On Bayesian Deep Learning and Deep Bayesian Learning

Yee Whye Teh

Dept of Statistics
University of Oxford

DeepMind

http://csml.stats.ox.ac.uk/people/teh/

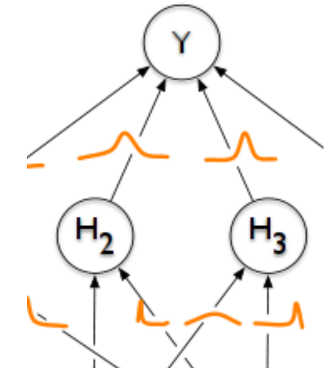
painting credit: DeepArt.io

Collaborators


Andriy Mnih


Arnaud Doucet

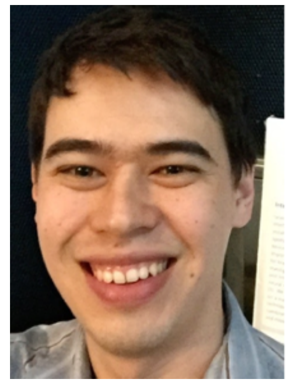

Balaji Lakshminarayanan

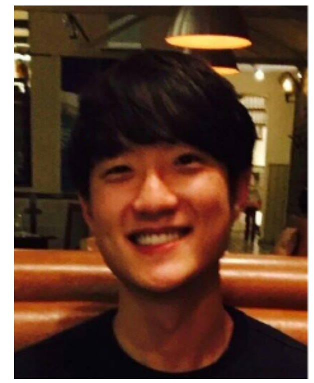


Charles Blundell


Dieterich Lawson


Chris Maddison

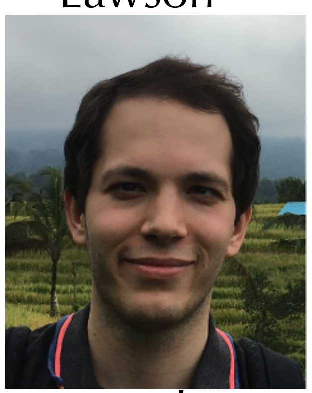

George Tucker


Hyunjik Kim


James Kirkpatrick


Jerome Connor

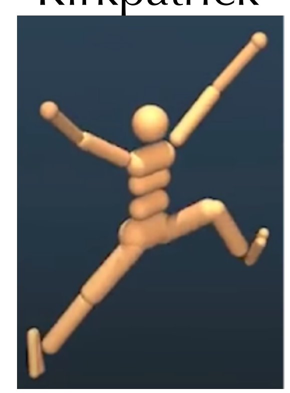

John Quan


Jonathan Schwarz


Leonard Hasenclever


Minjie Xu


Mohammad Norouzi


Nicolas Heess

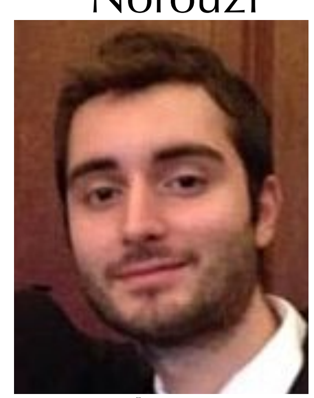

Raia Hadsell


Razvan Pascanu

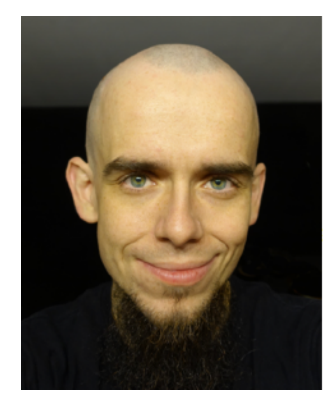

Sebastian Vollmer

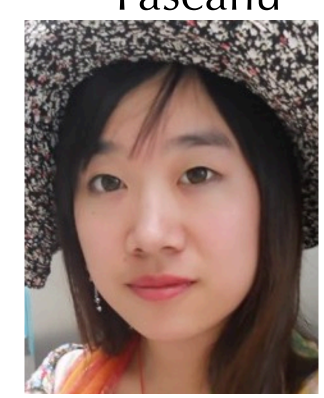

Stefan Webb


Thibaut Lienart

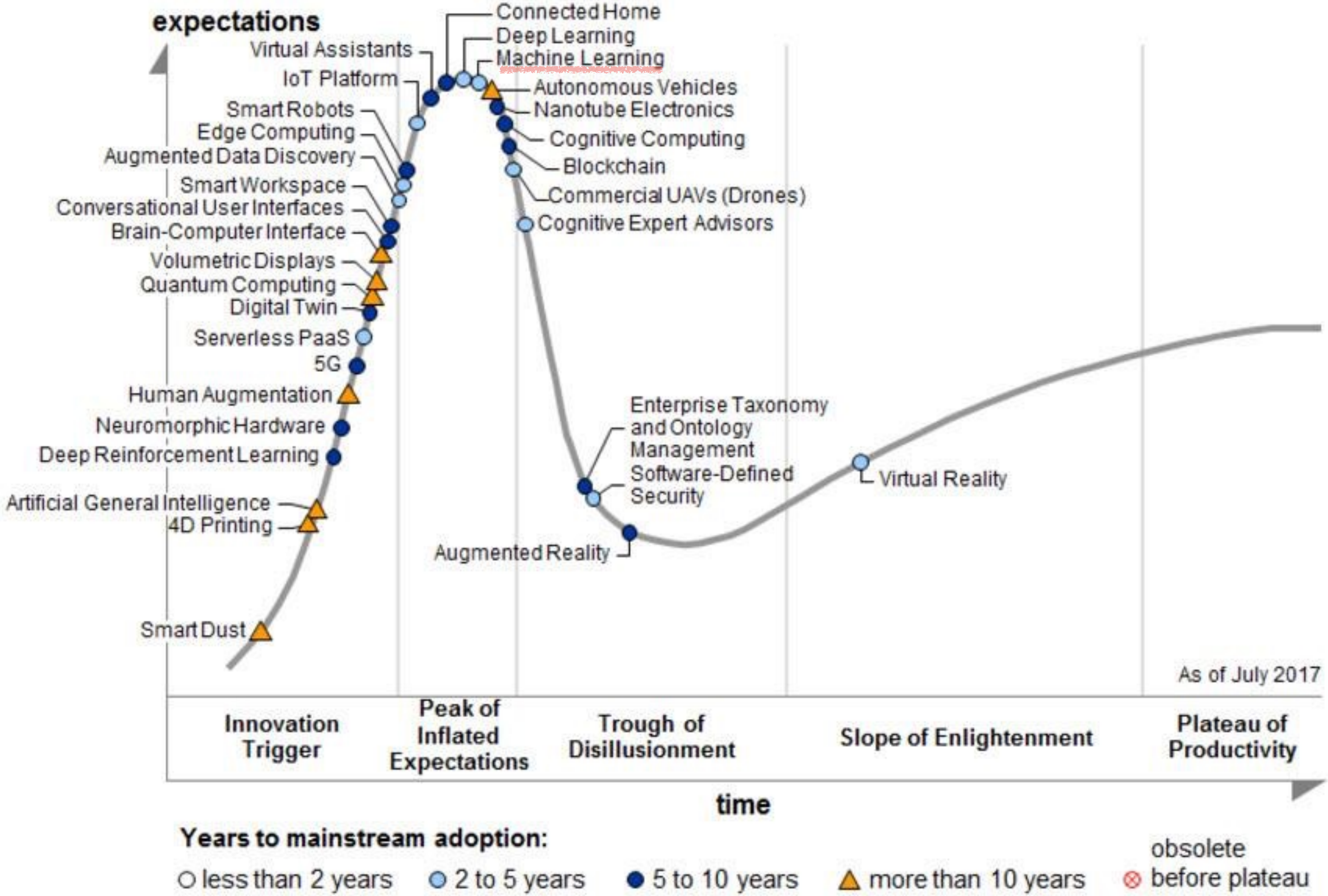

Valerio Perrone


Victor Bapst

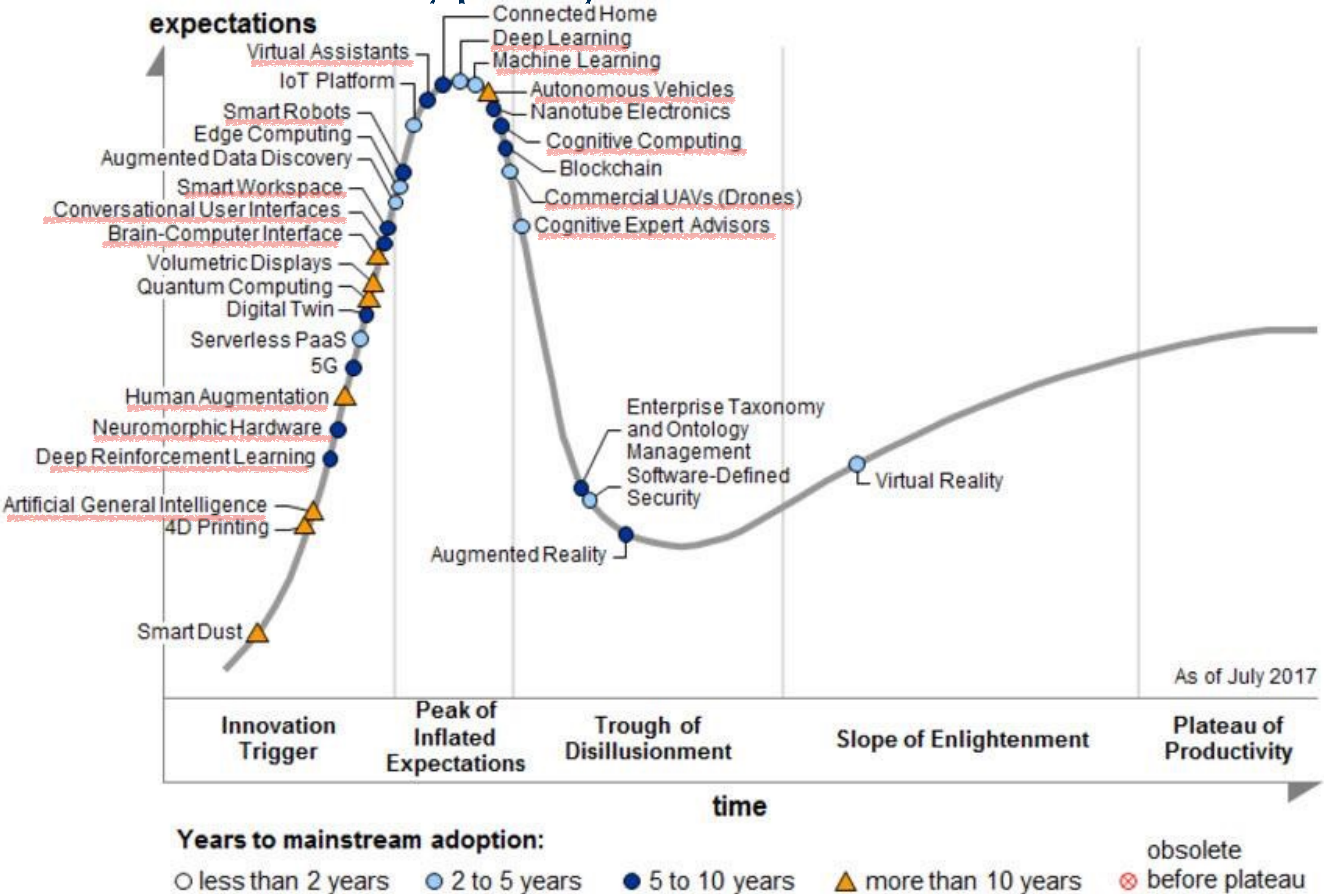

Wojtek Czarnecki


Xiaoyu Lu

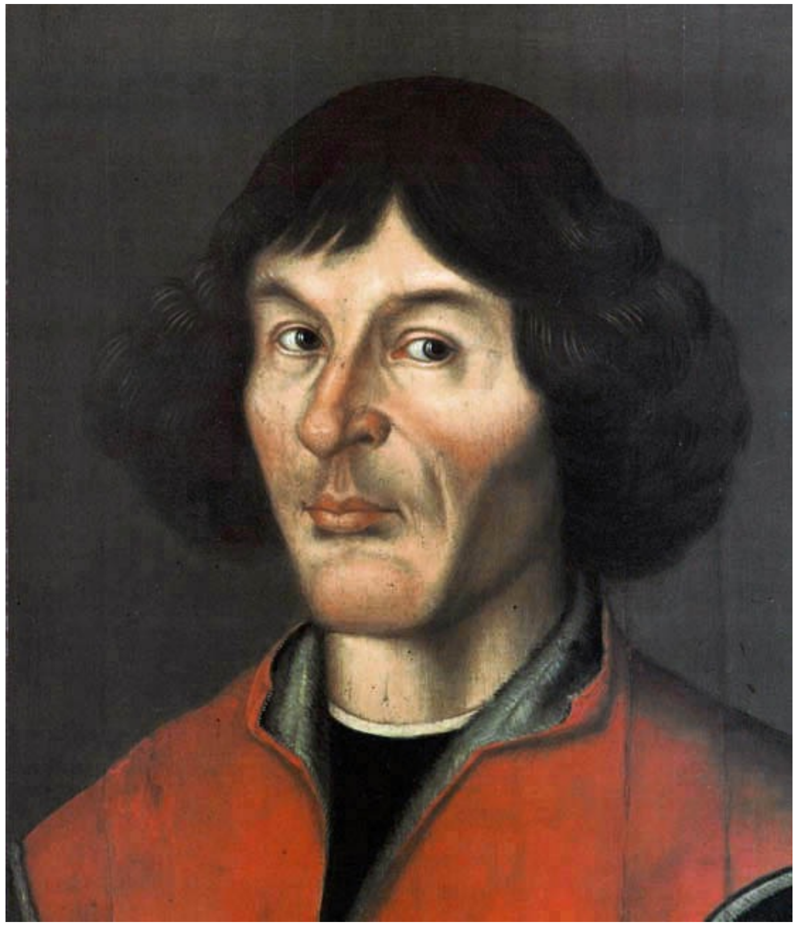
2017 Gartner Hype Cycle



2017 Gartner Hype Cycle



Copernican Revolution



Nicolaus Copernicus
(1473-1543)

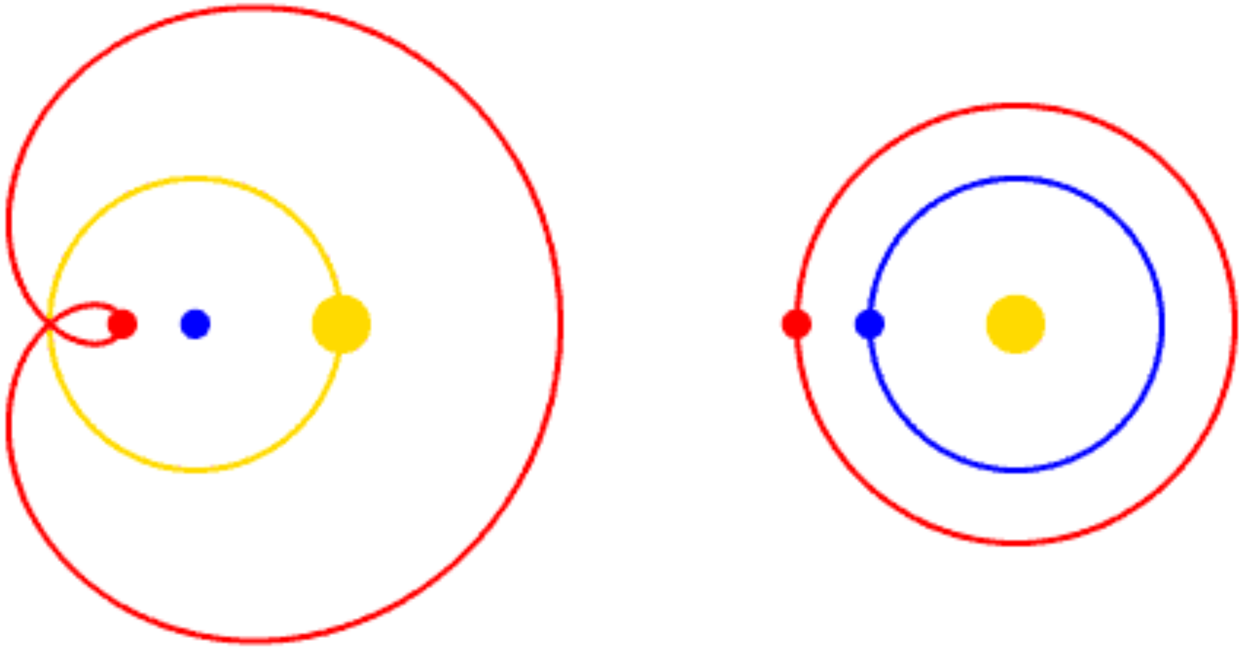


Figure credit: wikipedia

Theory-led Models

- Newton's Laws of Motion

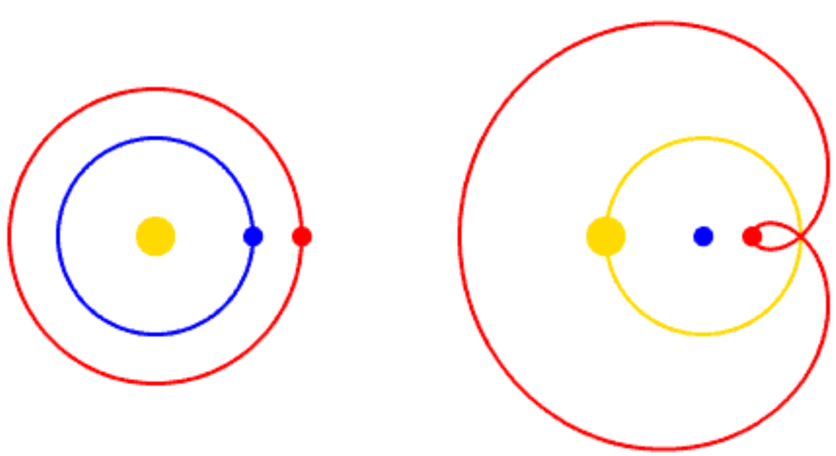
$$F = 0 \Leftrightarrow \frac{dv}{dt} = 0$$

$$F = ma$$

$$F_1 = -F_2$$

- Newton's Law of Universal Gravitation

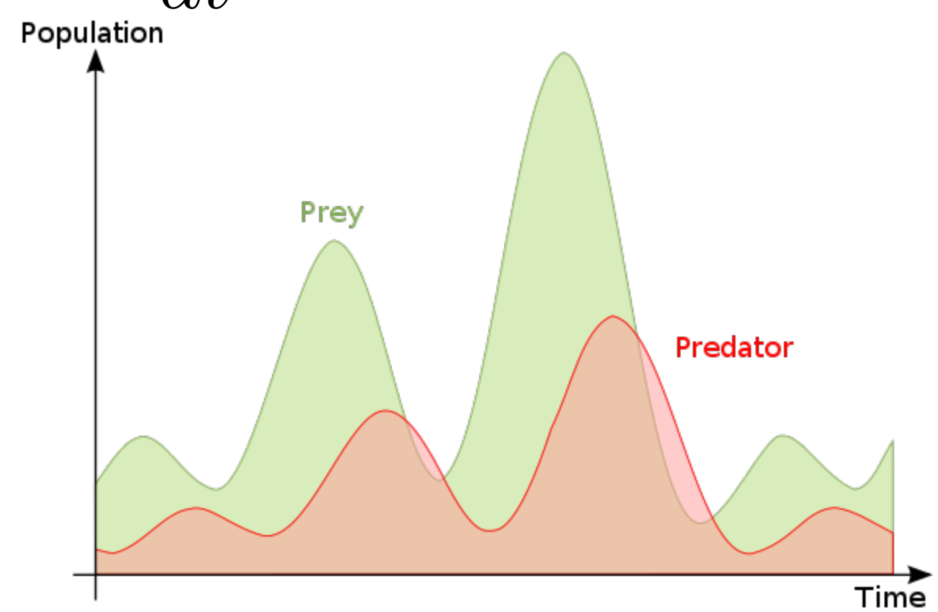
$$F = G\frac{m_1 m_2}{r^2}$$



- Lotka-Volterra equations

$$\frac{dx}{dt} = \alpha x - \beta xy$$

$$\frac{dy}{dt} = \delta xy - \gamma y$$

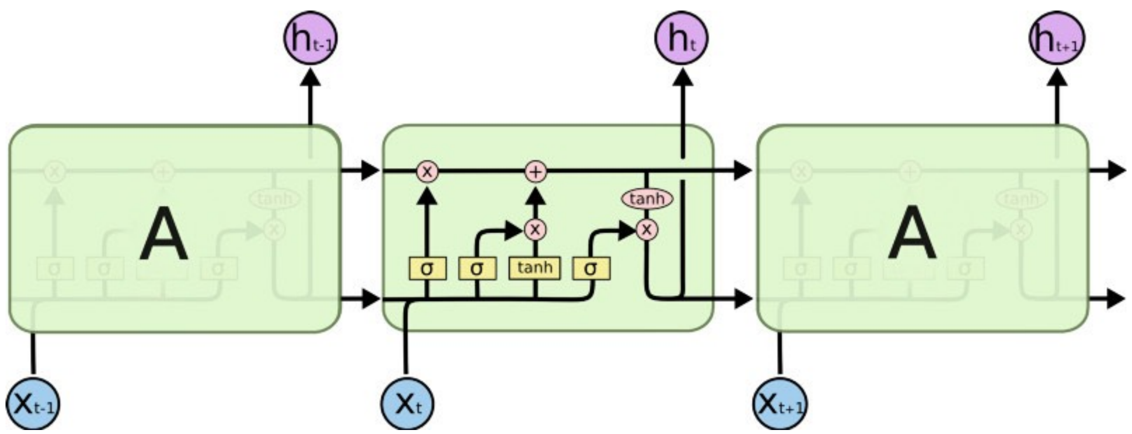


- Spatial residential-retail model

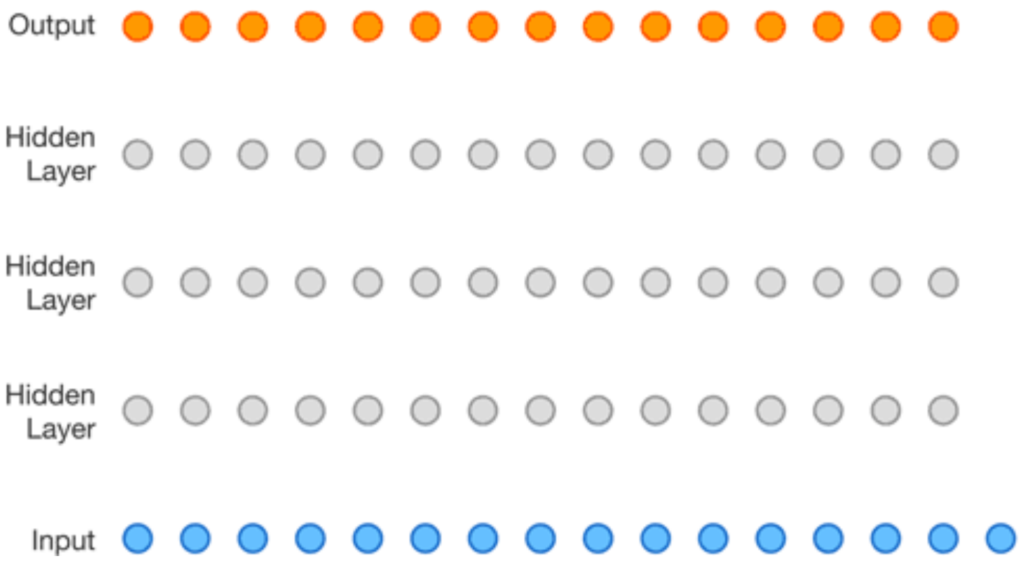
$$S_{ij} = \frac{I_i P_i W_j^{\alpha} e^{-\beta m_j c_{ij}}}{\sum_k W_k^{\alpha} e^{-\beta m_j c_{ij}}}$$

Figure credit: wikipedia

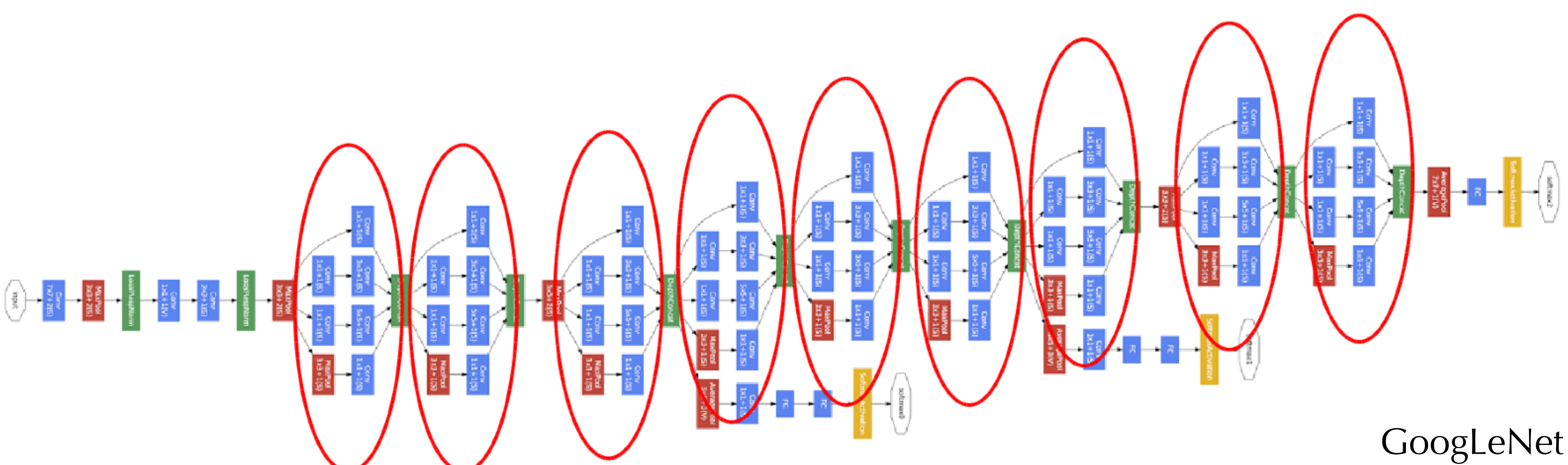
Data-led Models



LSTM
[Hochreiter & Schmidhuber 1997]

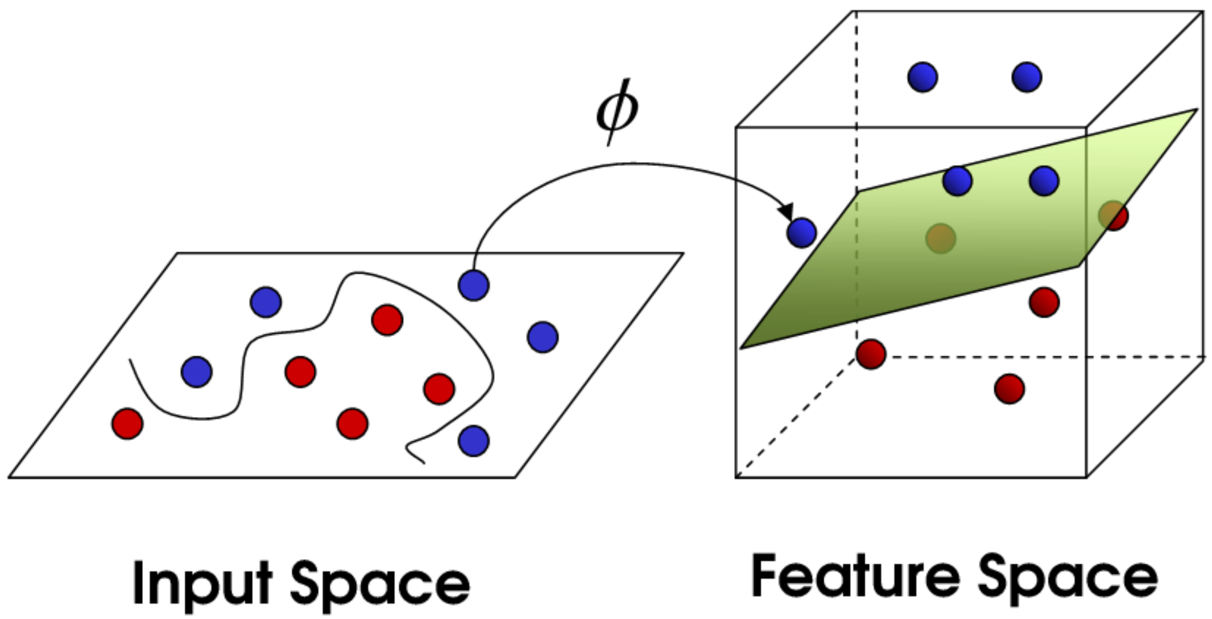


WaveNet
[van den Oord et al 2017]

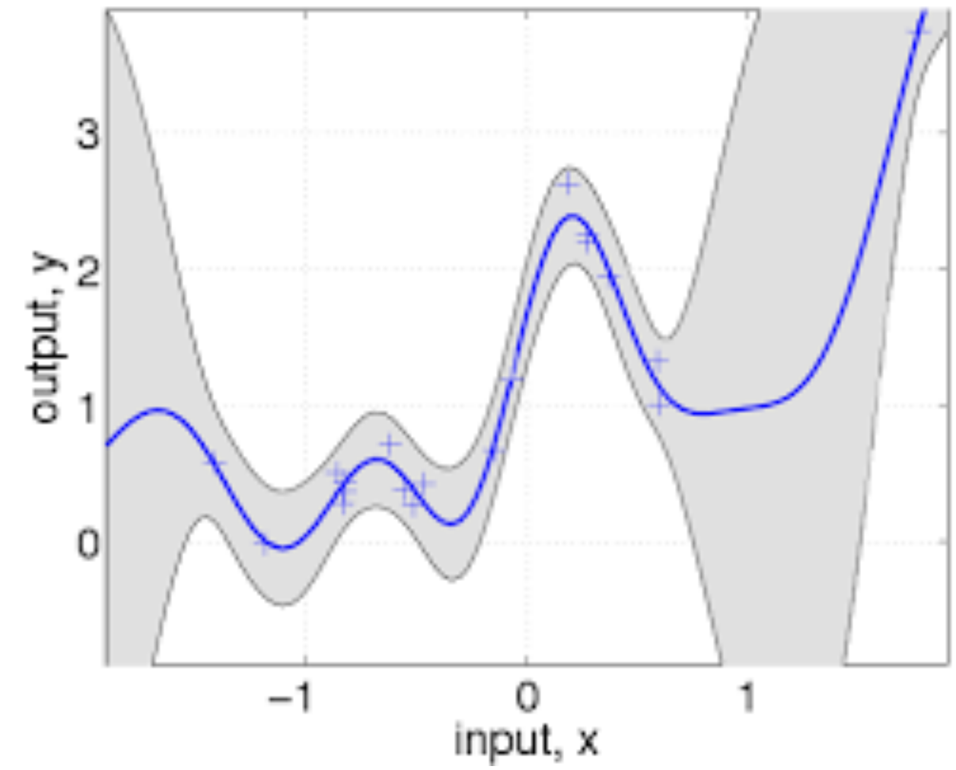


GoogLeNet
[Szegedy et al 2015]

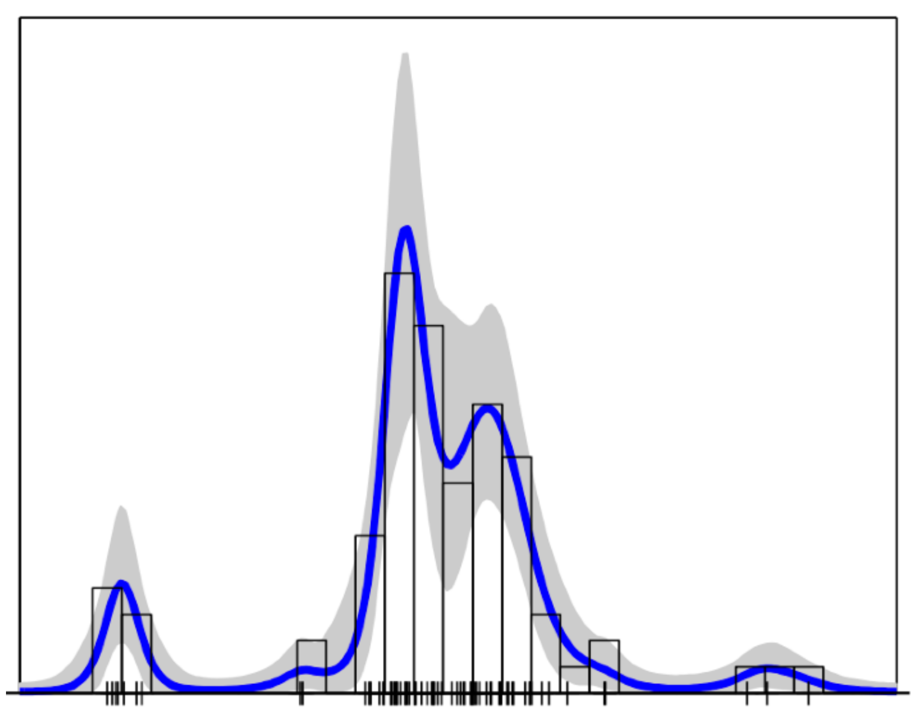
Ever Increasing Flexibility



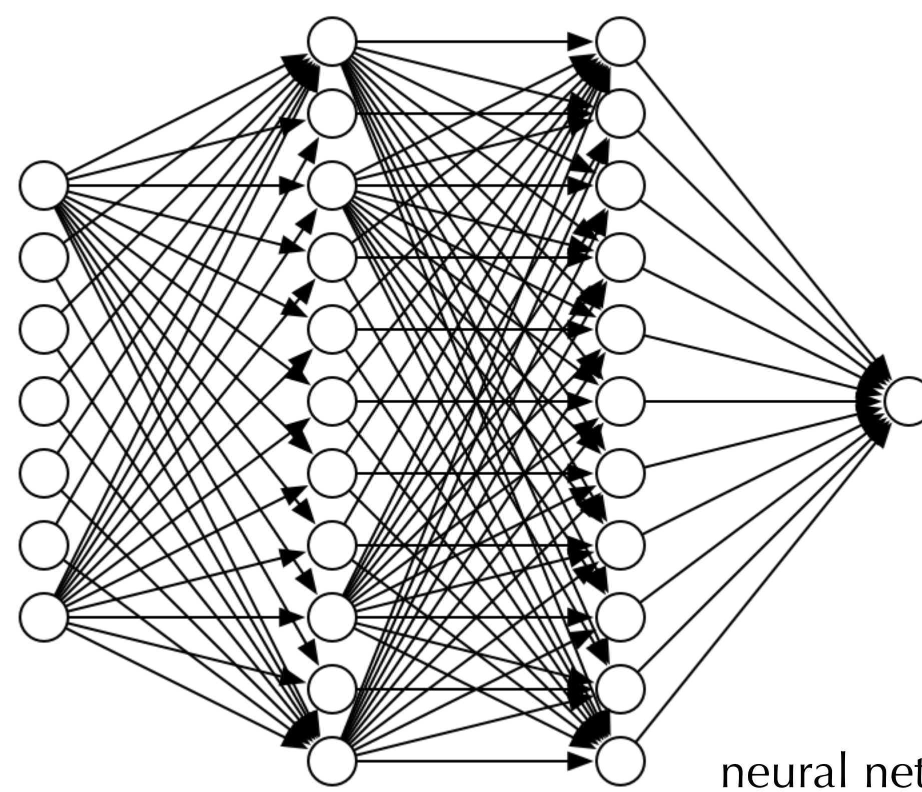
kernel trick [StackOverflow]



[gaussianprocess.org]

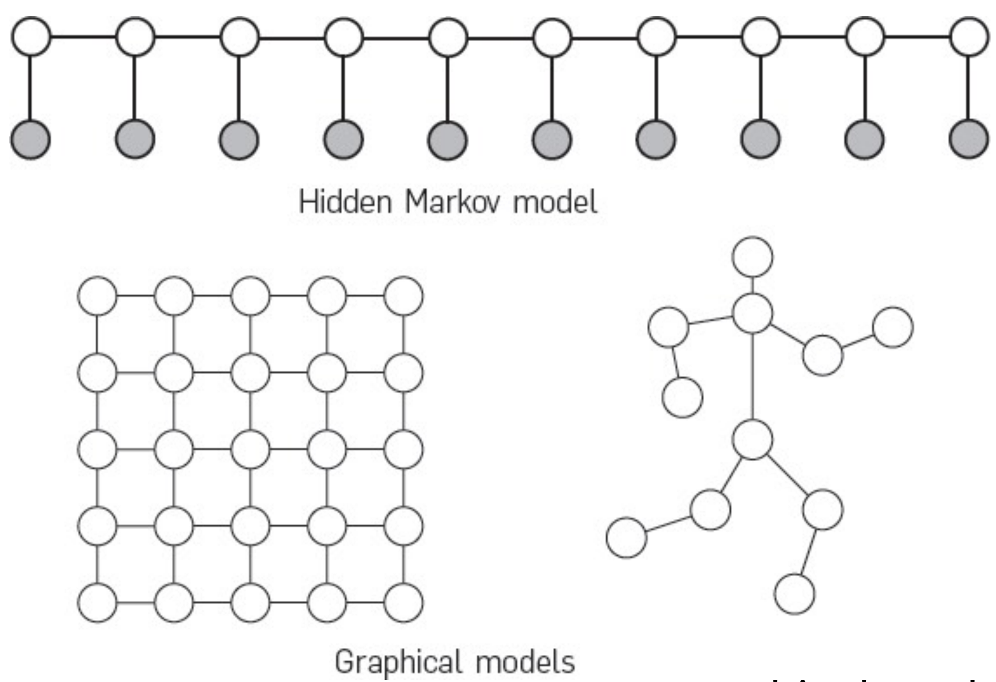


Bayesian nonparametrics



neural networks

Ever Increasing Complexity



graphical models
[Nonparametric BP Sudderth et al 2010]

```c
#include "probabilistic.h"

int main(int argc, char **argv) {

    double var = 2;
    double mu = normal_rng(1, 5);

    observe(normal_lnp(9, mu, var));
    observe(normal_lnp(8, mu, var));

    predictf("mu %f\n", mu);

    return 0;
}
```

probabilistic C
[Paige & Wood 2014]

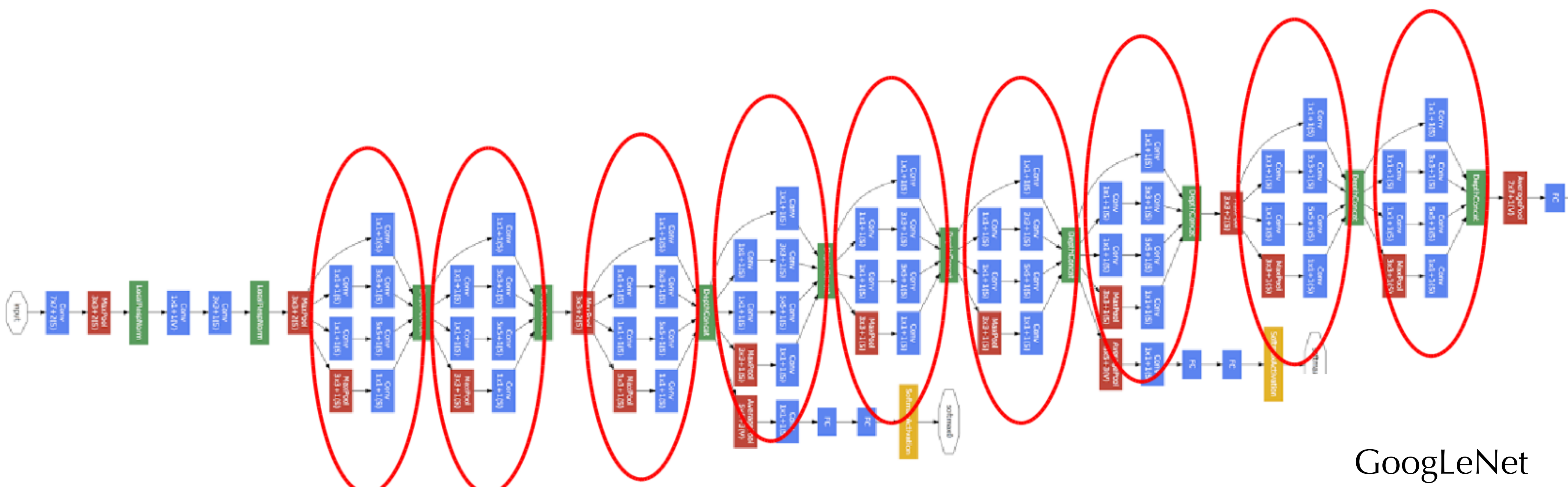
LibBi

Venture

theano

mxnet

dynet



GoogLeNet
[Szegedy et al 2015]

On Bayesian Learning and Deep Learning

Graphical models

NPBayes

GPs

BayesOpt

Variational inference

Monte Carlo

Thomas Bayes

Bayesian NNs

Deep generative models

VAEs

GANs

Autoregressive models

Geoffrey Hinton

Neural nets

ConvNets

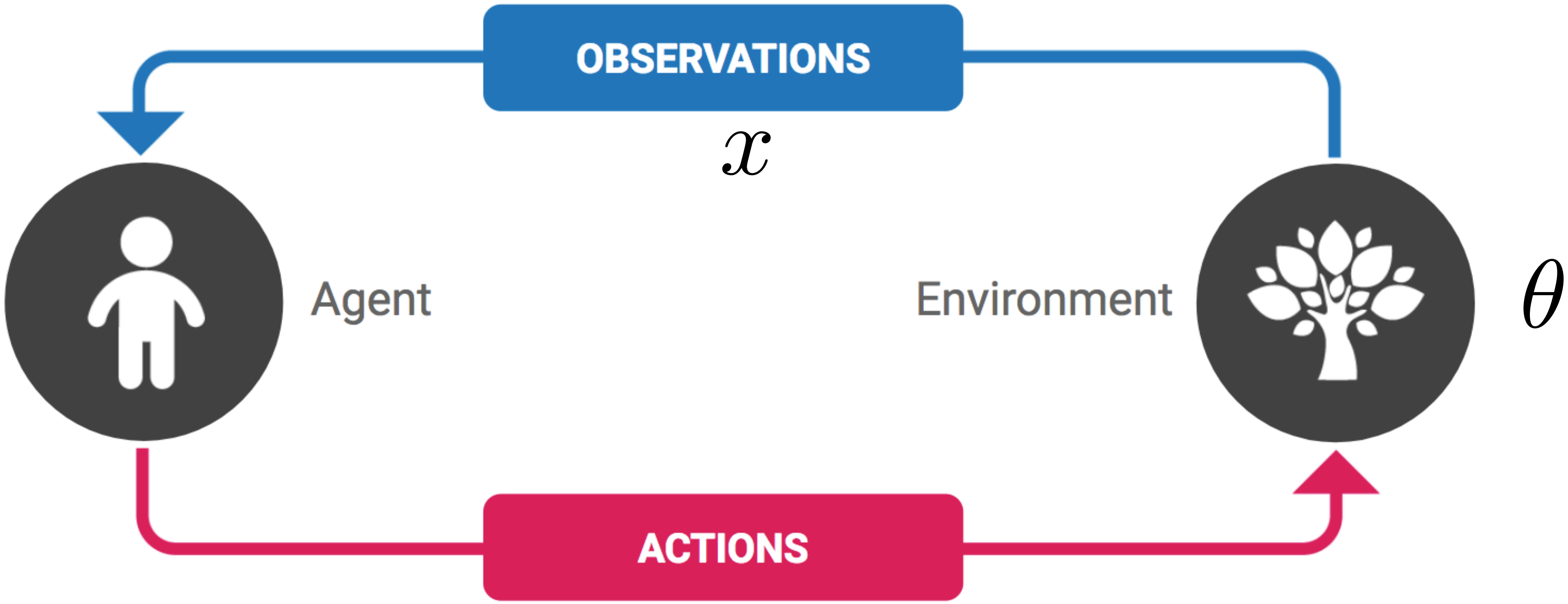
RNNs

Attention

SGD

Dropout

Bayesian Theory of Learning



Predict: $p(x_*|x) = \int p(x_*|\theta)p(\theta|x)d\theta$

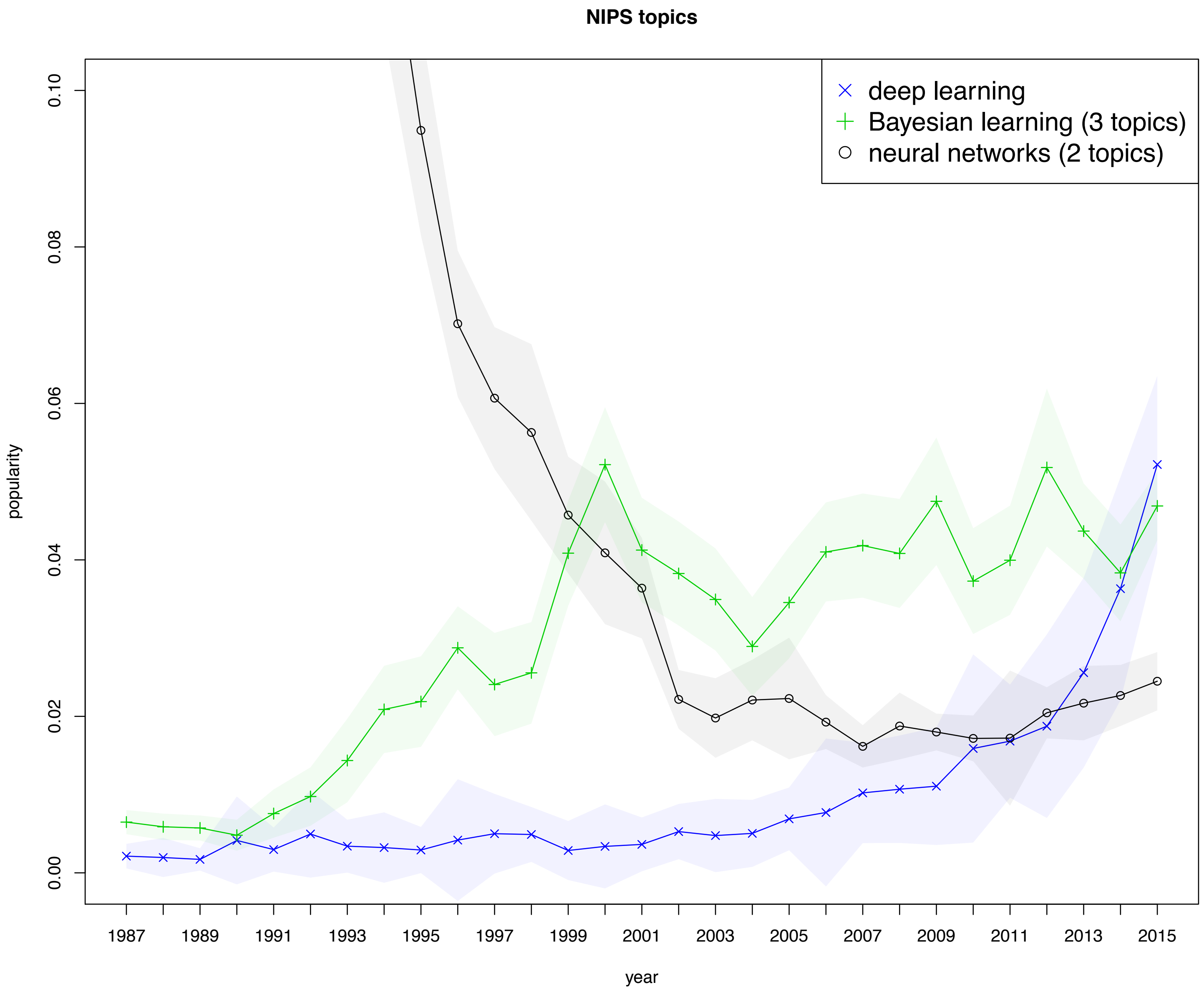
Act: $U(a|x) = \int U(a;\theta)p(\theta|x)d\theta$

Prior: $p(\theta)$

Likelihood: $p(x|\theta)$

Posterior: $p(\theta|x) = \dfrac{p(\theta)p(x|\theta)}{\sum_\theta p(x|\theta)p(\theta)}$

Bayesian and Deep Learning @ NIPS '87-'15



Top words in each topic:

deep learning

deep	rnn
layers	bengio
convolutional	train
mnist	hinton
sgd	unsupervised
rbm	boltzmann

neural networks

architecture	propagation
recurrent	feedforward
back	feedback
activation	backpropagation
outputs	hinton
forward	connectionist

net	simulation
connections	development
connection	topology
nets	represented
connected	structures
parallel	role

Bayesian learning

generative	bayes
priors	poisson
missing	inferred
gamma	jordan
dirichlet	counts
dependencies	multinomial

belief	messages
graphical	passing
propagation	assignment
marginal	potentials
message	pairwise
marginals	mrf

chain	predictive
gibbs	importance
carlo	hyperparameters
monte	particle
mcmc	stationary
sampler	proposal

Analysis from a Bayesian nonparametric dynamic topic model
Poisson random fields for dynamic feature models [Perrone et al 2016]

Bayesian Learning

- Strengths:
  - A normative account of “best” learning given model and data.
  - Explicit expression of all prior knowledge/inductive biases in model.
  - Unified treatment of uncertainties.
  - Common language with statistics, applied sciences.
- Rigidity issue:
  - Learning can be wrong if model is wrong.
  - Not all prior knowledge can be encoded as joint distributions.
  - Simple analytic forms for conditional distributions.
- Scalability issue:
  - approximations to intractable posterior $p(\theta|x)$.
  - no single posterior computation method that works well across all Bayesian models.

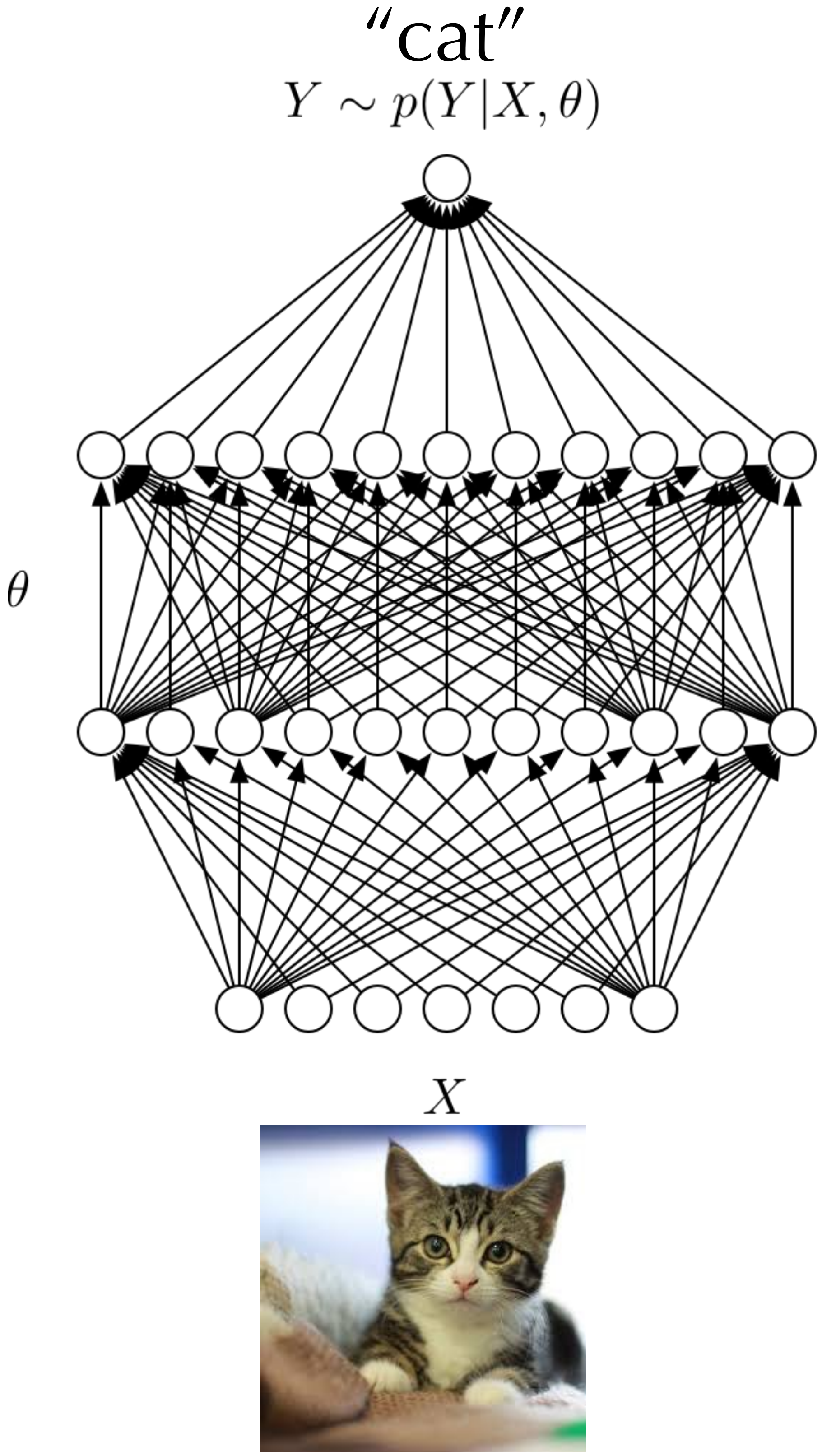
Talk Outline

The Posterior Server

The Concrete VAEs

FIVO: Filtered variational objectives

Bayesian Deep Learning



- Neural network as nonlinear function approximator:

$$p(y|x,\theta) = f(y,x,\theta)$$

- Weight decay is a Gaussian prior:

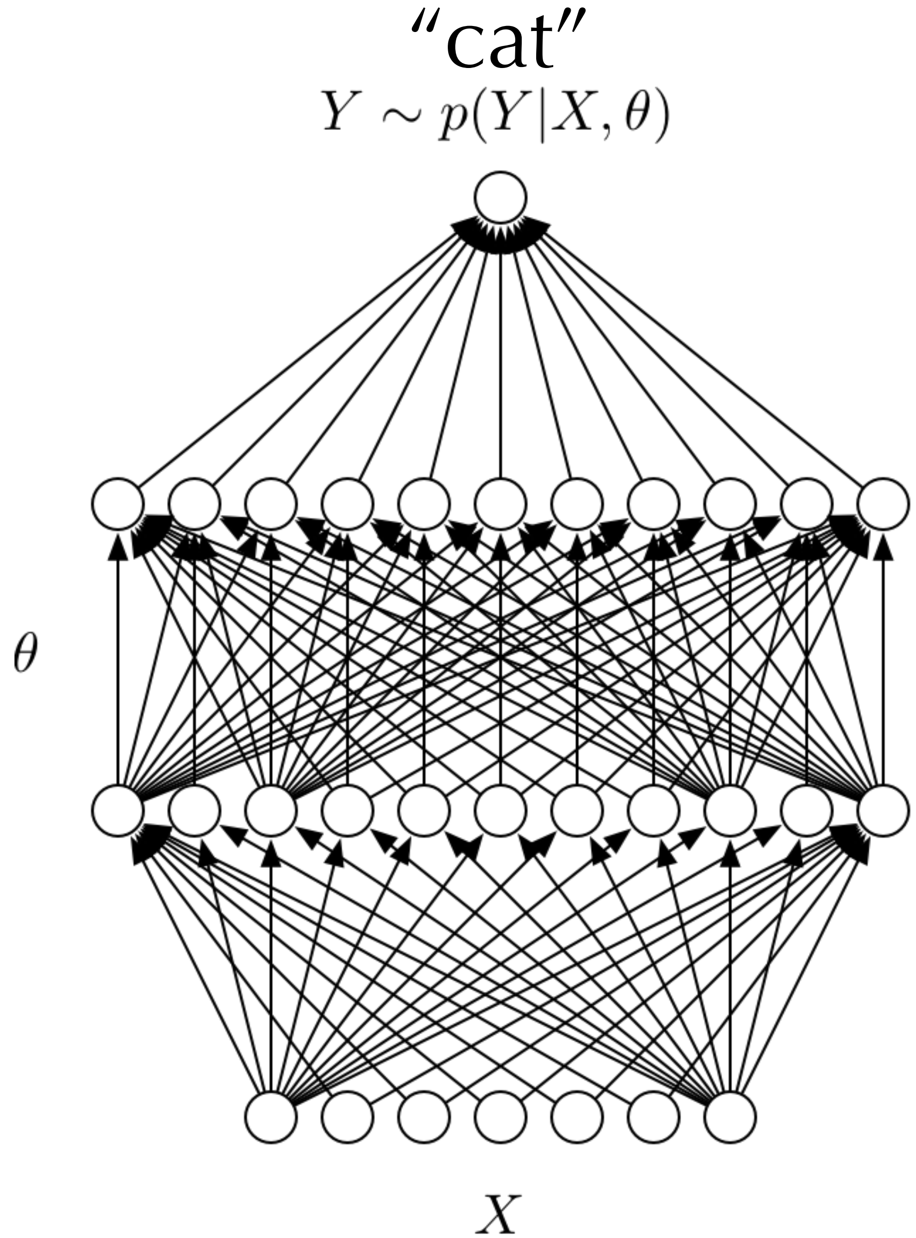
$$p(\theta) = \mathcal{N}(\theta; 0, \Lambda)$$

- Posterior distribution:

$$p(\theta|\text{data}) = \frac{p(\theta)\prod_{i=1}^n p(y_i|x_i,\theta)}{\int p(\theta)\prod_{i=1}^n p(y_i|x_i,\theta)d\theta}$$

- regularised ML estimator is a posterior mode.

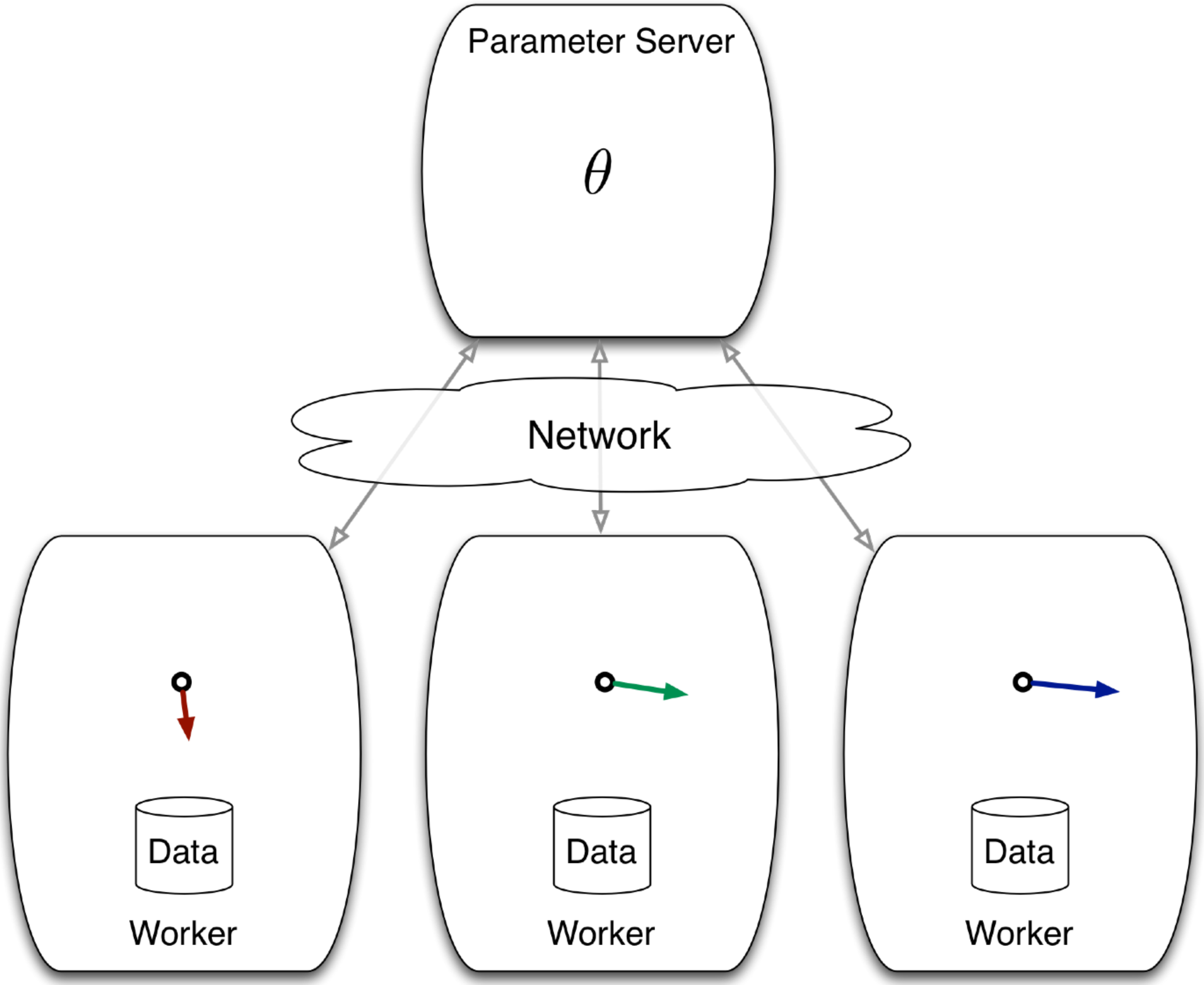
Bayesian Deep Learning



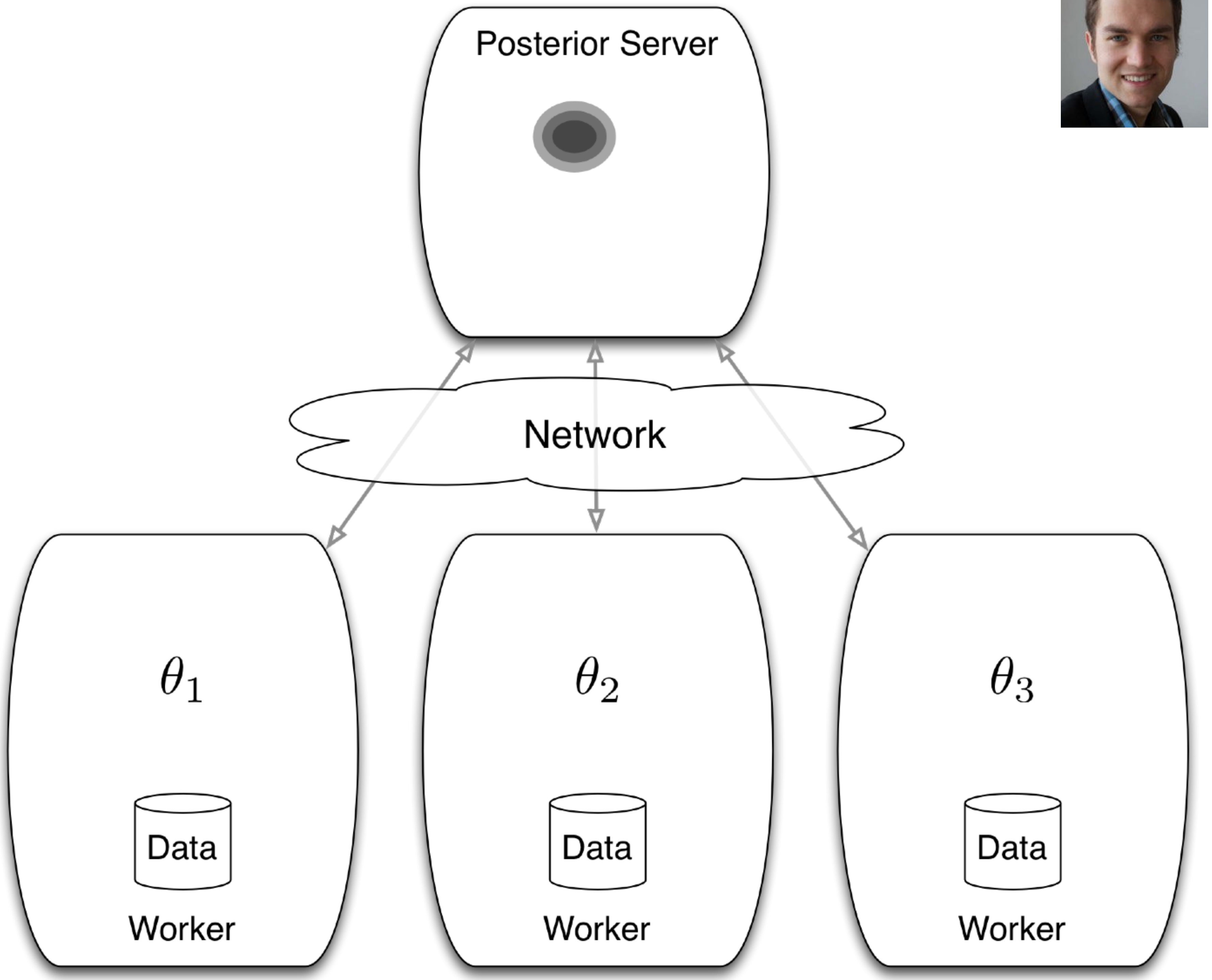
$$p(\theta|\text{data}) = \frac{p(\theta)\prod_{i=1}^{n} p(y_i|x_i,\theta)}{\int p(\theta)\prod_{i=1}^{n} p(y_i|x_i,\theta)d\theta}$$

- Markov chain Monte Carlo
  - Hamiltonian Monte Carlo [Neal 1994]
  - stochastic gradient Langevin dynamics (SGLD) and variants [Welling and Teh 2011, Ma et al 2015, Perrone et al 2017]
- Variational inference
  - Mean field [Hinton & van Camp 1993, Blundell et al 2015]
  - EP [Hernandez-Lobato & Adams 2015, Li et al 2015]

Distributed Learning

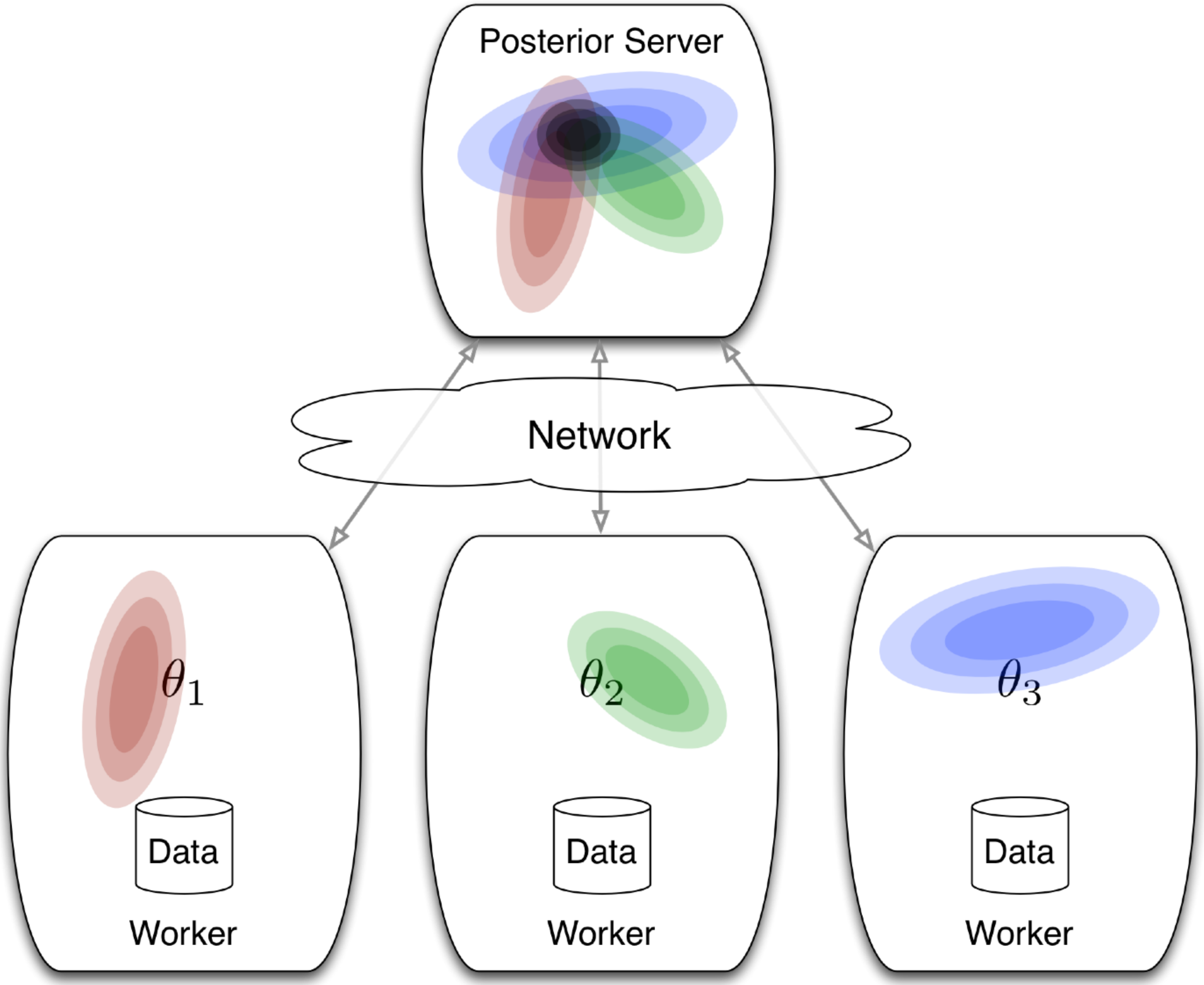


Distributed Bayesian Learning



[Hasenclever et al, JMLR 2017]

Distributed Bayesian Learning



[Hasenclever et al, JMLR 2017]

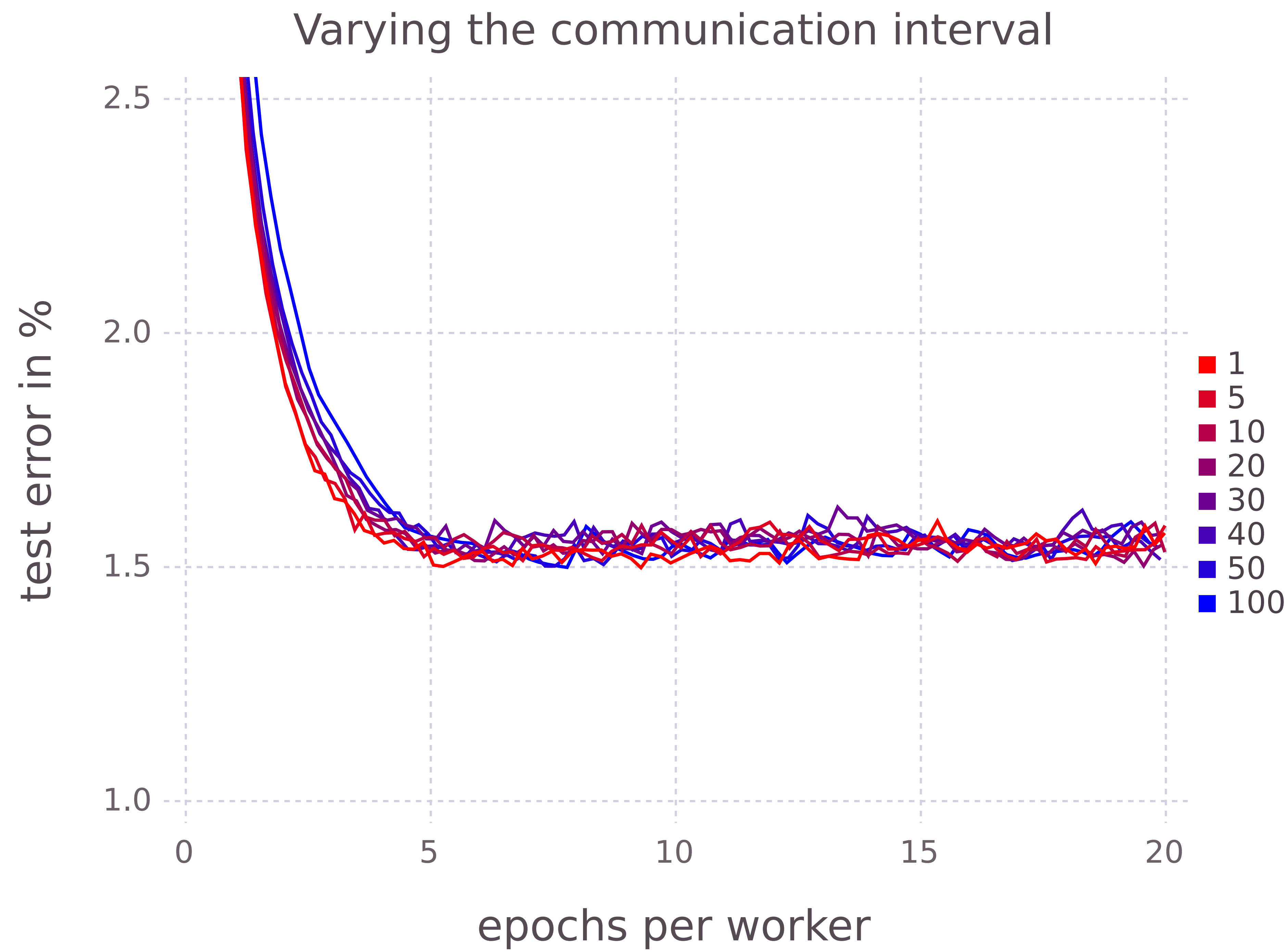
Distributed Bayesian Learning

- Each worker forms a Gaussian approximation to its likelihood using Expectation Propagation (EP) [Minka 2001].
  - Actually: alternative to EP called stochastic natural-gradient EP (SNEP) which can be guaranteed to converge.

- Required statistics are estimated using Markov chain Monte Carlo
  - We use stochastic gradient Langevin dynamics (SGLD) [Welling & Teh 2011]:
$$\theta^{(t+1)} = \theta^{(t)} + \epsilon_t \widehat{\nabla} \log p(\theta^{(t)}, \text{data}) + \sqrt{2\epsilon_t}\eta_t, \quad \eta_t \sim \mathcal{N}(0, I)$$
  - See also [Chen et al 2014, Ding et al 2014, Ma et al 2015, Leimkuhler & Shang 2015, Perrone et al 2017] for extensions, [Teh et al 2016, Vollmer et al 2016, Zhang et al 2017, Raginsky et al 2017] for theory.

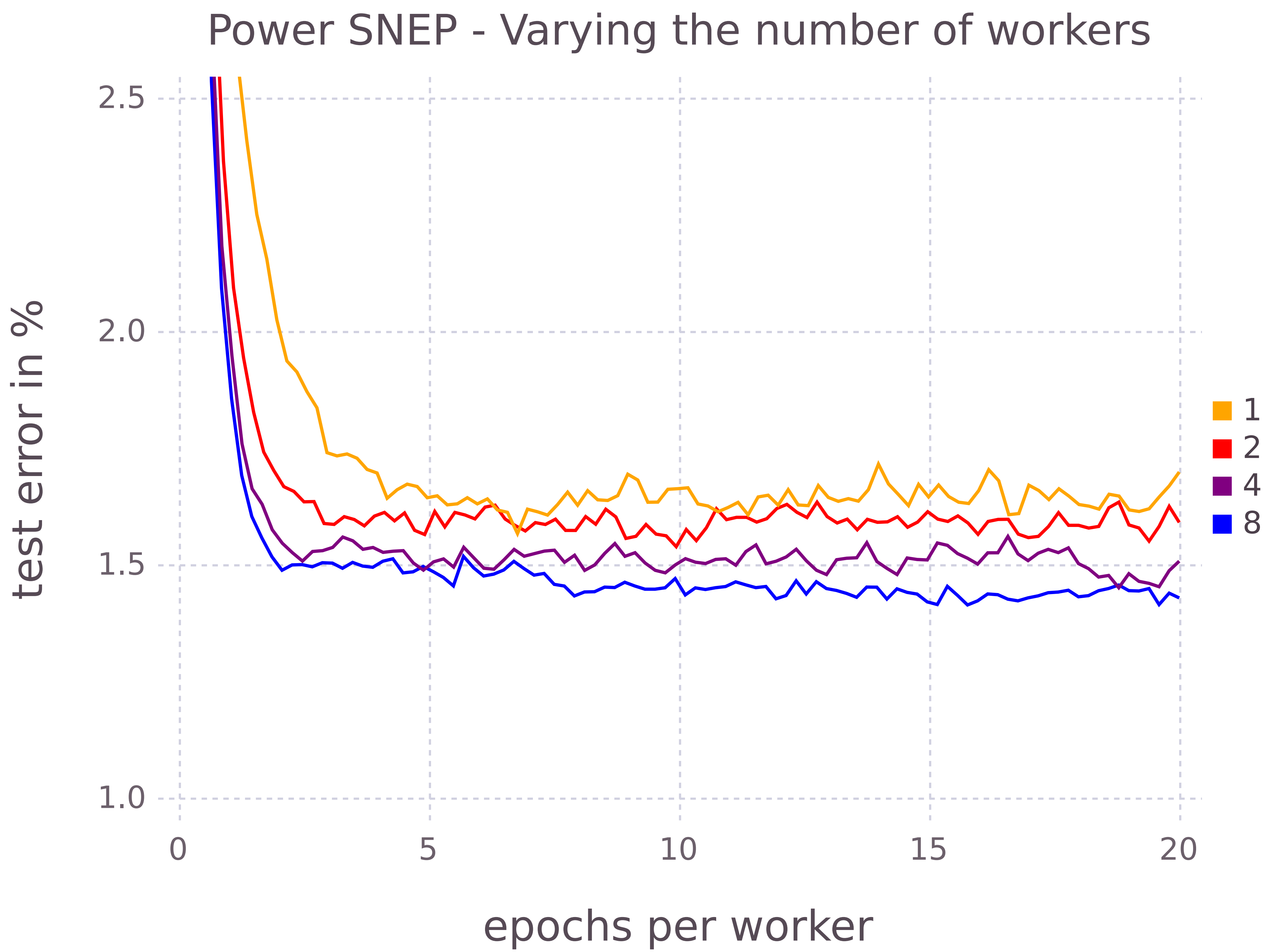
[Hasenclever et al, JMLR 2017]

MNIST 500x300



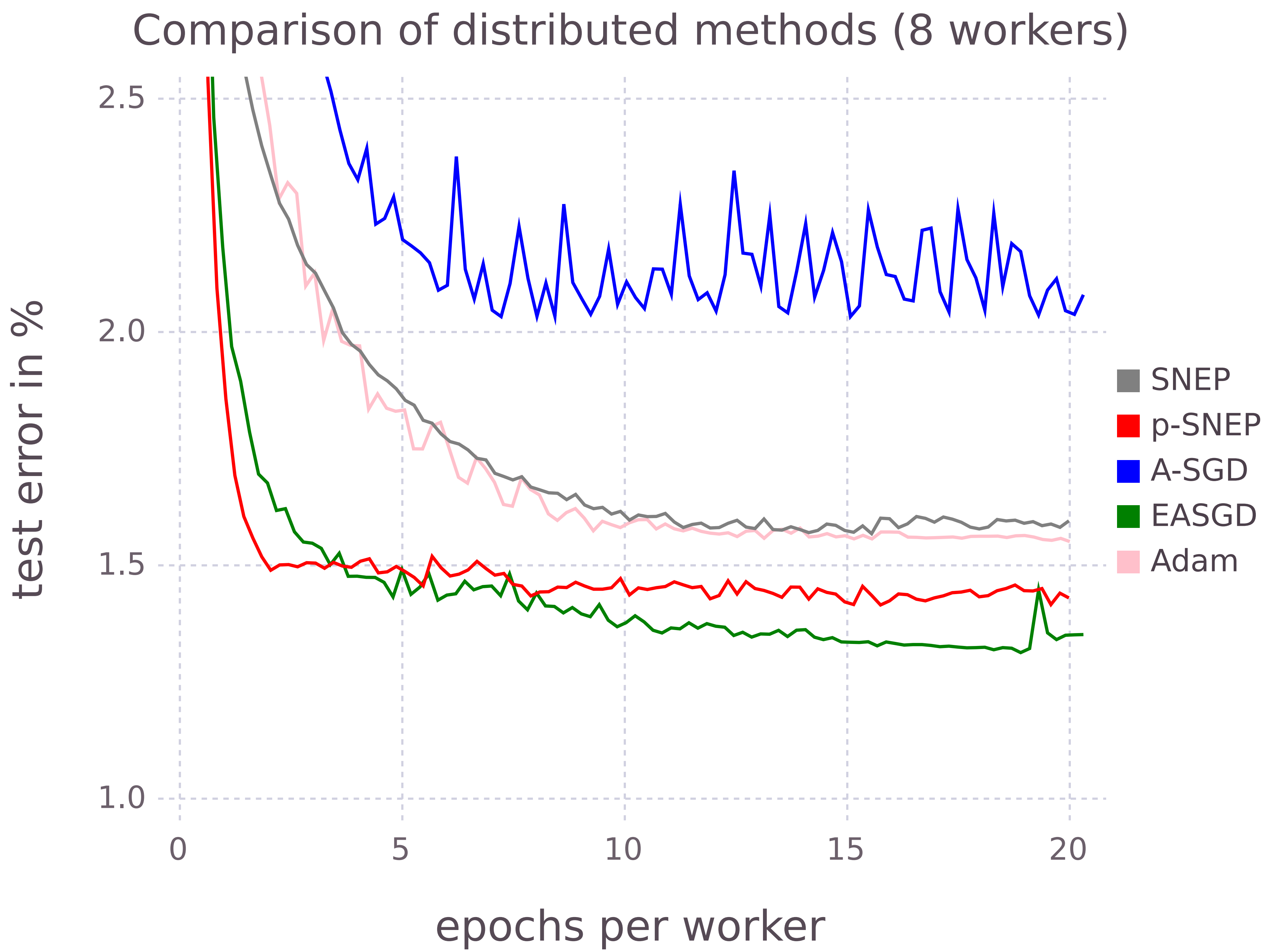
[Hasenclever et al, JMLR 2017]

MNIST 500x300



[Hasenclever et al, JMLR 2017]

MNIST 500x300

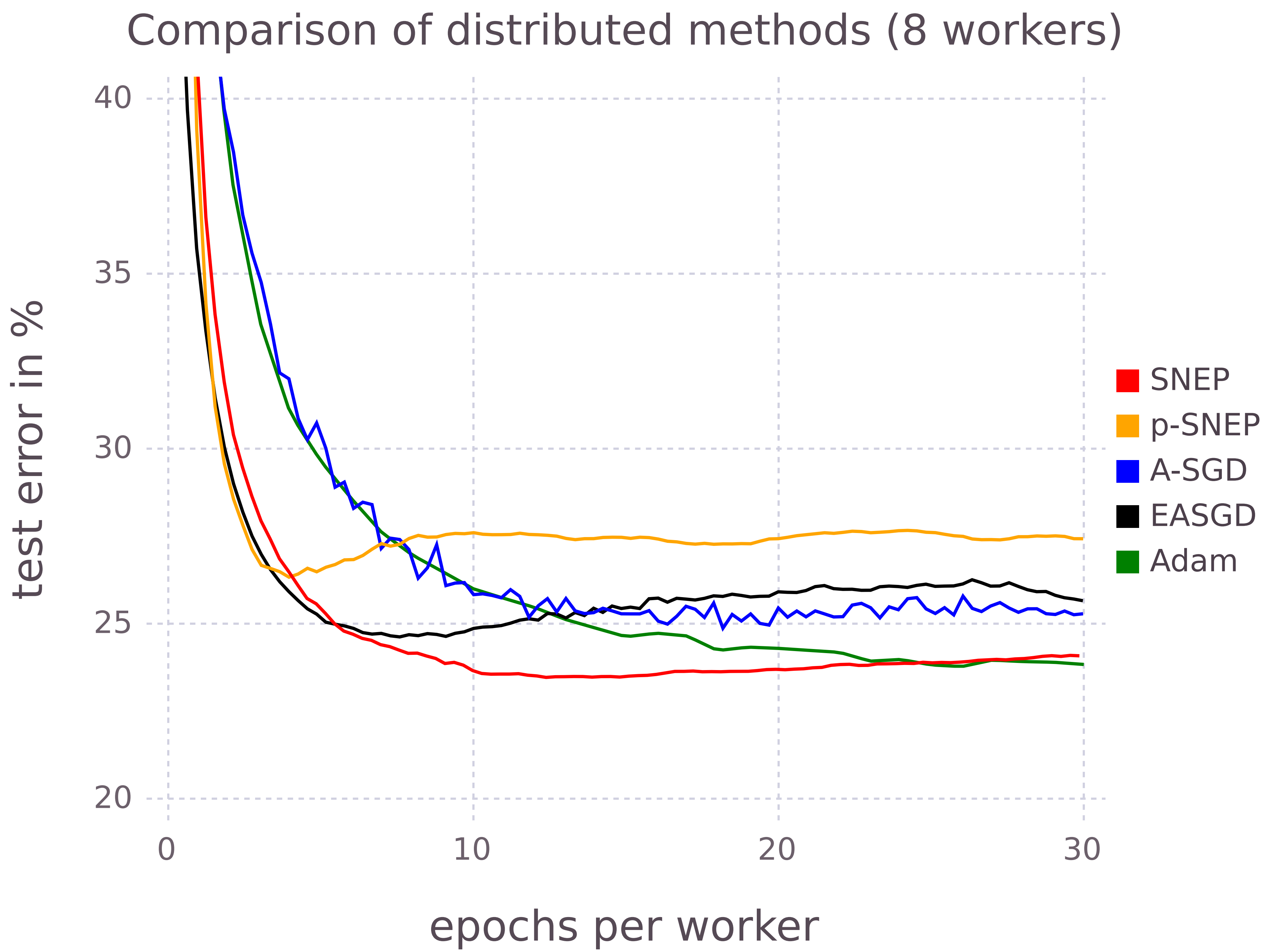


[Hasenclever et al, JMLR 2017]

MNIST 20 layer MLP

[Hasenclever et al, JMLR 2017]

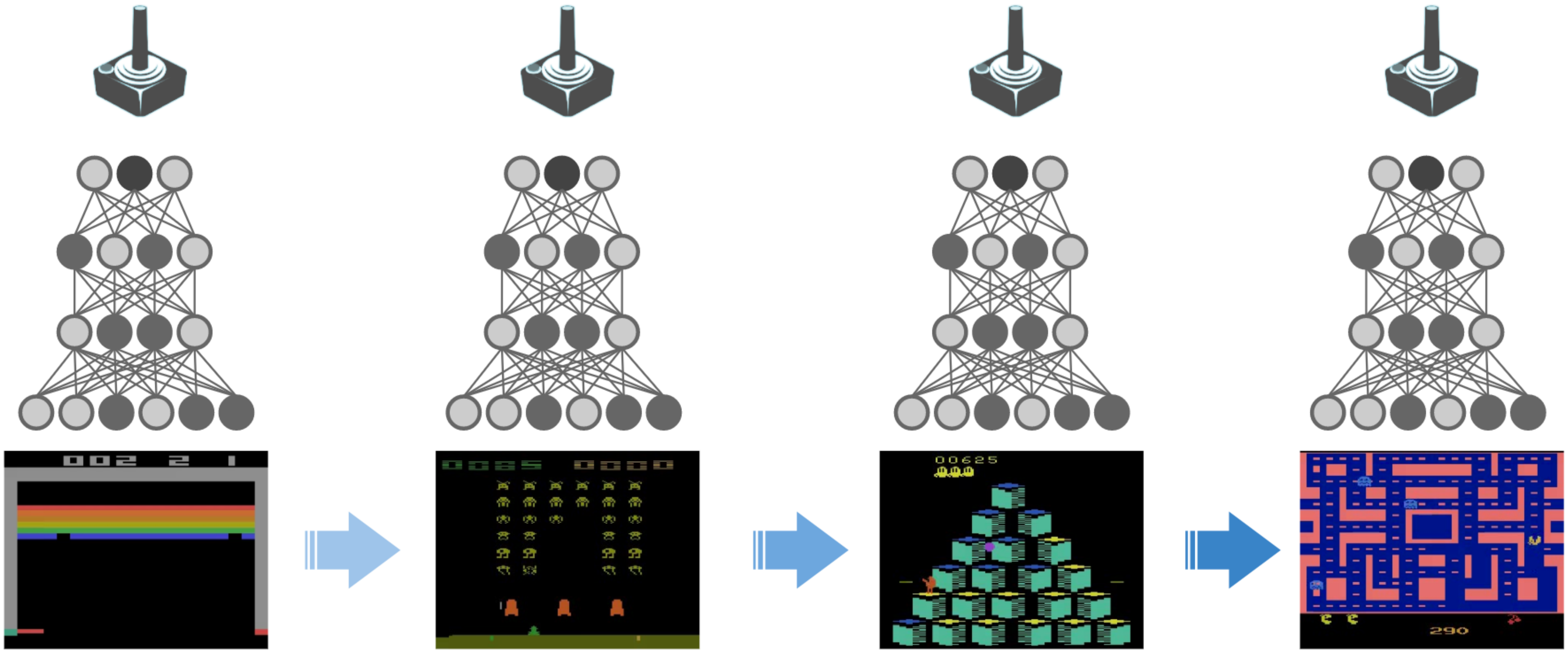
CIFAR10 ConvNet



[Hasenclever et al, JMLR 2017]

Towards AGI: Multitask and Continual Learning

- [Kirkpatrick et al, PNAS 2017]



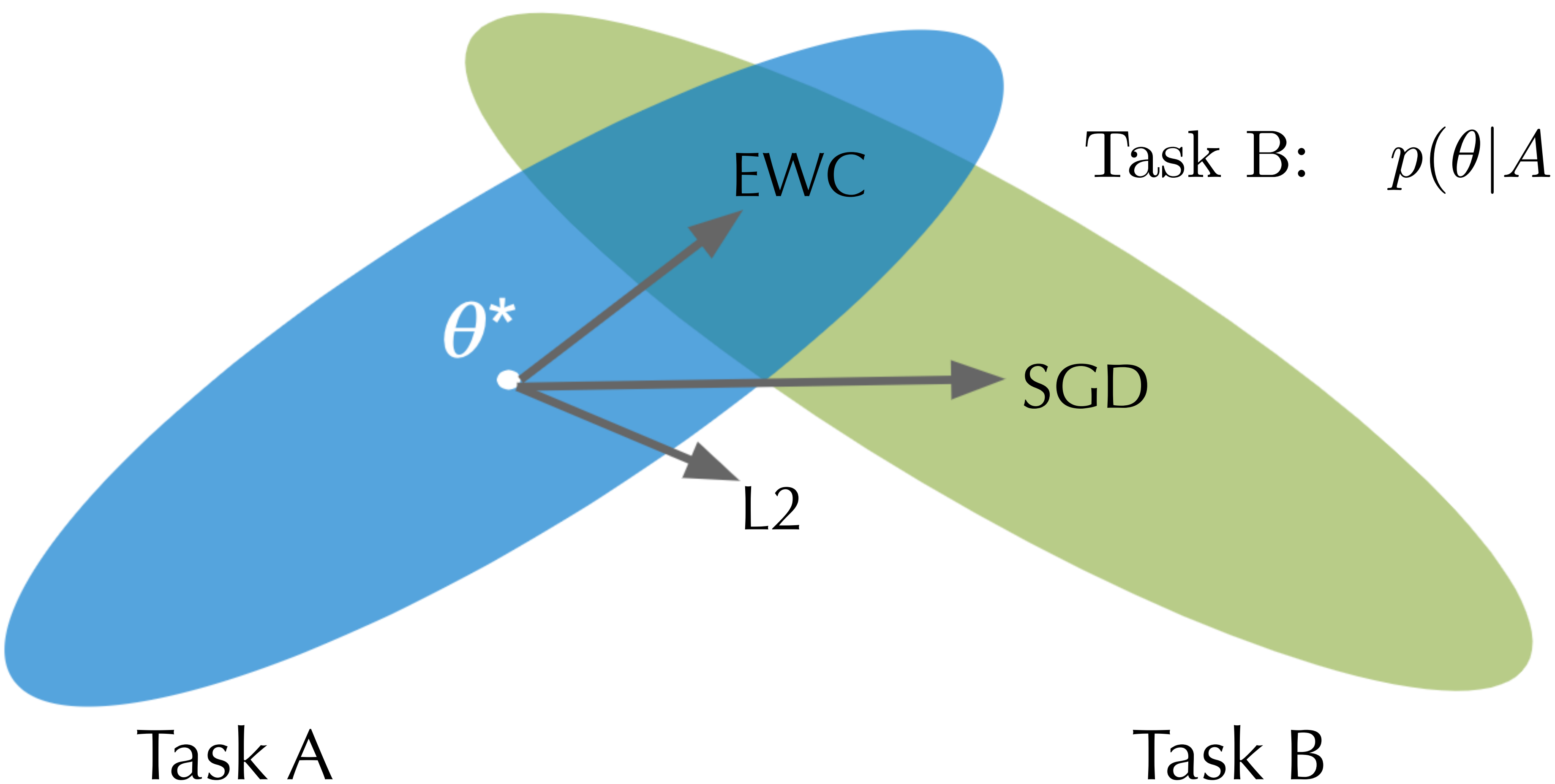
Elastic Weight Consolidation

- [Kirkpatrick et al, PNAS 2017]
- Continual learning by approximate Bayes.

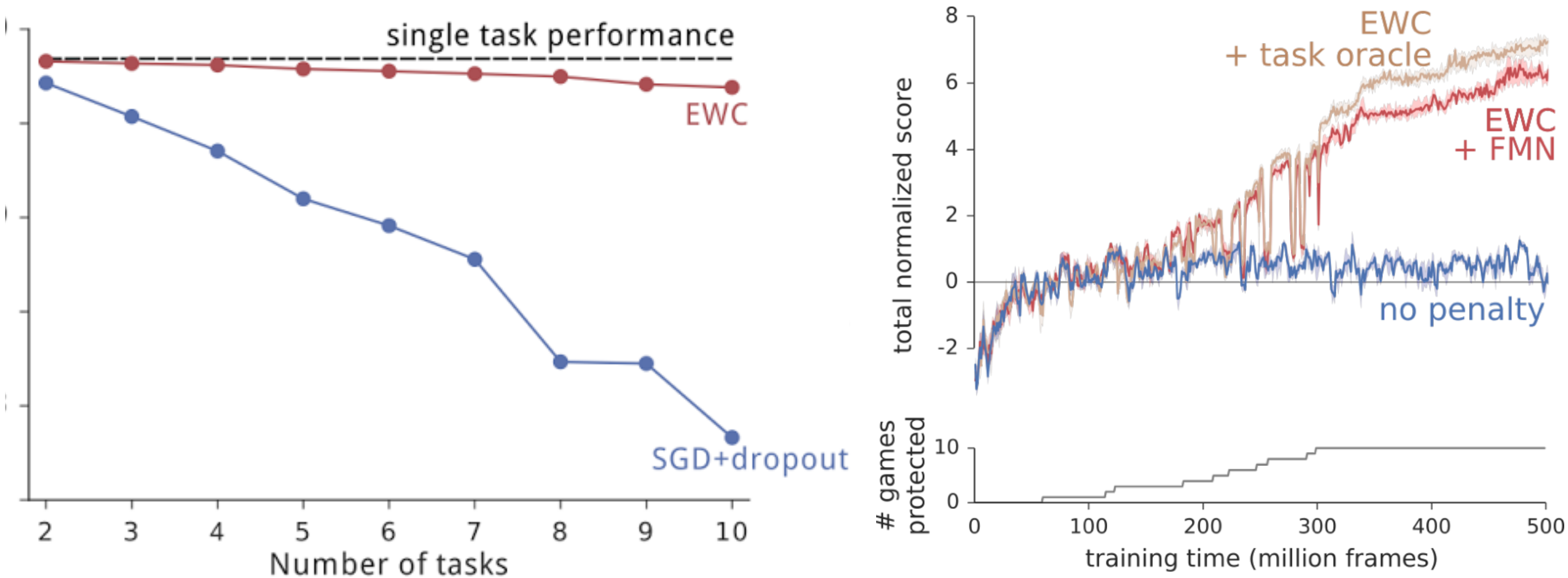
Prior: $p(\theta)$

Task A: $p(\theta|A) \propto p(\theta)p(A|\theta)$

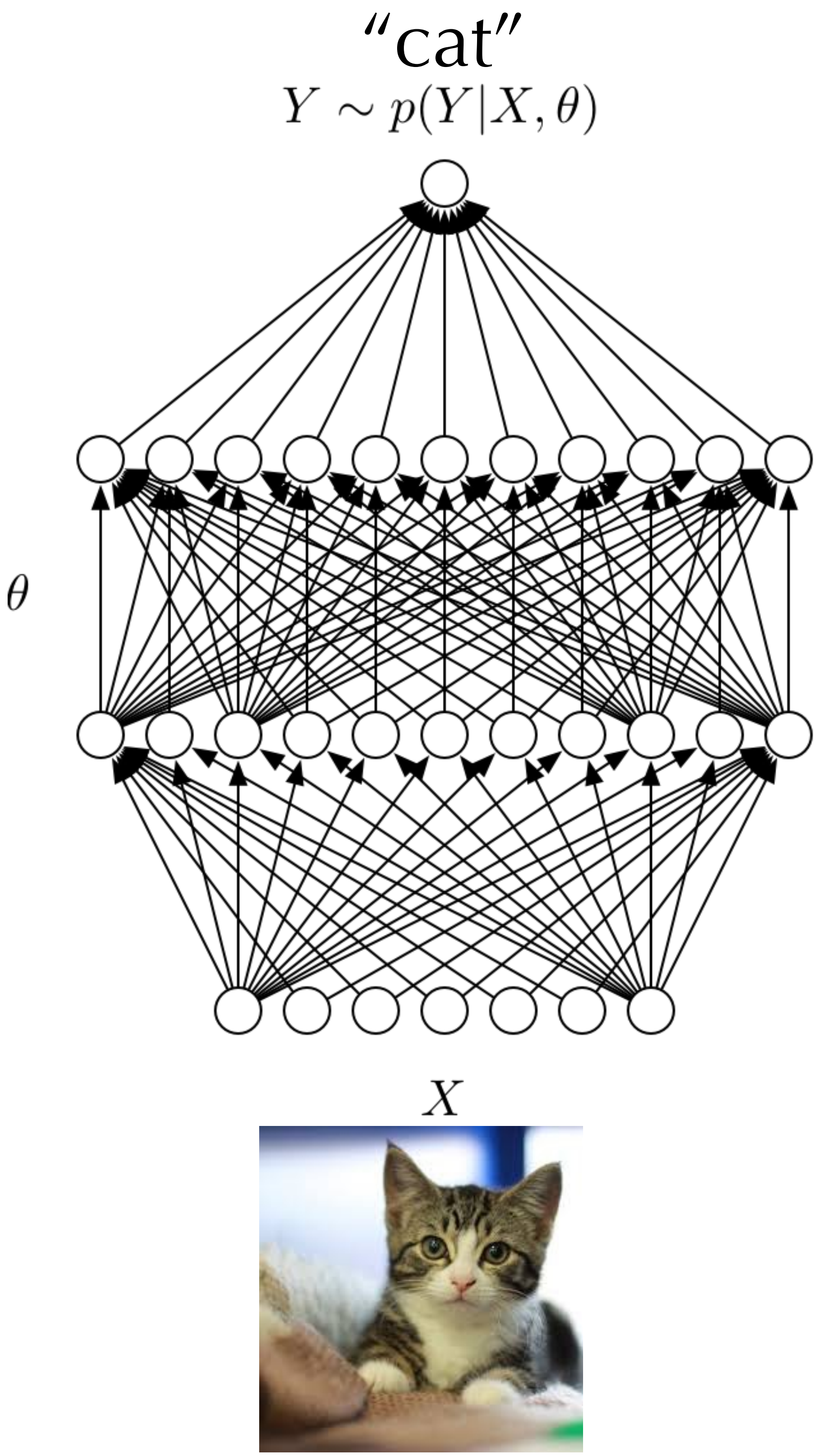
Task B: $$\begin{aligned} p(\theta|A,B) &\propto p(\theta)p(A|\theta)p(B|\theta) \\ &\propto p(\theta|A)p(B|\theta) \\ &\approx \widetilde{p}(\theta|A)p(B|\theta) \end{aligned}$$



Experimental Results



A Side Note on Parameters and Functions



$$p(\theta|\text{data}) = \frac{p(\theta)\prod_{i=1}^{n} p(y_i|x_i,\theta)}{\int p(\theta)\prod_{i=1}^{n} p(y_i|x_i,\theta)d\theta}$$

- Parameters in neural networks don't have meaning, are non-identifiable.
- It might be better to think in the space of functions instead.
  - See [Neal 1994, Rasmussen & Williams 2006, Damianou & Lawrence 2013, Pereyra et al 2017, Zhang et al 2017, Teh et al 2017]

Talk Outline

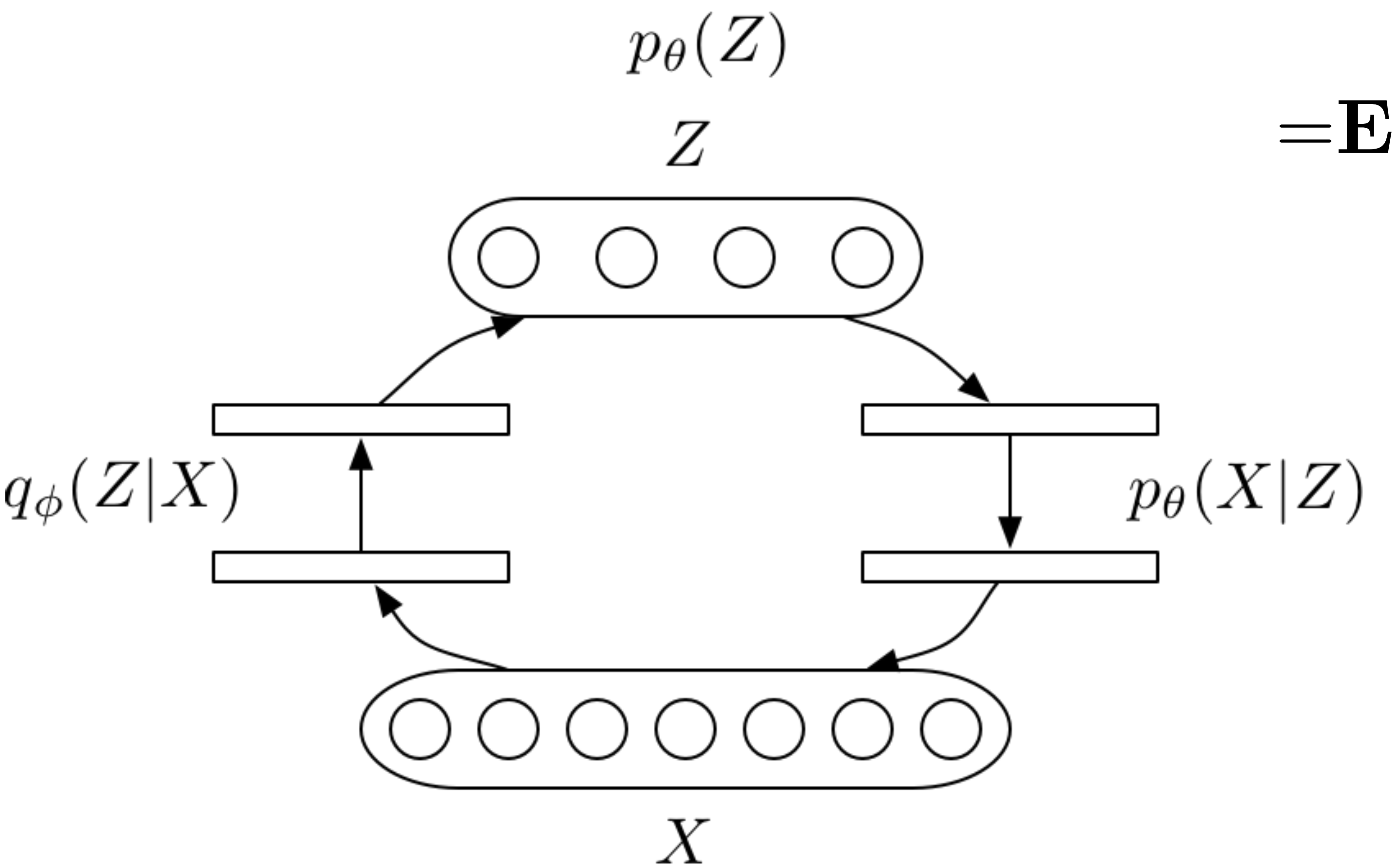
The Posterior Server

The Concrete VAE

FIVO: Filtered variational objectives

Variational Auto-encoders

- VAEs [Kingma and Welling 2014], [Rezende et al 2014]

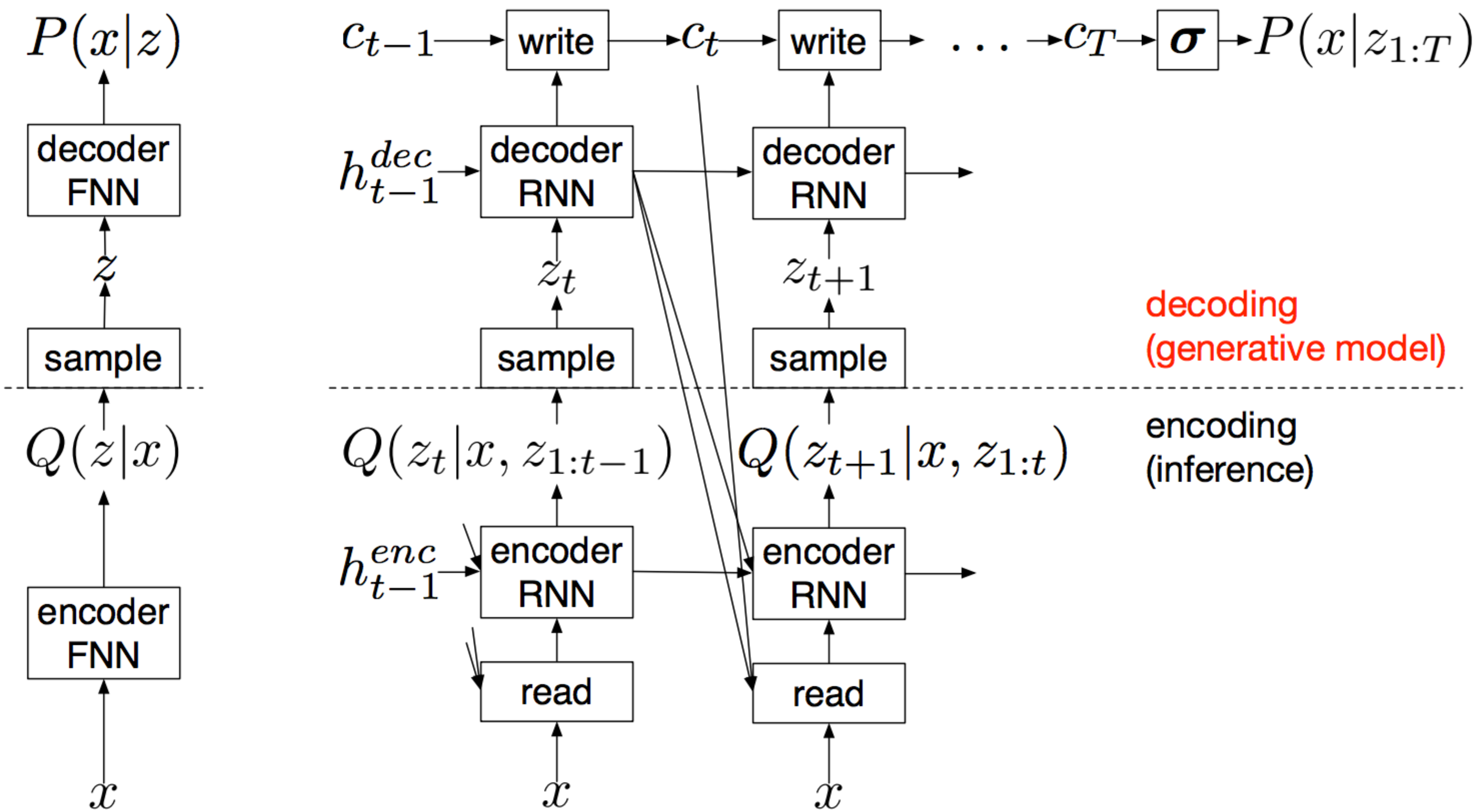


$$\log p_\theta(X) \geq \mathrm{ELBO}(\theta, \phi)$$

$$= \mathbf{E}_q \left[\log p_\theta(X|Z) + \log \frac{p_\theta(Z)}{q_\phi(Z|X)} \right]$$

DRAW: A RNN for Image Generation

- [Gregor et al 2015]

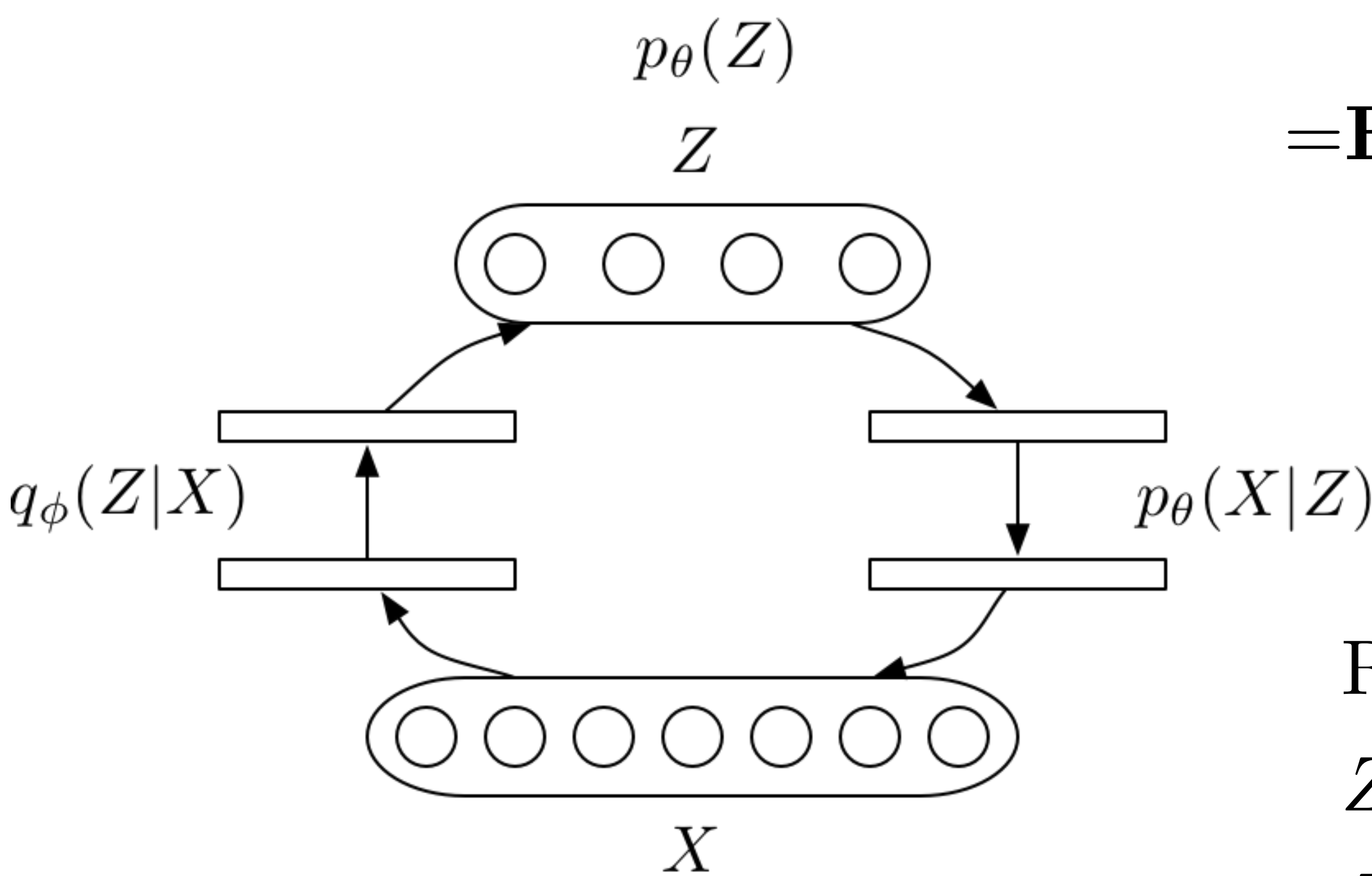


DRAW: A RNN for Image Generation

- [Gregor et al 2015]

Variational Auto-encoders

- VAEs [Kingma and Welling 2014], [Rezende et al 2014]



$$\log p_\theta(X) \geq \mathrm{ELBO}(\theta, \phi)$$

$$= \mathbf{E}_q \left[\log p_\theta(X|Z) + \log \frac{p_\theta(Z)}{q_\phi(Z|X)} \right]$$

Reparameterization Trick:

$$Z \sim q_\phi(Z|X)$$

$$\Leftrightarrow Z = f_\phi(S, X), S \sim N(0, I)$$

$$\mathrm{ELBO}(\theta, \phi) = \mathbb{E}_{S \sim N(0,I)} \left[\log p_\theta(X|f_\phi(S, X)) + \log \frac{p_\theta(f_\phi(S, X))}{q_\phi(f_\phi(S, X)|X)} \right]$$

VAEs with Discrete Latent Variables

- Reparameterization trick is crucial to VAEs.

- Many models naturally involve discrete latent variables:
  - presence or absence of features
  - attention mechanisms
  - stacks, queues and other discrete data structures
  - control flow

- Reparameterization trick for discrete latent variables?

Discrete Variables

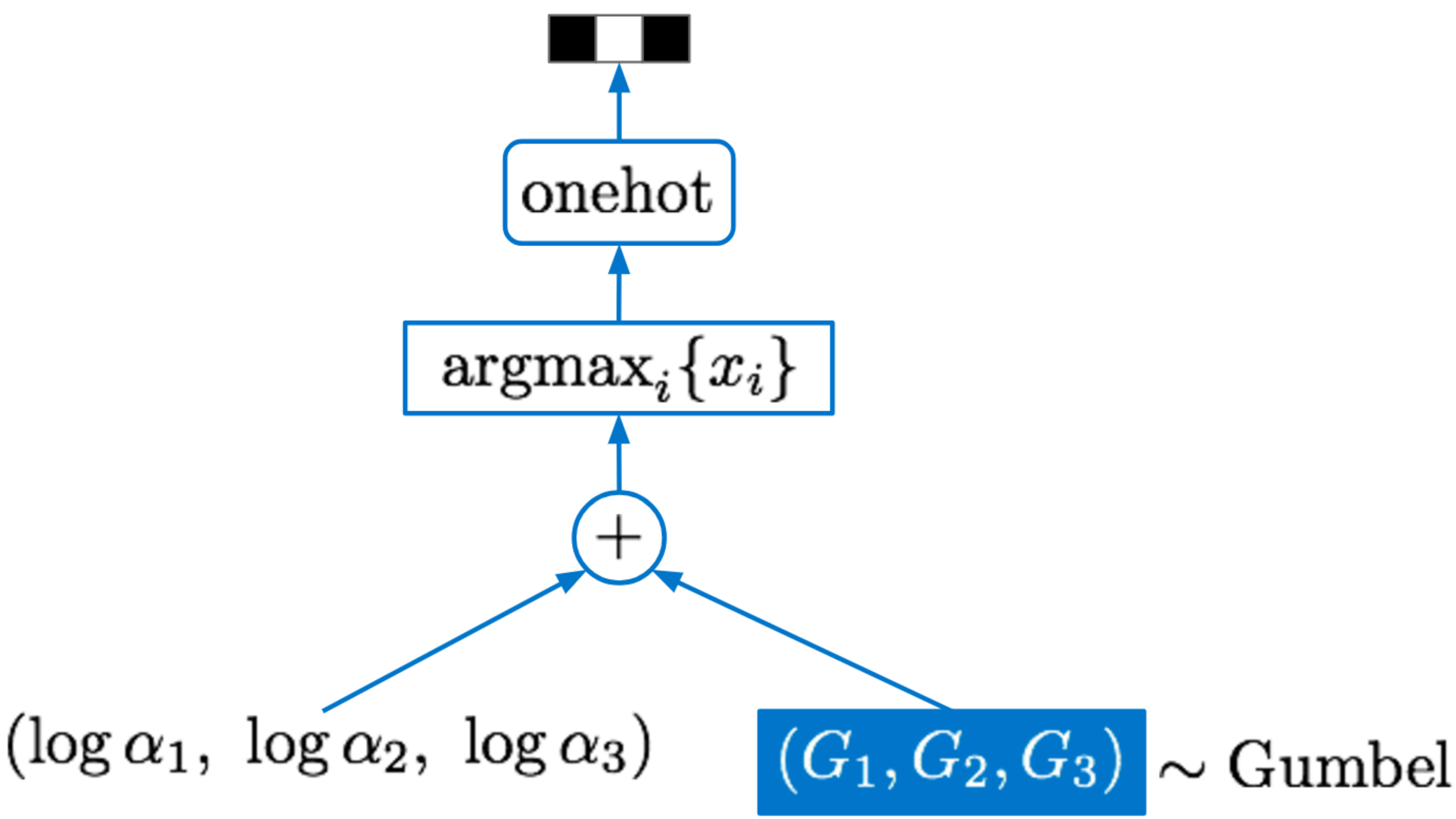
$$Z \sim \mathrm{Discrete}(\alpha_1, \alpha_2, \alpha_3)$$

$$\frac{\alpha_1}{\alpha_1+\alpha_2+\alpha_3} \qquad \frac{\alpha_2}{\alpha_1+\alpha_2+\alpha_3} \qquad \frac{\alpha_3}{\alpha_1+\alpha_2+\alpha_3}$$

[Maddison et al, ICLR 2017] concurrent work [Jang et al ICLR 2017]

Computation for Discrete Variables

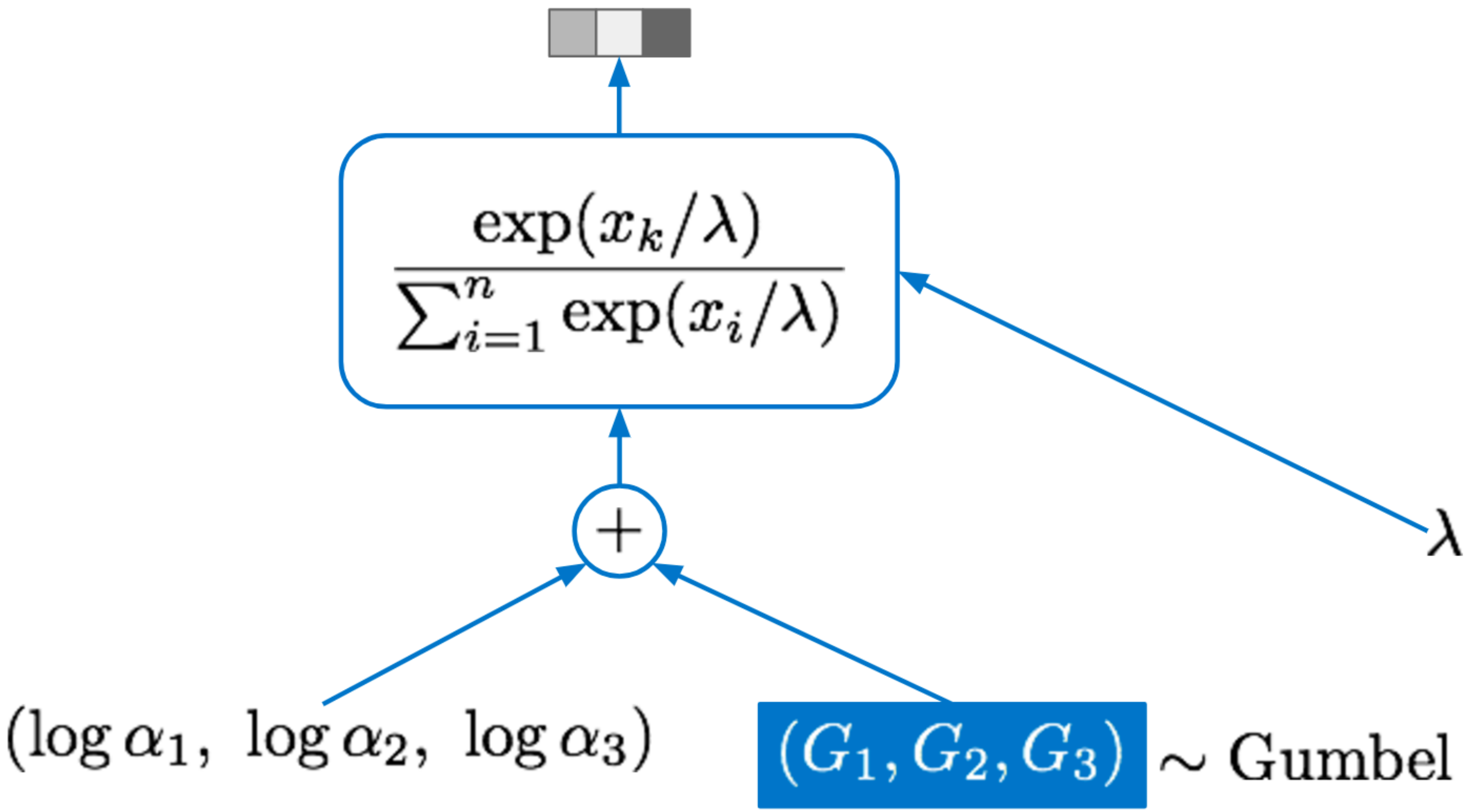
$$Z \sim \text{Discrete}(\alpha_1, \alpha_2, \alpha_3)$$



[Maddison et al, ICLR 2017] concurrent work [Jang et al ICLR 2017]

Computation for Concrete Variables

$$Z \sim \text{Concrete}(\alpha_1, \alpha_2, \alpha_3; \lambda)$$



[Maddison et al, ICLR 2017] concurrent work [Jang et al ICLR 2017]

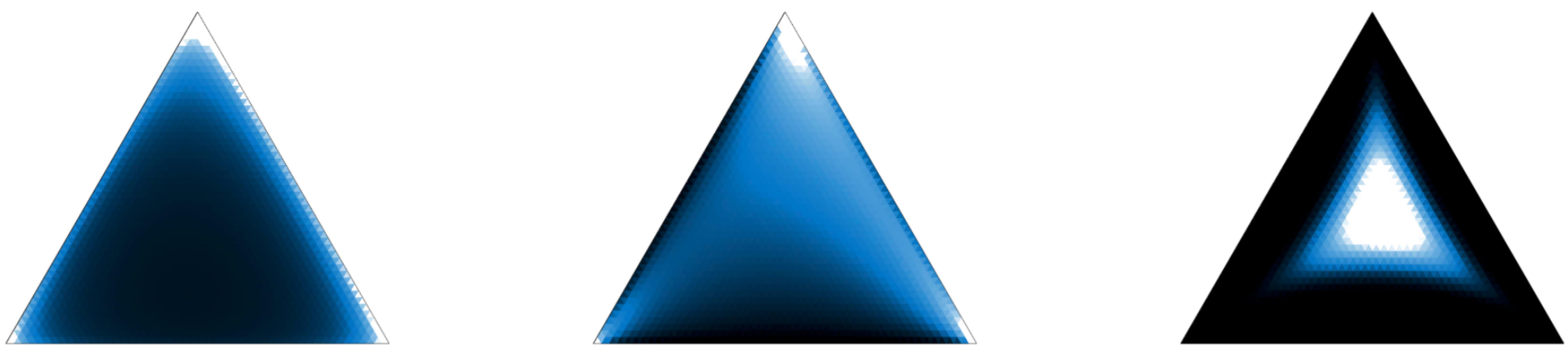
The Concrete Distribution

- **CON**tinuous relaxation of dis**CRETE** distributions.

$$\text{Concrete}(\alpha; \lambda_n) \to \text{Discrete}(\alpha) \text{ as } \lambda_n \to 0$$

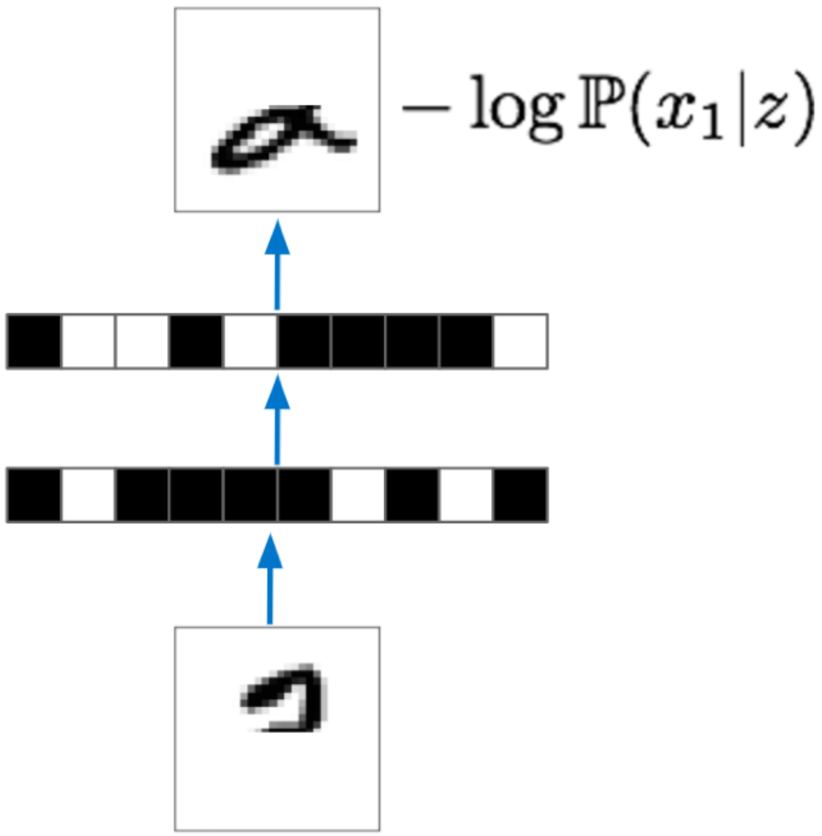
- Density:

$$(n-1)!\lambda^{n-1} \prod_{k=1}^{n} \frac{\alpha_k z_k^{-\lambda-1}}{\sum_{i=1}^{n} \alpha_i z_i^{-\lambda}}$$



[Maddison et al, ICLR 2017] concurrent work [Jang et al, ICLR 2017]

Structured Prediction with Concrete VAEs



binary model	m	Test NLL Concrete	Test NLL VIMCO	Train NLL Concrete	Train NLL VIMCO
(392V–240H –240H–392V)	1	**58.5**	61.4	**54.2**	59.3
	5	**54.3**	54.5	**49.2**	52.7
	50	53.4	**51.8**	**48.2**	49.6
(392V–240H –240H–240H –392V)	1	**56.3**	59.7	**51.6**	58.4
	5	**52.7**	53.5	**46.9**	51.6
	50	52.0	**50.2**	**45.9**	47.9

[Maddison et al, ICLR 2017] concurrent work [Jang et al, ICLR 2017]

Rebar: Reinforced Concrete

- [Tucker et al NIPS 2017]
- Reinforce trick [Williams 1992]:

$$\nabla_\theta \mathbf{E}_{q_\theta}[\mathcal{L}(Z)] = \mathbf{E}_{q_\theta}[\mathcal{L}(Z) \nabla_\theta \log q_\theta(Z)]$$

- Reinforce is unbiased but high variance.
- Concrete is low variance but biased.
- Use concrete estimator as control variate for reinforce!

- Relax [Grathwohl et al 2017] - generalize to learnt control variate.

Talk Outline

The Posterior Server

The Concrete VAE

FIVO: Filtered variational objectives

Importance Weighted Auto-Encoders

- Variational lower bound:

$$\log p(X|\theta) \geq \mathbf{E}_q \left[\log \frac{p(X|Z)p(Z)}{q(Z|X)} \right]$$

- IWAE [Burda et al 2015]: rederivation from importance sampling

$$p(X|\theta) = \mathbf{E}_{p(Z)} \left[p(X|Z) \right] = \mathbf{E}_{q(Z|X)} \left[\frac{p(X|Z)p(Z)}{q(Z|X)} \right]$$

- Better to use multiple samples

$$\log p(X|\theta) \geq \mathbf{E}_q \left[\log \frac{1}{N} \sum_{i=1}^{N} \frac{p(X|Z_i)p(Z_i)}{q(Z_i|X)} \right]$$

  - See also VIMCO [Mnih & Rezende 2016].

FIVO: Filtered Variational Objectives

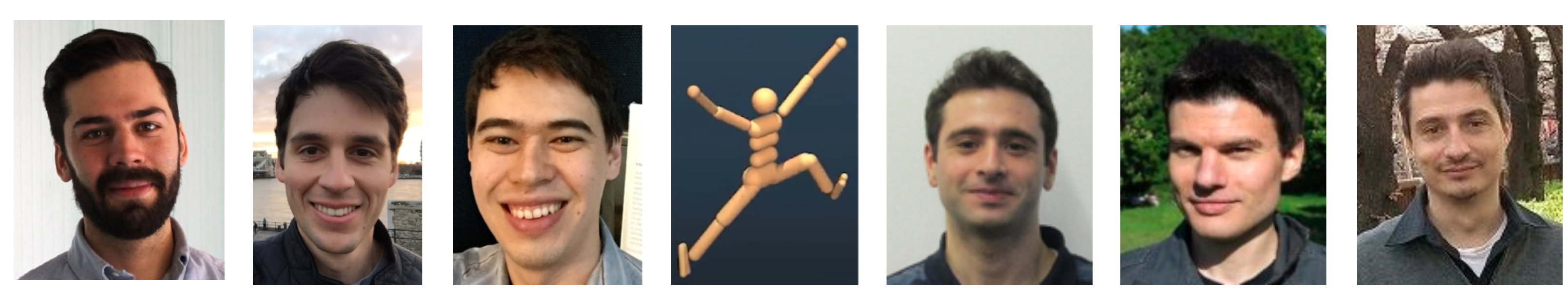
- We can use any unbiased estimator $\hat{p}(X)$ of marginal probability

$$p(X) = \mathbf{E}[\hat{p}(X)] \qquad\qquad \log p(X) \geq \mathbf{E}[\log \hat{p}(X)]$$

- Tightness of bound related to variance of estimator,

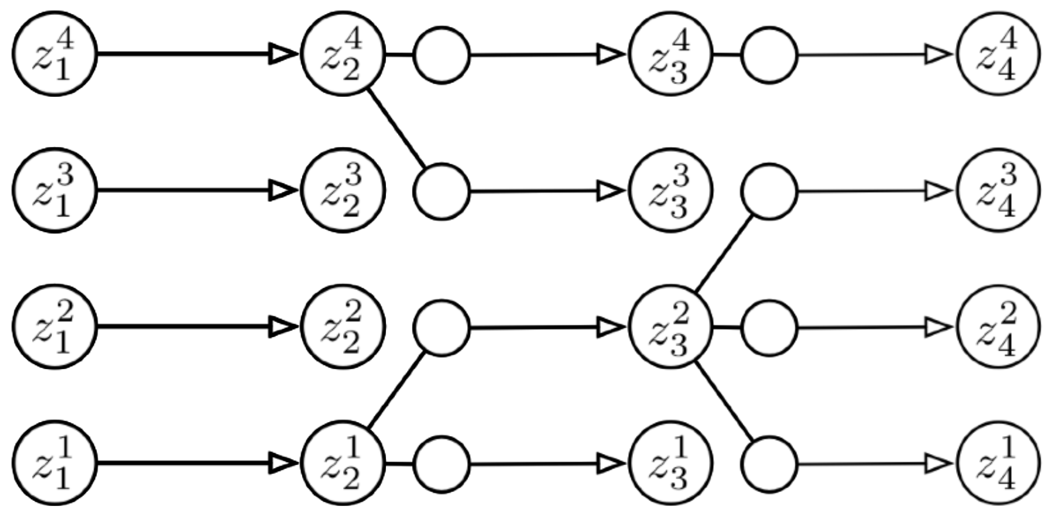
$$\log p(X) - \mathbf{E}[\log \hat{p}(X)] \approx \frac{1}{2}\mathrm{Var}\left[\frac{\hat{p}(X)}{p(X)}\right]$$

- For sequential models, we can use particle filters [Doucet et al 2009]:
  - produces unbiased estimator of marginal probability.
  - can have much lower variance than importance samplers.

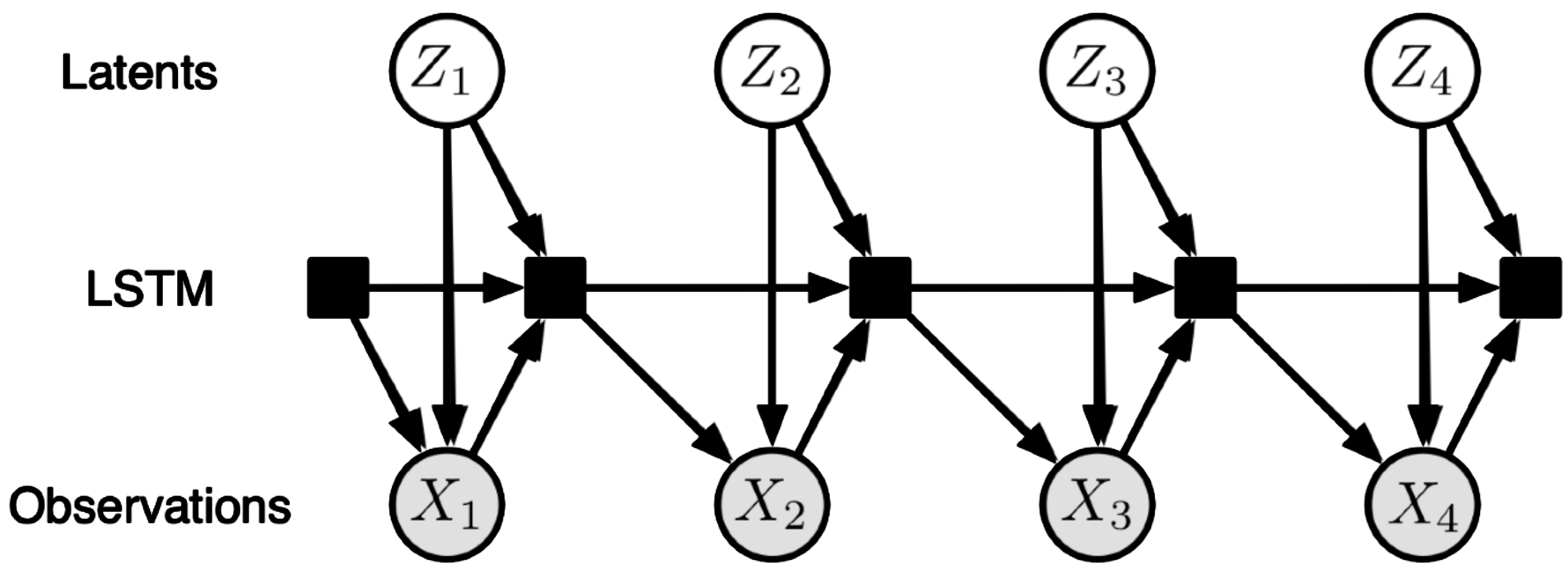


[Maddison et al, NIPS 2017] concurrent work [Le et al 2017, Naesseth et al 2017]

FIVO: Filtered Variational Objectives



$$\hat{p}_t(X_t|X_{1:t-1}) = \frac{1}{n}\sum_{t=1}^{N} w_t^i$$



$$\mathcal{L}_N^{\mathrm{FIVO}}$$

$$= \mathbb{E}\left[\sum_{t=1}^{T} \log \hat{p}_t(X_t|X_{1:t-1})\right]$$

[Maddison et al, NIPS 2017] concurrent work [Le et al 2017, Naesseth et al 2017]

FIVO: Filtered Variational Objectives

N	Bound	Nottingham	JSB	MuseData	piano-midi
4	ELBO	-3.23	-8.61	-7.12	-7.79
	IWAE	-3.21	-8.59	-7.17	-7.81
	FIVO	**-2.86**	**-6.95**	**-6.55**	**-7.72**
8	ELBO	-3.60	-8.60	-7.11	-7.83
	IWAE	-3.30	-7.53	-7.10	-7.81
	FIVO	**-2.62**	**-6.69**	**-6.36**	**-7.49**
16	ELBO	-3.54	-8.60	-7.17	-7.83
	IWAE	-2.95	-7.55	-7.08	-7.81
	FIVO	**-2.58**	**-6.60**	**-6.09**	**-7.19**

		TIMIT	
N	Bound	64 units	512 units
4	ELBO	35,908	36,981
	IWAE	35,984	34,067
	FIVO	**40,211**	**41,834**
8	ELBO	35,612	37,902
	IWAE	36,835	38,074
	FIVO	**40,912**	**41,666**

Table 1: Test set performance of models trained with different bounds and numbers of particles on polyphonic music and TIMIT.

[Maddison et al, NIPS 2017] concurrent work [Le et al 2017, Naesseth et al 2017]

Concluding Remarks

- Bringing management of uncertainties into deep learning
  - What uncertainties do we need in Bayesian deep learning for computer vision? [Kendall & Gal 2017], concrete dropout [Gal et al 2017]
  - Decomposition of uncertainty for active learning and reliable reinforcement learning in stochastic systems [Depeweg et al 2017]
  - Simple and scalable predictive uncertainty estimation using deep ensembles [Laksnminarayanan et al 2017]
- Bringing flexibility and scalability to Bayesian modelling
  - DRAW: A RNN for image generation [Gregor et al 2015]
  - WaveNet: A Generative Model for Raw Audio [van den Oord et al 2016]
  - Composing graphical models with neural networks for structured representations and fast inference [Johnson et al 2016]
  - Learning disentangled representations with semi-Supervised deep generative models [Narayanaswamy et al 2017]
  - A disentangled recognition and nonlinear dynamics model for unsupervised learning [Fraccaro et al 2017]

Concluding Remarks

- Development of deep probabilistic programming systems
  - Edward, Bayesflow, pyro, probtorch

- NIPS Workshops:
  - Advances in Approximate Bayesian Inference (Friday)
  - Bayesian Deep Learning (Saturday)

- Questions to think about:
  - Being Bayesian in the space of functions instead of parameters?
  - How to deal with uncertainties under model misspecification?

Thank you!

- NIPS organizers
- Funders

EPSRC Engineering and Physical Sciences Research Council

erc European Research Council Established by the European Commission

DeepMind

- Questions?

References

- Stochastic Natural-Gradient Expectation Propagation and the Posterior Server [Hasenclever et al JMLR 2017]
  - arXiv:1512.09327

- Concrete variational auto-encoders [Maddison et al ICLR 2017]
  - arXiv:1611.00712

- Filtered variational objectives [Maddison et al NIPS 2017]
  - arXiv:1705.09279