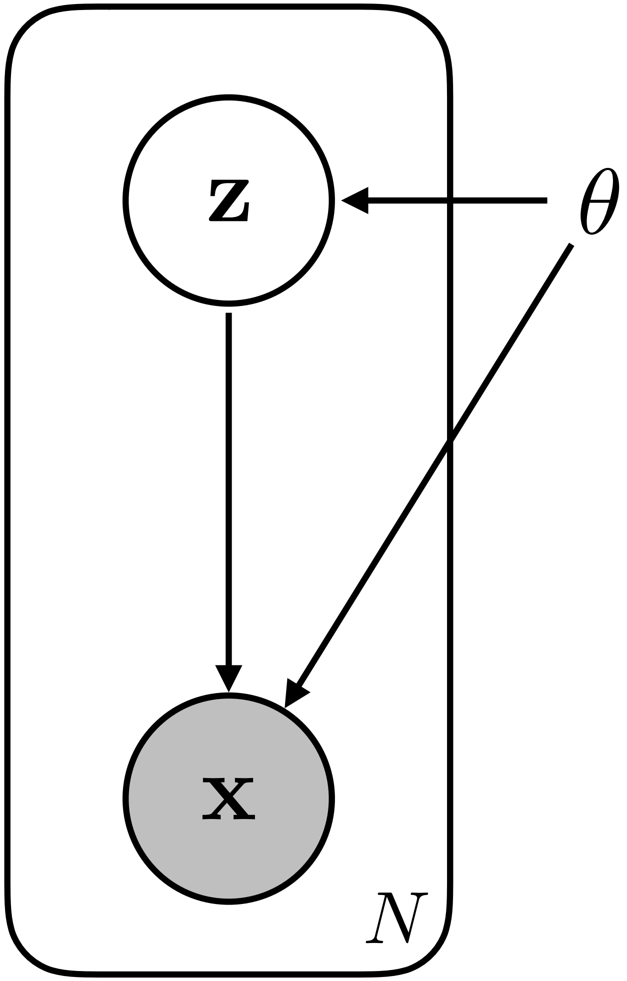
marginalization:

$$\underbrace{p_\theta(\mathbf{x})}_{\substack{\textit{marginal} \\ \textit{likelihood}}} = \int p_\theta(\mathbf{x}, \mathbf{z}) d\mathbf{z}$$

inference:

$$\underbrace{p_\theta(\mathbf{z}|\mathbf{x})}_{\textit{posterior}} = \frac{p_\theta(\mathbf{x}, \mathbf{z})}{p_\theta(\mathbf{x})}$$

LATENT VARIABLE MODELS



maximum likelihood is typically intractable

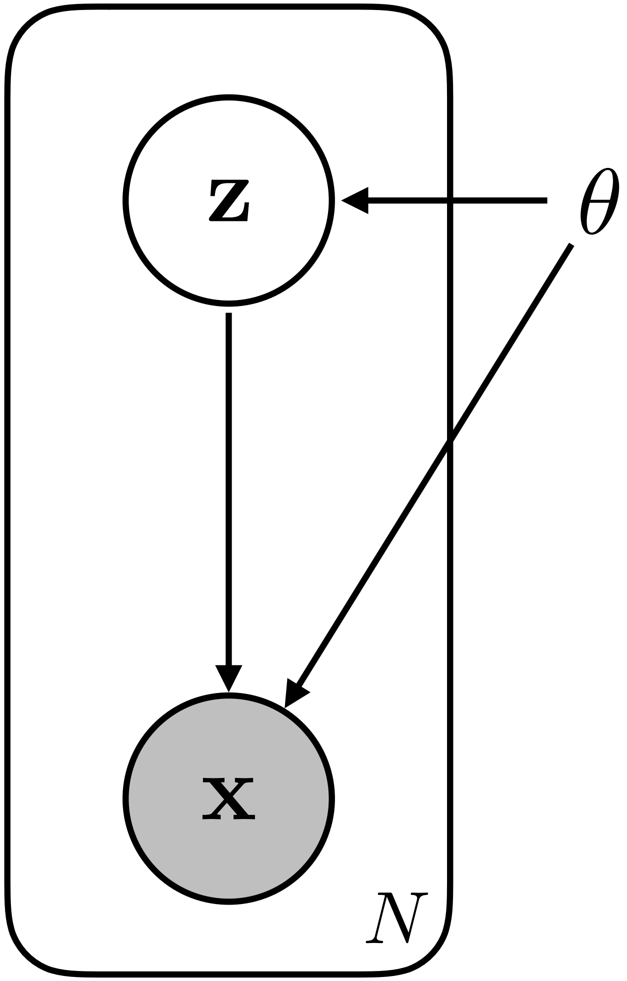
$$\theta^* = \arg\max_\theta \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\log p_\theta(\mathbf{x})\right]$$

$$\approx \arg\max_\theta \frac{1}{N}\sum_{i=1}^{N} \log p_\theta(\mathbf{x}^{(i)})$$

$$\approx \arg\max_\theta \frac{1}{N}\sum_{i=1}^{N} \log \underbrace{\left[\int p_\theta(\mathbf{x}^{(i)}, \mathbf{z}) d\mathbf{z}\right]}_{\text{intractable integral}}$$

must resort to approximation techniques

VARIATIONAL INFERENCE



approximate posterior $q(\mathbf{z}|\mathbf{x})$

variational lower bound

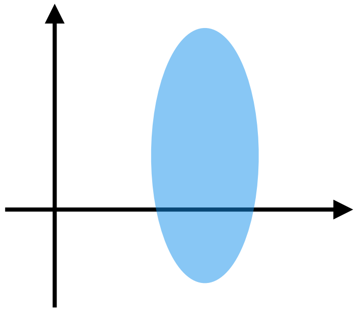
$$\log p_\theta(\mathbf{x}) \geq \mathcal{L}(\mathbf{x}; q) = \mathbb{E}_q \left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q(\mathbf{z}|\mathbf{x})} \right]$$

variational expectation maximization (EM)

tighten the bound: $q(\mathbf{z}|\mathbf{x}) \leftarrow \arg\max_q \mathcal{L}(\mathbf{x}; q)$

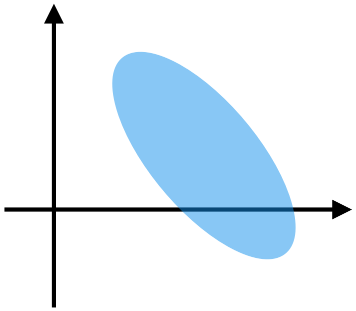
improve the model: $\theta \leftarrow \theta + \alpha \nabla_\theta \mathcal{L}(\mathbf{x}; q)$

STRUCTURED VARIATIONAL INFERENCE



mean field

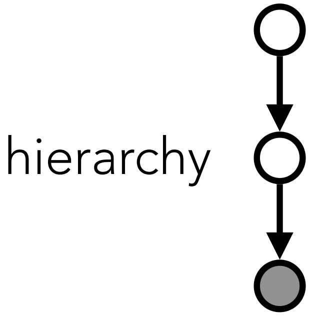
$$q(\mathbf{z}|\mathbf{x}) = \prod_j q(z_j|\mathbf{x})$$

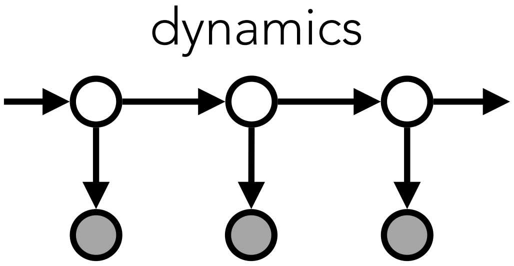


structured (auto-regressive)

$$q(\mathbf{z}|\mathbf{x}) = \prod_j q(z_j|\mathbf{x}, \mathbf{z}_{<j})$$

structured approximate posteriors are important for capturing latent dependencies within the model





AMORTIZED VARIATIONAL INFERENCE

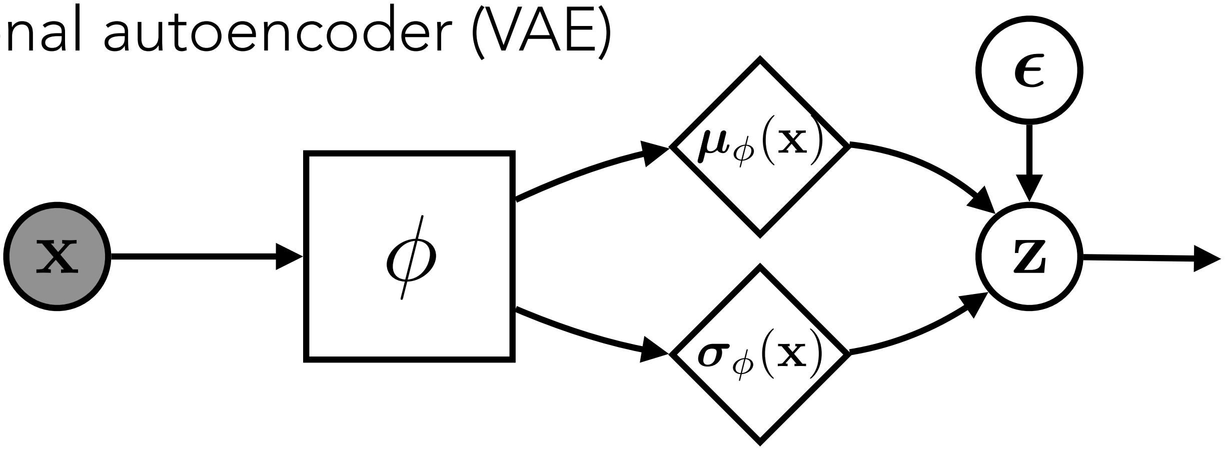
parameterize $q_\phi(\mathbf{z}|\mathbf{x})$ using a learned model,
shared (*amortized*) across data examples

example: $q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_\phi(\mathbf{x}), \boldsymbol{\sigma}^2_\phi(\mathbf{x}))$

learn the model through gradient descent,
using the reparameterization trick

$\mathbf{z} = \boldsymbol{\mu}_\phi(\mathbf{x}) + \boldsymbol{\sigma}_\phi(\mathbf{x}) \odot \boldsymbol{\epsilon}$ where $p(\boldsymbol{\epsilon}) = \mathcal{N}(\boldsymbol{\epsilon}; \mathbf{0}, \mathbf{I})$

variational autoencoder (VAE)


$\boldsymbol{\epsilon}$
$\boldsymbol{\mu}_\phi(\mathbf{x})$
$\mathbf{x}$
ϕ
$\mathbf{z}$
$\boldsymbol{\sigma}_\phi(\mathbf{x})$

Kingma & Welling, 2014
Rezende *et al.*, 2014

AMORTIZED VARIATIONAL INFERENCE

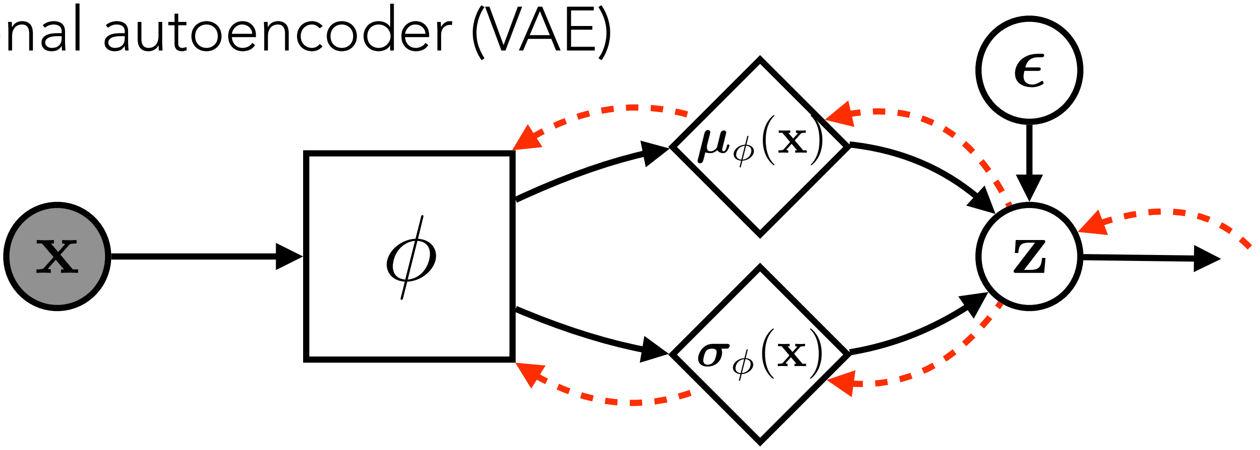
parameterize $q_\phi(\mathbf{z}|\mathbf{x})$ using a learned model, shared (*amortized*) across data examples

example: $$q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}_\phi(\mathbf{x}), \boldsymbol{\sigma}^2_\phi(\mathbf{x}))$$

learn the model through gradient descent, using the reparameterization trick

$$\mathbf{z} = \boldsymbol{\mu}_\phi(\mathbf{x}) + \boldsymbol{\sigma}_\phi(\mathbf{x}) \odot \boldsymbol{\epsilon} \quad \text{where} \quad p(\boldsymbol{\epsilon}) = \mathcal{N}(\boldsymbol{\epsilon}; \mathbf{0}, \mathbf{I})$$

variational autoencoder (VAE)



$\mathbf{x}$
ϕ
$\boldsymbol{\mu}_\phi(\mathbf{x})$
$\boldsymbol{\sigma}_\phi(\mathbf{x})$
$\boldsymbol{\epsilon}$
$\mathbf{z}$

Kingma & Welling, 2014
Rezende *et al.*, 2014

AMORTIZED VARIATIONAL INFERENCE

let $\boldsymbol{\lambda}$ be the distribution parameters of $q(\mathbf{z}|\mathbf{x})$, for example, $\boldsymbol{\lambda} = \{\boldsymbol{\mu}, \boldsymbol{\sigma}^2\}$

inference optimization: $q(\mathbf{z}|\mathbf{x}) \leftarrow \arg\max_q \mathcal{L}(\mathbf{x}; q)$

BLACK-BOX VARIATIONAL INFERENCE

gradient-based optimization

$$\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \eta \nabla_{\boldsymbol{\lambda}} \mathcal{L}$$

DIRECT AMORTIZED INFERENCE

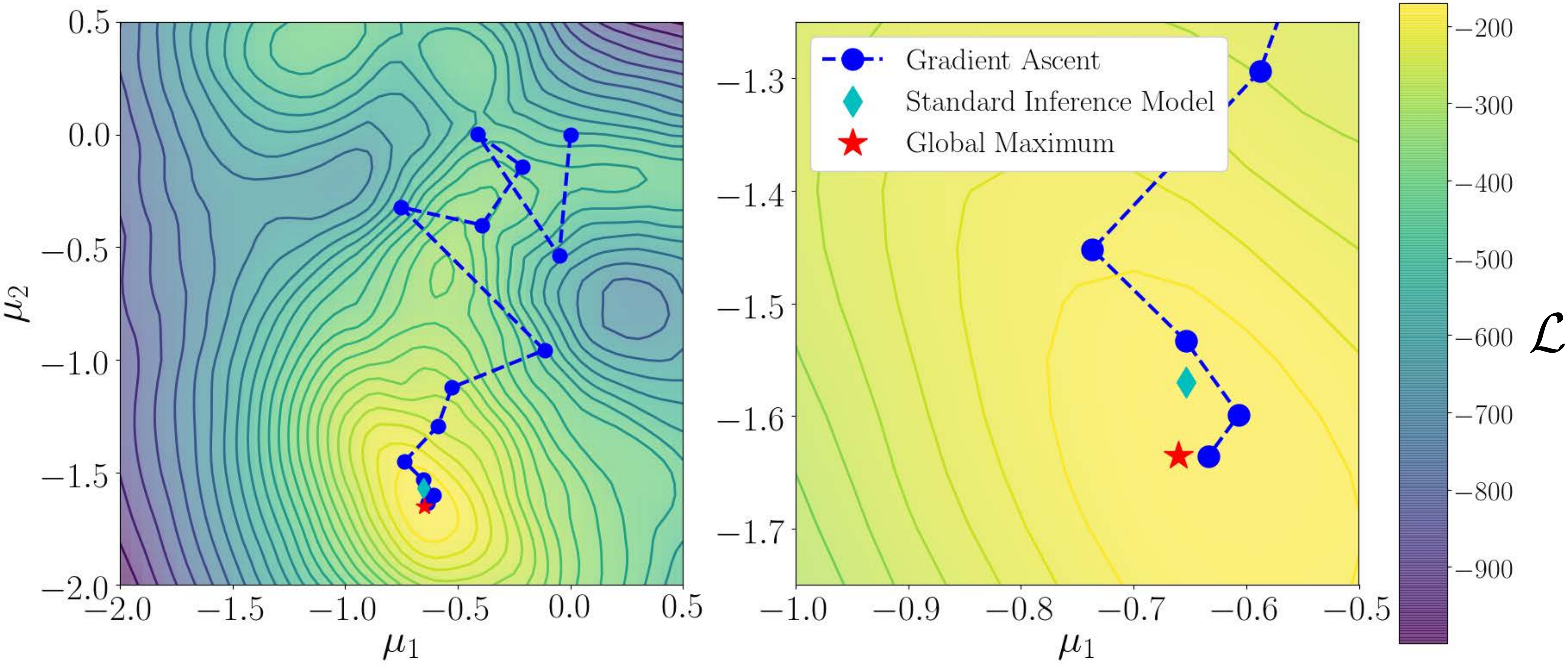
standard amortized inference models learn a direct mapping

$$\boldsymbol{\lambda} \leftarrow f_\phi(\mathbf{x})$$

efficient, but potentially inaccurate

INFERENCE OPTIMIZATION

2D model, MNIST



inference models may not reach fully optimized estimates

see also: **Inference Suboptimality in Variational Autoencoders**, Cremer *et al.*, 2018

Marino *et al.*, 2018a

ITERATIVE AMORTIZED INFERENCE

let $\boldsymbol{\lambda}$ be the distribution parameters of $q(\mathbf{z}|\mathbf{x})$, for example, $\boldsymbol{\lambda} = \{\boldsymbol{\mu}, \boldsymbol{\sigma}^2\}$

inference optimization: $q(\mathbf{z}|\mathbf{x}) \leftarrow \arg\max_q \mathcal{L}(\mathbf{x}; q)$

ITERATIVE AMORTIZED INFERENCE

iterative amortized inference models learn an iterative mapping

$$\boldsymbol{\lambda} \leftarrow f_\phi(\boldsymbol{\lambda}, \nabla_{\boldsymbol{\lambda}} \mathcal{L})$$

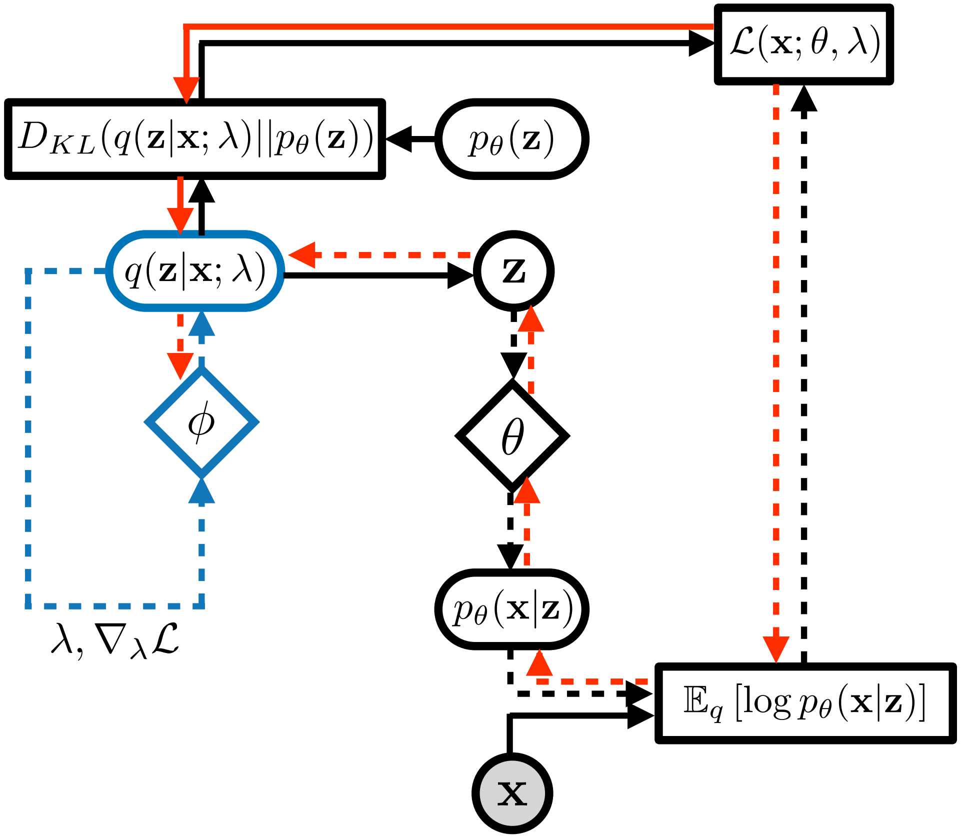
retain efficiency, with a more flexible mapping

Marino *et al.*, 2018a

ITERATIVE AMORTIZED INFERENCE

iterative amortized inference models learn an iterative mapping

$$\boldsymbol{\lambda} \leftarrow f_\phi(\boldsymbol{\lambda}, \nabla_{\boldsymbol{\lambda}} \mathcal{L})$$



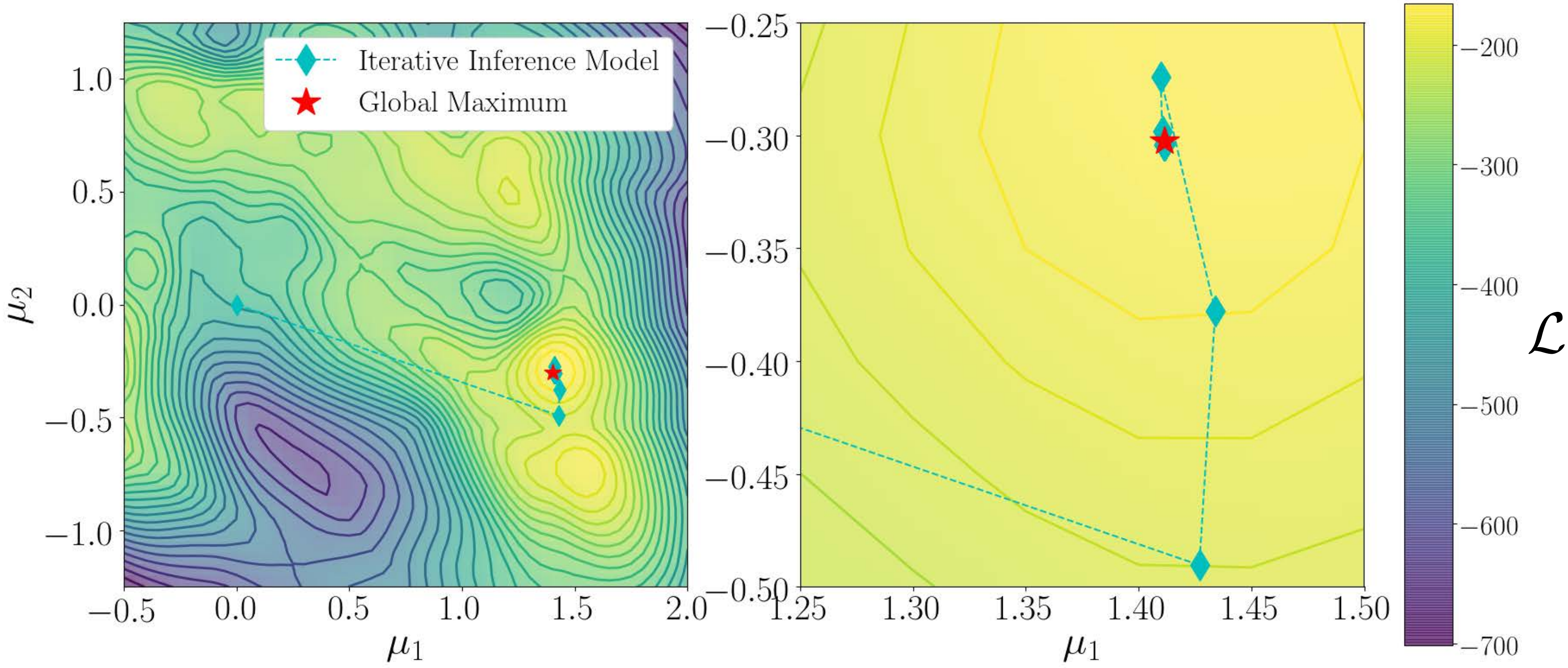
$\mathcal{L}(\mathbf{x}; \theta, \lambda)$
$D_{KL}(q(\mathbf{z}|\mathbf{x};\lambda)||p_\theta(\mathbf{z}))$
$p_\theta(\mathbf{z})$
$q(\mathbf{z}|\mathbf{x};\lambda)$
$\mathbf{z}$
ϕ
θ
$\lambda, \nabla_\lambda \mathcal{L}$
$p_\theta(\mathbf{x}|\mathbf{z})$
$\mathbb{E}_q[\log p_\theta(\mathbf{x}|\mathbf{z})]$
$\mathbf{x}$

Marino *et al.*, 2018a

INFERENCE OPTIMIZATION

directly visualize inference in the optimization landscape

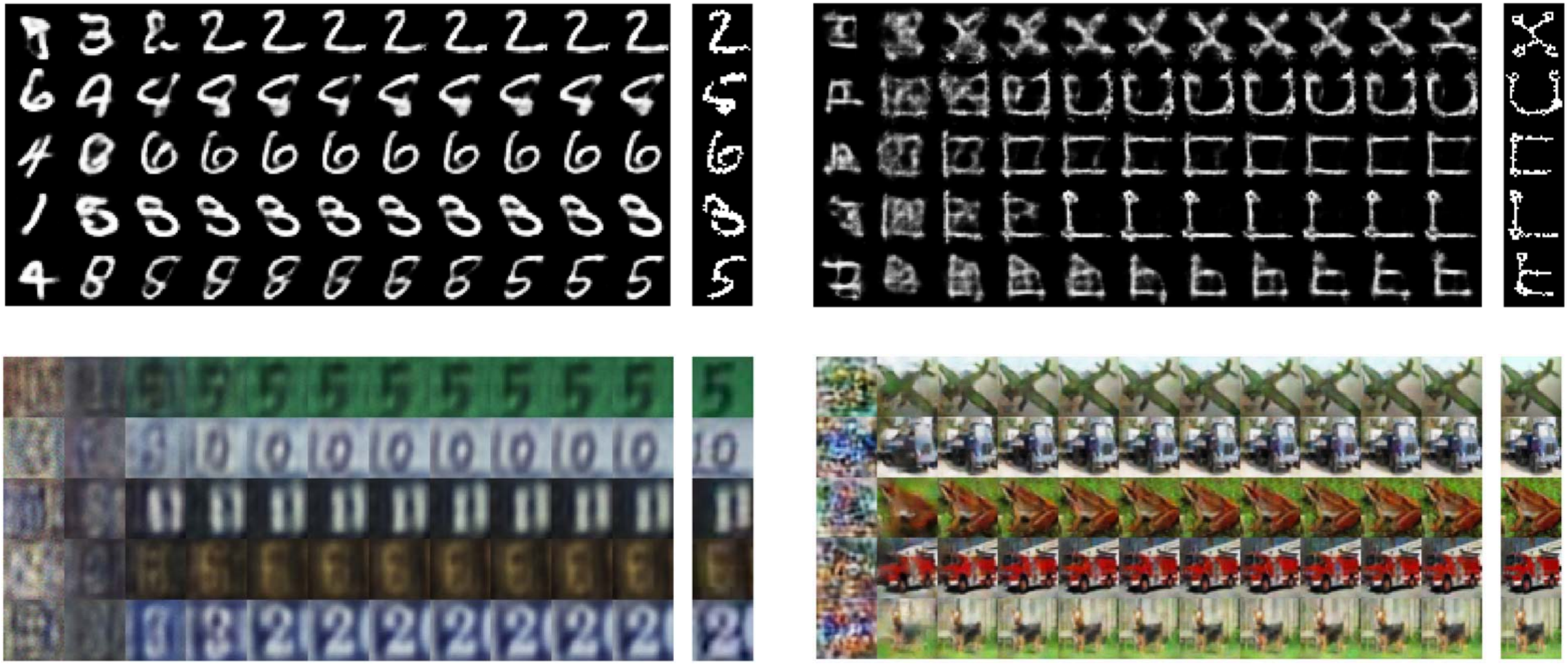
2D model, MNIST



Marino *et al.*, 2018a

INFERENCE OPTIMIZATION

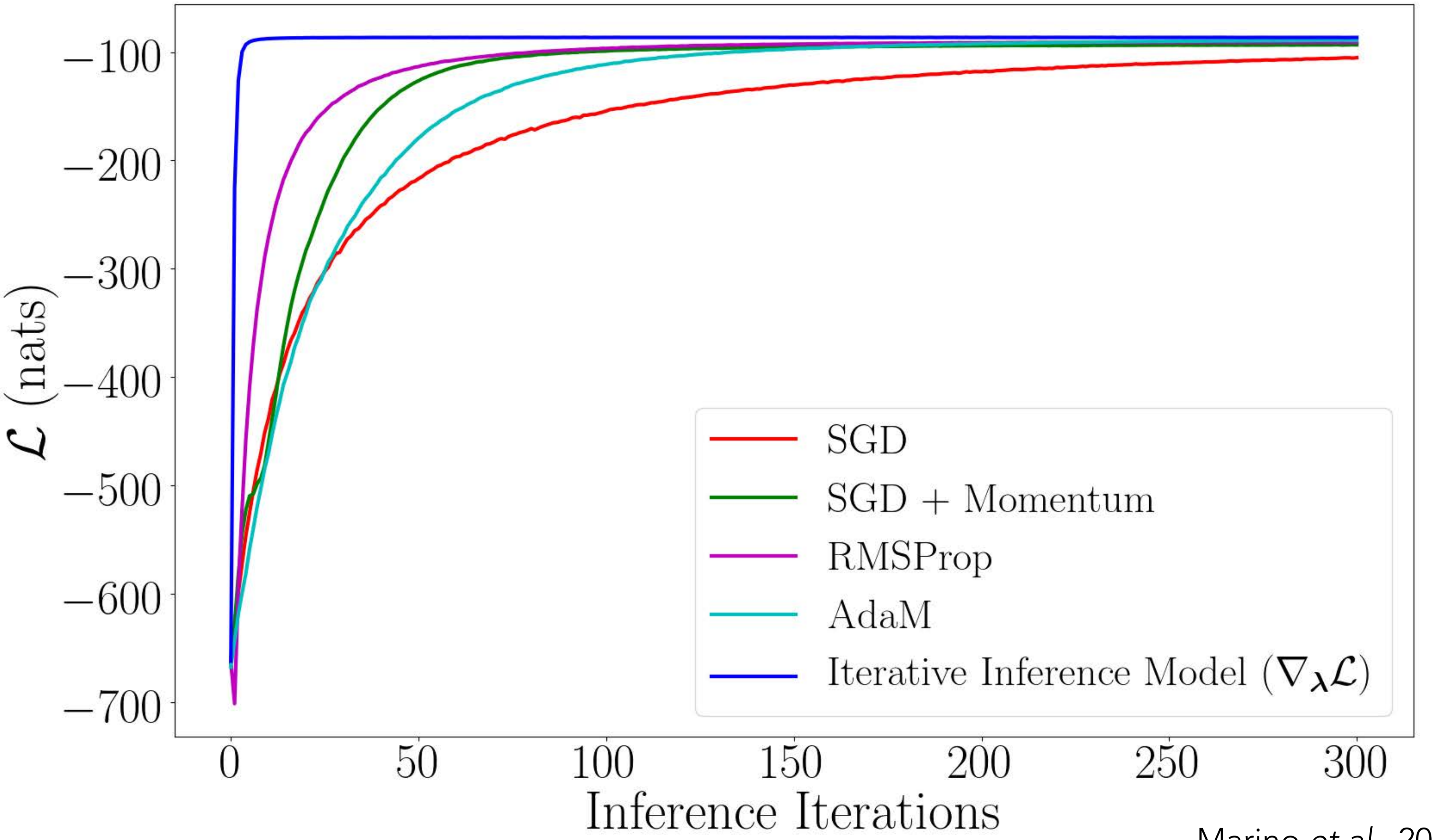
visualize data reconstructions over inference iterations



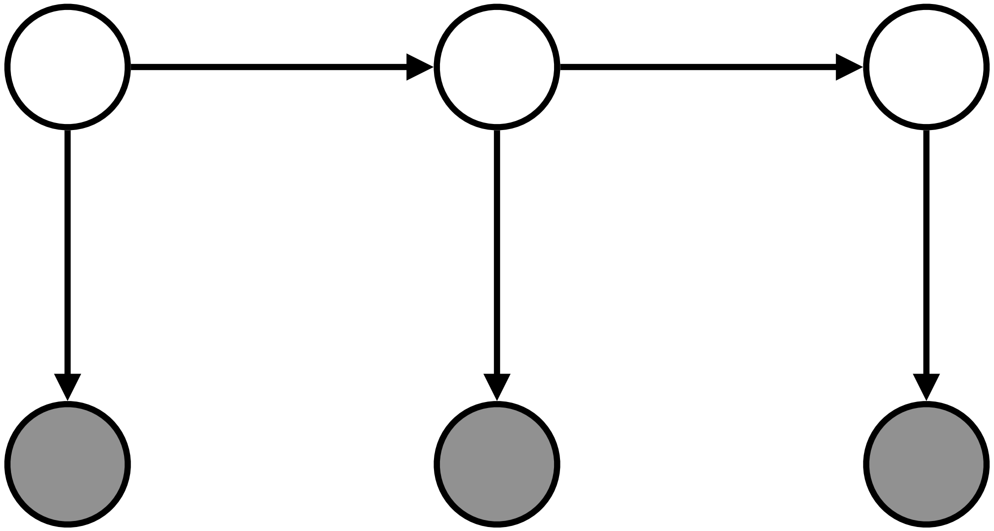
Marino *et al.*, 2018a

INFERENCE OPTIMIZATION

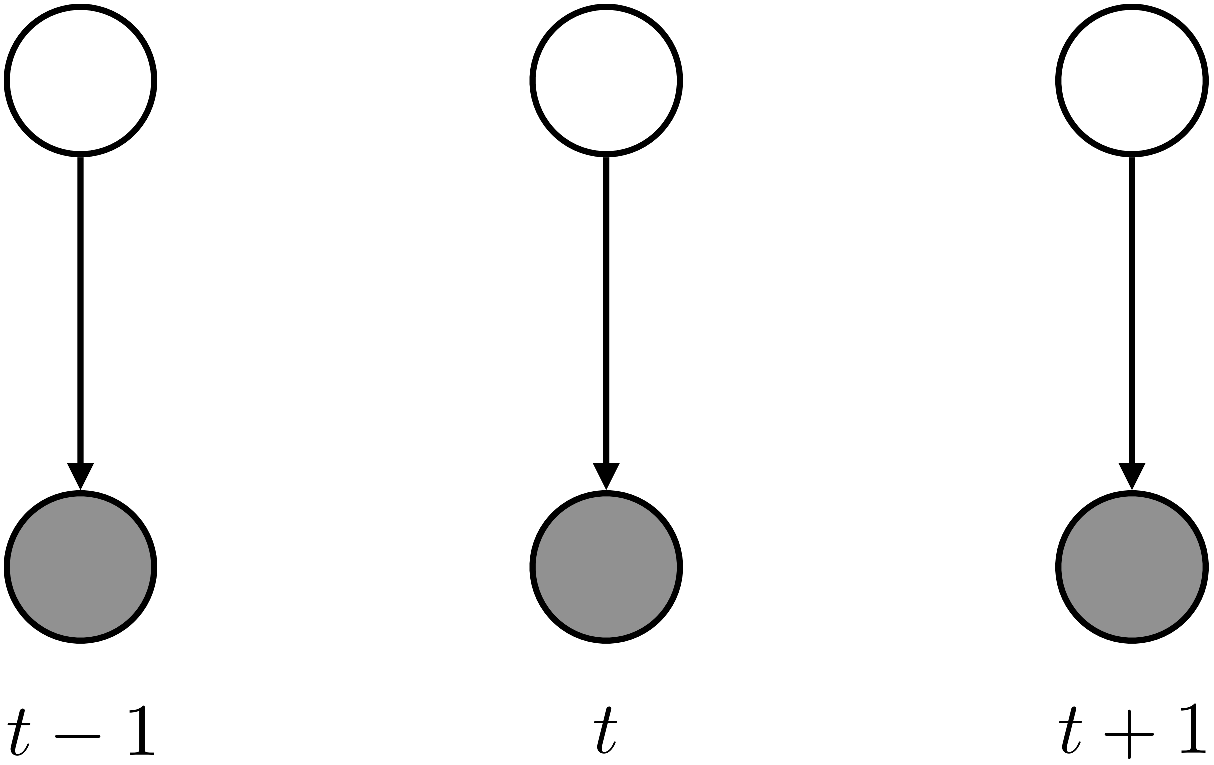
plot the ELBO over inference iterations



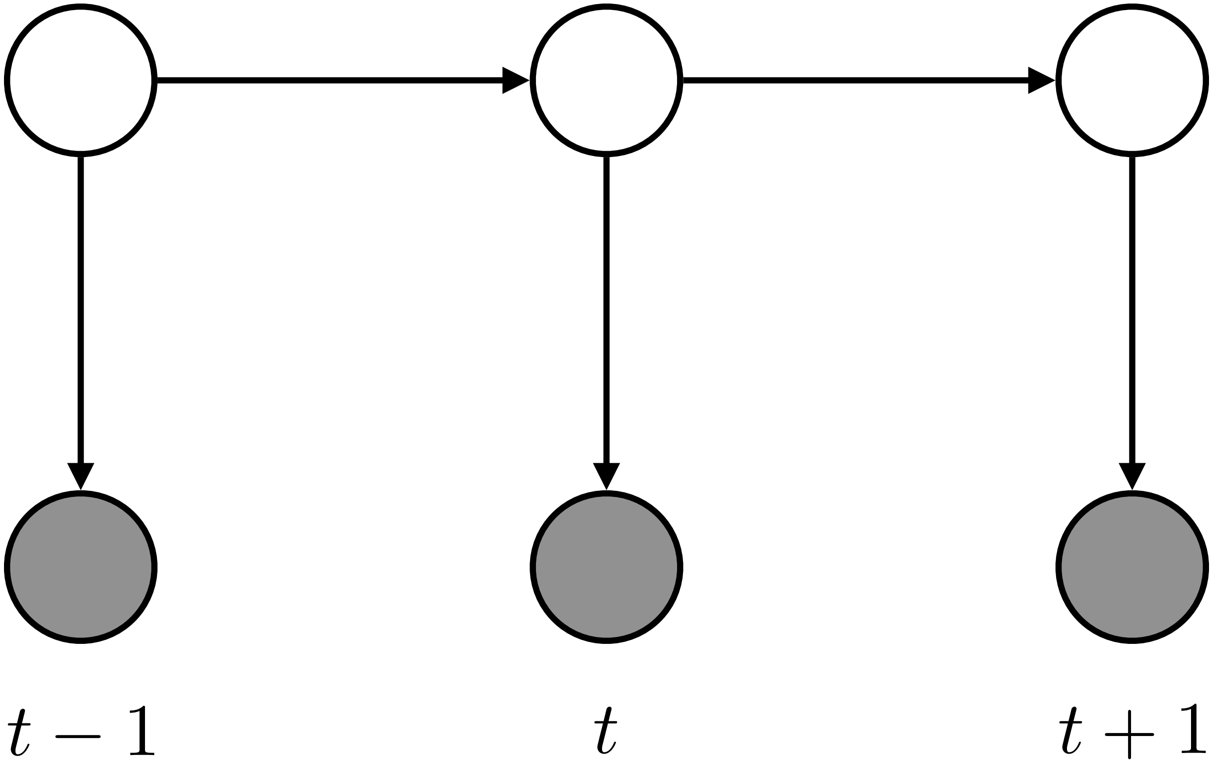
Marino *et al.*, 2018a



DEEP SEQUENTIAL LATENT VARIABLE MODELS


$t-1$
t
$t+1$

use information from other time steps to estimate current state



model temporal dependencies

SEQUENTIAL LATENT VARIABLE MODELS

general form:

$$p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T}) = \prod_{t=1}^{T} \underbrace{p_\theta(\mathbf{x}_t | \mathbf{x}_{<t}, \mathbf{z}_{\leq t})}_{\text{likelihood/emission}} \underbrace{p_\theta(\mathbf{z}_t | \mathbf{x}_{<t}, \mathbf{z}_{<t})}_{\text{prior/dynamics}}$$

where $\mathbf{x}_{\leq T}$ is a sequence of T observed variables

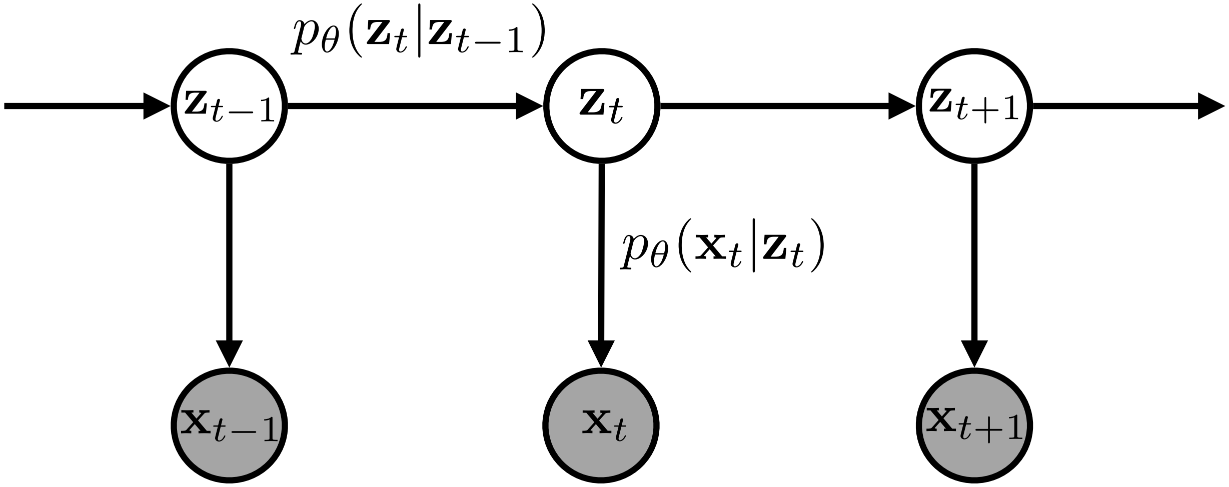
$\mathbf{z}_{\leq T}$ is a sequence of T latent variables

SEQUENTIAL LATENT VARIABLE MODELS

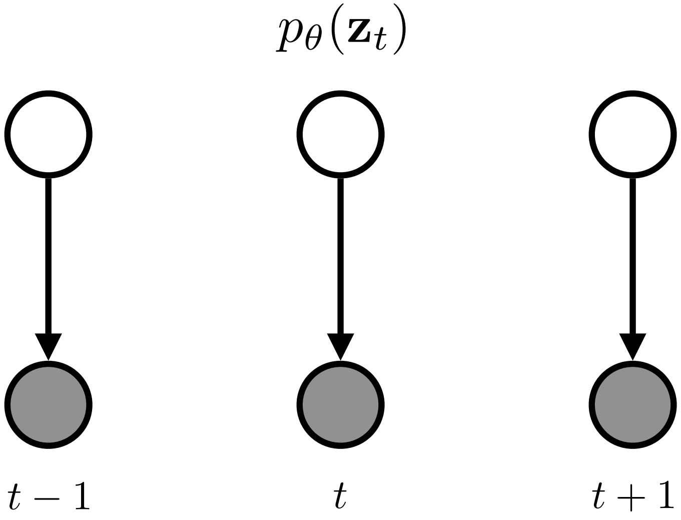
general form:

$$p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T}) = \prod_{t=1}^{T} \underbrace{p_\theta(\mathbf{x}_t | \mathbf{x}_{<t}, \mathbf{z}_{\leq t})}_{\text{likelihood/emission}} \underbrace{p_\theta(\mathbf{z}_t | \mathbf{x}_{<t}, \mathbf{z}_{<t})}_{\text{prior/dynamics}}$$

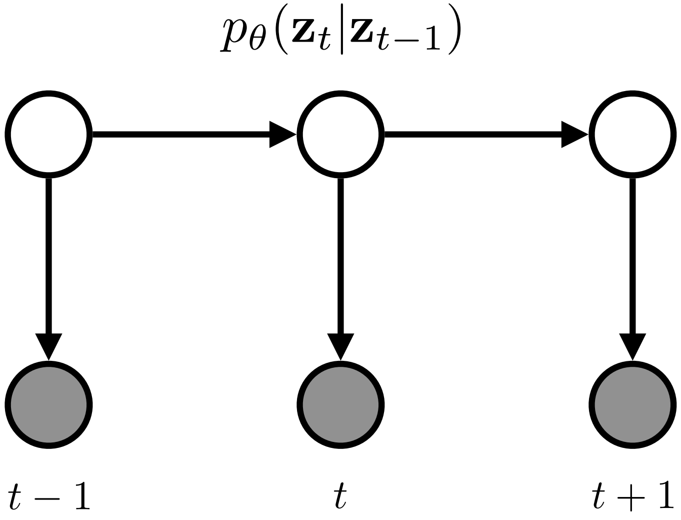
simplified case (hidden Markov model):



SEQUENTIAL DEPENDENCIES



static model



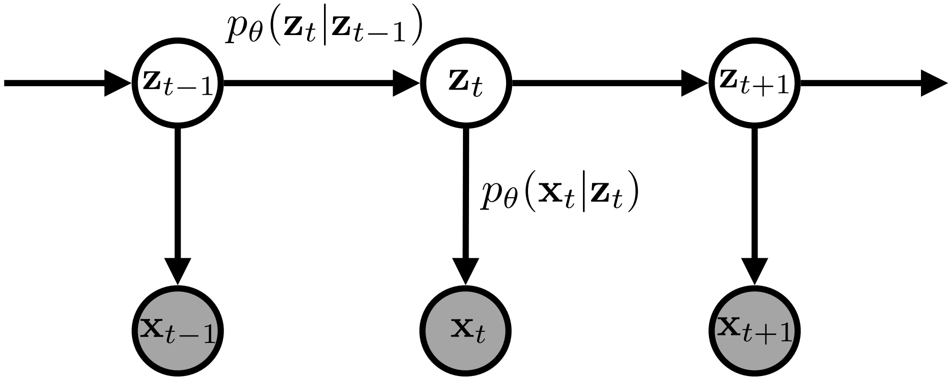
sequential model

$p_\theta(\mathbf{z}_t) = \int p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1})p_\theta(\mathbf{z}_{t-1})d\mathbf{z}_{t-1}$ is more flexible than a static $p_\theta(\mathbf{z}_t)$

can fit the data better if relationships exist between time steps

SEQUENTIAL LATENT VARIABLE MODELS

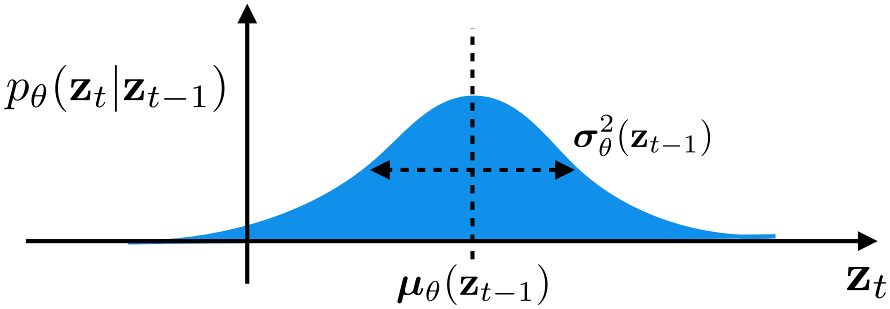
Markov model:



Parameterization:

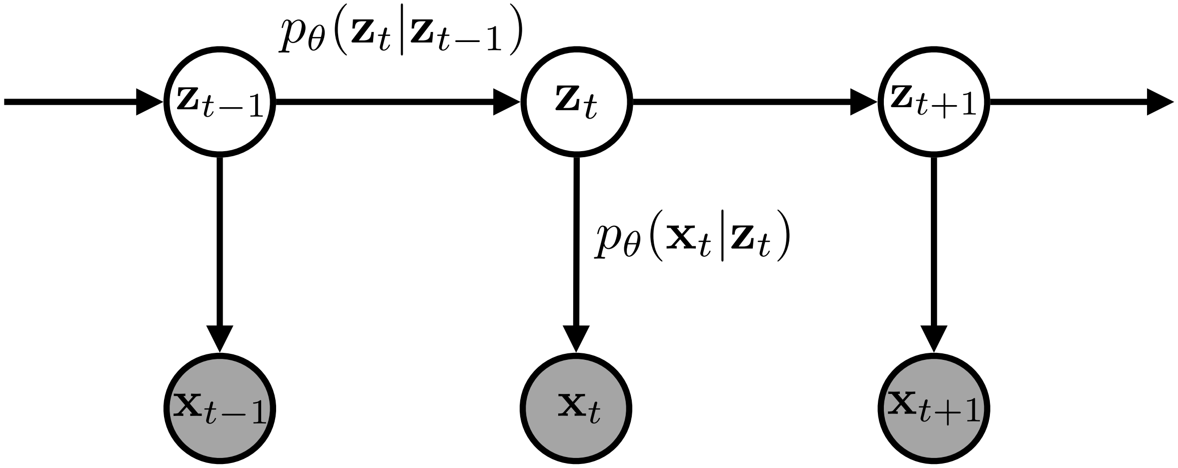
$p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1})$ is typically an analytical distribution

for example, $p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1}) = \mathcal{N}(\mathbf{z}_t; \boldsymbol{\mu}_\theta(\mathbf{z}_{t-1}), \text{diag}(\boldsymbol{\sigma}^2_\theta(\mathbf{z}_{t-1})))$



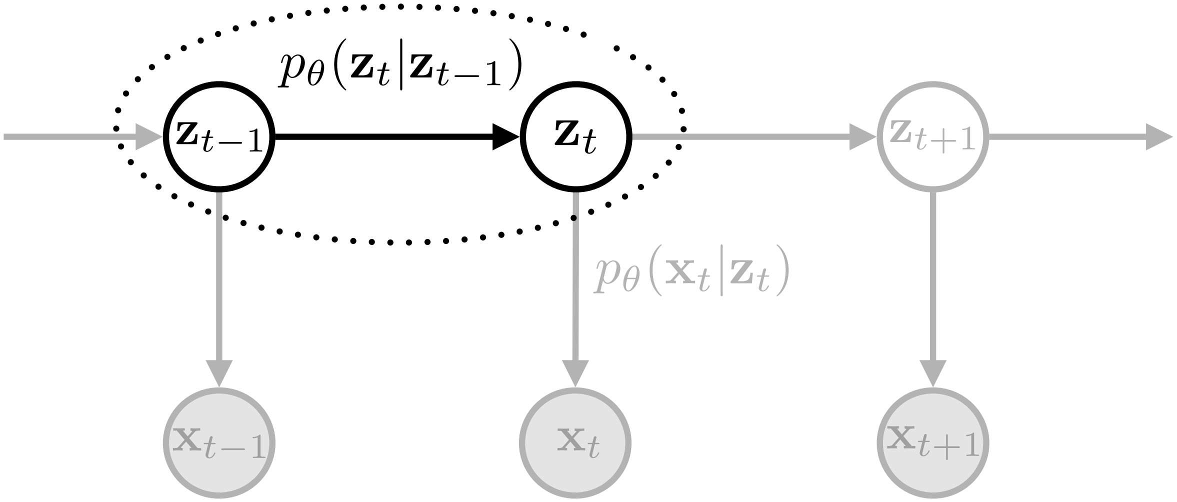
SEQUENTIAL LATENT VARIABLE MODELS

the parameters of these analytical distributions are functions, often *deep networks*



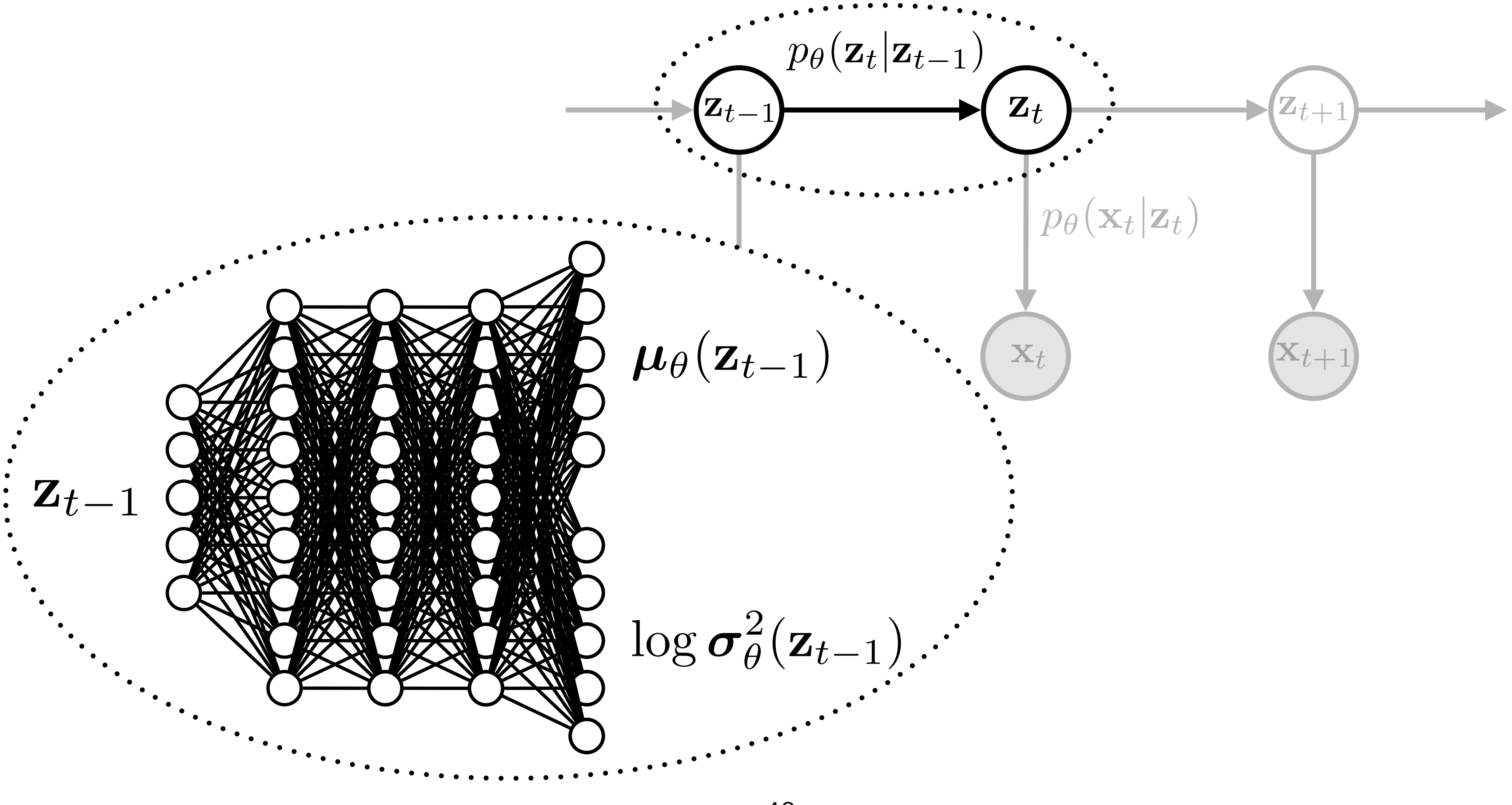
SEQUENTIAL LATENT VARIABLE MODELS

the parameters of these analytical distributions are functions, often *deep networks*



SEQUENTIAL LATENT VARIABLE MODELS

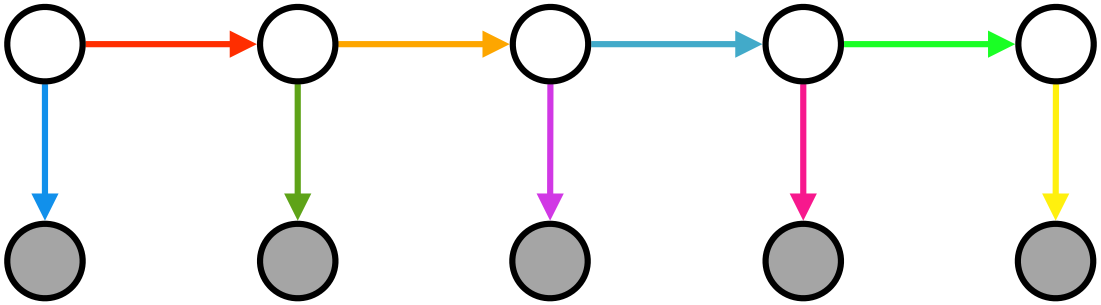
the parameters of these analytical distributions are functions, often *deep networks*



$p_\theta(\mathbf{z}_t|\mathbf{z}_{t-1})$
$\mathbf{z}_{t-1}$
$\mathbf{z}_t$
$\mathbf{z}_{t+1}$
$p_\theta(\mathbf{x}_t|\mathbf{z}_t)$
$\mathbf{x}_t$
$\mathbf{x}_{t+1}$
$\mathbf{z}_{t-1}$
$\boldsymbol{\mu}_\theta(\mathbf{z}_{t-1})$
$\log \boldsymbol{\sigma}^2_\theta(\mathbf{z}_{t-1})$

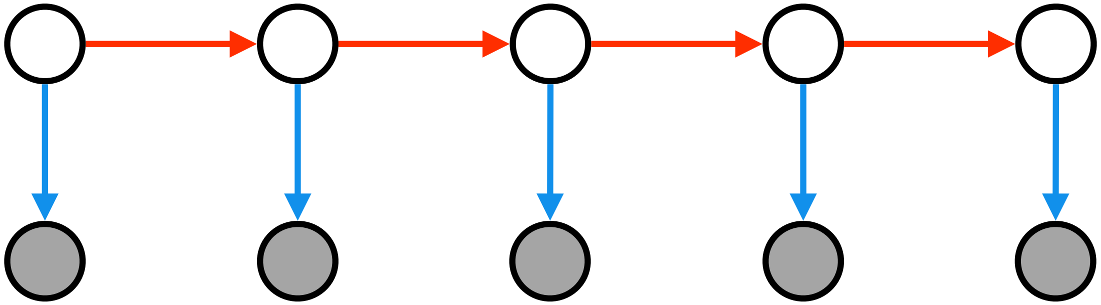
WEIGHT SHARING

could use a separate network for each conditional dependence



number of parameters grows linearly with time

share weights for similar conditional dependencies

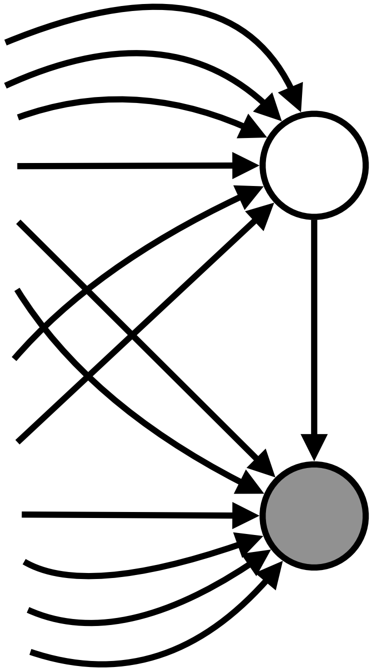


fixed number of parameters

LONG-TERM DEPENDENCIES

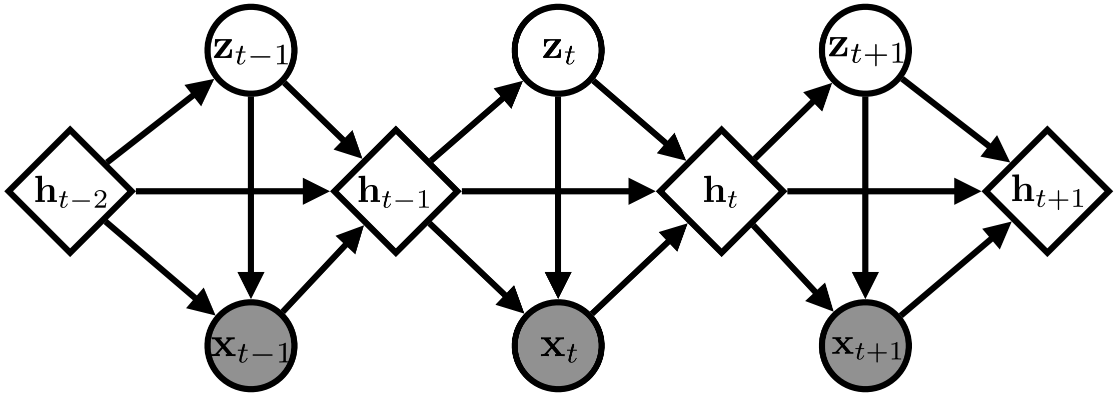
general model form $p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T}) = \prod_{t=1}^{T} p_\theta(\mathbf{x}_t|\mathbf{x}_{<t}, \mathbf{z}_{\leq t}) p_\theta(\mathbf{z}_t|\mathbf{x}_{<t}, \mathbf{z}_{<t})$

how do we model long-term dependencies?



use a *recurrent network*

◇ deterministic

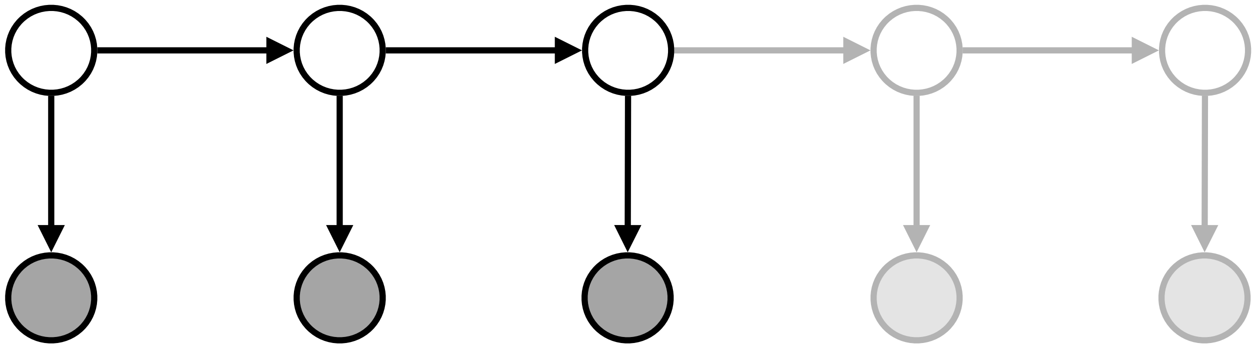


$$p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T}) = \prod_{t=1}^{T} p_\theta(\mathbf{x}_t|\mathbf{h}_{t-1}, \mathbf{z}_t) p_\theta(\mathbf{z}_t|\mathbf{h}_{t-1})$$

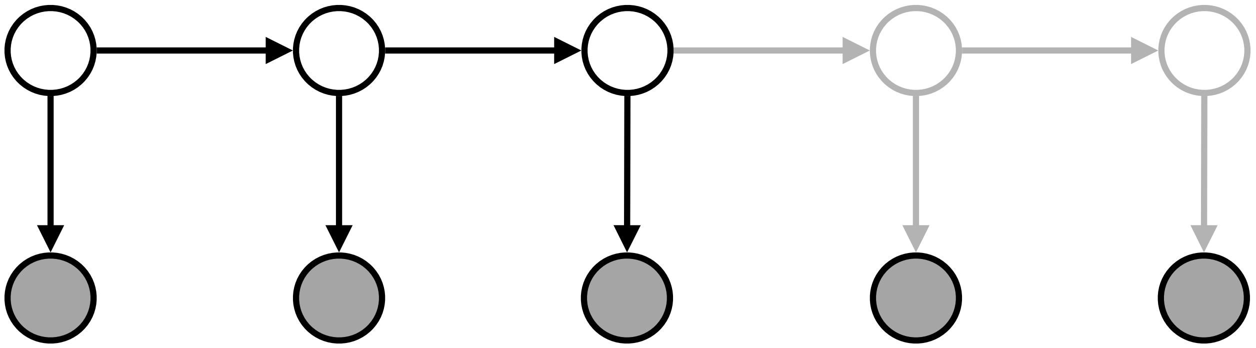
INFERENCE

given a sequence of observations, $\mathbf{x}_{\leq T}$, infer $p_\theta(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T})$

filtering inference



smoothing inference



VARIATIONAL INFERENCE IN SEQUENTIAL MODELS

introduce an approximate posterior $q(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T})$

$$\boxed{\text{ELBO:} \quad \mathcal{L}(\mathbf{x}_{\leq T}, q) = \mathbb{E}_q \left[\log \frac{p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T})}{q(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T})} \right]}$$

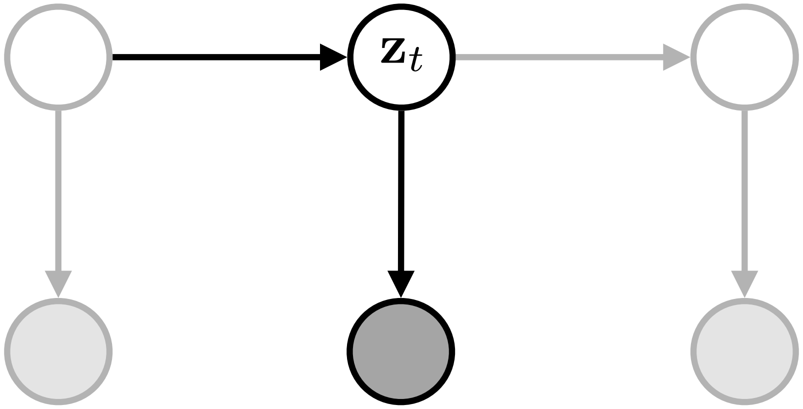
choices about the form of $q(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T})$ determine how we evaluate $\mathcal{L}$

⟶ often $q(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T})$ is *structured*

STRUCTURED VARIATIONAL INFERENCE

the model contains temporal dependencies

$p_\theta(\mathbf{z}_t|\mathbf{x}_{<t}, \mathbf{z}_{<t})$



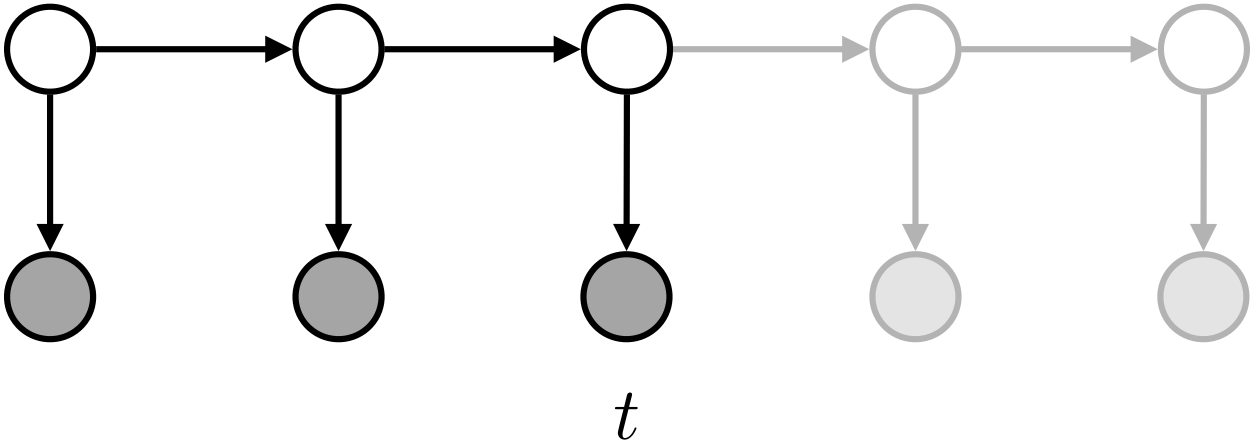
the approximate posterior should account for these dependencies

⟶ if we use $q(\mathbf{z}_t|\mathbf{x}_t)$, we cannot account for $\mathbf{x}_{<t}$ and $\mathbf{z}_{<t}$

FILTERING INFERENCE

filtering approximate posterior

$$q(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T}) = \prod_{t=1}^{T} q(\mathbf{z}_t | \mathbf{x}_{\leq t}, \mathbf{z}_{<t})$$

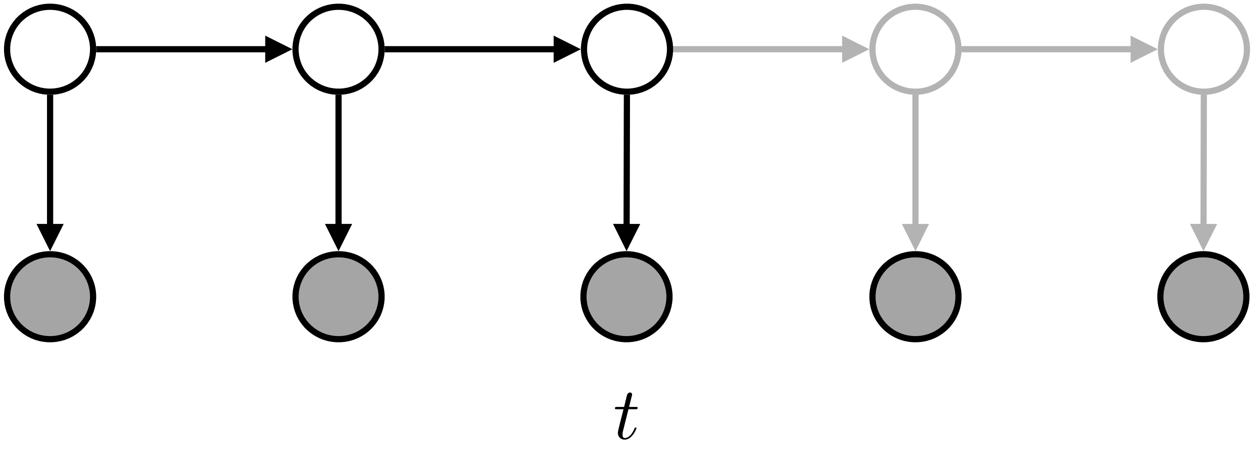


condition on observations at past and present time steps

SMOOTHING INFERENCE

smoothing approximate posterior

$$q(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T}) = \prod_{t=1}^{T} q(\mathbf{z}_t | \mathbf{x}_{\leq T}, \mathbf{z}_{<t})$$



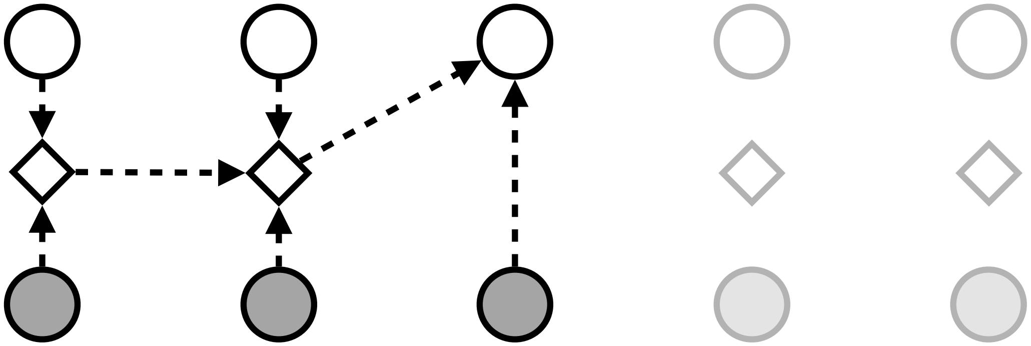
condition on observations at all time steps

AMORTIZED VARIATIONAL INFERENCE

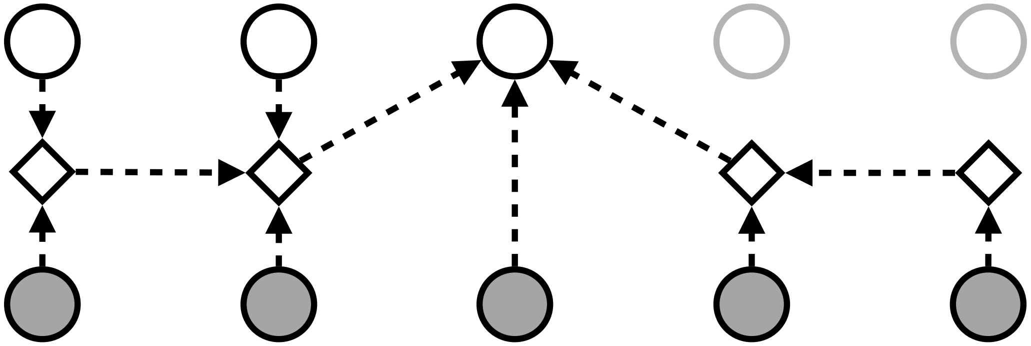
how do we amortize inference in sequential models?

typical approach:

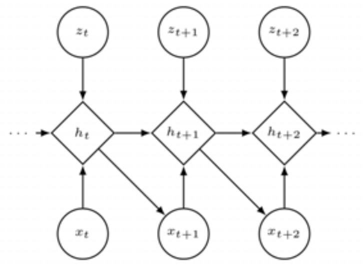
filtering: use a recurrent network



smoothing: use a bi-directional recurrent network

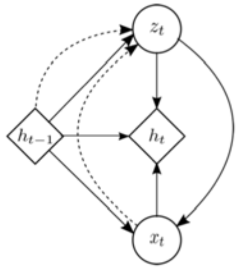


RECENT MODELS



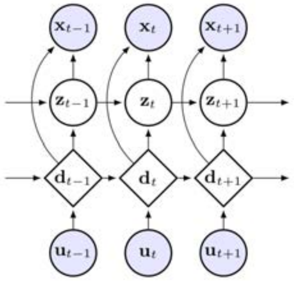
STORN

Bayer & Osendorfer, 2014



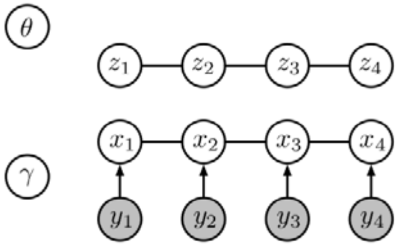
VRNN

Chung *et al.*, 2015



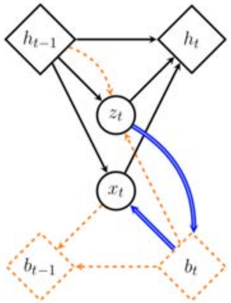
SRNN

Fraccaro *et al.*, 2016



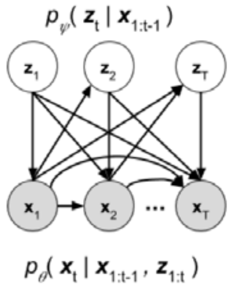
Structured VAE

Johnson *et al.*, 2016



Z-Forcing

Goyal *et al.*, 2017


$p_\psi(\mathbf{z}_t \mid \mathbf{x}_{1:t-1})$
$\mathbf{z}_1$
$\mathbf{z}_2$
$\mathbf{z}_T$
$\mathbf{x}_1$
$\mathbf{x}_2$
...
$\mathbf{x}_T$
$p_\theta(\mathbf{x}_t \mid \mathbf{x}_{1:t-1}, \mathbf{z}_{1:t})$

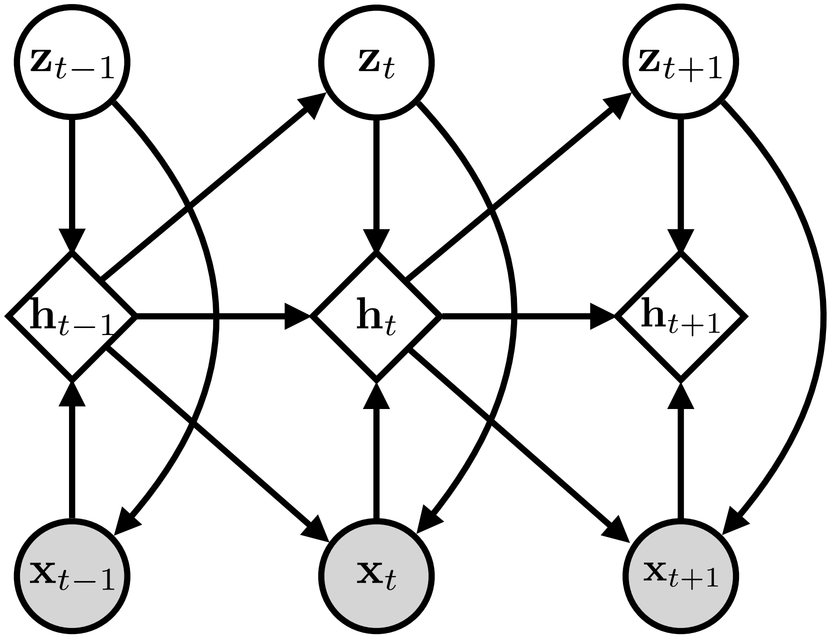
SVG

Denton & Fergus, 2018

• • •

VRNN

generative model



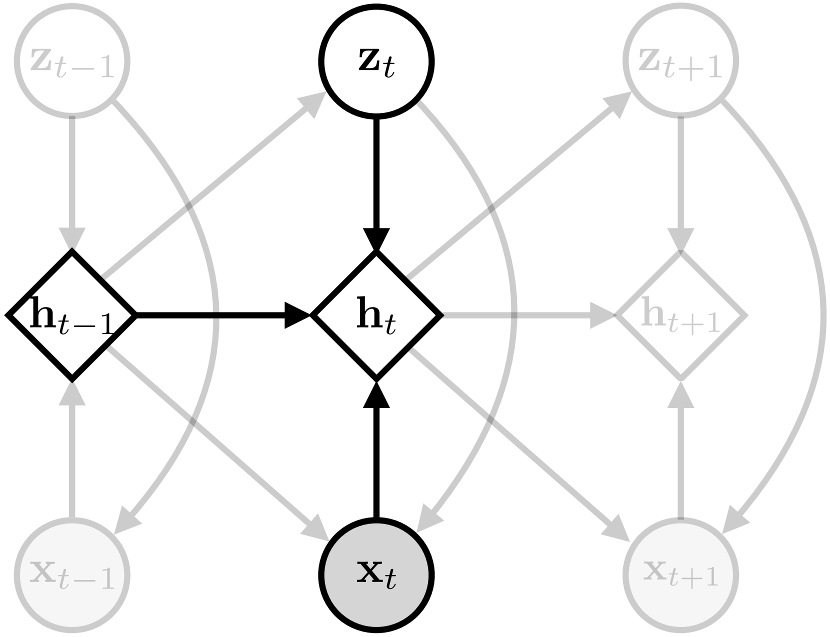
general model form $$p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T}) = \prod_{t=1}^{T} p_\theta(\mathbf{x}_t|\mathbf{x}_{<t}, \mathbf{z}_{\leq t}) p_\theta(\mathbf{z}_t|\mathbf{x}_{<t}, \mathbf{z}_{<t})$$

VRNN model form $$= \prod_{t=1}^{T} p_\theta(\mathbf{x}_t|\mathbf{z}_t, \mathbf{h}_{t-1}) p_\theta(\mathbf{z}_t|\mathbf{h}_{t-1})$$

Chung *et al.*, 2015

VRNN

generative model



recurrence:

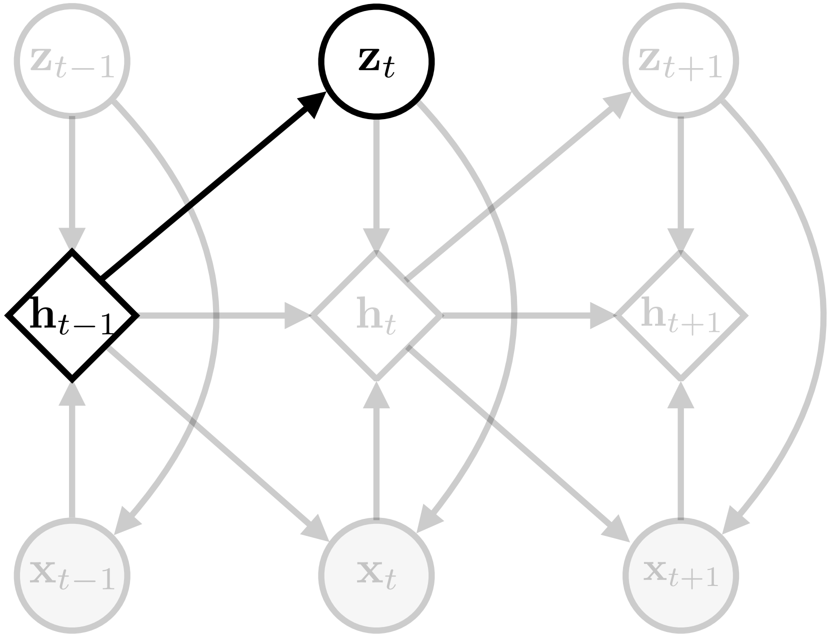
$$\mathbf{h}_t = \mathrm{LSTM}([\varphi_{\mathbf{x}}(\mathbf{x}_t), \varphi_{\mathbf{z}}(\mathbf{z}_t)], \mathbf{h}_{t-1})$$

φ are fully-connected networks

Chung *et al.*, 2015

VRNN

generative model



prior:

$$p_\theta(\mathbf{z}_t|\mathbf{h}_{t-1}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{z},t}, \mathrm{diag}(\boldsymbol{\sigma}^2_{\mathbf{z},t}))$$

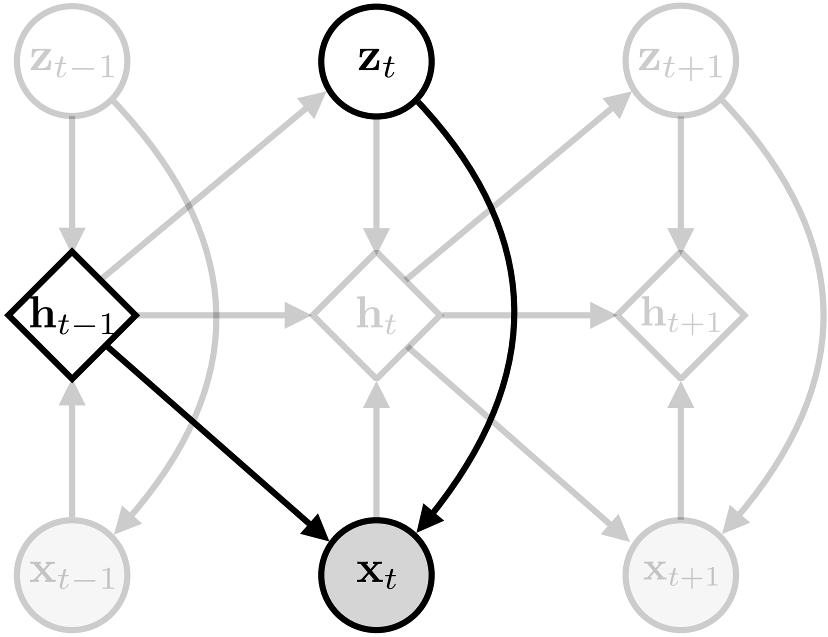
where $[\boldsymbol{\mu}_{\mathbf{z},t}, \boldsymbol{\sigma}_{\mathbf{z},t})] = \varphi_{\mathrm{prior}}(\mathbf{h}_{t-1})$

φ are fully-connected networks

Chung *et al.*, 2015

VRNN

generative model



conditional likelihood:

$$p_\theta(\mathbf{x}_t|\mathbf{z}_t, \mathbf{h}_{t-1}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x},t}, \mathrm{diag}(\boldsymbol{\sigma}^2_{\mathbf{x},t}))$$

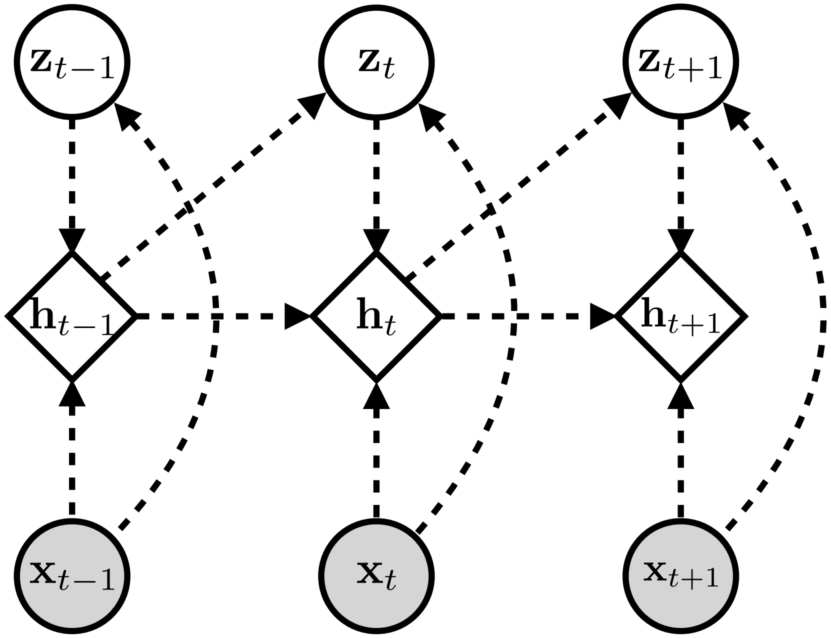
where $[\boldsymbol{\mu}_{\mathbf{x},t}, \boldsymbol{\sigma}_{\mathbf{x},t})] = \varphi_{\mathrm{dec}}(\varphi_{\mathbf{z}}(\mathbf{z}_t), \mathbf{h}_{t-1})$

φ are fully-connected networks

Chung *et al.*, 2015

VRNN

inference model



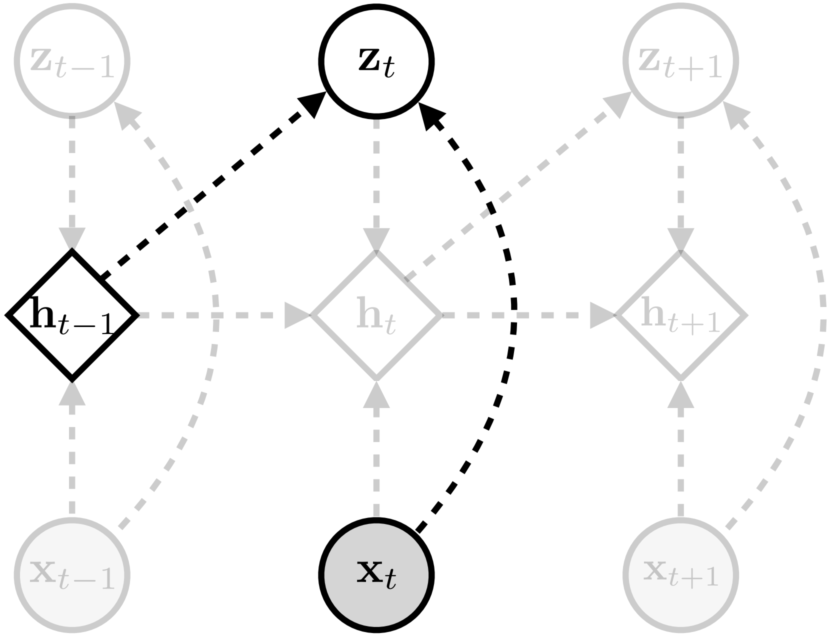
filtering inference $$q(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{<t})$$

VRNN inference model form $$= \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{x}_t, \mathbf{h}_{t-1})$$

Chung *et al.*, 2015

VRNN

inference model



approximate posterior:

$$q(\mathbf{z}_t|\mathbf{x}_t, \mathbf{h}_{t-1}) = \mathcal{N}(\boldsymbol{\mu}_{\mathbf{z},t}, \mathrm{diag}(\boldsymbol{\sigma}^2_{\mathbf{z},t}))$$

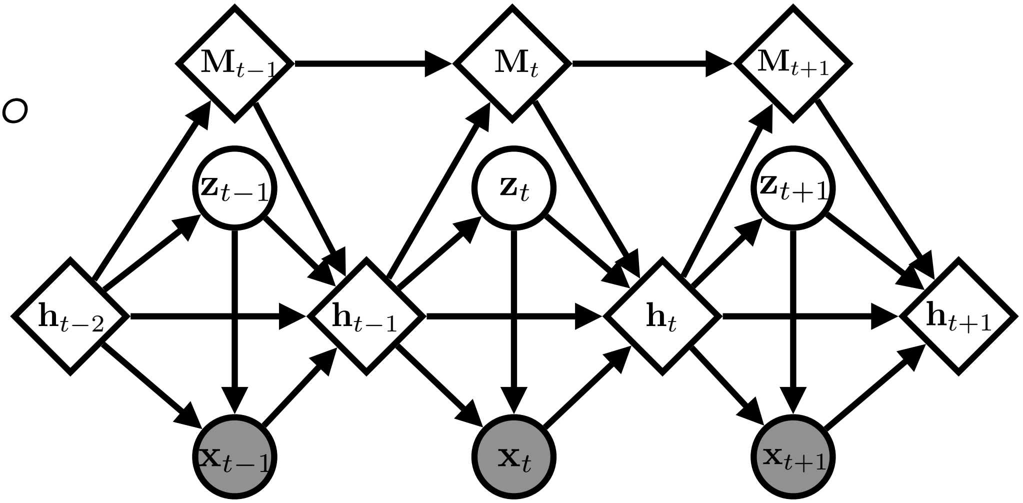
where $[\boldsymbol{\mu}_{\mathbf{z},t}, \boldsymbol{\sigma}_{\mathbf{z},t})] = \varphi_{\mathrm{enc}}(\varphi_{\mathbf{x}}(\mathbf{x}_t), \mathbf{h}_{t-1})$

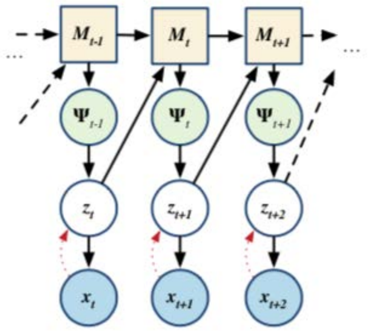
φ are fully-connected networks

Chung *et al.*, 2015

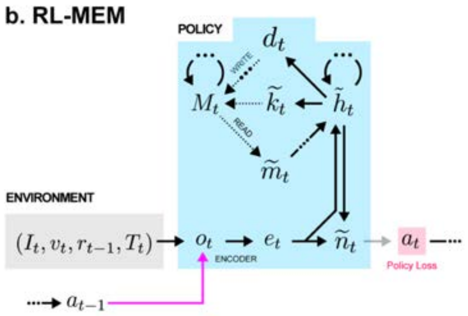
MEMORY

use a specialized memory module to model longer-term dependencies

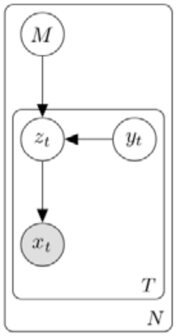




GTMM
Gemici *et al.*, 2017



MERLIN
Wayne *et al.*, 2018

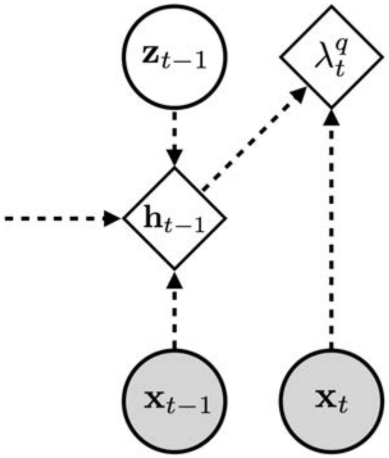


Kanerva Machine
Wu *et al.*, 2018

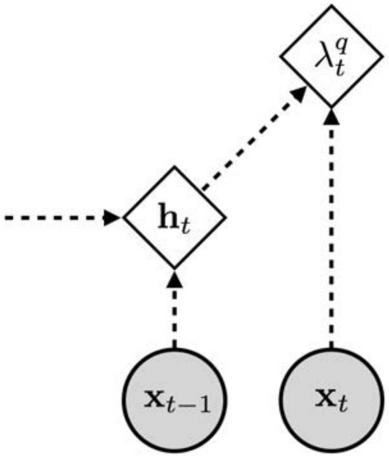
● ● ●

FILTERING INFERENCE MODELS

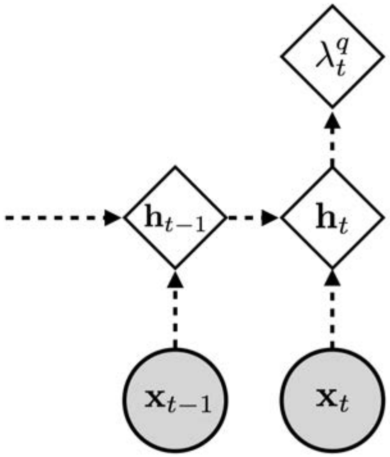
approx. posterior parameters $\boldsymbol{\lambda}_t^q$



VRNN



SRNN



SVG

custom-designed

FILTERING VARIATIONAL LOWER BOUND

definition of lower bound

$$\mathcal{L} \equiv \mathbb{E}_{q(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T})}\left[\log \frac{p_\theta(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T})}{q(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T})}\right]$$

under a ***filtering*** approximate posterior

$$q(\mathbf{z}_{\leq T}|\mathbf{x}_{\leq T}) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{<t}).$$

the variational lower bound is

$$\mathcal{L} = \sum_{t=1}^{T} \mathbb{E}_{q(\mathbf{z}_{<t}|\mathbf{x}_{<t},\mathbf{z}_{<t-1})}\left[\mathcal{L}_t\right]$$

where

$$\mathcal{L}_t \equiv \mathbb{E}_{q(\mathbf{z}_t|\mathbf{x}_{\leq t},\mathbf{z}_{<t})}\left[\log \frac{p_\theta(\mathbf{x}_t, \mathbf{z}_t|\mathbf{x}_{<t}, \mathbf{z}_{<t})}{q(\mathbf{z}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{<t})}\right]$$

Gemici *et al.*, 2017
Marino *et al.*, 2018b

FILTERING VARIATIONAL LOWER BOUND

define $\tilde{\mathcal{L}}_t := \mathbb{E}_{q(\mathbf{z}_{<t}|\mathbf{x}_{<t}, \mathbf{z}_{<t-1})} \left[\mathcal{L}_t \right]$

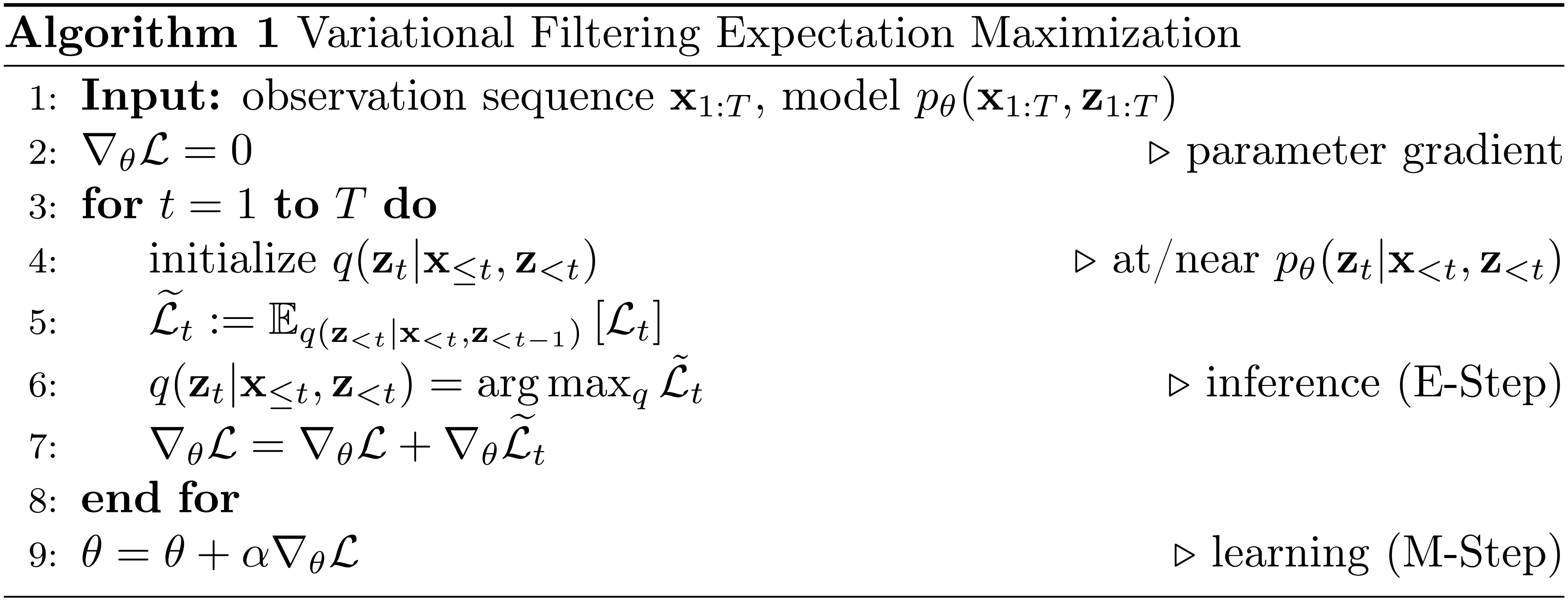
$$\mathcal{L} = \underbrace{\tilde{\mathcal{L}}_1 + \tilde{\mathcal{L}}_2 + \cdots + \tilde{\mathcal{L}}_{t-1} + \rlap{$\overbrace{\phantom{\tilde{\mathcal{L}}_t + \tilde{\mathcal{L}}_{t+1} + \cdots + \tilde{\mathcal{L}}_{T-1} + \tilde{\mathcal{L}}_T}}^{\text{terms in which } q(\mathbf{z}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{<t}) \text{ appears}}$}\tilde{\mathcal{L}}_t}_{\text{steps on which } q(\mathbf{z}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{<t}) \text{ depends}} + \tilde{\mathcal{L}}_{t+1} + \cdots + \tilde{\mathcal{L}}_{T-1} + \tilde{\mathcal{L}}_T$$

sequentially optimize $\mathcal{L}_t$ w.r.t. $q(\mathbf{z}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{<t})$, holding past expectations fixed

$$q^*(\mathbf{z}_t|\mathbf{x}_{\leq t}, \mathbf{z}_{<t}) \leftarrow \arg\max_q \tilde{\mathcal{L}}_t$$

Marino *et al.*, 2018b

FILTERING VARIATIONAL LOWER BOUND

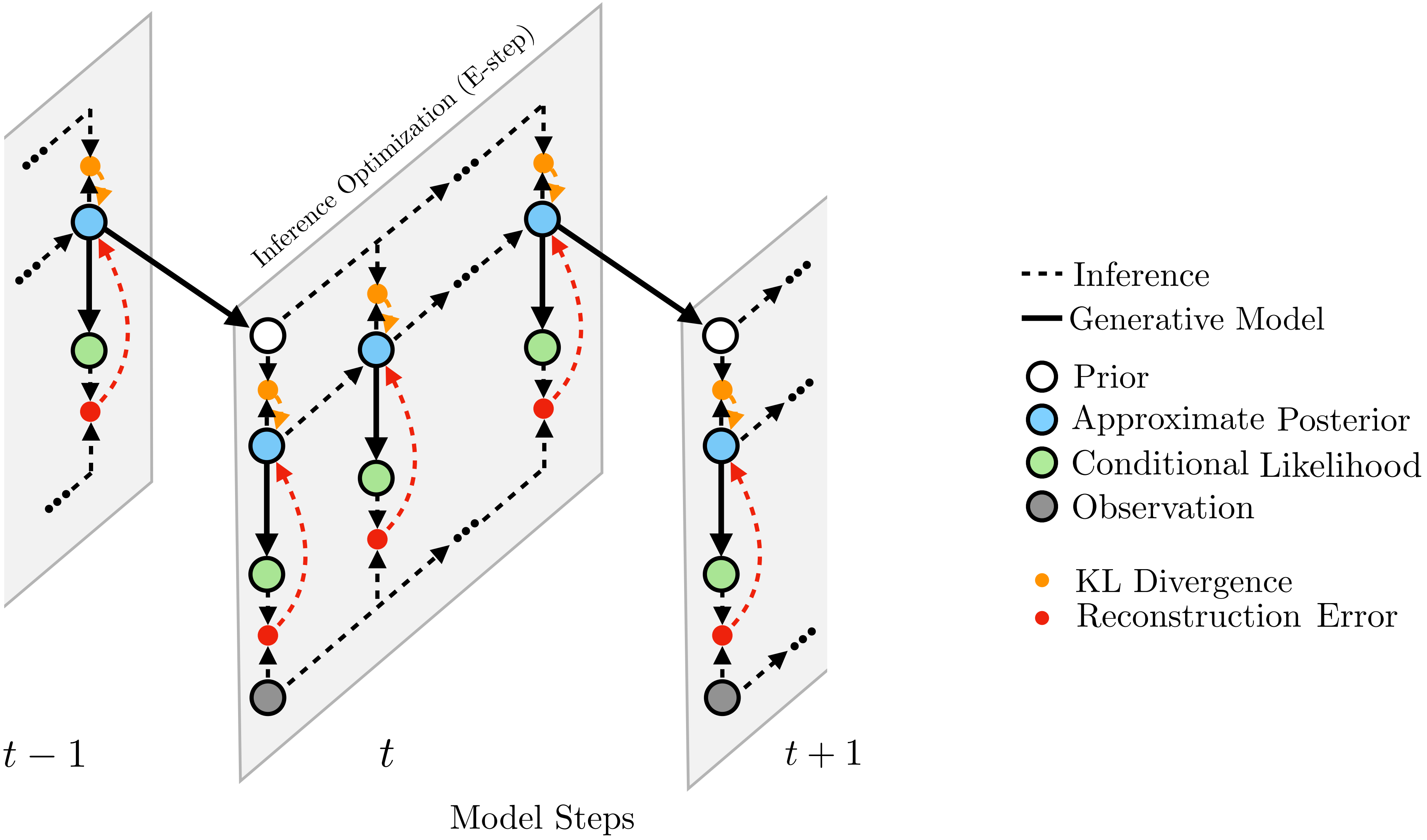


Algorithm 1 Variational Filtering Expectation Maximization

```
**Input:** observation sequence $\mathbf{x}_{1:T}$, model $p_\theta(\mathbf{x}_{1:T}, \mathbf{z}_{1:T})$
$\nabla_\theta \mathcal{L} = 0$                                                  ▷ parameter gradient
**for** $t = 1$ **to** $T$ **do**
    initialize $q(\mathbf{z}_t | \mathbf{x}_{\leq t}, \mathbf{z}_{<t})$                ▷ at/near $p_\theta(\mathbf{z}_t | \mathbf{x}_{<t}, \mathbf{z}_{<t})$
    $\widetilde{\mathcal{L}}_t := \mathbb{E}_{q(\mathbf{z}_{<t} | \mathbf{x}_{<t}, \mathbf{z}_{<t-1})} [\mathcal{L}_t]$
    $q(\mathbf{z}_t | \mathbf{x}_{\leq t}, \mathbf{z}_{<t}) = \arg\max_q \tilde{\mathcal{L}}_t$      ▷ inference (E-Step)
    $\nabla_\theta \mathcal{L} = \nabla_\theta \mathcal{L} + \nabla_\theta \widetilde{\mathcal{L}}_t$
**end for**
$\theta = \theta + \alpha \nabla_\theta \mathcal{L}$                                       ▷ learning (M-Step)
```

Marino *et al.*, 2018b

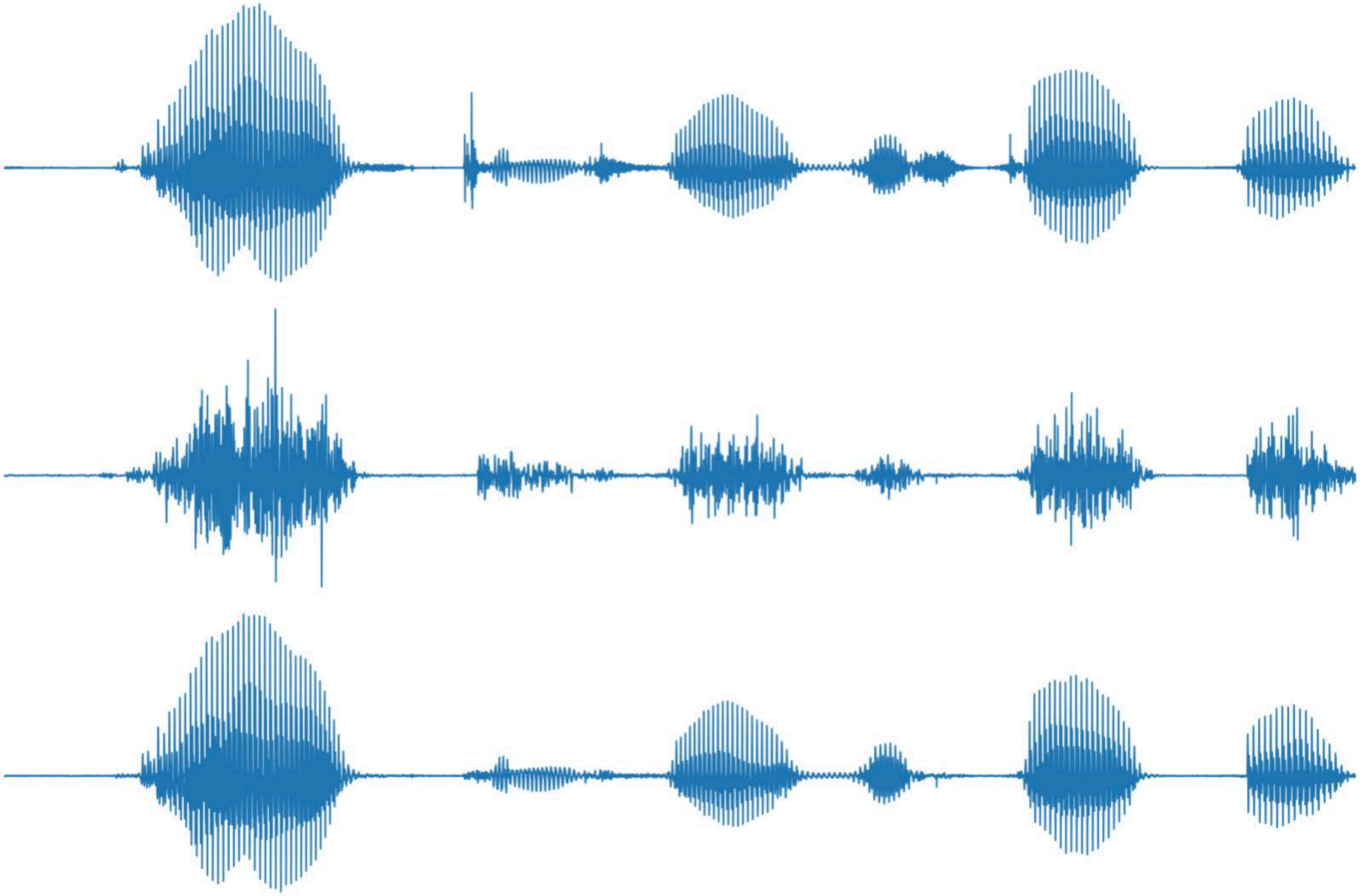
AMORTIZED VARIATIONAL FILTERING



Marino *et al.*, 2018b

VISUALIZING INFERENCE IMPROVEMENT

TIMIT audio waveforms



observation

model output, iteration 0

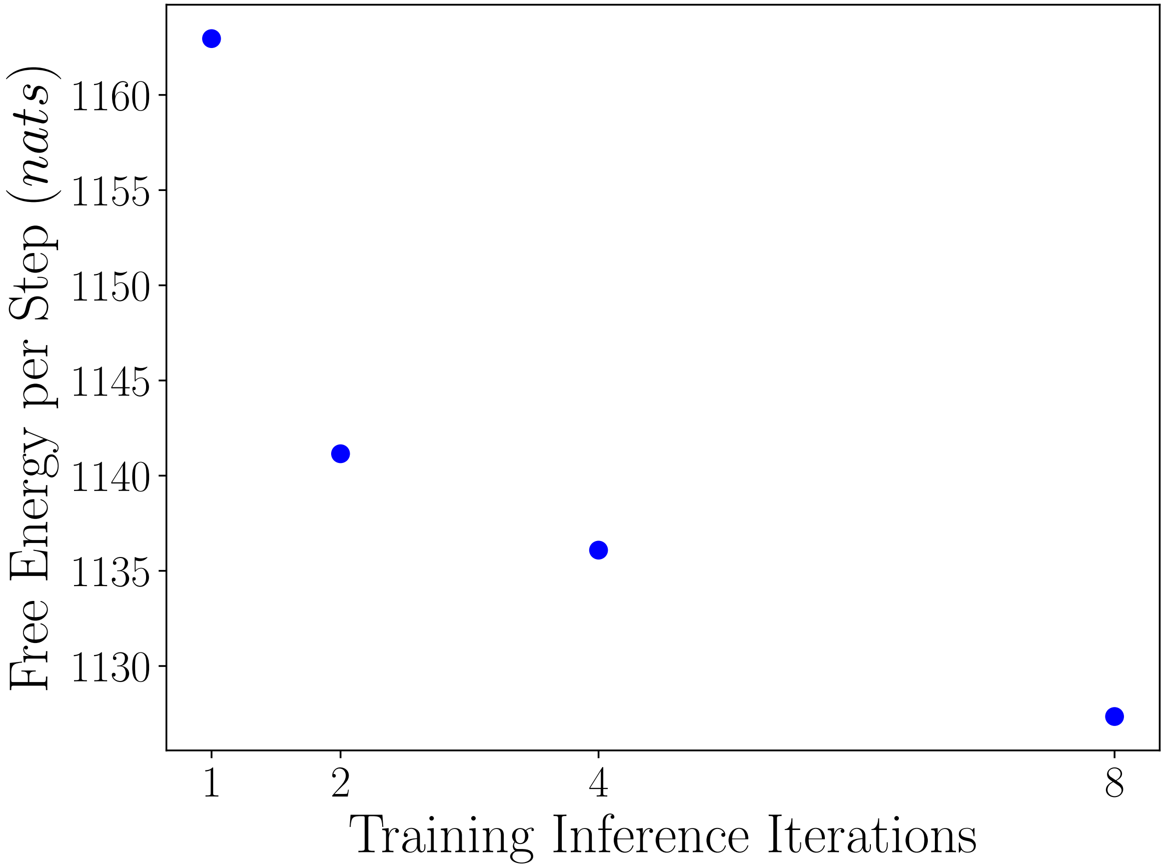
model output, iteration 1

Marino *et al.*, 2018b

INFERENCE ITERATIONS

training with additional inference iterations results in improved performance

ON TIMIT VAL SET

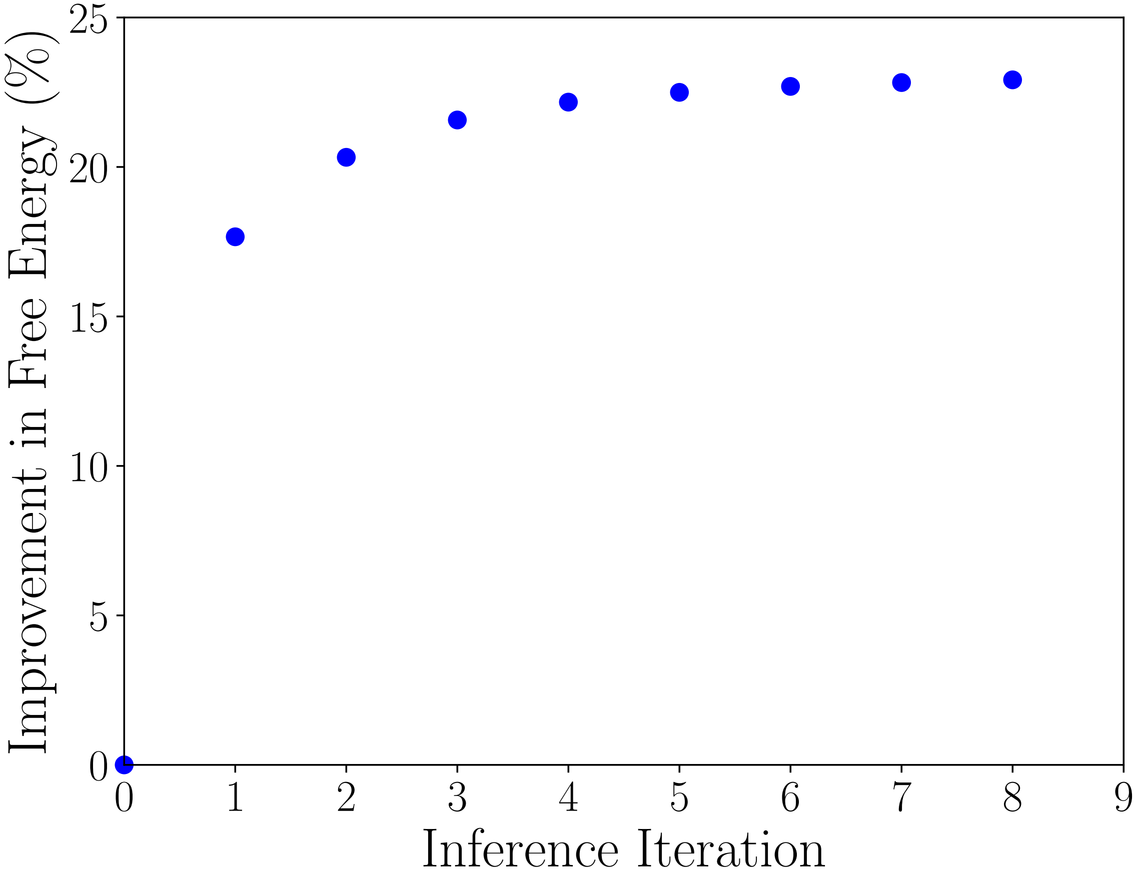


Marino *et al.*, 2018b

INFERENCE ITERATIONS

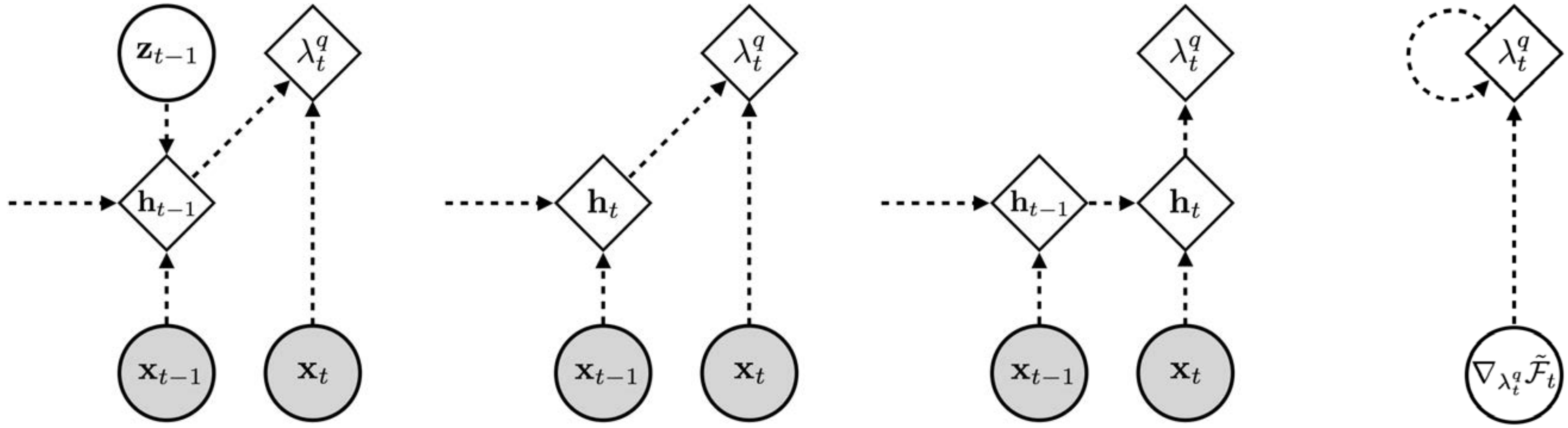
each inference iteration yields decreasing relative improvement

ON TIMIT VAL SET



Marino *et al.*, 2018b

FILTERING INFERENCE MODELS



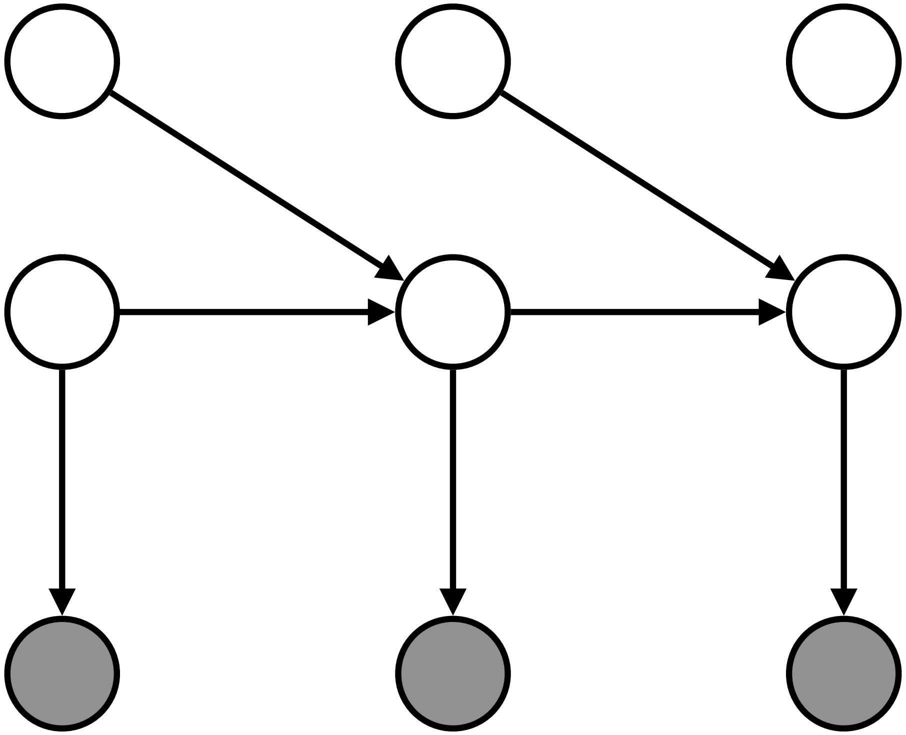
VRNN SRNN SVG AVF

$$\underbrace{\hspace{12em}}$$

custom-designed

general-purpose

Marino *et al.*, 2018b



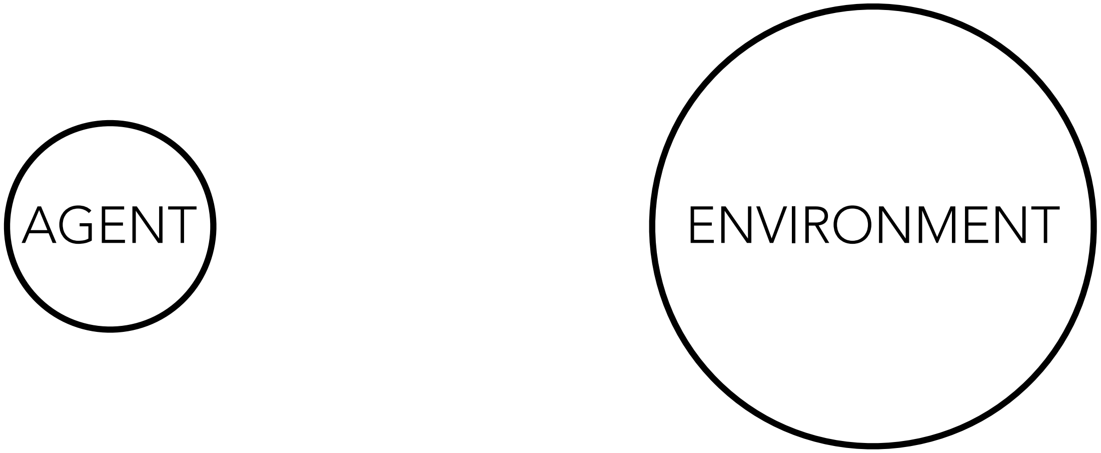
MODEL-BASED REINFORCEMENT LEARNING

REINFORCEMENT LEARNING

sequential decision making by maximizing expected future reward

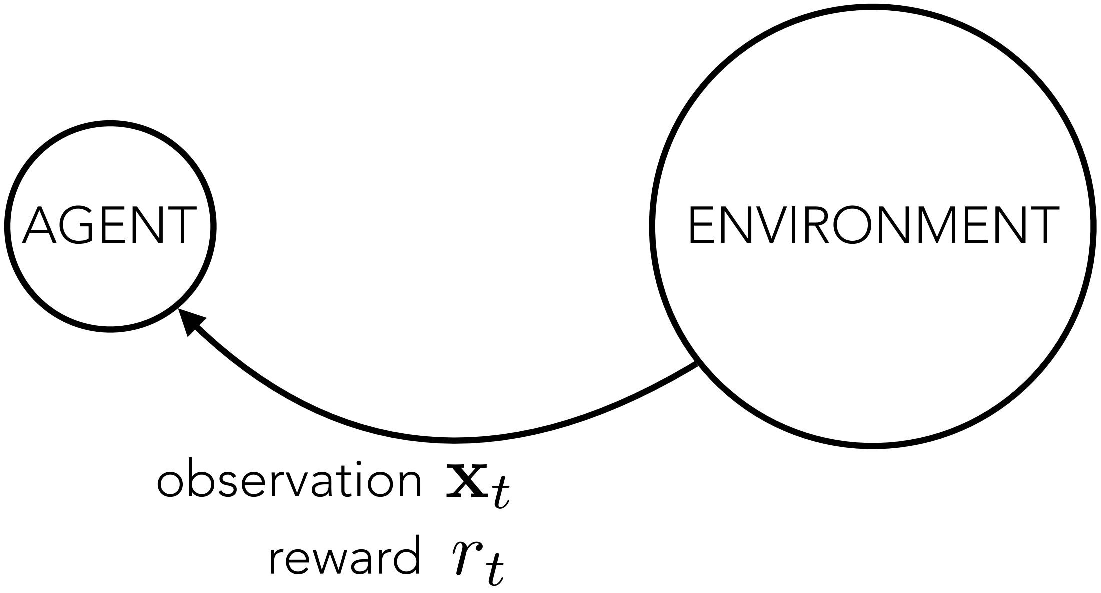
REINFORCEMENT LEARNING

agent-environment interaction



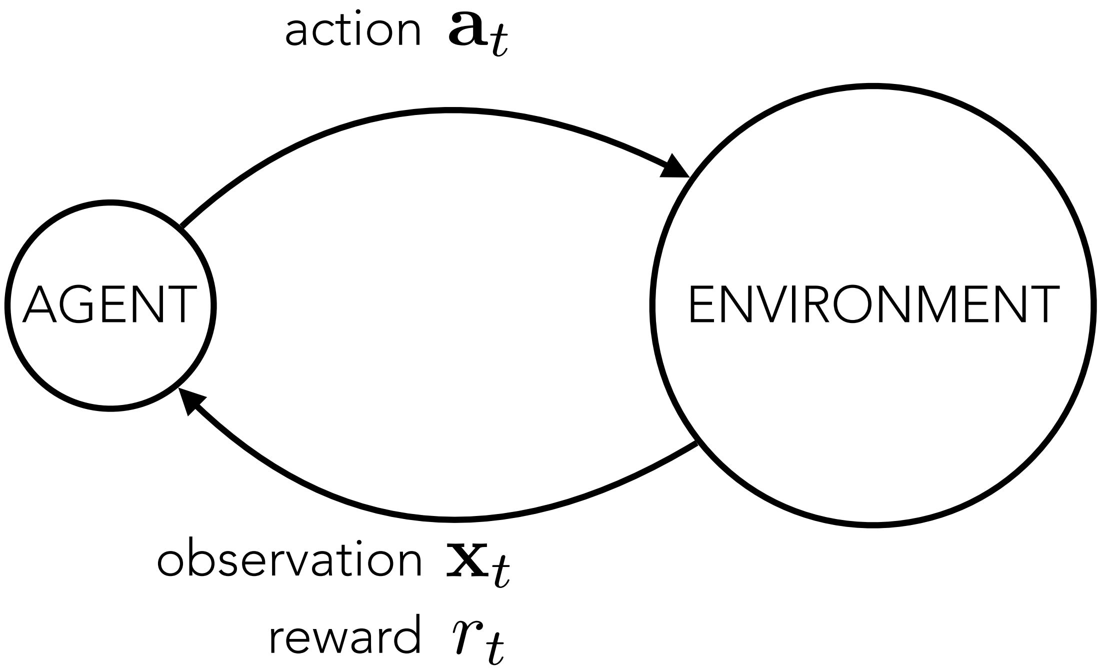
REINFORCEMENT LEARNING

agent-environment interaction



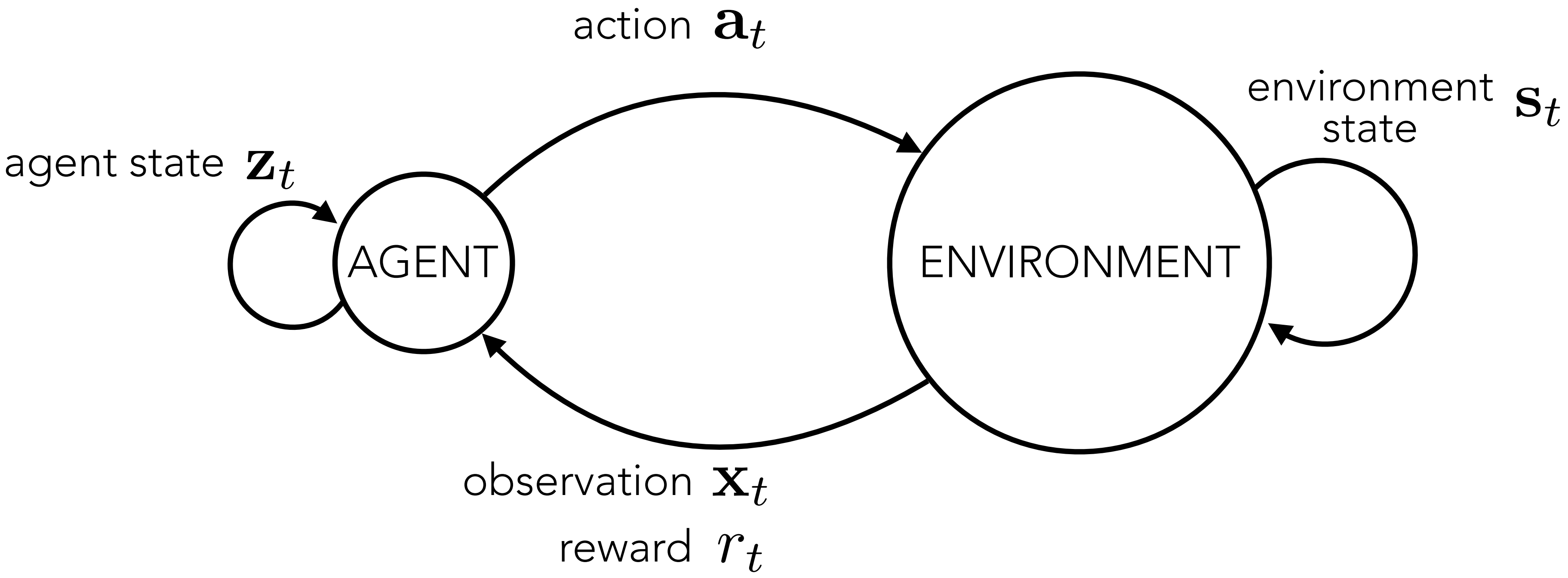
REINFORCEMENT LEARNING

agent-environment interaction



REINFORCEMENT LEARNING

agent-environment interaction


action $\mathbf{a}_t$
environment state $\mathbf{s}_t$
agent state $\mathbf{z}_t$
AGENT
ENVIRONMENT
observation $\mathbf{x}_t$
reward r_t

REINFORCEMENT LEARNING

a *policy* is a probability distribution over actions: $\mathbf{a} \sim \pi(\mathbf{a}|\cdot)$

RL objective:

maximize the expected sum of rewards (*return*)

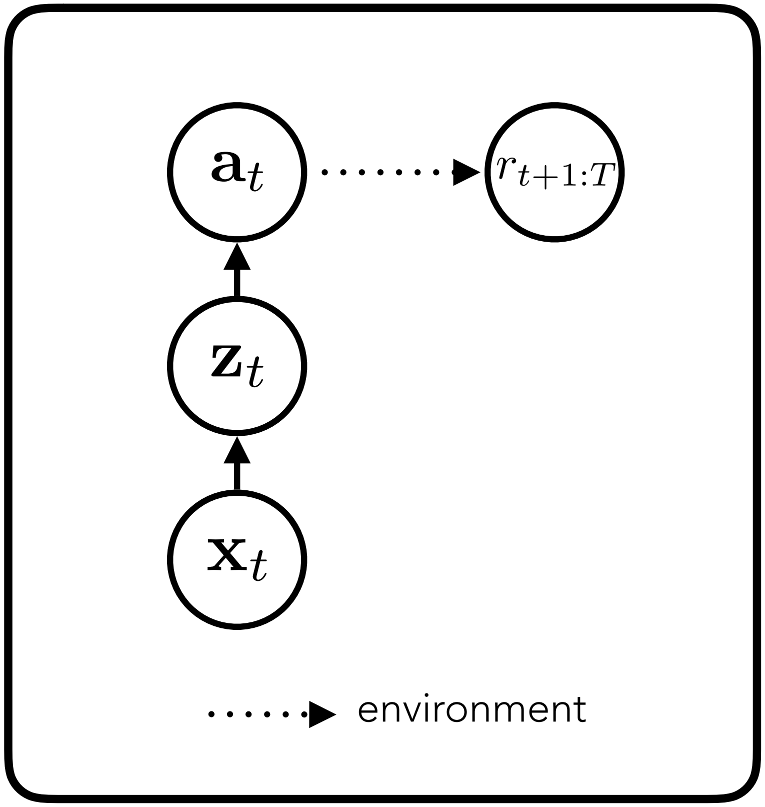
$$\pi(\mathbf{a}|\cdot) \leftarrow \arg\max_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=1}^{T} r_t \right]$$

REINFORCEMENT LEARNING

approaches to optimizing the RL objective

model-free

direct mapping to actions

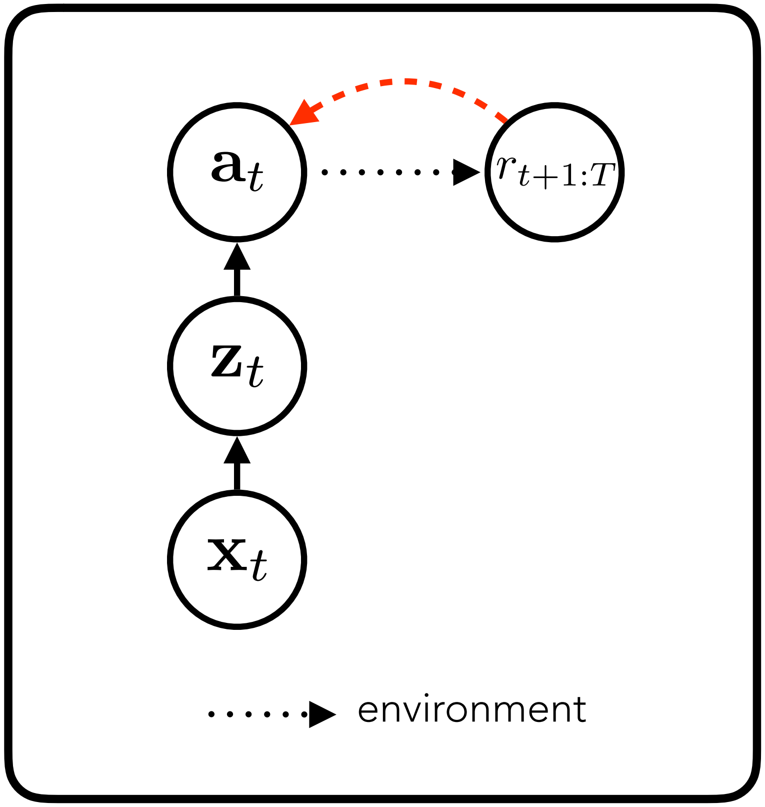


REINFORCEMENT LEARNING

approaches to optimizing the RL objective

model-free

direct mapping to actions

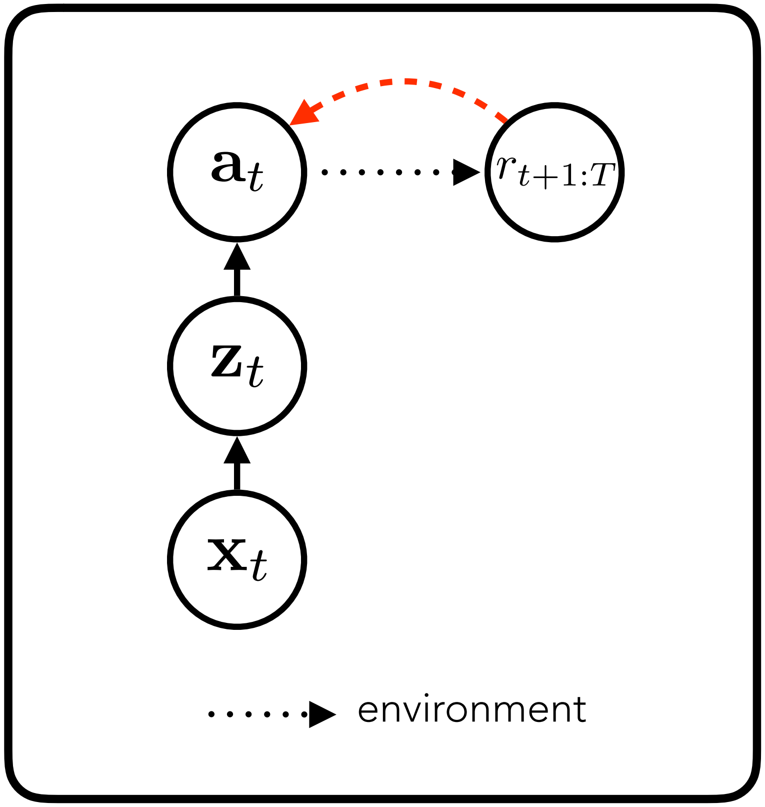


REINFORCEMENT LEARNING

approaches to optimizing the RL objective

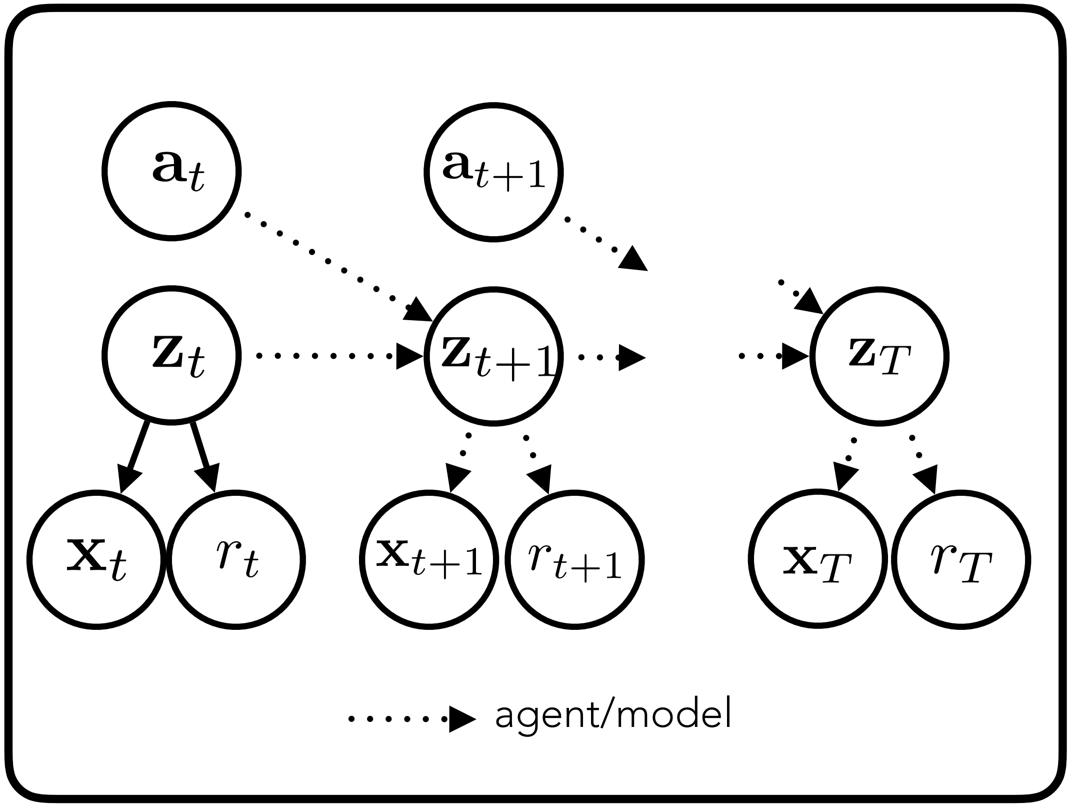
model-free

direct mapping to actions



model-based

unroll model to evaluate actions

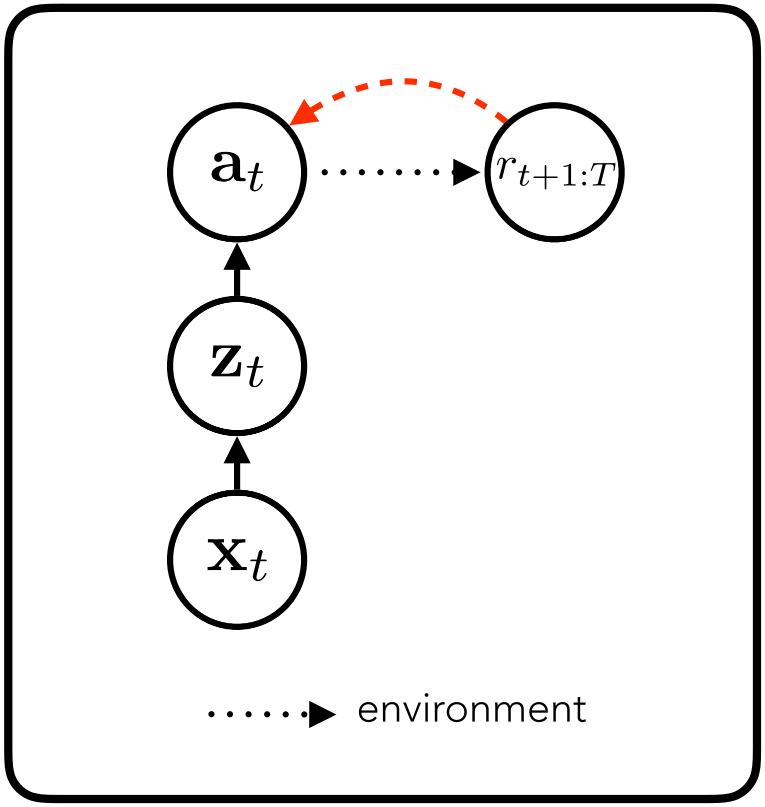


REINFORCEMENT LEARNING

approaches to optimizing the RL objective

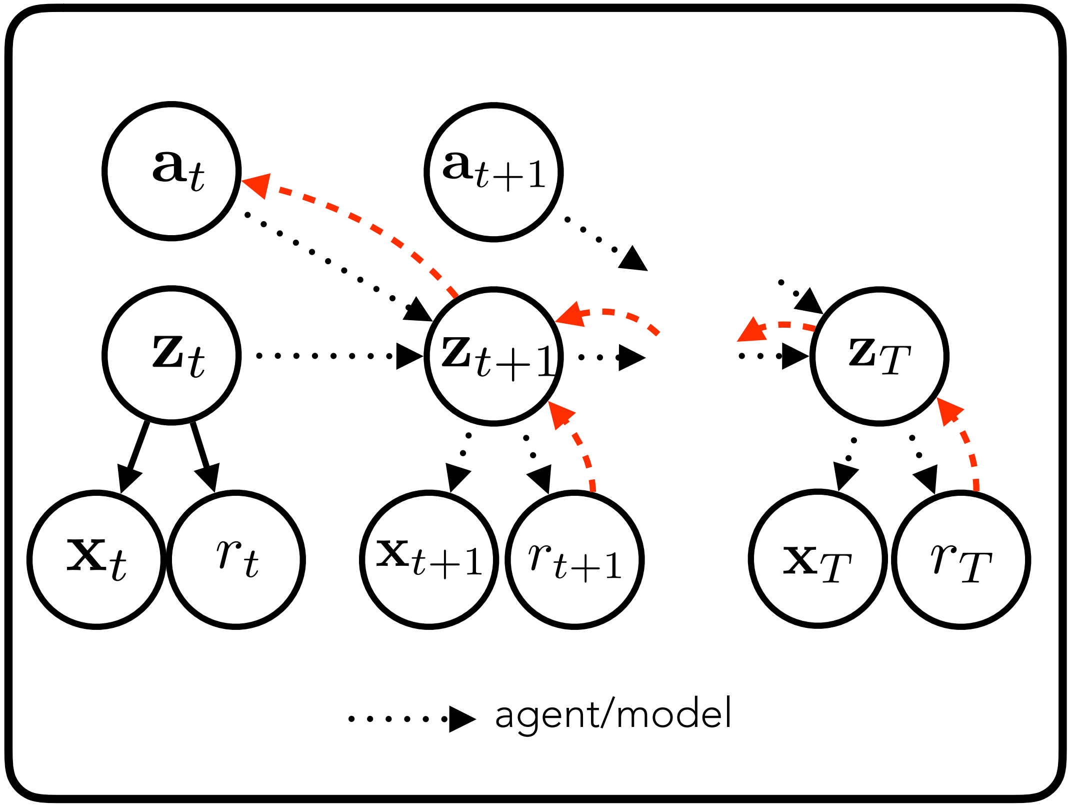
model-free

direct mapping to actions



model-based

unroll model to evaluate actions

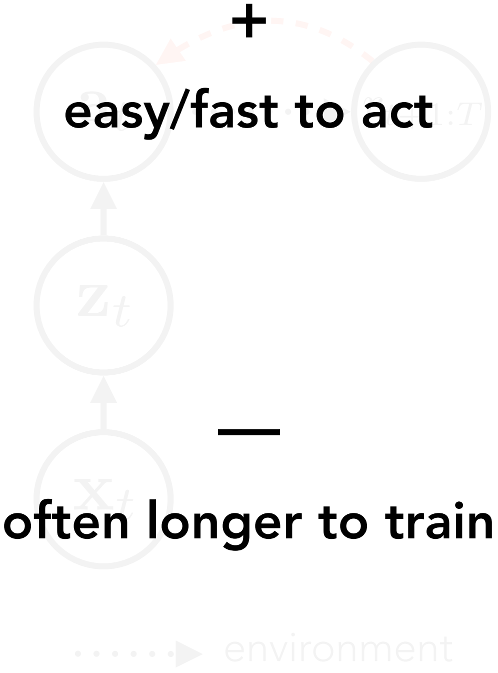


REINFORCEMENT LEARNING

approaches to optimizing the RL objective

model-free

direct mapping to actions



+

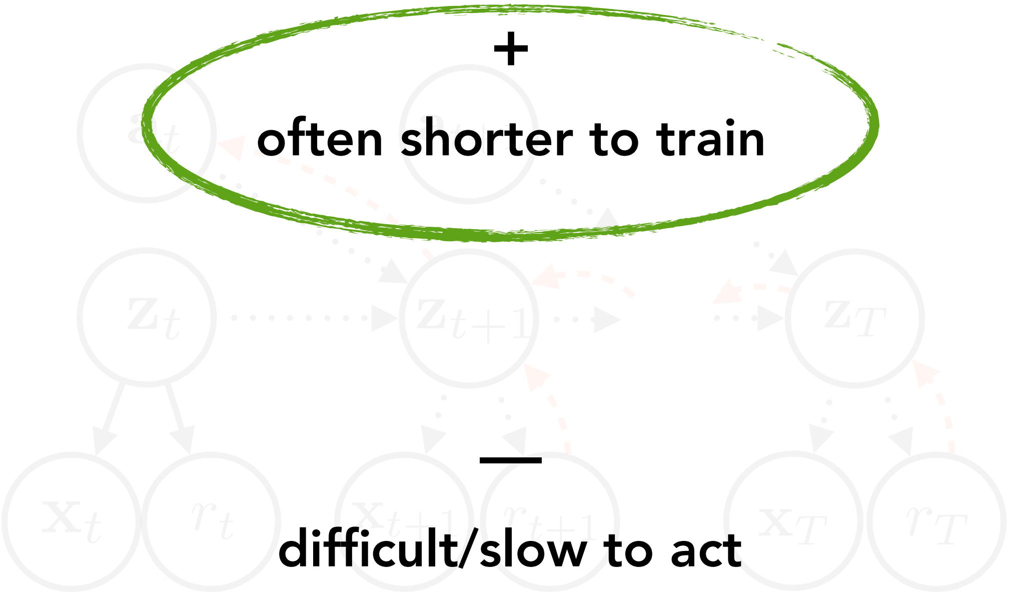
easy/fast to act

—

often longer to train

model-based

unroll model to evaluate actions



+

often shorter to train

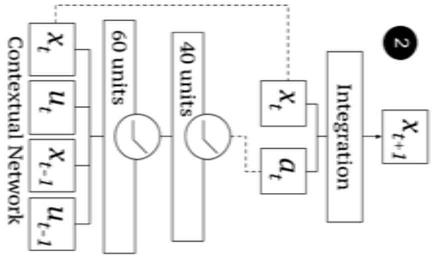
—

difficult/slow to act

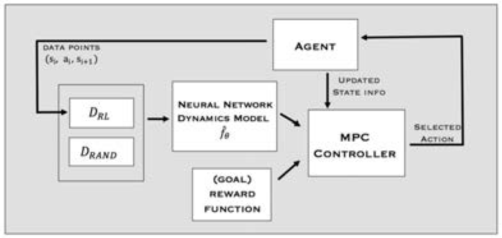
also, **only as good as the model**

RECENT APPROACHES TO MODEL-BASED RL

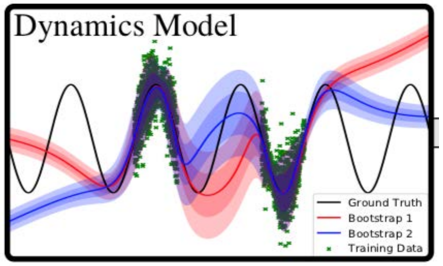
without latent variables:



Fu *et al.*, 2016



Nagabandi *et al.*, 2018

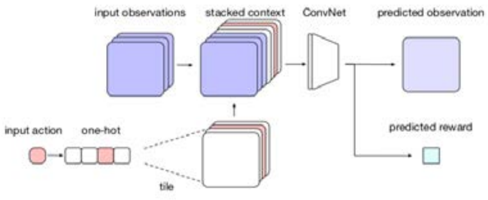


PETS

Chua *et al.*, 2018

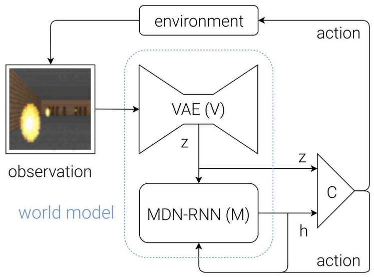
• • •

with latent variables:



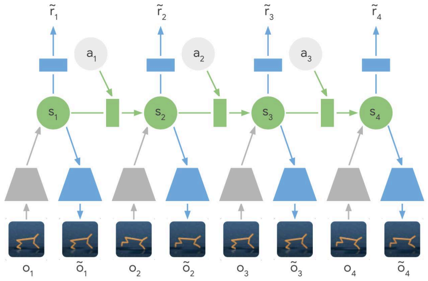
I2A

Weber *et al.*, 2017



World Models

Ha & Schmidhuber, 2018



Planning From Pixels

Hafner *et al.*, 2019

• • •

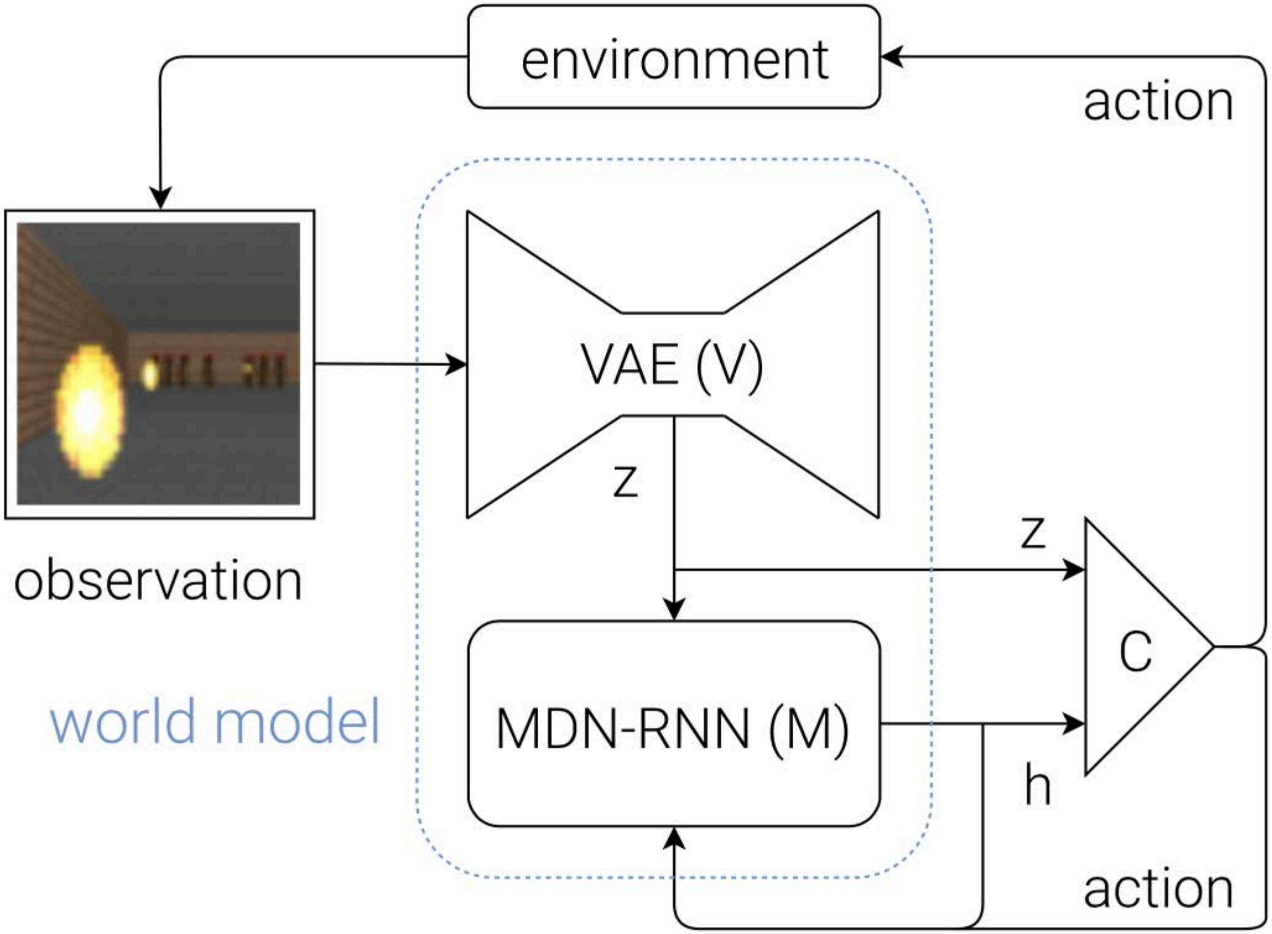
WORLD MODELS

- learn a generative model of environment from pixel observations
- use the model as a simulator to learn actions

Ha & Schmidhuber, 2018

WORLD MODELS

the model:

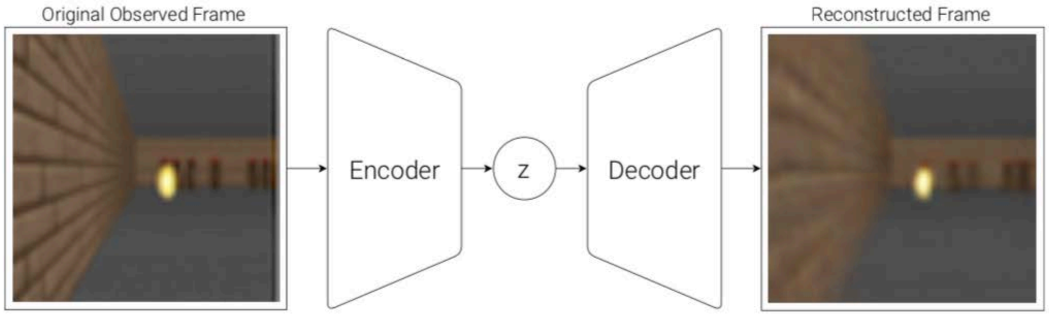


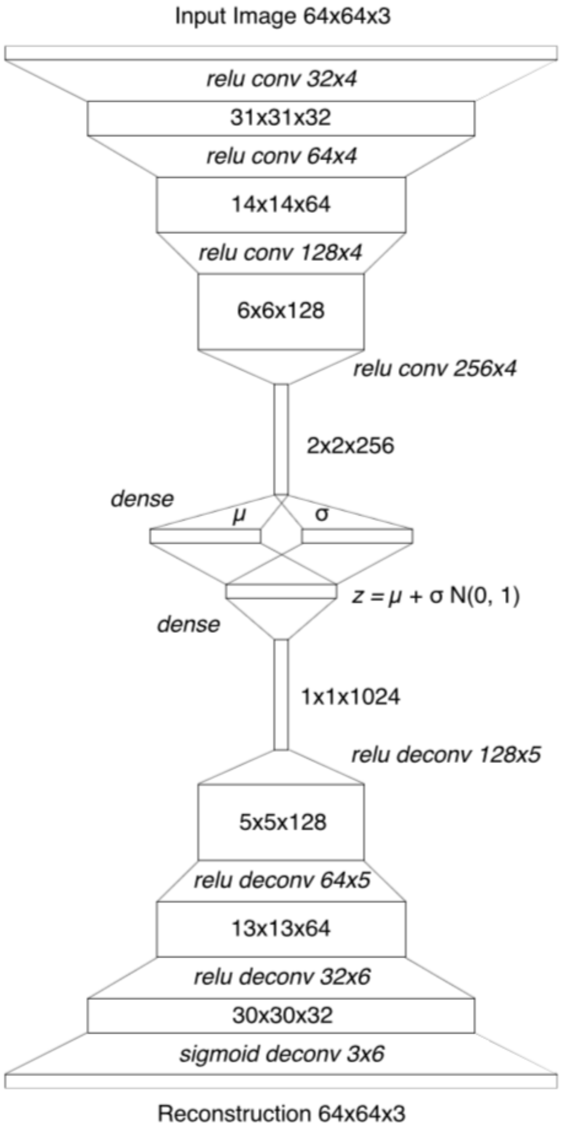
Ha & Schmidhuber, 2018

WORLD MODELS

the model (vision):

compress the observations



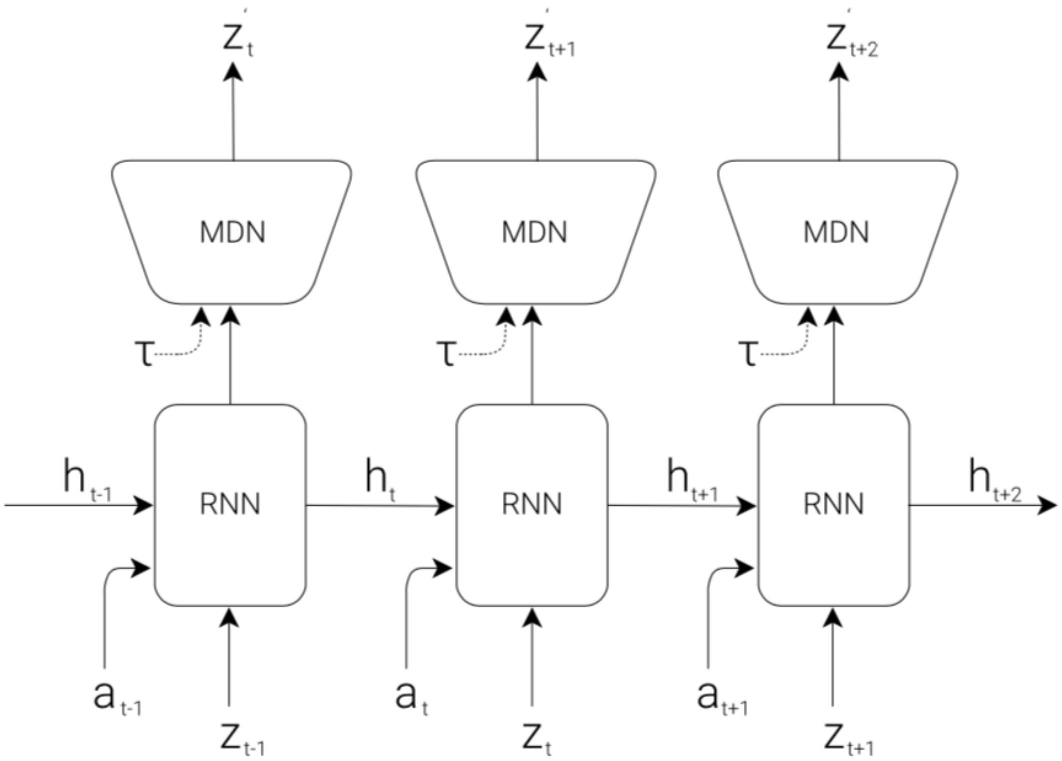


Ha & Schmidhuber, 2018

WORLD MODELS

the model (dynamics):

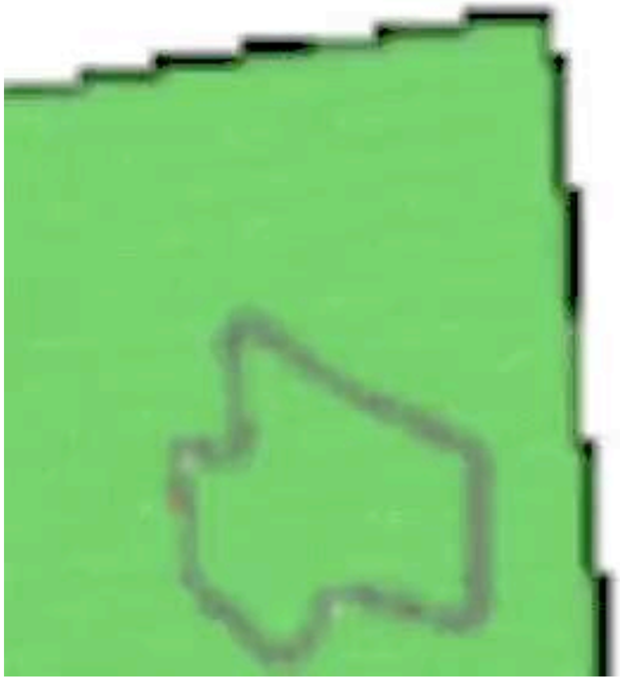
learn the dynamics of compressed state representations



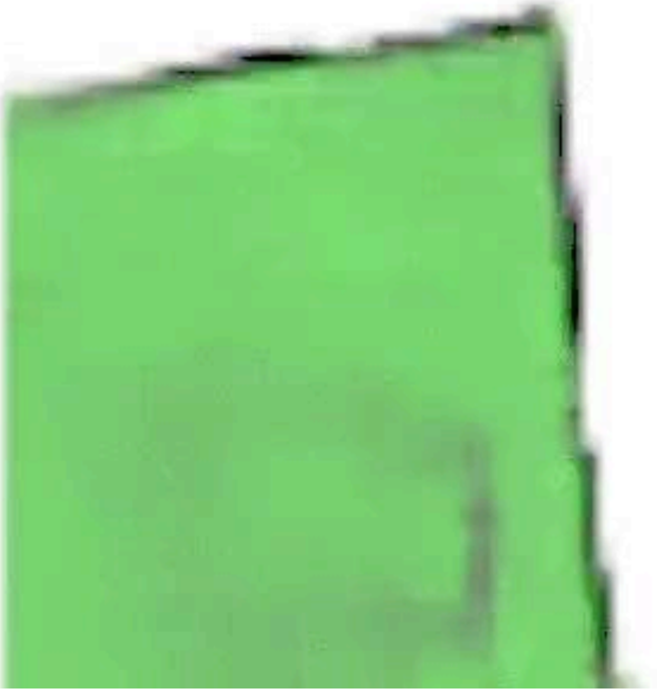
Ha & Schmidhuber, 2018

WORLD MODELS

`CarRacing-v0`



observations

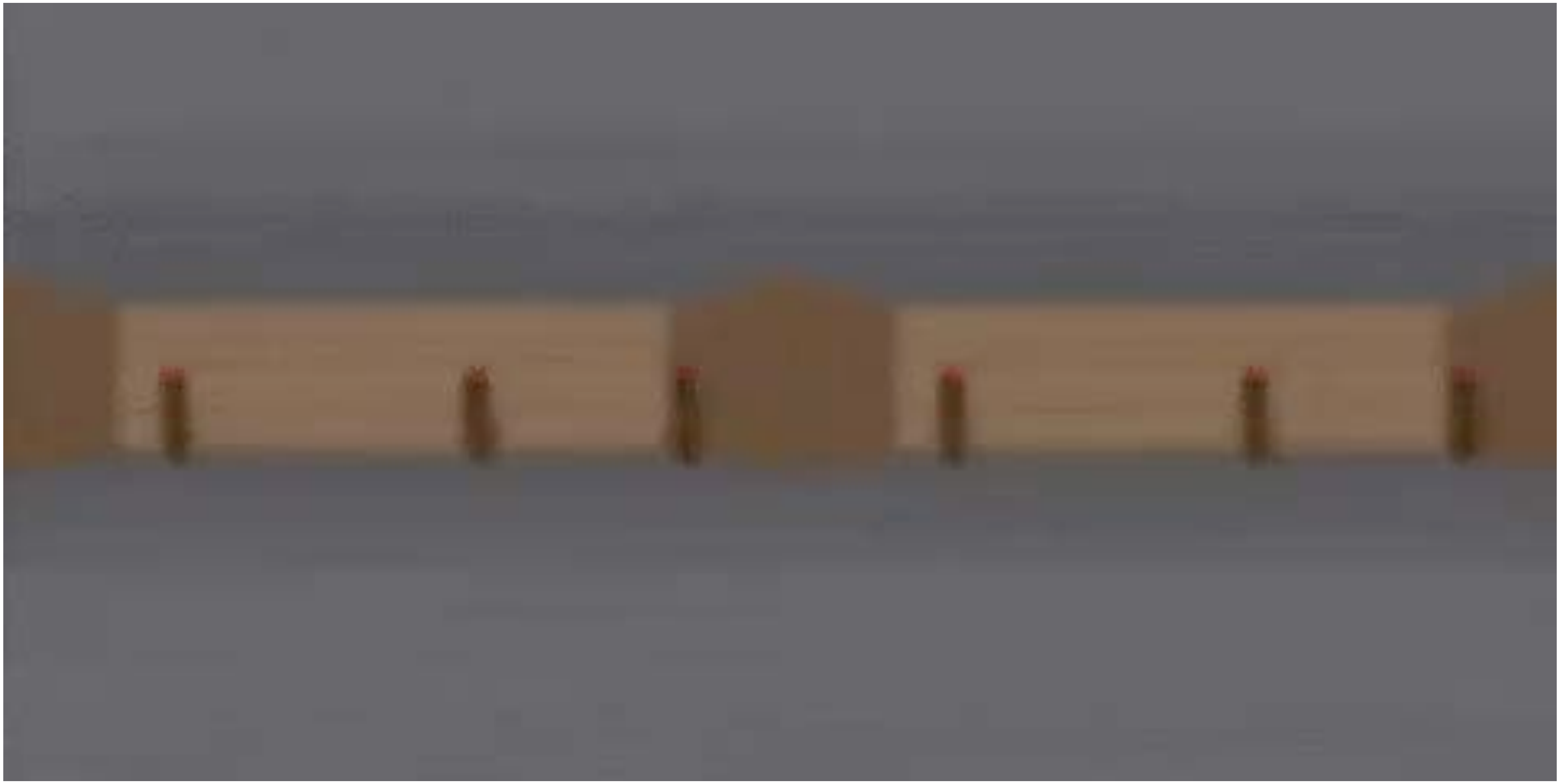


reconstructions

Ha & Schmidhuber, 2018

WORLD MODELS

`VizDoomTakeCover`



observations reconstructions

Ha & Schmidhuber, 2018

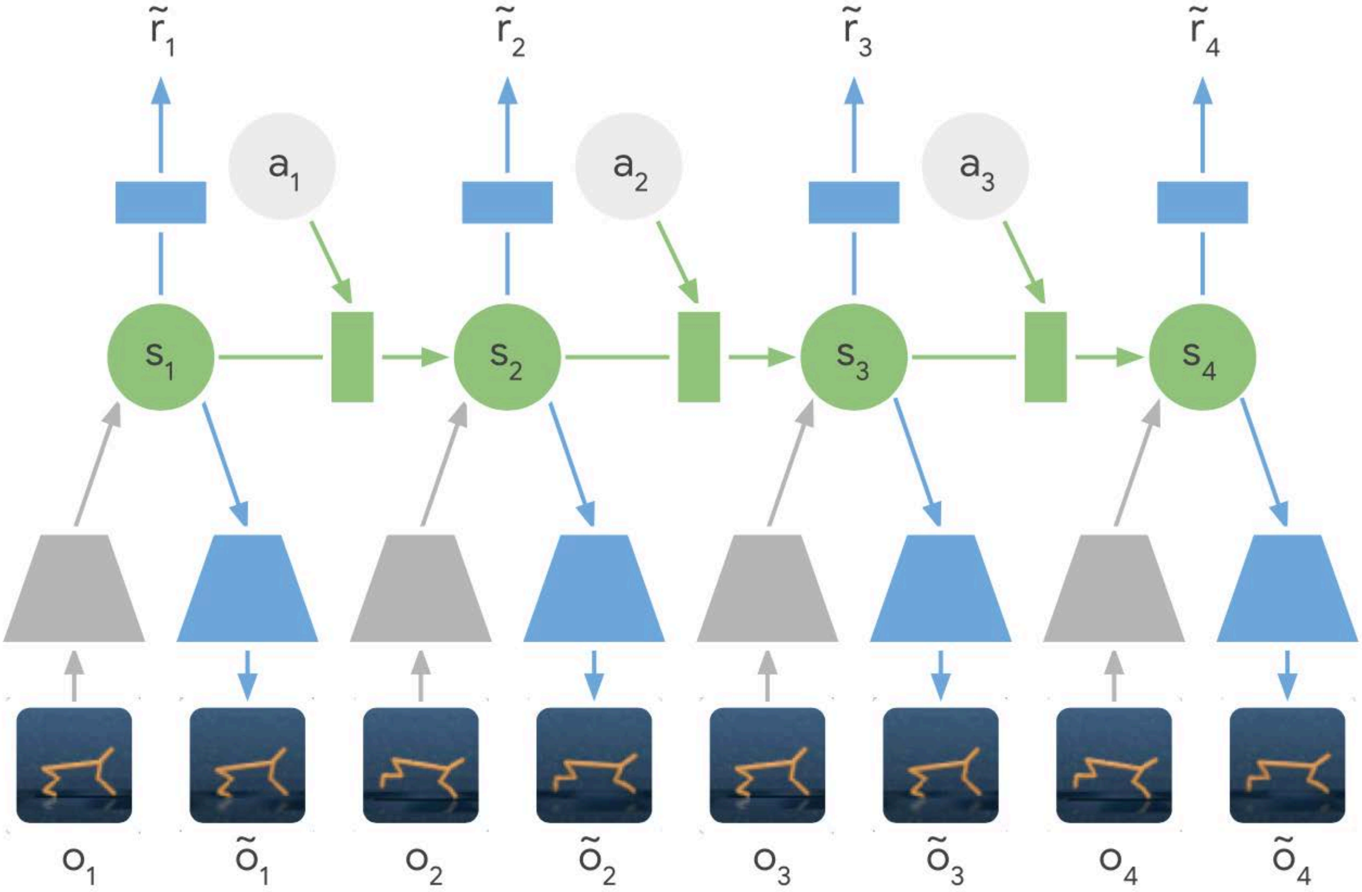
PLANNING FROM PIXELS

- learn a generative model of environment from pixel observations
- use the model for planning actions

Hafner *et al.*, 2019

PLANNING FROM PIXELS

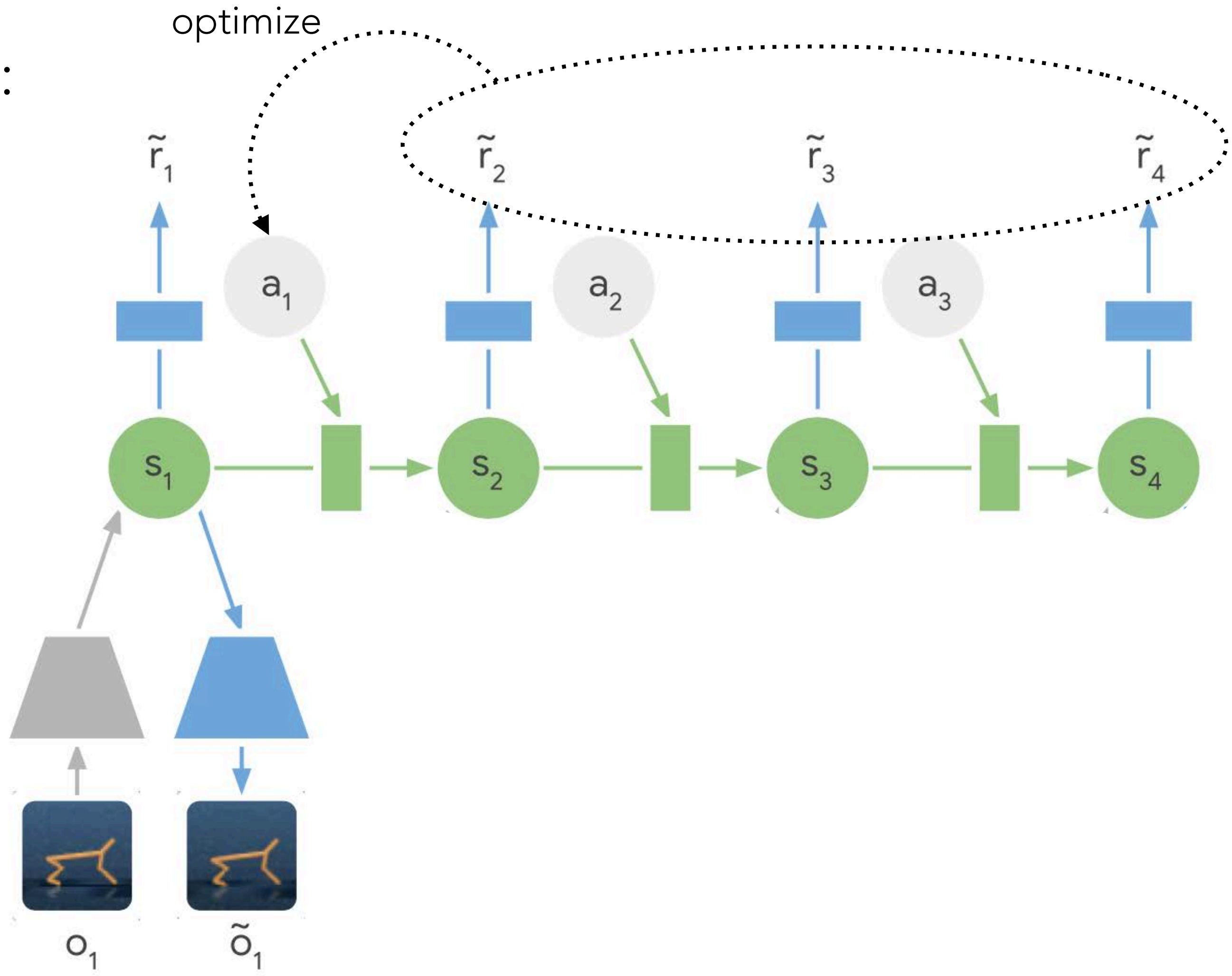
the model:



Hafner *et al.*, 2019

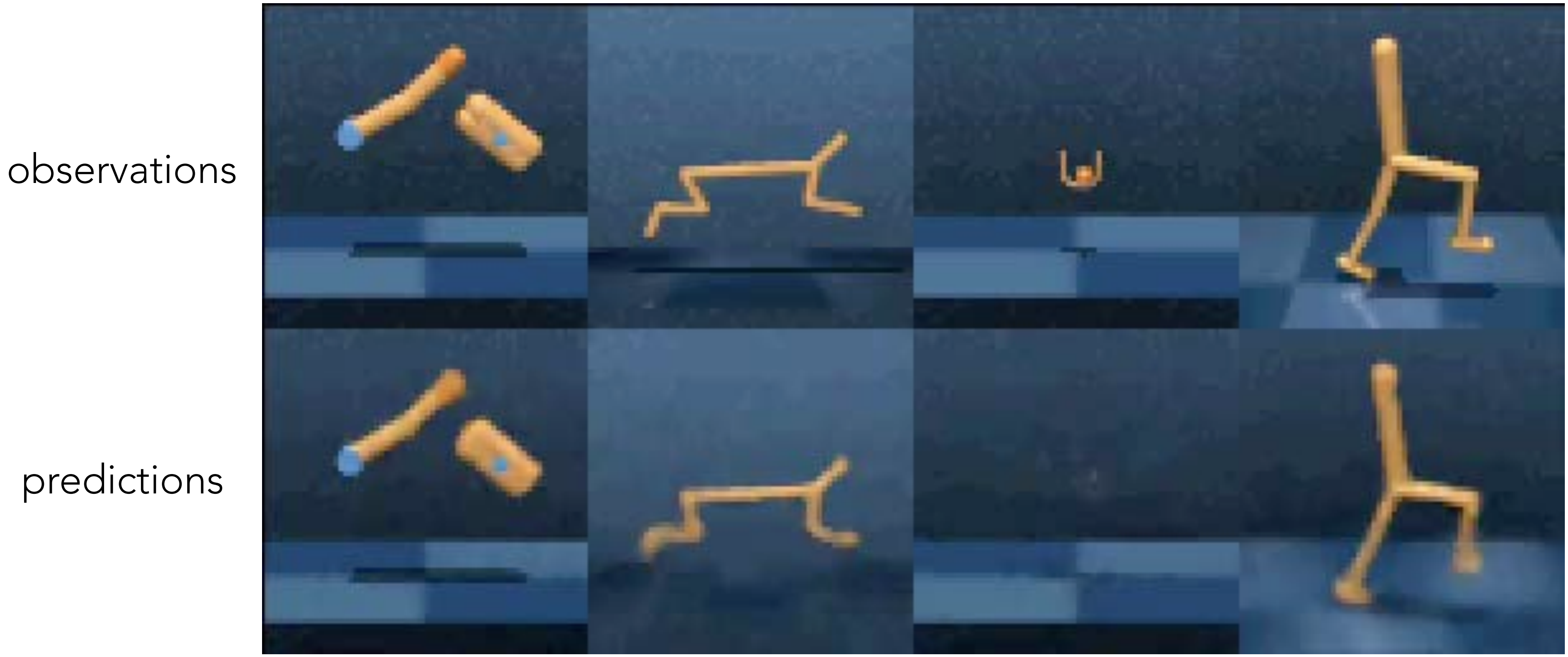
PLANNING FROM PIXELS

planning:



Hafner *et al.*, 2019

PLANNING FROM PIXELS

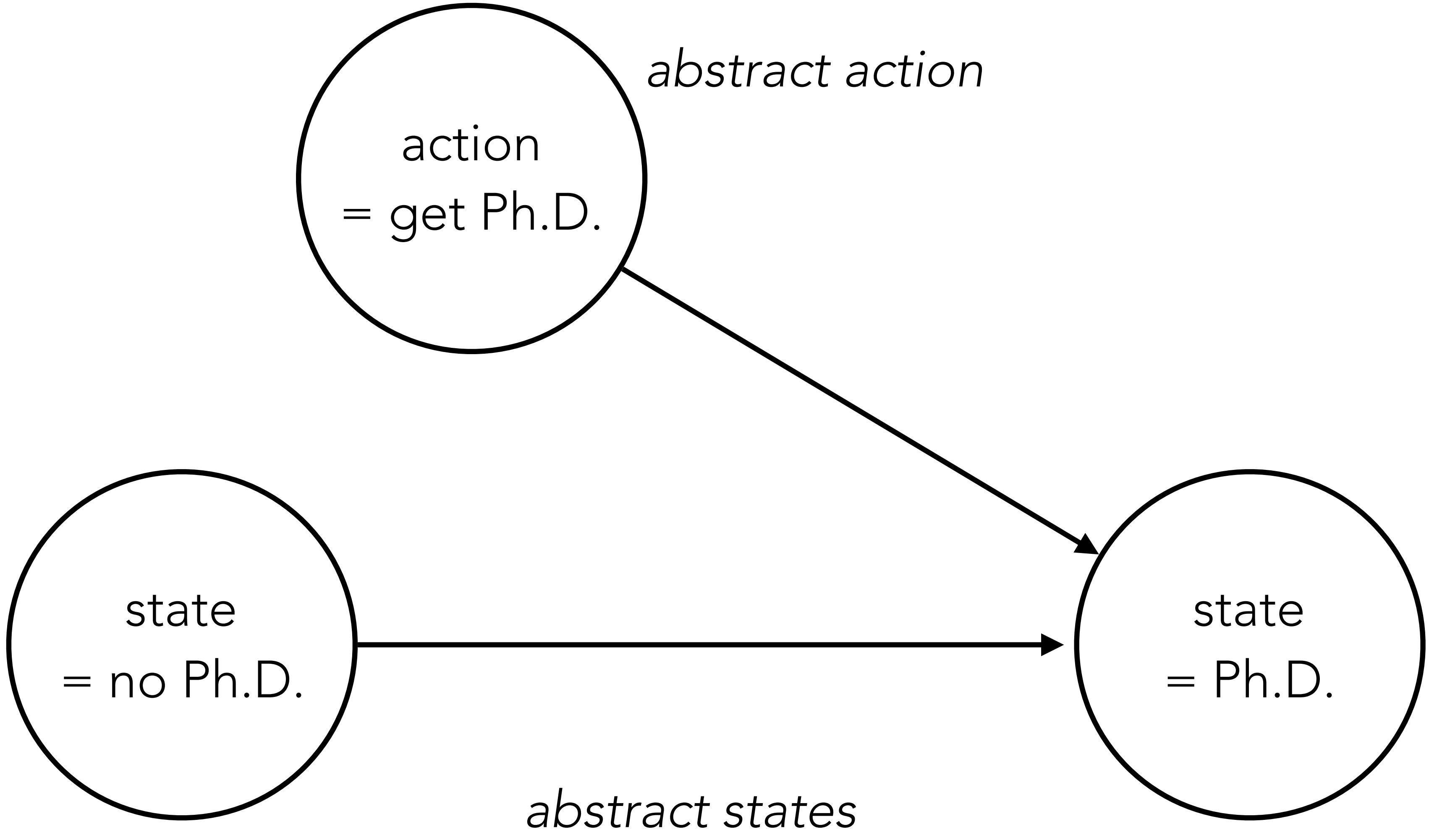


Hafner *et al.*, 2019

OPEN RESEARCH AREAS IN MODEL-BASED RL

TEMPORAL ABSTRACTION

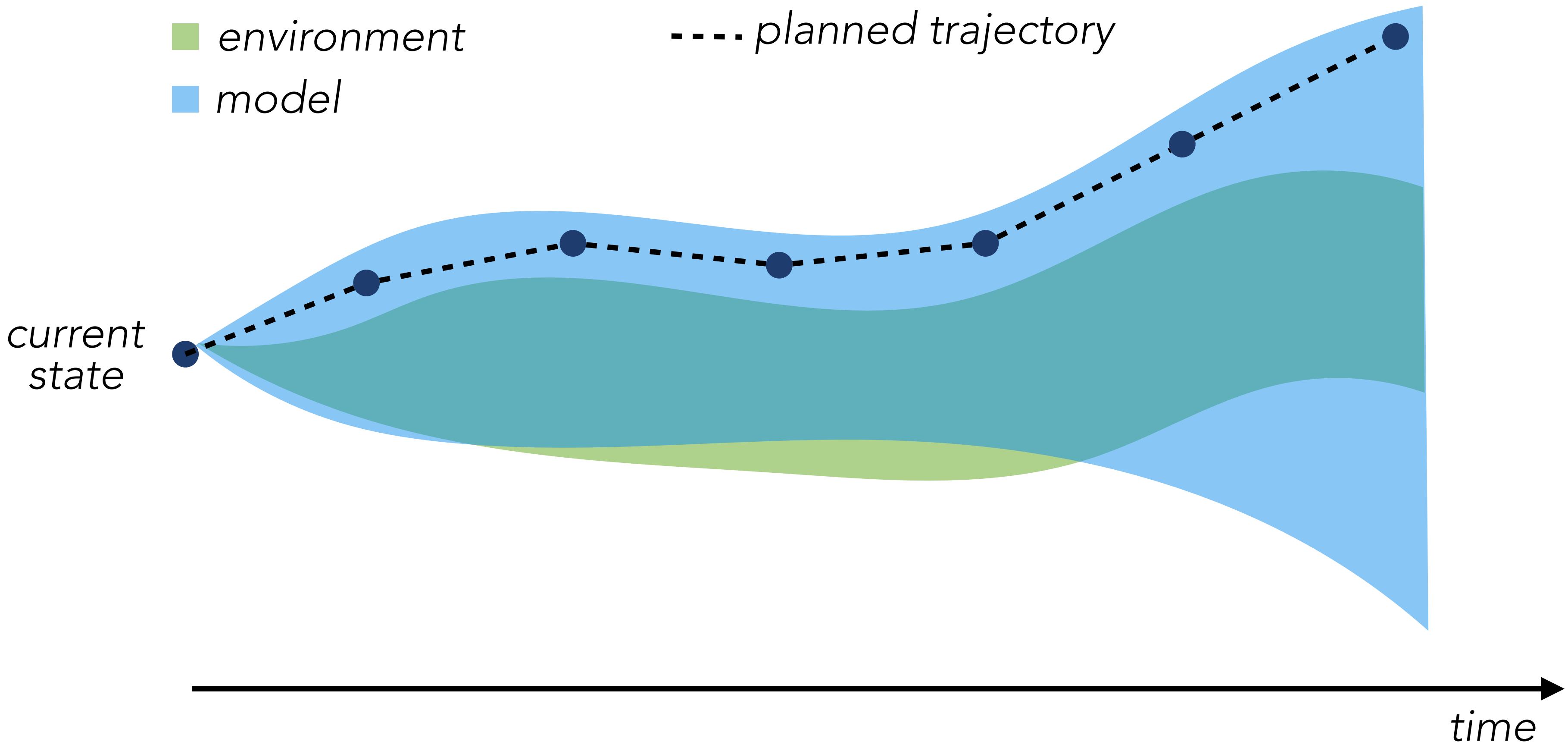
hierarchy of states and actions



UNCERTAINTY ESTIMATION

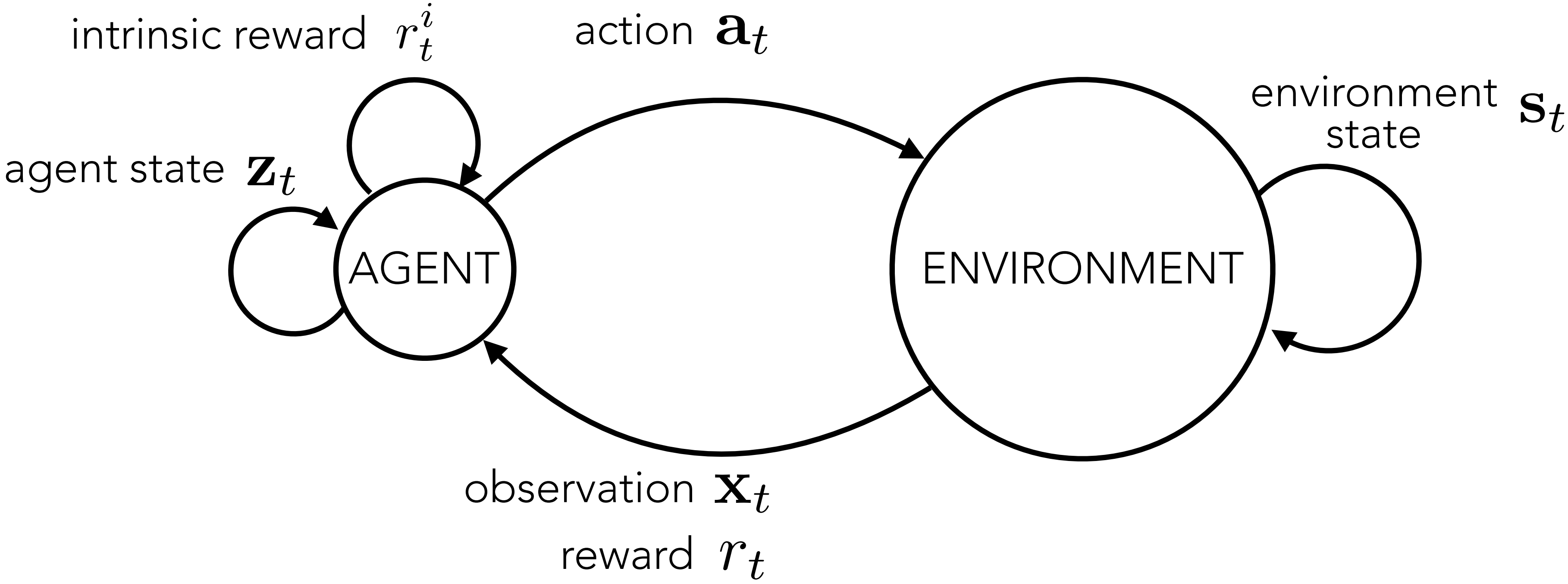
distinguish between model uncertainty and environment stochasticity

prevent regions of exploitability in the model



INTRINSIC MOTIVATION

learning from intrinsic (non-environmental) rewards

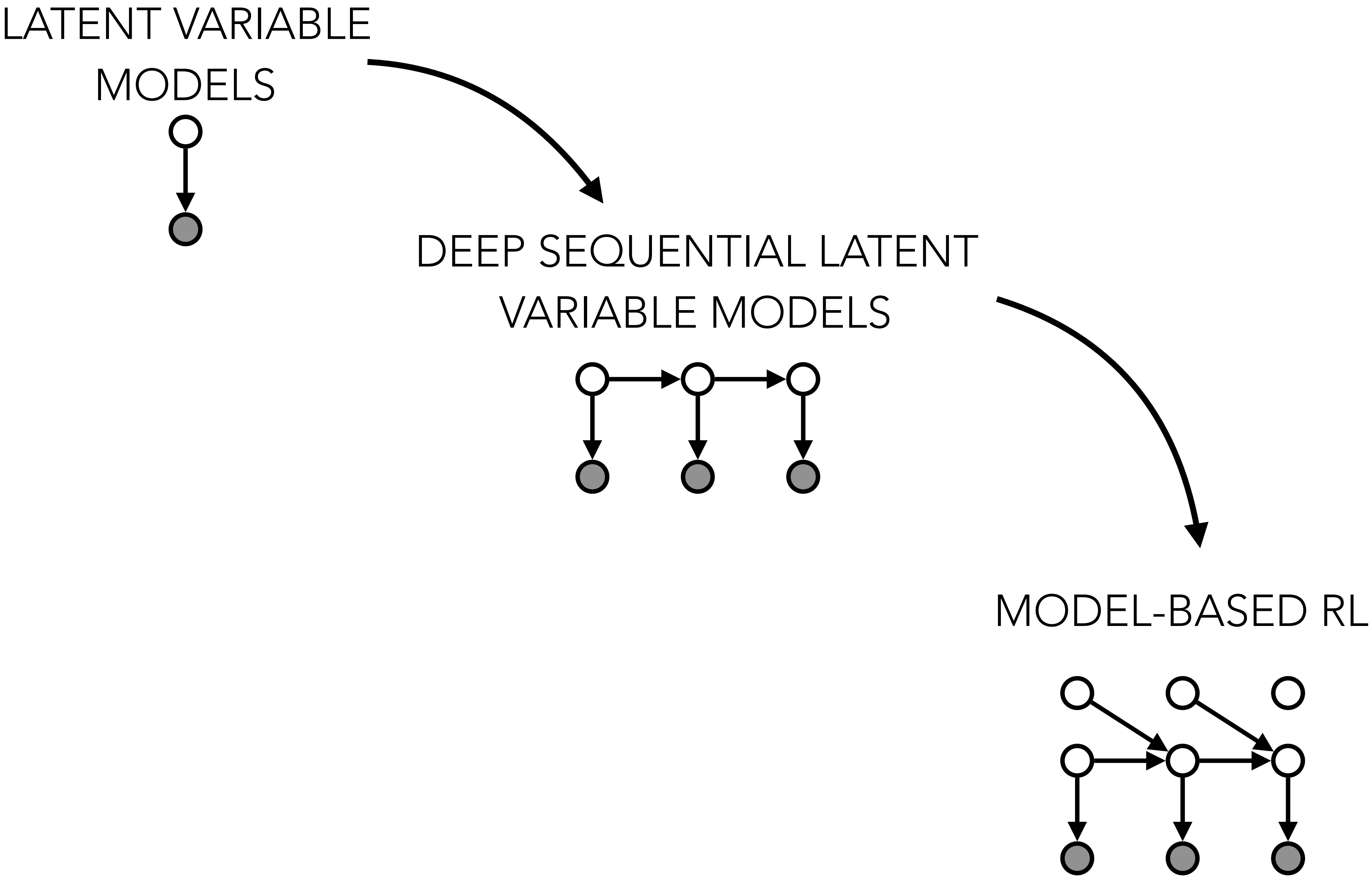

intrinsic reward r_t^i
action $\mathbf{a}_t$
environment state $\mathbf{s}_t$
agent state $\mathbf{z}_t$
AGENT
ENVIRONMENT
observation $\mathbf{x}_t$
reward r_t

intrinsic reward signals:

surprise, empowerment, learning improvement, etc.

often helpful to have a model of the environment to estimate these quantities

OVERVIEW



LATENT VARIABLE MODELS

DEEP SEQUENTIAL LATENT VARIABLE MODELS

MODEL-BASED RL