PRIORS FOR DEEP INFINITE NETWORKS

SAM SCHOENHOLZ

GOOGLE BRAIN

CARGÈSE, 2018

COLLABORATION

Jeffrey Pennington

Jascha Sohl-Dickstein

Surya Ganguli

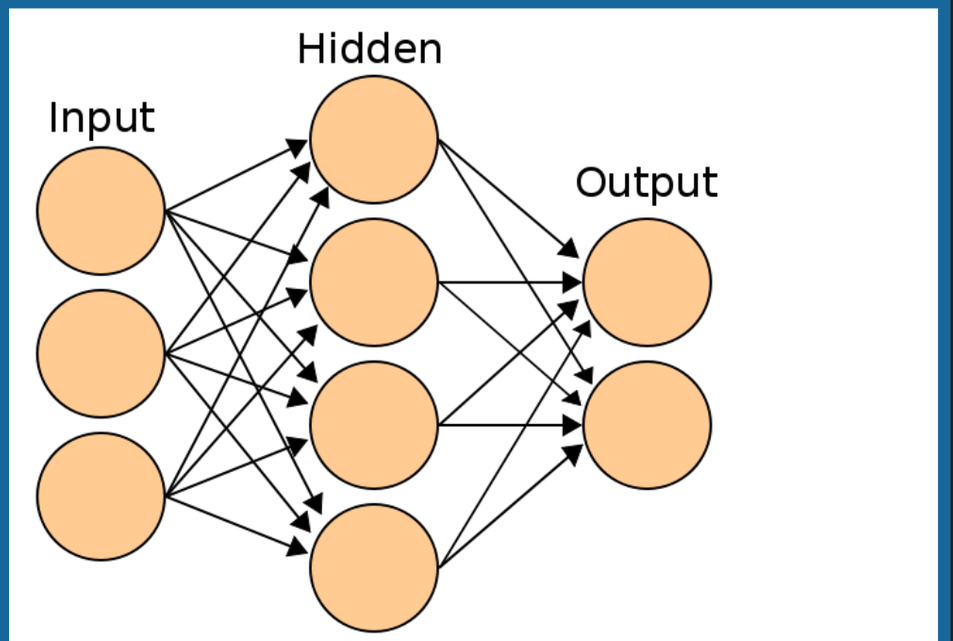
Greg Yang

Lechao Xiao

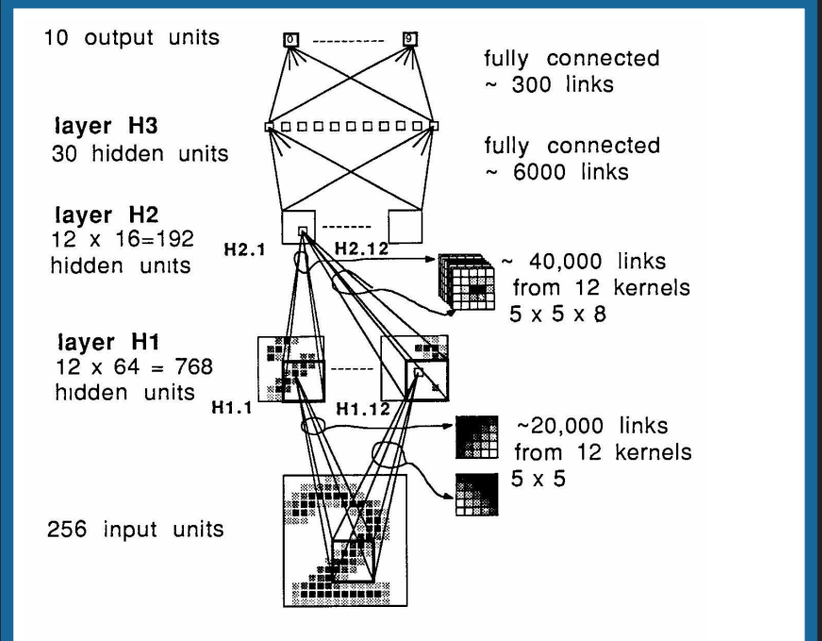
Minmin Chen

Yasaman Bahri

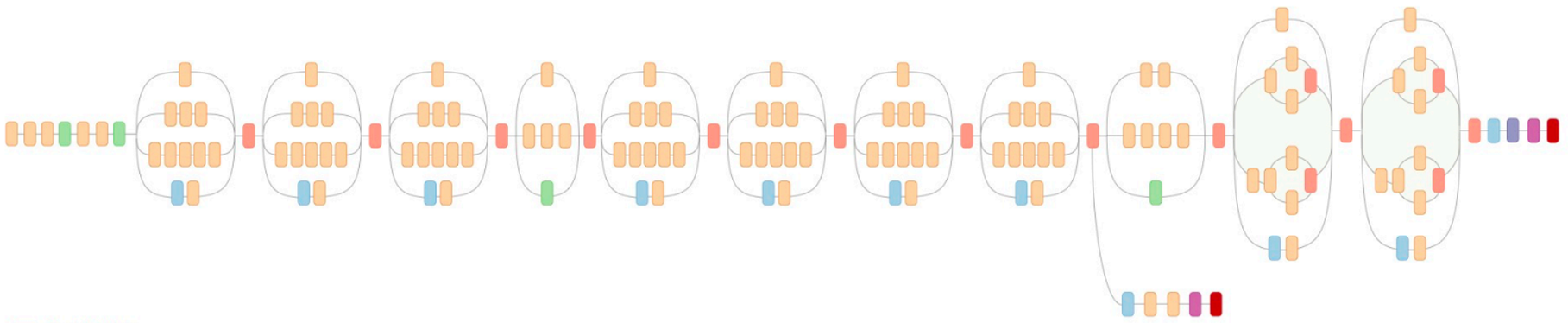
PROLIFERATION OF MODELS IN DEEP LEARNING



Fully Connected

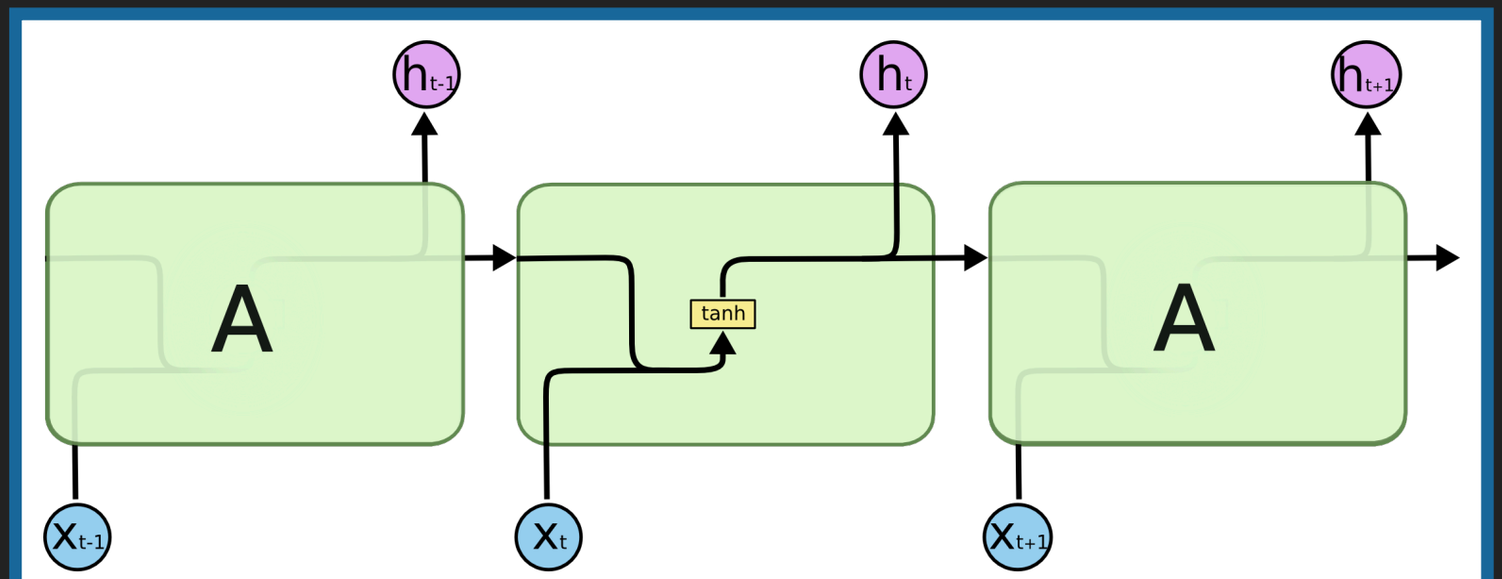


Convolutions (LeCun et al. ’82)

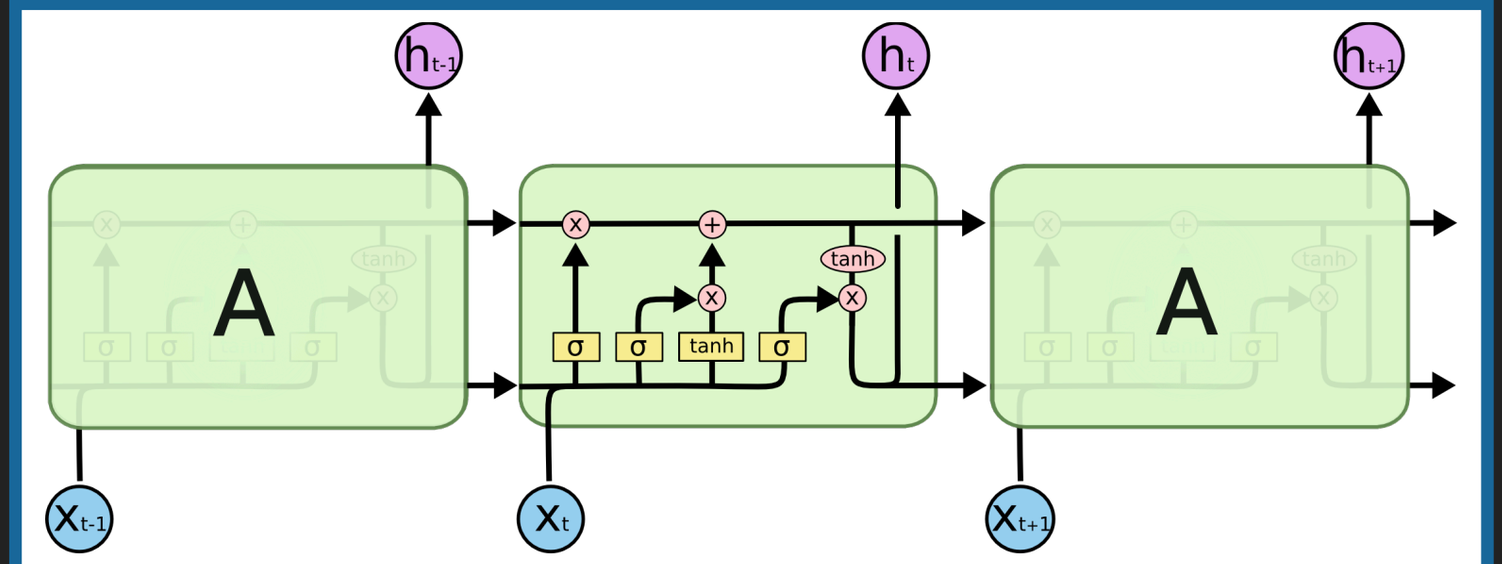


Inception v3 (Szagedy et al. ’15)

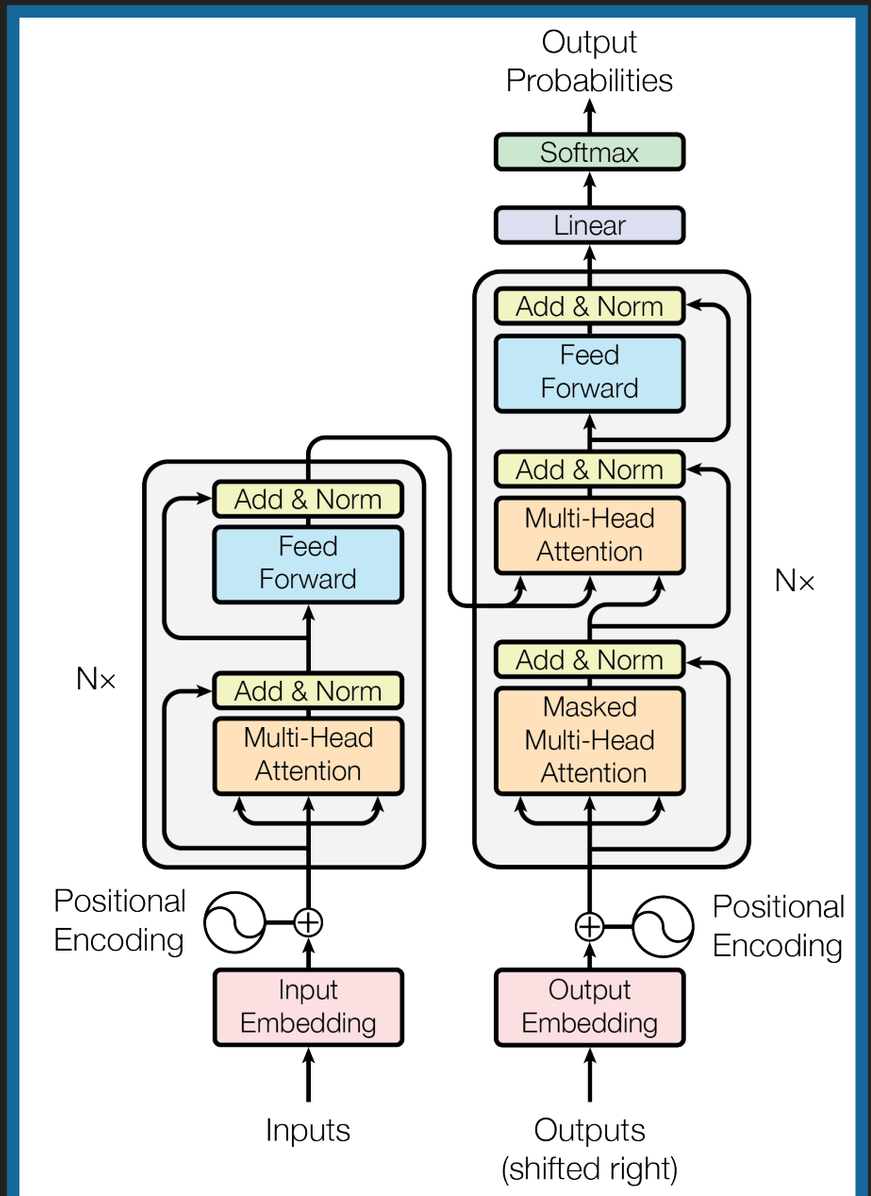
PROLIFERATION OF MODELS IN DEEP LEARNING



Recurrent Neural Network Cell (Hopfield '82)



Long Short Term Memory Cell (Hochreiter and Schmidhuber '97)



Transformer (Vaswani et al. '17)

MOTIVATION

Given a problem / dataset

1. Propose a class of architectures
2. Select a set of hyperparameters
3. Randomly initialize network
4. Train network and evaluate performance
5. Repeat from 2. until saturated computational budget
6. Repeat from 1. until happy (or deadline)

Problem: Conflates model performance with trainability

Can we do something more principled?

MOTIVATION

Given a problem / dataset

1. Propose a class of architectures
2. Select a set of hyperparameters
3. Randomly initialize network
4. Train network and evaluate performance
5. Repeat from 2. until saturated computational budget
6. Repeat from 1. until happy (or deadline)

Problem: Conflates model performance with trainability

Can we do something more principled?

NEURAL NETWORK PRIORS

Instead of training a single random network, train an ensemble

Equivalent to taking a prior over functions to a “posterior”

$$p(\mathcal{D}|\theta;\omega)p(\theta;\omega) \xrightarrow{\text{SGD}} q(\theta|\mathcal{D};\omega)$$

Good hyperparameters correspond to selecting good priors

Generally, these priors are intractable

In the infinite “size” limit we can use mean field theory and random matrix theory

Theory is (approx.) tractable for many network topologies

THE SINGLE HIDDEN LAYER CASE

Fully connected, single hidden layer [Radford Neal '94]

Inputs: $x_a \in \mathbb{R}^{N_0}$ with input index *a=1,2* $\quad \Sigma^0_{ab} = \frac{1}{N_0} \sum_i x_{ia} x_{ib}$

Parameters: $W^l_{ij} \in \mathbb{R}^{N_{l-1} \times N_l} \qquad b^l_i \in \mathbb{R}^{N_l}$

Prior: $\quad W^l_{ij} \sim \mathcal{N}(0, \sigma_w^2 / N_{l-1}) \qquad b^l_i \sim \mathcal{N}(0, \sigma_b^2)$

Network:

Weighted sum of Gaussians

$$z^1_{ia} = \sum_j W^1_{ij} x_{ja} + b^1_i \qquad (z^1_{ia}, z^1_{jb})^T \sim \mathcal{N}(0, \Sigma^1_{ab} \delta_{ij})$$

$$y_{ia} = \sum_j W^2_{ij} \phi(z^1_{ja}) + b^2_i \qquad (y_{ia}, y_{jb})^T \xrightarrow{N_1 \to \infty} \mathcal{N}(0, \Sigma^2_{ab} \delta_{ij})$$

Sum of i.i.d. random variables

THE SINGLE HIDDEN LAYER CASE

Infinitely wide neural networks are Gaussian Processes

$$z^1_{ia} = \sum_j W^1_{ij} x_{ja} + b^1_i \qquad (z^1_{ia}, z^1_{jb})^T \sim \mathcal{N}(0, \Sigma^1_{ab}\delta_{ij})$$

$$y_{ia} = \sum_j W^2_{ij} \phi(z^1_{ja}) + b^2_i \qquad (y_{ia}, y_{jb})^T \xrightarrow{N_1 \to \infty} \mathcal{N}(0, \Sigma^2_{ab}\delta_{ij})$$

Completely defined by covariance matrices

$$\Sigma^1 = \sigma_w^2 \Sigma^0 + \sigma_b^2$$

$$\Sigma^2 = \sigma_w^2 \int \mathcal{D}_{\Sigma^1} \boldsymbol{z} \phi(\boldsymbol{z}) \phi(\boldsymbol{z})^T + \sigma_b^2 \qquad \mathcal{D}_\Sigma \boldsymbol{z} = \frac{1}{2\pi\sqrt{|\Sigma|}} d\boldsymbol{z} e^{-\frac{1}{2}\boldsymbol{z}^T \Sigma^{-1} \boldsymbol{z}}$$

Significant simplification

DEEP NETWORKS

Extension to deep networks

$$z^1_{ia} = \sum_j W^1_{ij} x_{ja} + b^1_i \longrightarrow \Sigma^1 = \sigma_w^2 \Sigma^0 + \sigma_b^2$$

$$\downarrow$$

$$z^2_{ia} = \sum_j W^2_{ij} \phi(z^1_{ja}) + b^2_i \xrightarrow{N_1 \to \infty} \Sigma^2 = \sigma_w^2 \int \mathcal{D}_{\Sigma^1} \boldsymbol{z} \phi(\boldsymbol{z}) \phi(\boldsymbol{z})^T + \sigma_b^2$$

$$\downarrow$$

$$\vdots$$

$$\downarrow$$

$$z^l_{ia} = \sum_j W^l_{ij} \phi(z^{l-1}_{ja}) + b^l_i \xrightarrow{N_{l-1} \to \infty} \Sigma^l = \sigma_w^2 \int \mathcal{D}_{\Sigma^{l-1}} \boldsymbol{z} \phi(\boldsymbol{z}) \phi(\boldsymbol{z})^T + \sigma_b^2$$

Neural network induces dynamical system over covariances

Understanding prior equivalent to studying dynamics

SIMPLIFYING ASSUMPTIONS

We can choose to normalize the inputs

$$||\boldsymbol{x}_a||^2 = N_0 q^0$$

By symmetry this implies that

$$\Sigma^l = q^l \begin{pmatrix} 1 & c^l \\ c^l & 1 \end{pmatrix} \qquad c^l = \frac{1}{q^l} \boldsymbol{z}_a^l \cdot \boldsymbol{z}_b^l = \cos\theta^l$$

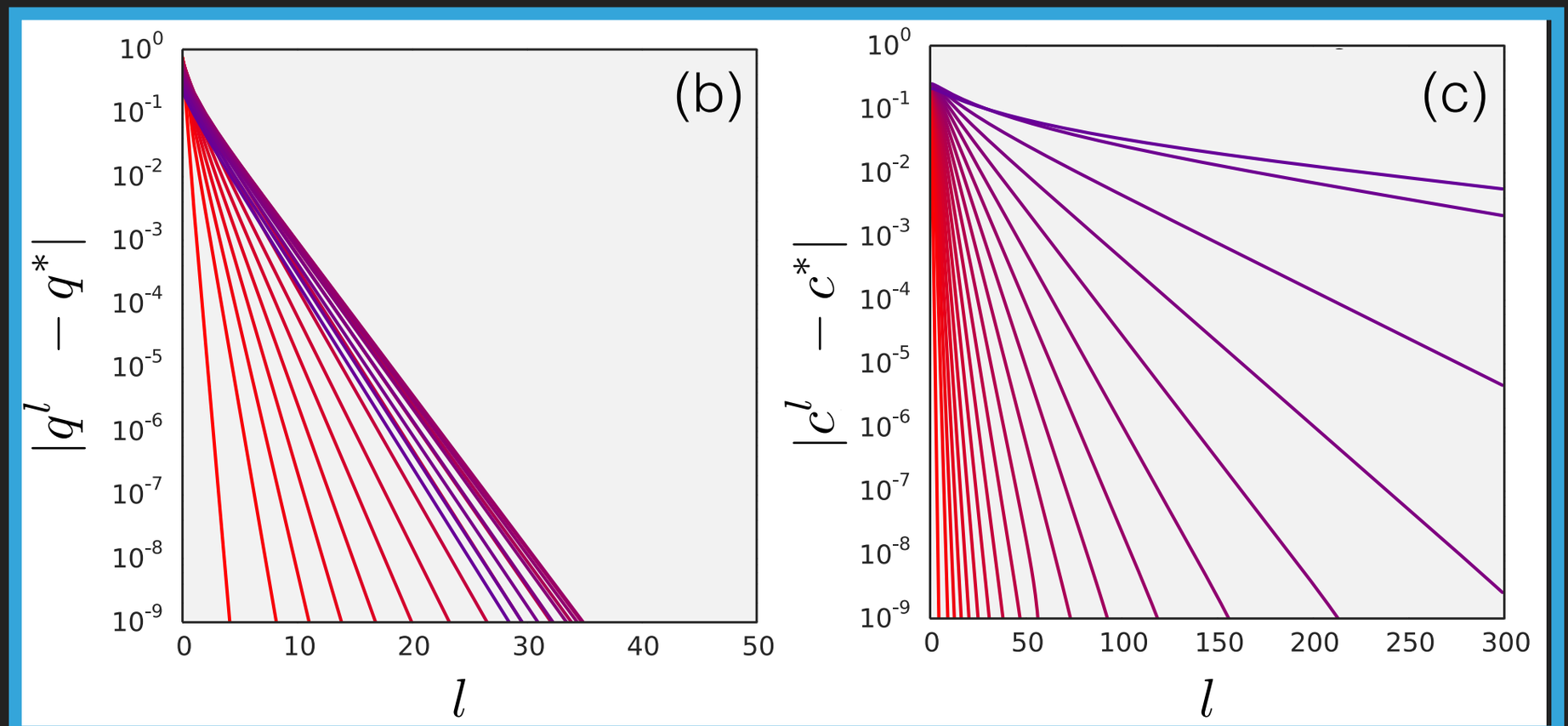
Variance Correlation

For $\phi = \tanh$ dynamical system has a fixed point $\lim_{l\to\infty} \Sigma^l = \Sigma^*$

$$\Sigma^* = q^* \begin{pmatrix} 1 & c^* \\ c^* & 1 \end{pmatrix}$$

SIMPLIFYING ASSUMPTIONS

Observe exponential convergence to fixed point



Convergence of q^l fast / uninteresting

Ignore it by choosing the normalization $q^0 = q^*$

CORRELATION DYNAMICS

Converted to studying one dimensional dynamics

$$c^{l+1} = \frac{1}{q^*}\left(\sigma_w^2 \int \mathcal{D}_{\Sigma^{l-1}} \boldsymbol{z} \phi(z_1)\phi(z_2) + \sigma_b^2\right)$$

Linearize the dynamics near the fixed point $c^l = c^* + \epsilon^l$

$$\epsilon^{l+1} = \chi_{c^*}\epsilon^l + \mathcal{O}((\epsilon^l)^2) \qquad \chi_{c^*} = \sigma_w^2 \int \mathcal{D}_{\Sigma^*} \boldsymbol{z} \phi'(z_1)\phi'(z_2)$$

Recover exponential dynamics $\epsilon^l \sim (\chi_{c^*})^l$
Stability if $\chi_{c^*} < 1$

Extract convergence depth scale $\epsilon^l \sim e^{-l/\xi_{c^*}} \quad \xi_{c^*}^{-1} = -\log \chi_{c^*}$

Depth diverges whenever a fixed point is marginally stable

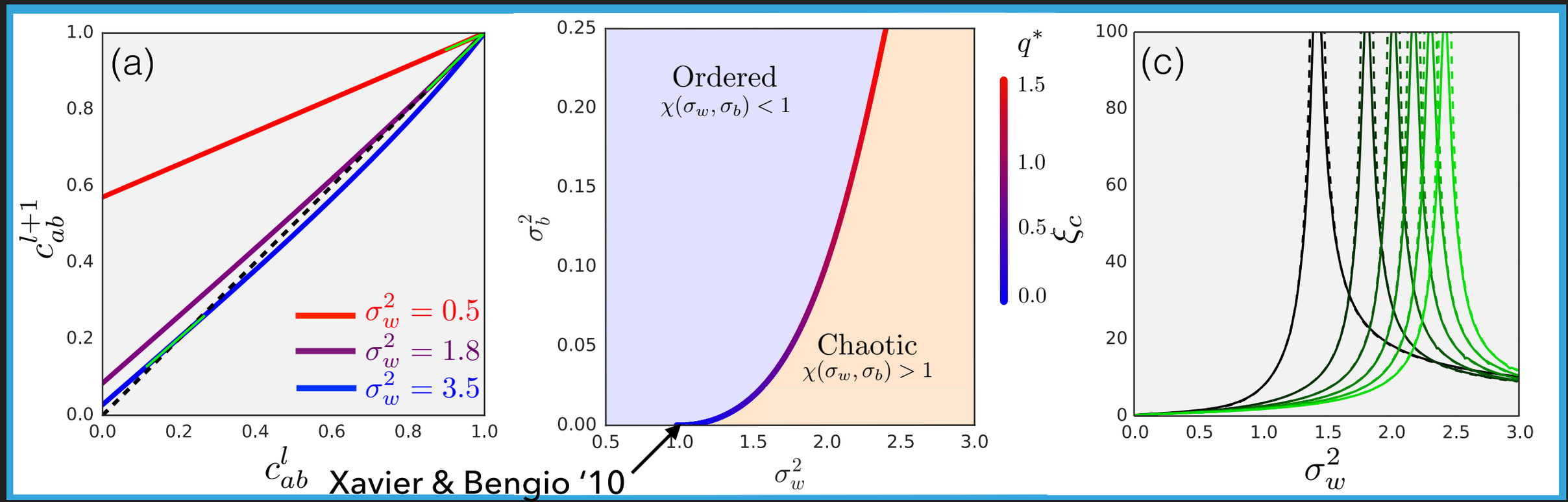
FIXED POINT STRUCTURE

Always a fixed point at $c^* = 1$

Stability of the fixed point defines an order-to-chaos transition

At phase transition depth-scale diverges

On critical like convergence is power-law



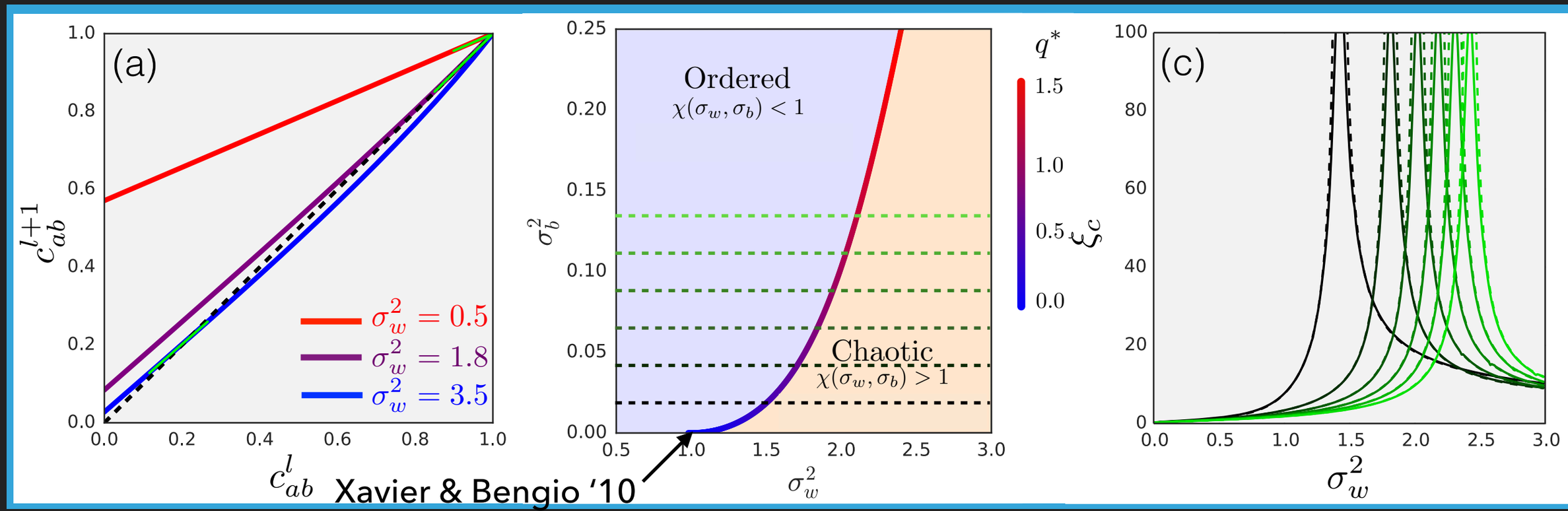
FIXED POINT STRUCTURE

Always a fixed point at $c^* = 1$

Stability of the fixed point defines an order-to-chaos transition

At phase transition depth-scale diverges

On critical like convergence is power-law



MEAN FIELD GRADIENTS

There is a duality between forward propagation of signal and back-propagation of gradients

Given a loss $\mathcal{L}$, back-propagation gives

$$\frac{\partial \mathcal{L}}{\partial W_{ij}^l} = \delta_i^l \phi(z_j^{l-1}) \quad \delta_i^l = \phi'(z_i^l) \sum_j \delta_j^{l+1} W_{ji}^{l+1} \qquad \delta_i^l = \frac{\partial L}{\partial z_i^l}$$

Gradient scale will be proportional to $\tilde{q}^l = \mathbb{E}[(\delta_i^l)^2]$

Assume independence

MEAN FIELD GRADIENTS

There is a duality between forward propagation of signal and back-propagation of gradients

Given a loss $\mathcal{L}$, back-propagation gives

$$\frac{\partial \mathcal{L}}{\partial W_{ij}^l} = \delta_i^l \phi(z_j^{l-1}) \quad \delta_i^l = \phi'(z_i^l) \sum_j \delta_j^{l+1} W_{ji}^{l+1} \qquad \delta_i^l = \frac{\partial L}{\partial z_i^l}$$

Gradient scale will be proportional to $\tilde{q}^l = \mathbb{E}[(\delta_i^l)^2]$

Assume independence

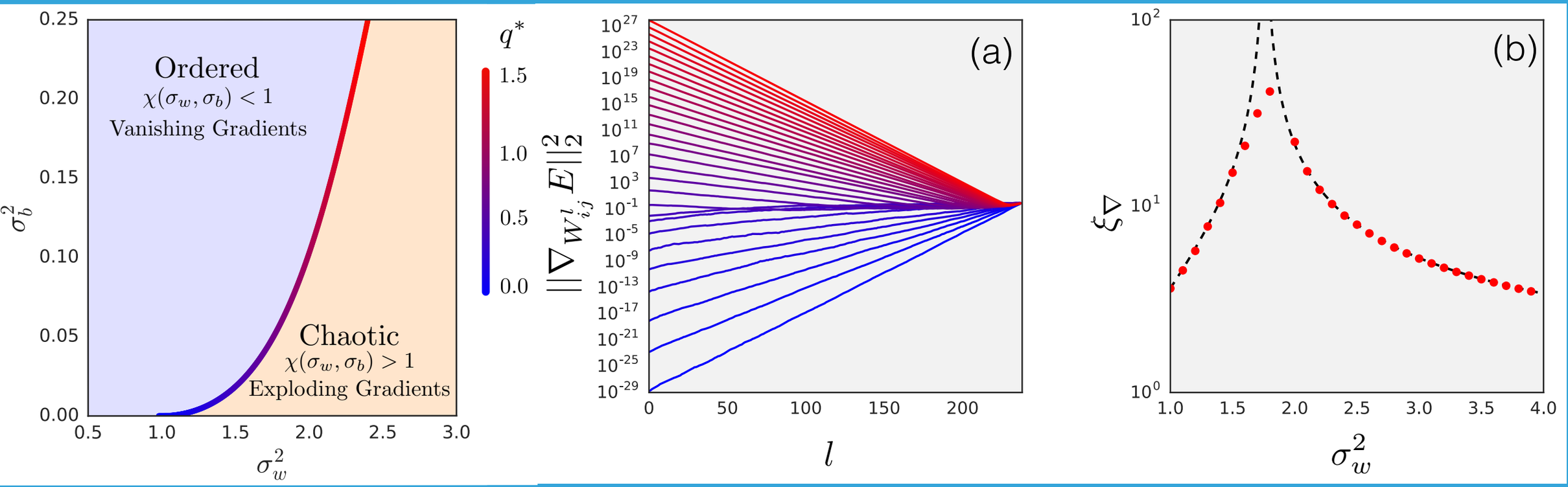
$$\tilde{q}^l = \frac{N_{l+1}}{N_{l+2}} \chi_1 \tilde{q}^{l+1}$$

MEAN FIELD GRADIENTS

Constant width networks $\mathbb{E}[(\nabla_{W_{ij}^l}\mathcal{L})^2] \sim \chi_1^{L-l}$

Ordered Phase - Vanishing Gradients
Chaotic Phase - Exploding Gradients

Grow / Vanish exponentially over depth $\xi_\nabla^{-1} = |\log \chi_1|$



TRAINABILITY

If $L \gg \xi$

$$P(y|x;\theta,\omega) = P(y;\theta,\omega)$$

Independent of the input

Signal cannot pass through the network

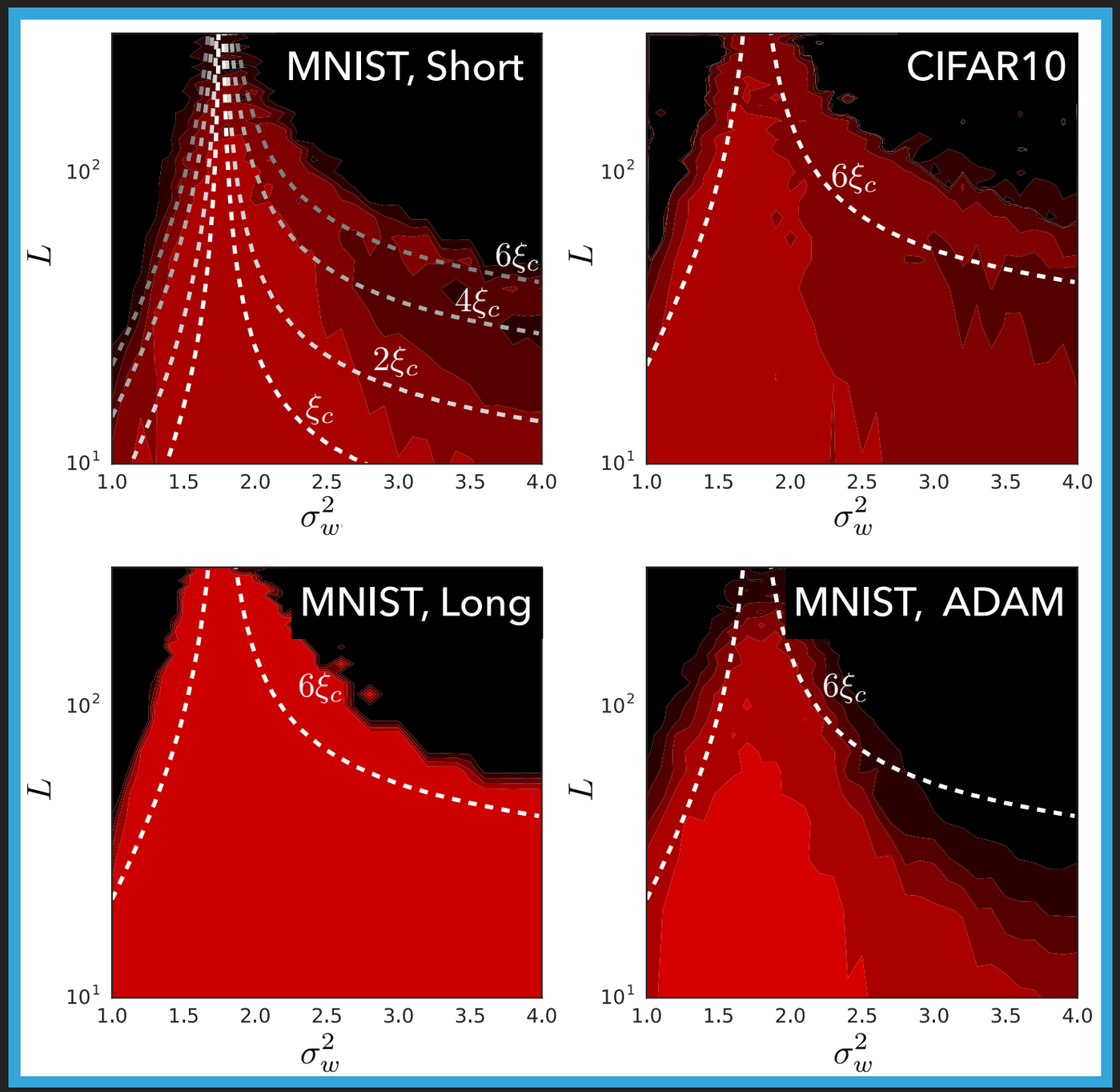
Gradient are poorly behaved

Network should not be trainable

This result is data independent

Consequence: at order-to-chaos transition networks should be trainable at large depths

TRAINABILITY



DROPOUT

Mean field formalism can be extended to include dropout

$$z_{ia}^l = \frac{1}{\rho} \sum_j W_{ij}^l p_{ja} \phi(z_{ja}^{l-1}) + b_i^l \qquad p_{ia} \sim \text{Bernoulli}(\rho)$$

Diagonal and off-diagonal terms have different scaling

$$\Sigma_{aa}^l = \frac{\sigma_w^2}{\rho} \int \mathcal{D}_{\Sigma^{l-1}} \boldsymbol{z} \phi(z_1)^2 + \sigma_b^2$$

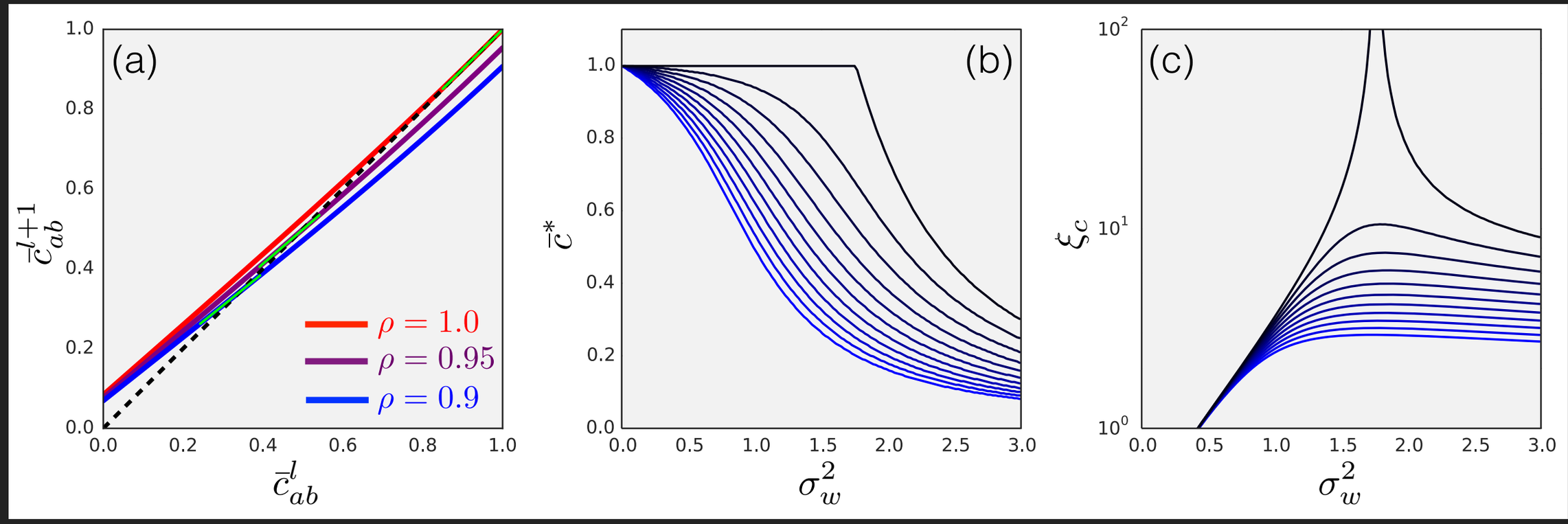
$$\Sigma_{ab}^l = \sigma_w^2 \int \mathcal{D}_{\Sigma^{l-1}} \boldsymbol{z} \phi(z_1) \phi(z_2) + \sigma_b^2$$

Implies that $c^* = 1$ is no longer a fixed point

DROPOUT

Infinitesimal amounts of dropout destroys order-to-chaos transition

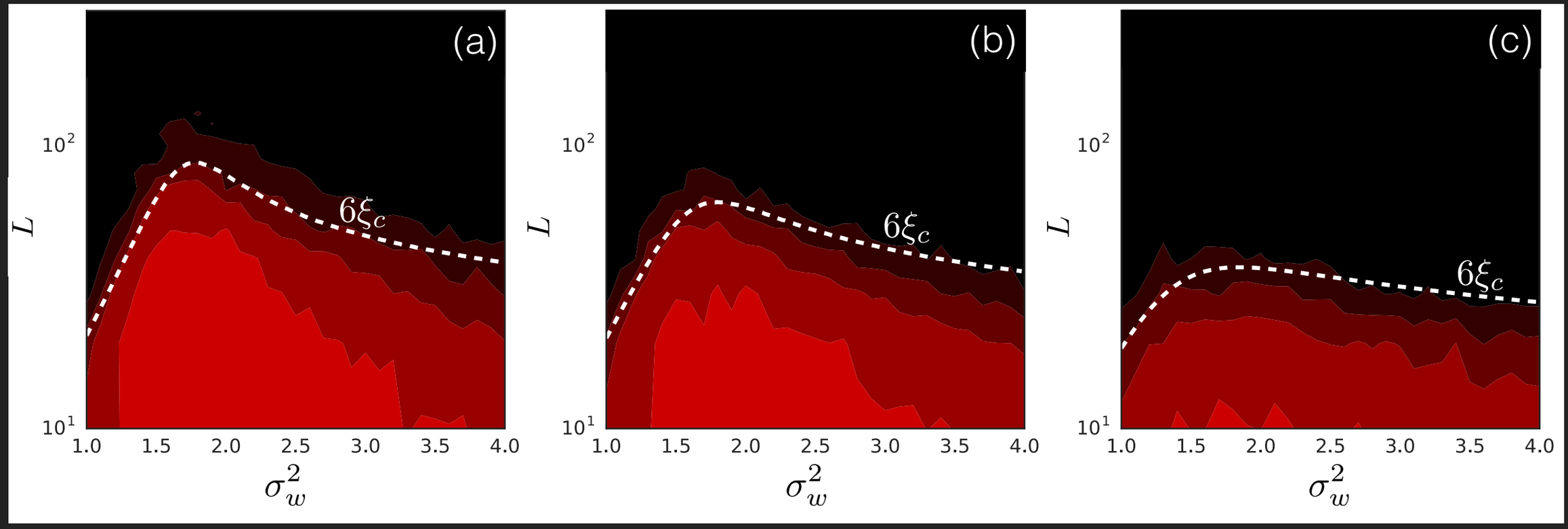
Significantly limits signal propagation depth



DROPOUT

Infinitesimal amounts of dropout destroys order-to-chaos transition

Significantly limits maximum trainable depth



RECTIFIED LINEAR NETWORKS

Can compute dynamics in closed form

$$q^{l+1} = \frac{1}{2}\sigma_w^2 q^l + \sigma_b^2 \qquad q^* = \frac{\sigma_b^2}{1 - \sigma_w^2/2}$$

$$c^{l+1} = \frac{\sigma_w^2}{2\pi}\frac{q^l}{q^{l+1}}\left[\sqrt{1-(c^l)^2} + c^l\left(\frac{\pi}{2} + \arcsin(c^l)\right)\right] + \frac{\sigma_b^2}{q^{l+1}}$$

Phase transition from bounded to unbounded q^l at $\sigma_w^2 = 2$

RECTIFIED LINEAR NETWORKS

Can compute dynamics in closed form

$$q^{l+1} = \frac{1}{2}\sigma_w^2 q^l + \sigma_b^2 \qquad q^* = \frac{\sigma_b^2}{1 - \sigma_w^2/2}$$

$$c^{l+1} = \frac{\sigma_w^2}{2\pi}\frac{q^l}{q^{l+1}}\left[\sqrt{1-(c^l)^2} + c^l\left(\frac{\pi}{2} + \arcsin(c^l)\right)\right] + \frac{\sigma_b^2}{q^{l+1}}$$

Phase transition from bounded to unbounded q^l at $\sigma_w^2 = 2$

Bounded phase: scale dependent, exponential convergence

$$q^* - q^l \sim (\chi_1)^l \qquad c^* - c^l \sim (\chi_1)^l \qquad \chi_1 = \sigma_w^2/2$$

RECTIFIED LINEAR NETWORKS

Can compute dynamics in closed form

$$q^{l+1} = \frac{1}{2}\sigma_w^2 q^l + \sigma_b^2 \qquad q^* = \frac{\sigma_b^2}{1 - \sigma_w^2/2}$$

$$c^{l+1} = \frac{\sigma_w^2}{2\pi}\frac{q^l}{q^{l+1}}\left[\sqrt{1-(c^l)^2} + c^l\left(\frac{\pi}{2} + \arcsin(c^l)\right)\right] + \frac{\sigma_b^2}{q^{l+1}}$$

Phase transition from bounded to unbounded q^l at $\sigma_w^2 = 2$

Bounded phase: scale dependent, exponential convergence

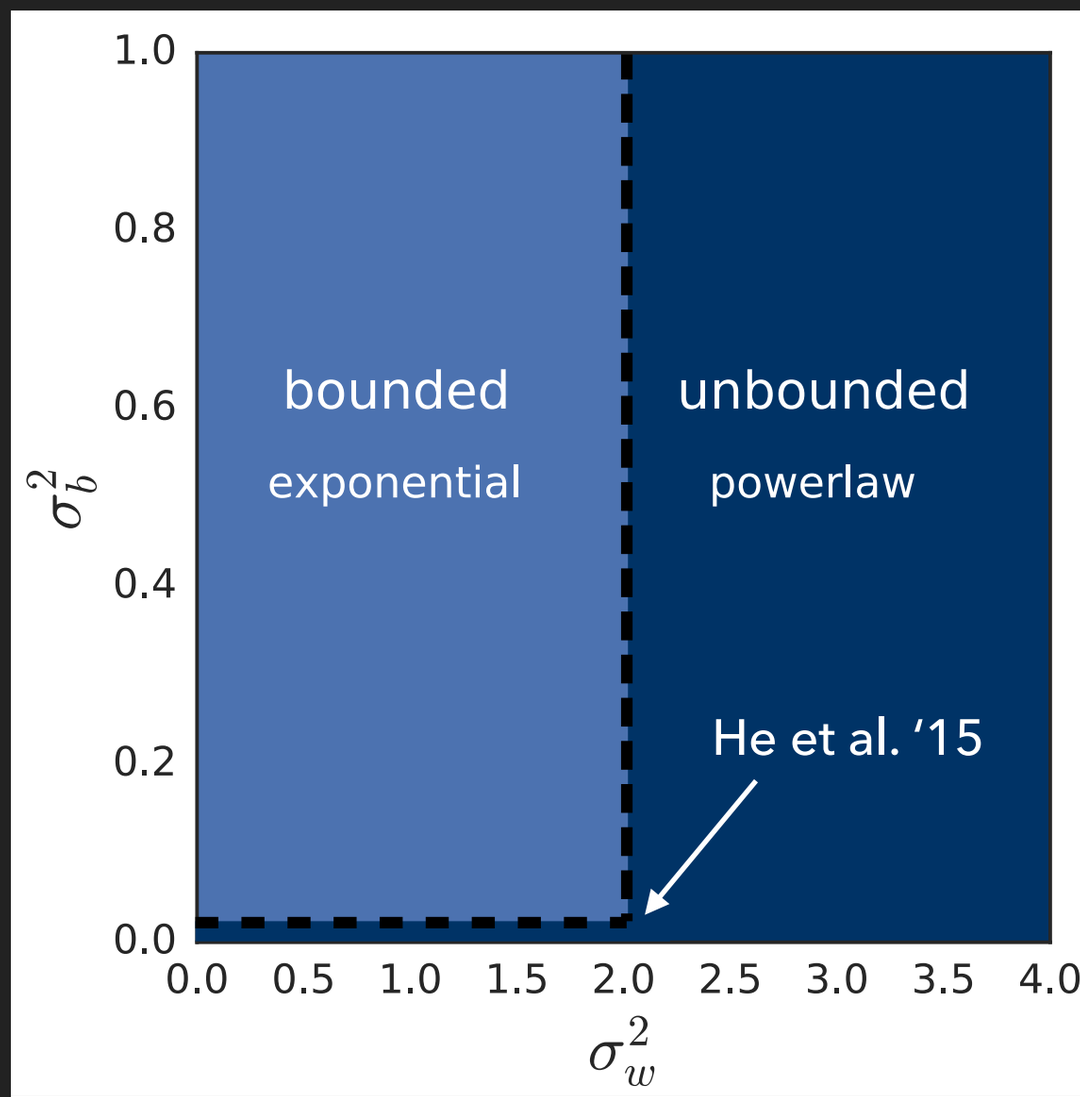
$$q^* - q^l \sim (\chi_1)^l \qquad c^* - c^l \sim (\chi_1)^l \qquad \chi_1 = \sigma_w^2/2$$

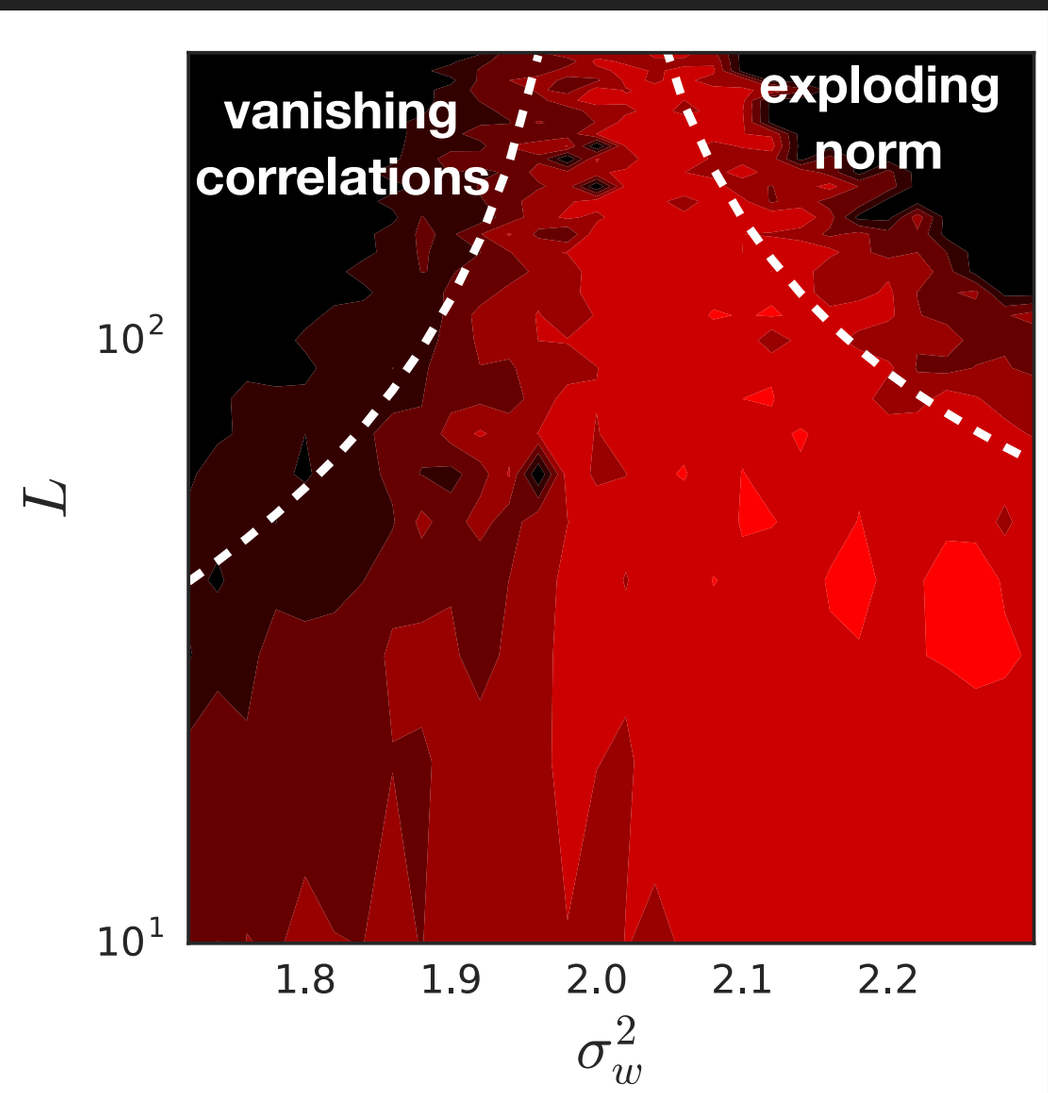
Unbounded phase: scale free, power law convergence

$$q^l \sim (\chi_1)^l \qquad c^* - c^l \sim \left(\frac{\xi_0}{l}\right)^2 \qquad \xi_0 = \frac{3\pi}{4\sqrt{2}}$$

RECTIFIED LINEAR NETWORKS

Phase transition at widely used normalization





RESIDUAL CONNECTIONS

Residual networks

$$z_{ia}^{l} = z_{ia}^{l-1} + \sum_{j} V_{ij}^{l} \phi\left(\sum_{k} W_{jk}^{l} z_{ka}^{l-1} + b_{j}^{l}\right) + a_{i}^{l}$$

Can derive the following relationships

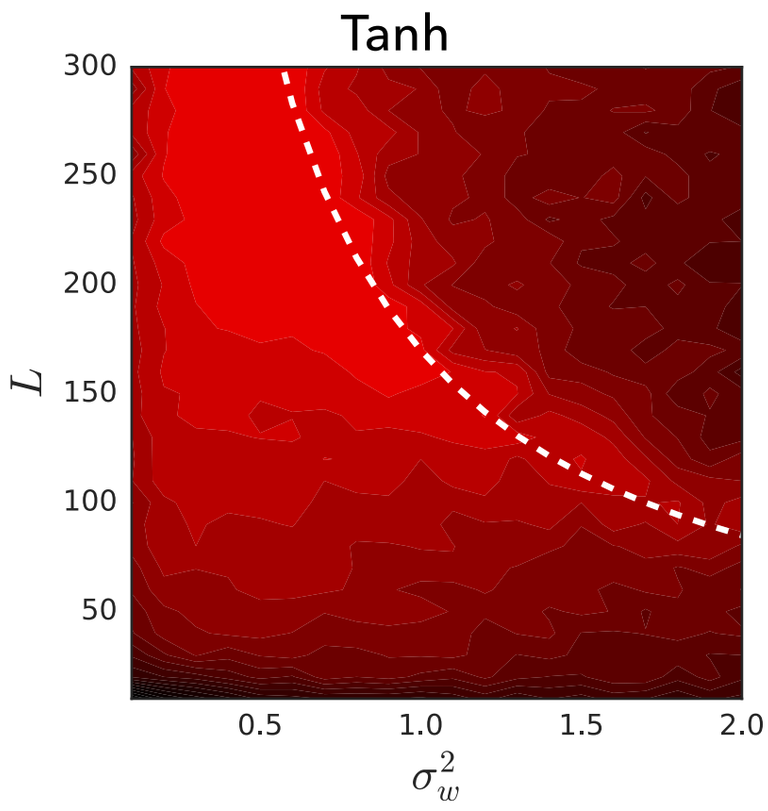
	Tanh/FRN		ReLU/FRN	
q^l	$\Theta(l)$,	B.9	$\exp(\Theta(l))$,	B.16
$c^* - c^l$	$l^{-\delta}$	B.11	$\Theta(l^{-2})$,	B.17
$\tilde{q}^l$	$\exp(\Theta(\sqrt{l}))$,	B.12	$\exp(\Theta(l))$,	B.20

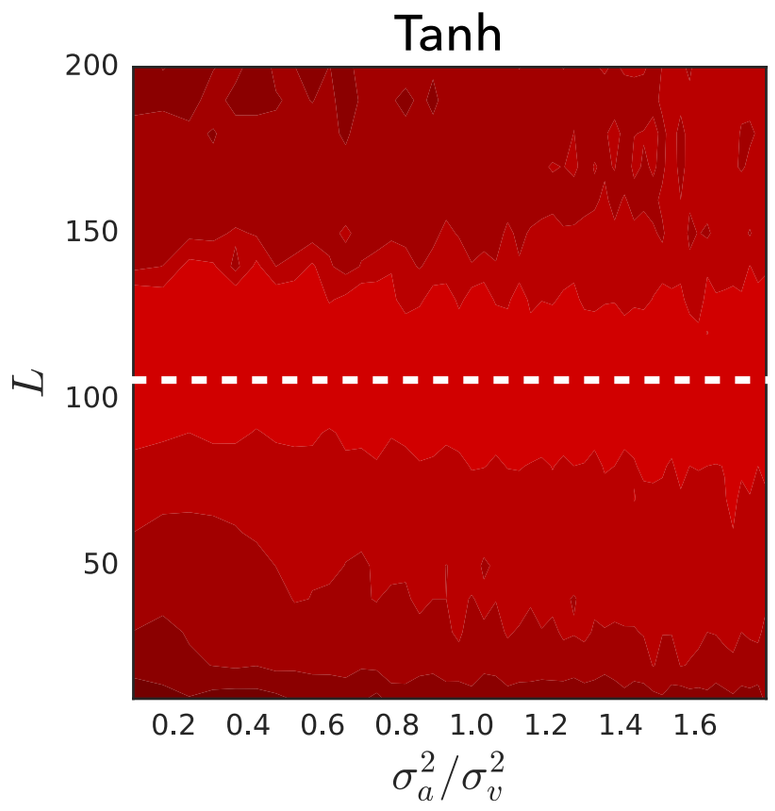
RESIDUAL CONNECTIONS

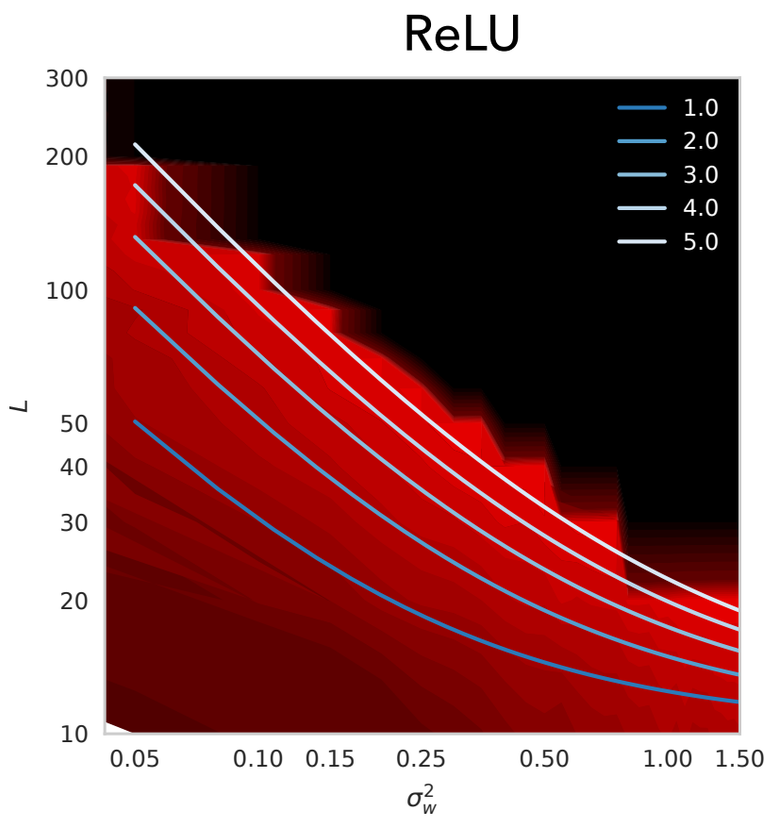
Trainability mostly limited by behavior of gradients

Each plot shows curves of constant $\tilde{q}^0/\tilde{q}^L$

May hint at the utility of batch normalization







BEYOND MEAN FIELD THEORY I

Can we compute finite N corrections?

Start with prior over weights

$$Q = \int [dW][db] \exp\left[-\sum_{l=0}^{L}\left(\frac{N_l}{2\sigma_w^2}\sum_{ij}(W_{ij}^l)^2 + \frac{1}{2\sigma_b^2}\sum_i (b_i^l)^2\right)\right]$$

Change variables by inserting δ - functions

$$Q = \int [dz] \exp\left[-\frac{1}{2}\sum_{l=0}^{L}\left(\sum_i\sum_{ab} z_{ia}^l (\mathbf{\Sigma}^l)_{ab}^{-1} z_{ib}^l + \log|\mathbf{\Sigma}^l|\right)\right]$$

Where

$$\Sigma_{ab}^l = \sum_i \frac{\sigma_w^2}{N_{l-1}}\phi(z_{ia}^{l-1})\phi(z_{ib}^{l-1}) + \sigma_b^2$$

BEYOND MEAN FIELD THEORY I

Can we compute finite N corrections?

Start with prior over weights

$$Q = \int [dW][db] \exp\left[-\sum_{l=0}^{L}\left(\frac{N_l}{2\sigma_w^2}\sum_{ij}(W_{ij}^l)^2 + \frac{1}{2\sigma_b^2}\sum_i (b_i^l)^2\right)\right]$$

Change variables by inserting δ - functions

$$Q = \int [dz] \exp\left[-\frac{1}{2}\sum_{l=0}^{L}\left(\sum_i\sum_{ab} z_{ia}^l (\mathbf{\Sigma}^l)_{ab}^{-1} z_{ib}^l + \log|\mathbf{\Sigma}^l|\right)\right]$$

Where

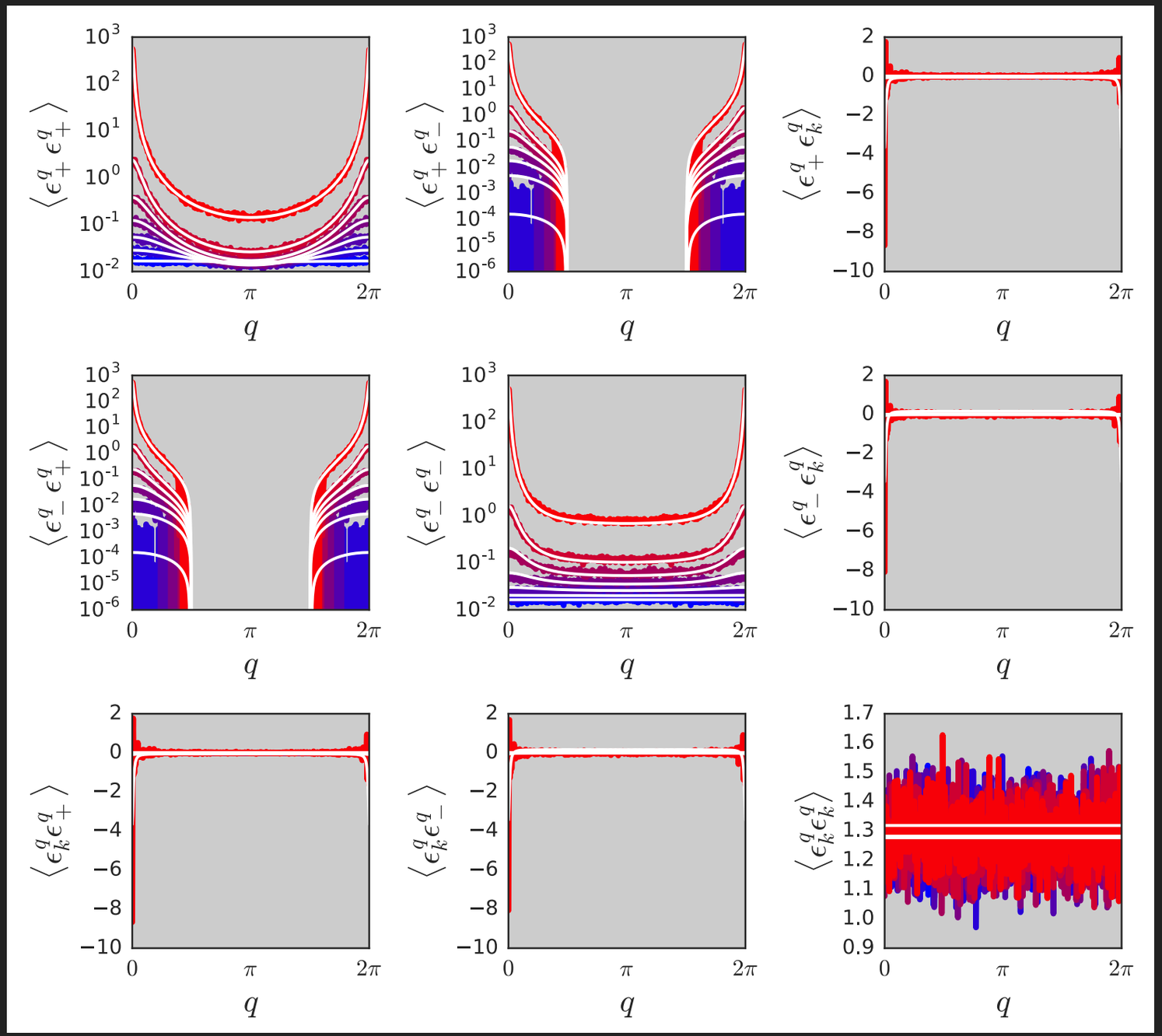
$$\Sigma_{ab}^l = \sum_i \frac{\sigma_w^2}{N_{l-1}} \underbrace{\langle \phi(z_{ia}^{l-1})\phi(z_{ib}^{l-1})\rangle}_{\text{MFT}} + \sigma_b^2$$

BEYOND MEAN FIELD THEORY I

Single example ReLU network can compute saddle point

Well defined EFT

Multiple examples?



BEYOND MEAN FIELD THEORY II

We have worked out behavior of gradients on average

Fluctuations are also important

Study the end-to-end Jacobian

$$\boldsymbol{J} = \frac{\partial \boldsymbol{z}^L}{\partial \boldsymbol{z}^0} = \prod_l \boldsymbol{D}^l \boldsymbol{W}^l \qquad D^l_{ij} = \phi'(z^l_i)\delta_{ij}$$

Diagonal Matrix

A few relations that make this interesting

$$\boldsymbol{\delta}^0 = \boldsymbol{J}\boldsymbol{\delta}^L \qquad f(\boldsymbol{x}+\boldsymbol{\delta}) \approx f(\boldsymbol{x}) + \boldsymbol{J}^T\boldsymbol{\delta} \qquad G = \boldsymbol{J}^T\boldsymbol{J}$$

Gradients | Linear Response | Induced Metric

What is the appropriate prior for $\boldsymbol{J}$? Isometry

The mean field calculation showed that $\mathbb{E}[\mathrm{tr}(\boldsymbol{J}^T\boldsymbol{J})] = \chi_1^L$

JP, SSS, SG ('17); JP SS, SG ('18)

COMPUTING THE SPECTRUM

We can use RMT + Free Probability

$$\rho_X(\lambda) = \left\langle \frac{1}{N} \sum_{i=1}^{N} \delta(\lambda - \lambda_i) \right\rangle_X$$

Define Stieltjes transform

$$G_X(z) \equiv \int_{\boldsymbol{R}} \frac{\rho_X(t)}{z-t} dt \quad \longleftrightarrow \quad \rho_X(\lambda) = -\frac{1}{\pi} \lim_{\epsilon \to 0^+} \mathrm{Im} G_X(\lambda + i\epsilon)$$

Related to moment generating function $M_X(z) \equiv z G_X(z) - 1$

And S-transform

$$S_X(z) = \frac{1+z}{z M_X^{-1}(z)} \longrightarrow S_{AB}(z) = S_A(z) S_B(z)$$

A, B freely independent

JP, SSS, SG ('17); JP SS, SG ('18)

COMPUTING THE SPECTRUM

With our normalization of $q^0 = q^*$

$$S_{JJ^T} = \prod_{l=1}^{L} S_{(D^l)^2} S_{(W^l)^T W^l} = S_{D^2}^L S_{W^T W}^L$$

The spectrum of D^2 is known and $W^T W$ is given

In principal this can be inverted, but there is an easier way

$$M_{JJ^T}(z) = M_{D^2}\left(z^{\frac{1}{L}} F\left(M_{JJ^T}(z)\right)\right)$$

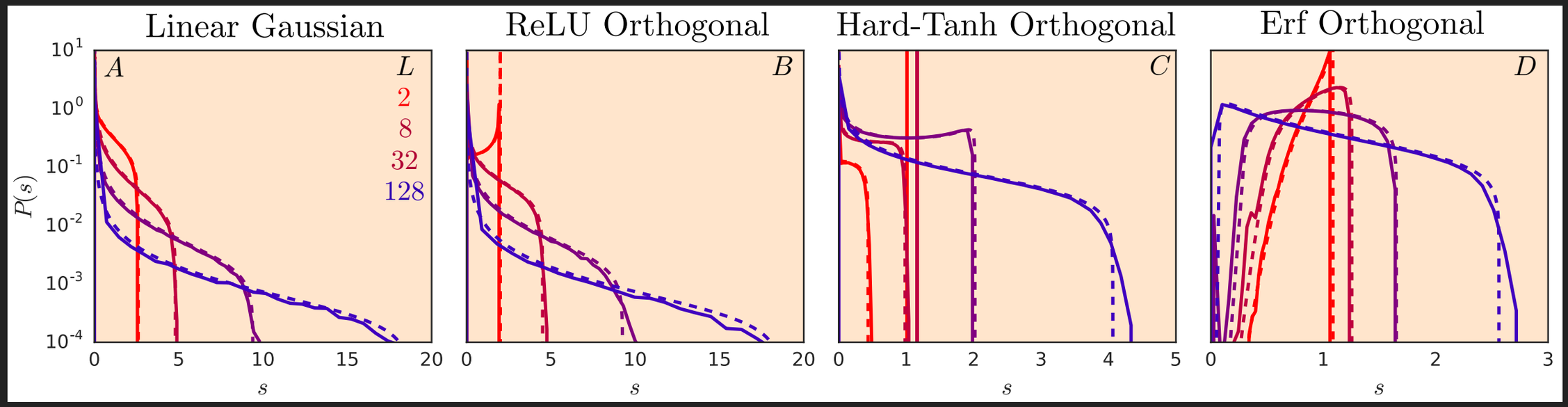
$$zG_{JJ^T}(z) - 1 = M_{D^2}\left(z^{\frac{1}{L}} F\left(zG_{JJ^T}(z) - 1\right)\right)$$

Closed set of functional equations

JP, SSS, SG ('17); JP SS, SG ('18)

COMPUTING THE SPECTRUM

Can solve numerically



Or expanded self-consistently

$$\sigma^2_{JJ^T} = \chi_1^{2L} L \left(\frac{\mu_2}{\mu_1^2} - 1 - s_1 \right)$$

MFT

Statistics of D

Statistics of W

Gaussian W

$s_1 = 0$

Orthogonal W

$s_1 = -1$

JP, SSS, SG ('17); JP SS, SG ('18)

EFFECTS OF THE SPECTRUM

Standard ReLU is never well conditioned

Gaussian W

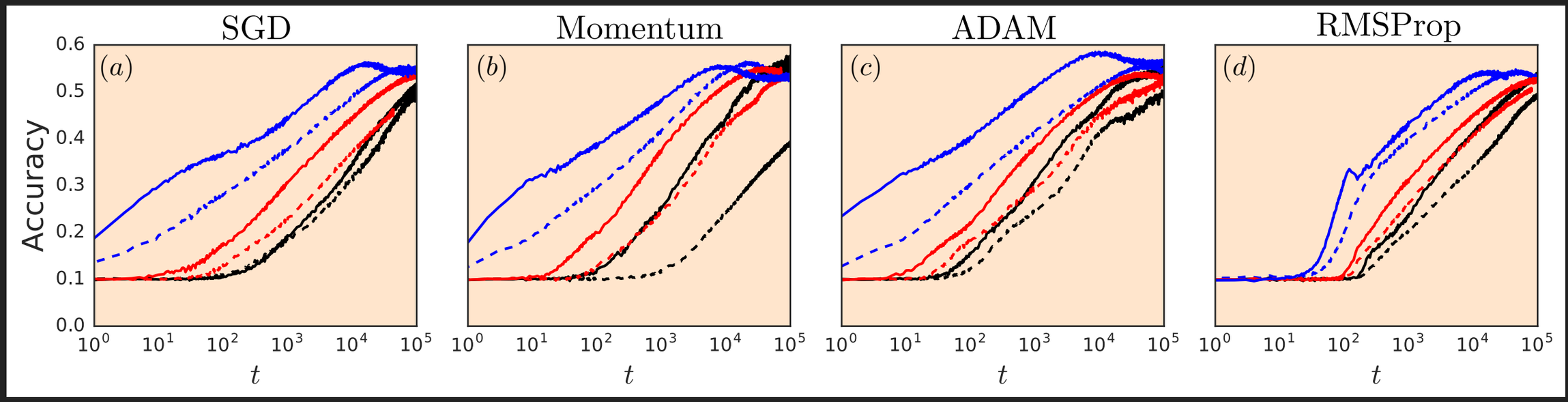
$s_1 = 0$

Orthogonal W

$s_1 = -1$

	$\phi(h)$	$M_{D^2}(z)$	μ_k	σ_w^2	$\sigma_{JJ^T}^2$
Linear	h	$\frac{1}{z-1}$	1	1	$L\left(-s_1\right)$
ReLu	$[h]_+$	$\frac{1}{2}\frac{1}{z-1}$	$\frac{1}{2}$	2	$L\left(1-s_1\right)$
Hard Tanh	$[h+1]_+ - [h-1]_+ - 1$	$\mathrm{erf}(\frac{1}{\sqrt{2q^*}})\frac{1}{z-1}$	$\mathrm{erf}(\frac{1}{\sqrt{2q^*}})$	$\frac{1}{\mathrm{erf}(\frac{1}{\sqrt{2q^*}})}$	$L\left(\frac{1}{\mathrm{erf}(\frac{1}{\sqrt{2q^*}})}-1-s_1\right)$
Erf	$\mathrm{erf}(\frac{\sqrt{\pi}}{2}h)$	$\frac{1}{\sqrt{\pi q^* z}}\Phi\left(\frac{1}{z},\frac{1}{2},\frac{1+\pi q_*}{\pi q_*}\right)$	$\frac{1}{\sqrt{1+\pi k q_*}}$	$\sqrt{1+\pi q^*}$	$L\left(\frac{1+\pi q^*}{\sqrt{1+2\pi q^*}}-1-s_1\right)$

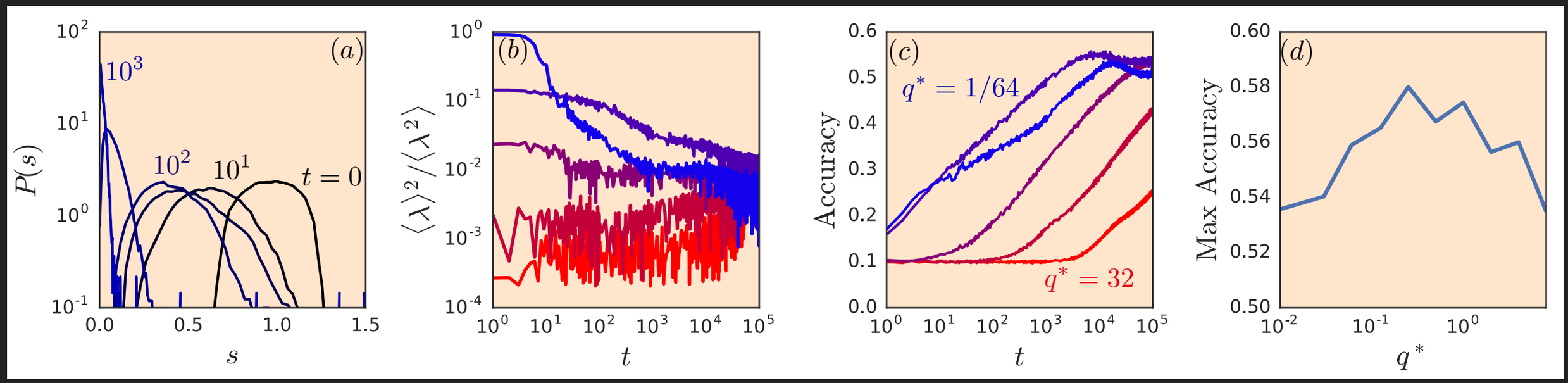
Optimizer independent effect on training



JP, SSS, SG ('17); JP SS, SG ('18)

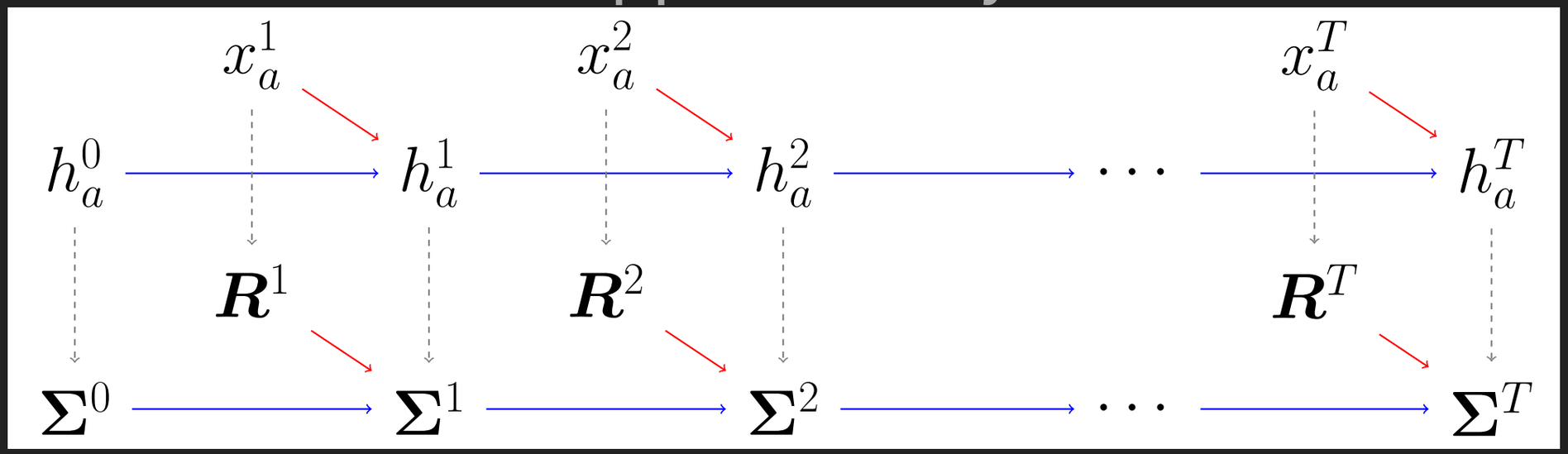
EFFECTS OF THE SPECTRUM

Conditioning effects generalization



(GATED) RECURRENT NETWORKS

Theory can be extended (approximately) to RNNs

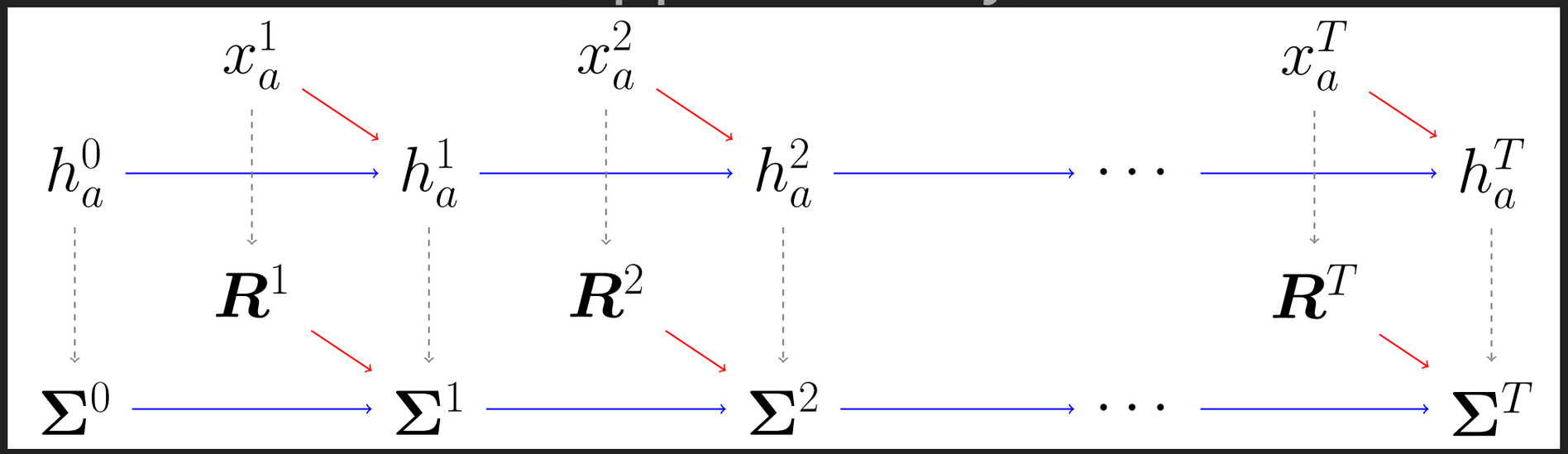


Two major differences from feed-forward case:

1. Weights are shared between layers (violates independence)
2. Inputs are fed into the network at each step (no fixed point)

(GATED) RECURRENT NETWORKS

Theory can be extended (approximately) to RNNs



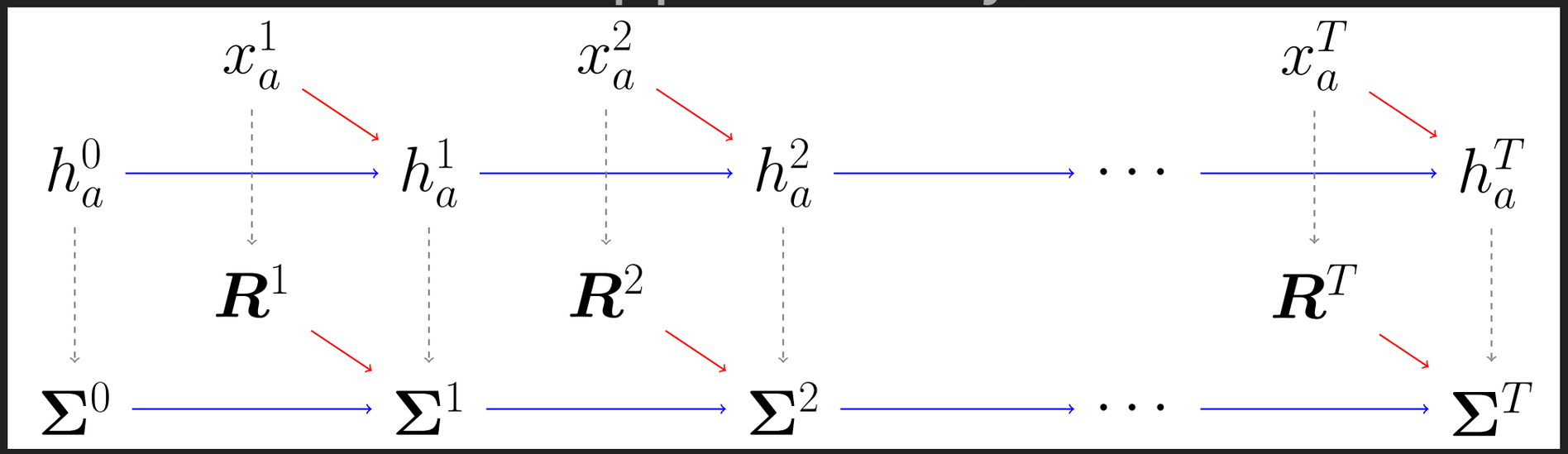
Two major differences from feed-forward case:

1. Weights are shared between layers
   (violates independence) Ignore it (use AMP trick, maybe)

2. Inputs are fed into the network at each step
   (no fixed point)

(GATED) RECURRENT NETWORKS

Theory can be extended (approximately) to RNNs



Two major differences from feed-forward case:

1. Weights are shared between layers
   (violates independence) Ignore it (use AMP trick, maybe)

2. Inputs are fed into the network at each step
   (no fixed point) Consider steady state $R^t = R$

(GATED) RECURRENT NETWORKS

What does steady state correspond to?

Imagine two copies of network that start off with different "memories"

Input random noise and measure how long state persists

As in feed-forward networks $c^* - c^t \sim e^{-t/\tau}$ $\tau = -\frac{1}{\log \chi_{c^*}}$

h_{t-1}

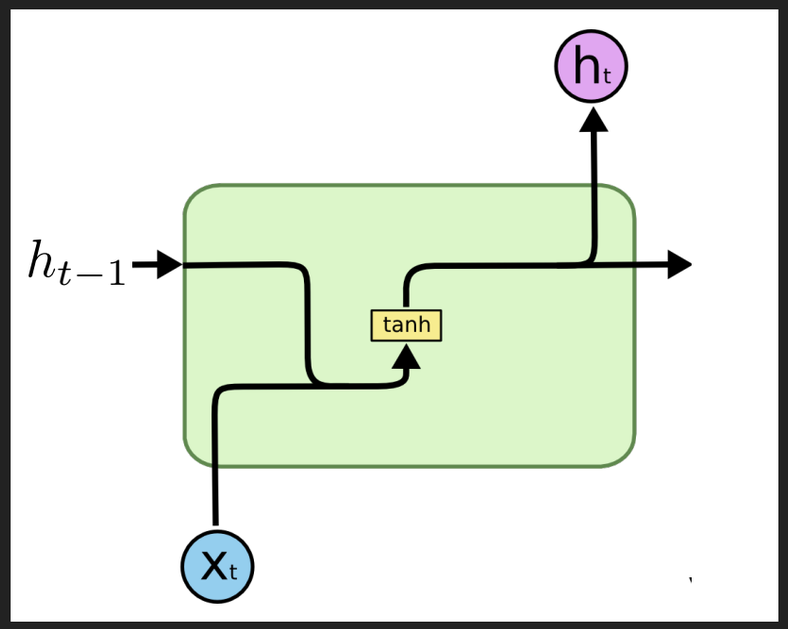
(GATED) RECURRENT NETWORKS

What does steady state correspond to?

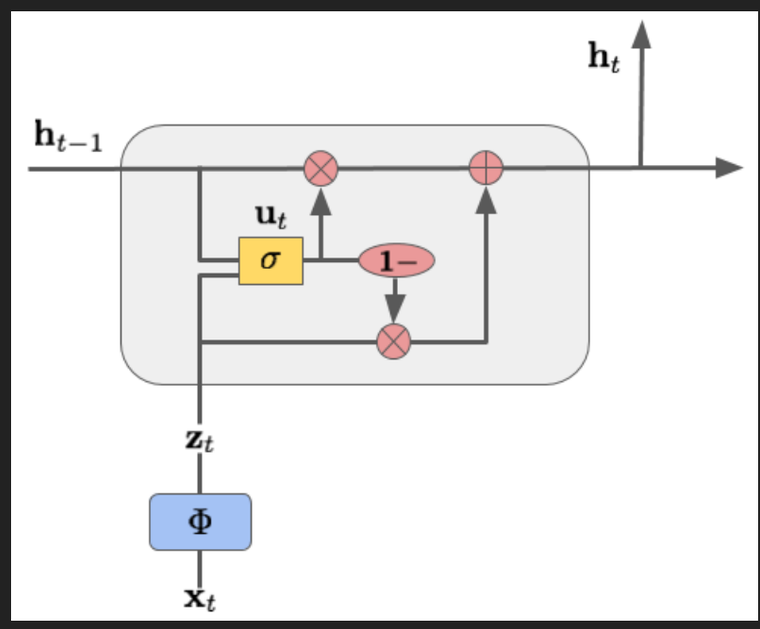
Imagine two copies of network that start off with different "memories"

Input random noise and measure how long state persists

As in feed-forward networks $c^* - c^t \sim e^{-t/\tau} \quad \tau = -\frac{1}{\log \chi_{c^*}}$



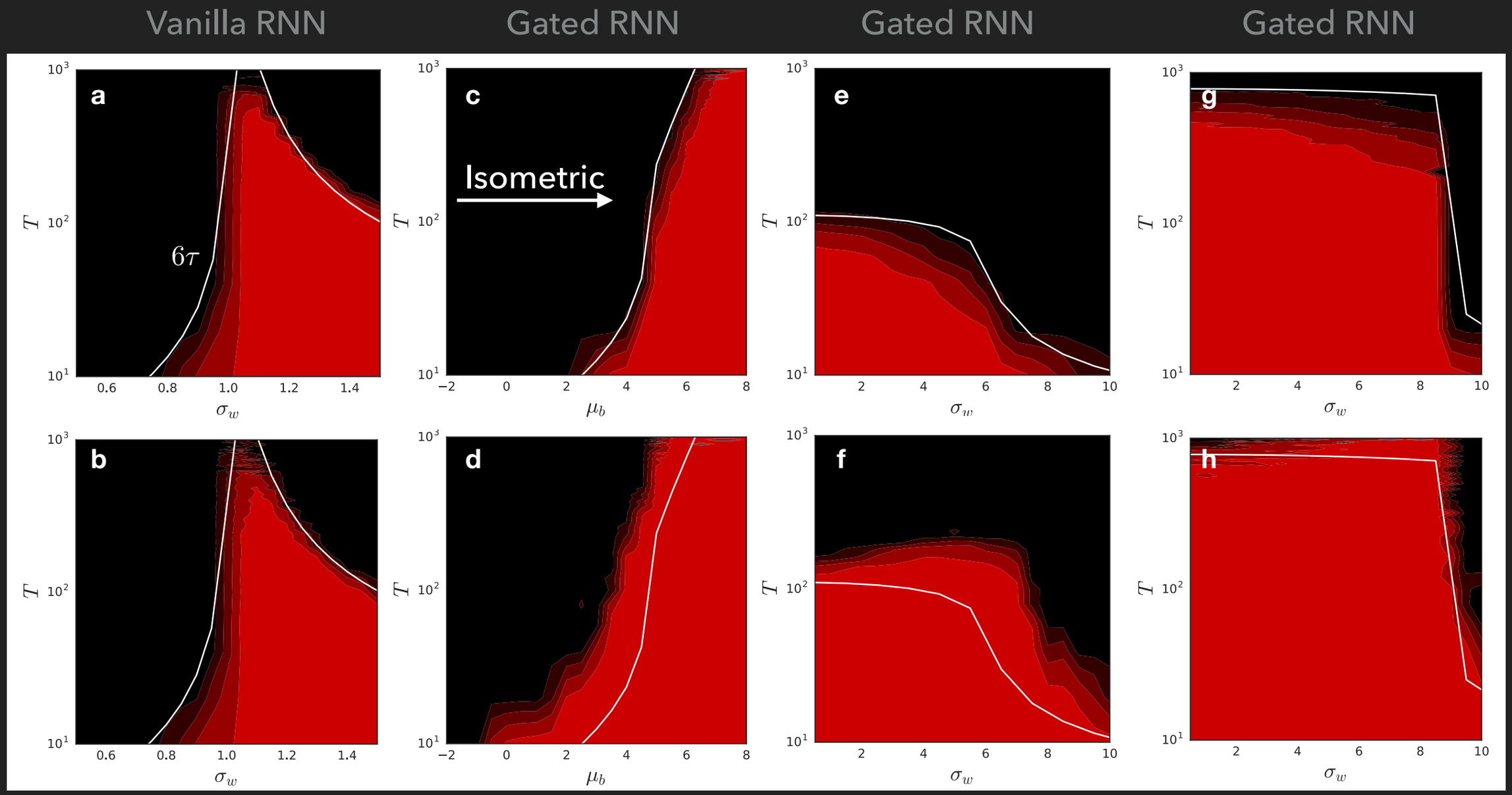
Vanilla RNN



Gated RNN

(GATED) RECURRENT NETWORKS

Test on toy task: feed MNIST digit at t = 0, then noise for T steps



Gated architectures have larger trainable volume

CONVOLUTIONAL NETWORKS

Convolutional networks have filters $W^l \in \mathbb{R}^{(2k+1)\times c\times c}$

$$h_c^l(x) = \sum_{\delta x}\sum_{c'} W_{cc'}^l(\delta x)\phi(h_{c'}^{l-1}(x+\delta x)) + b_c^l$$

As channel count get large

$$\Sigma^{l+1}(x,x') = \mathcal{A}\star\boxed{\left(\sigma_w^2\int \mathcal{D}_{\Sigma^l(x,x')}\phi(z)\phi(z)^T + \sigma_b^2\right)}$$

Fully Connected

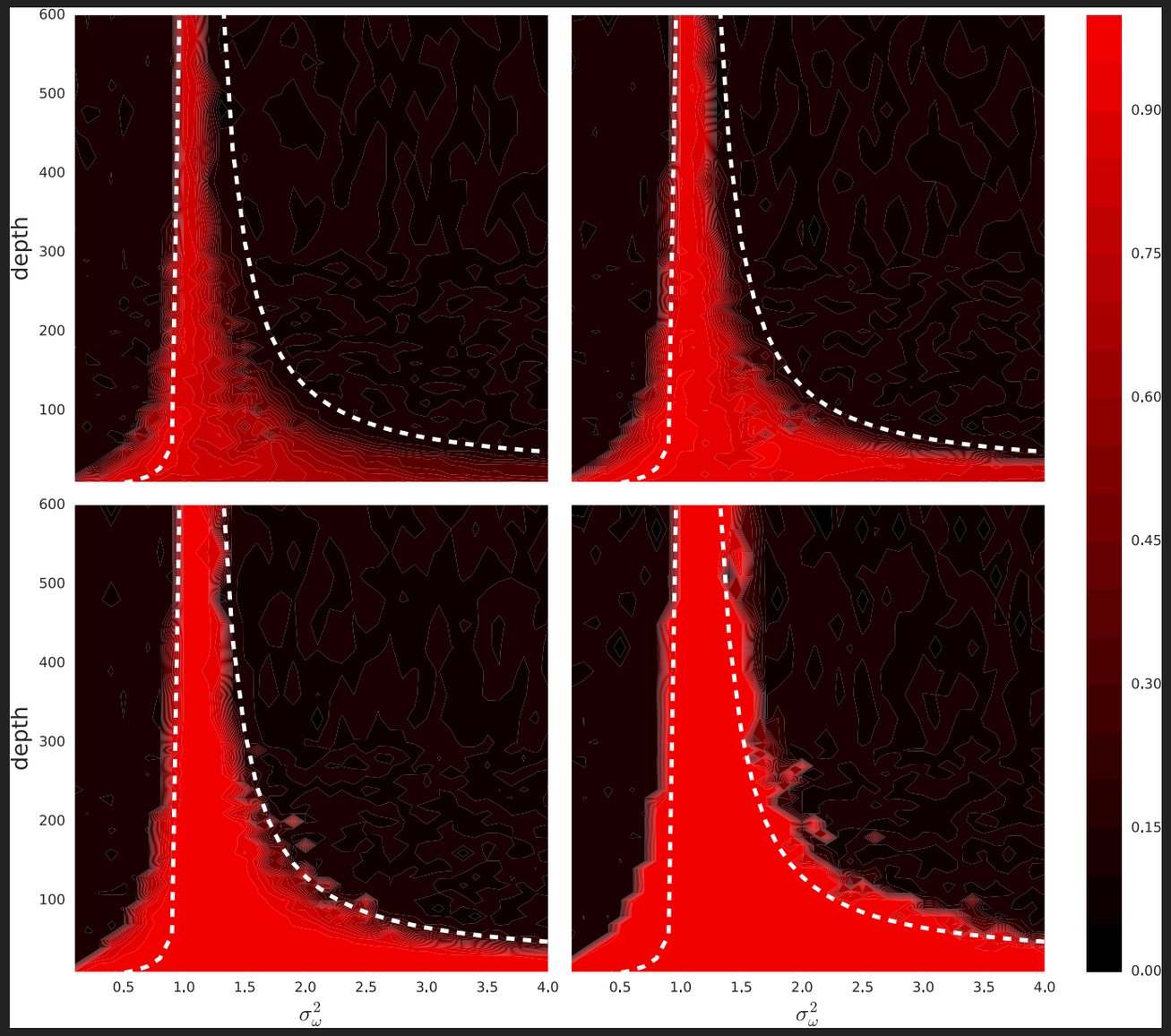
Dynamics near fixed point diagonalized by Fourier transform

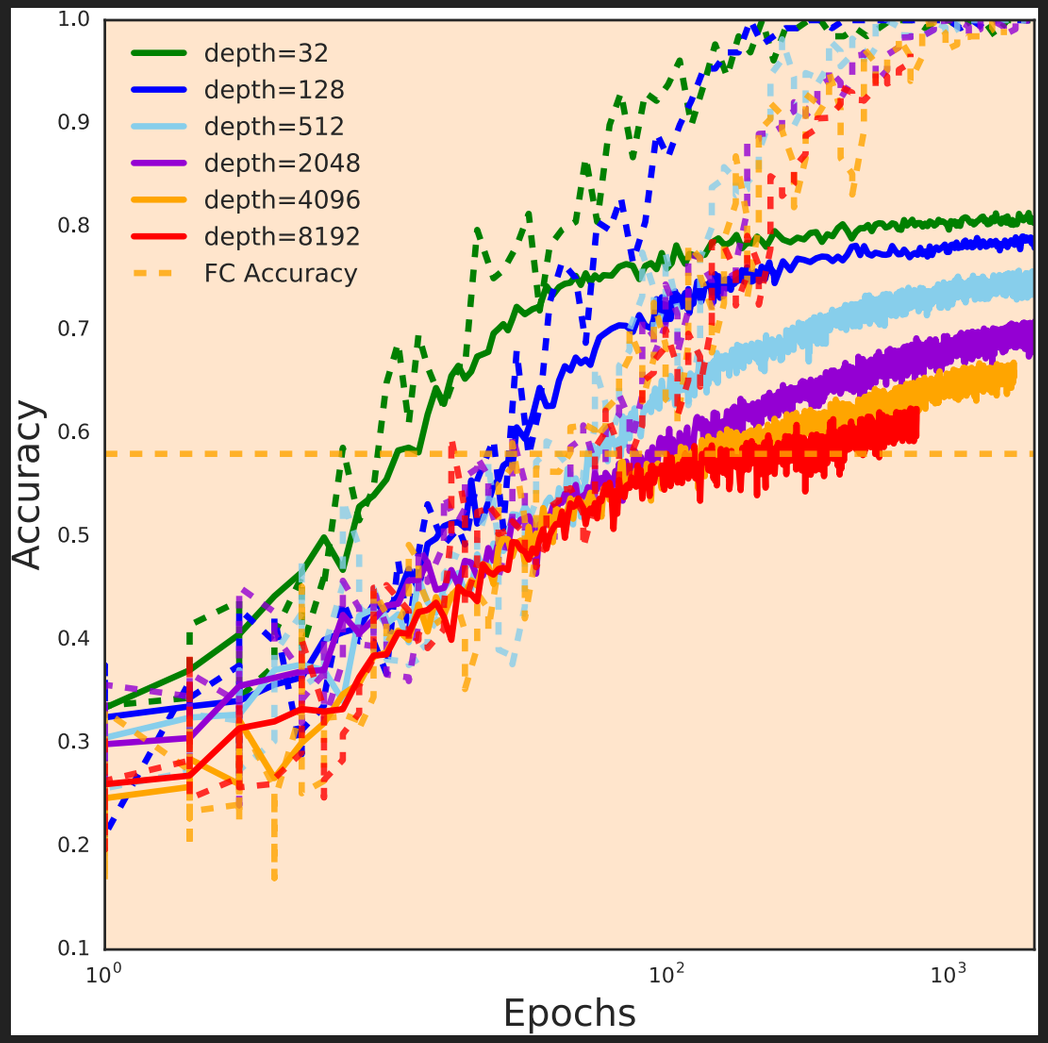
$$\tilde{\epsilon}_{ff'}^{l+1} \sim (\lambda_{ff'}\chi_{c^*})^l \qquad \lambda_{ff'} \leq 1$$

Determined by A

CONVOLUTIONAL NETWORKS

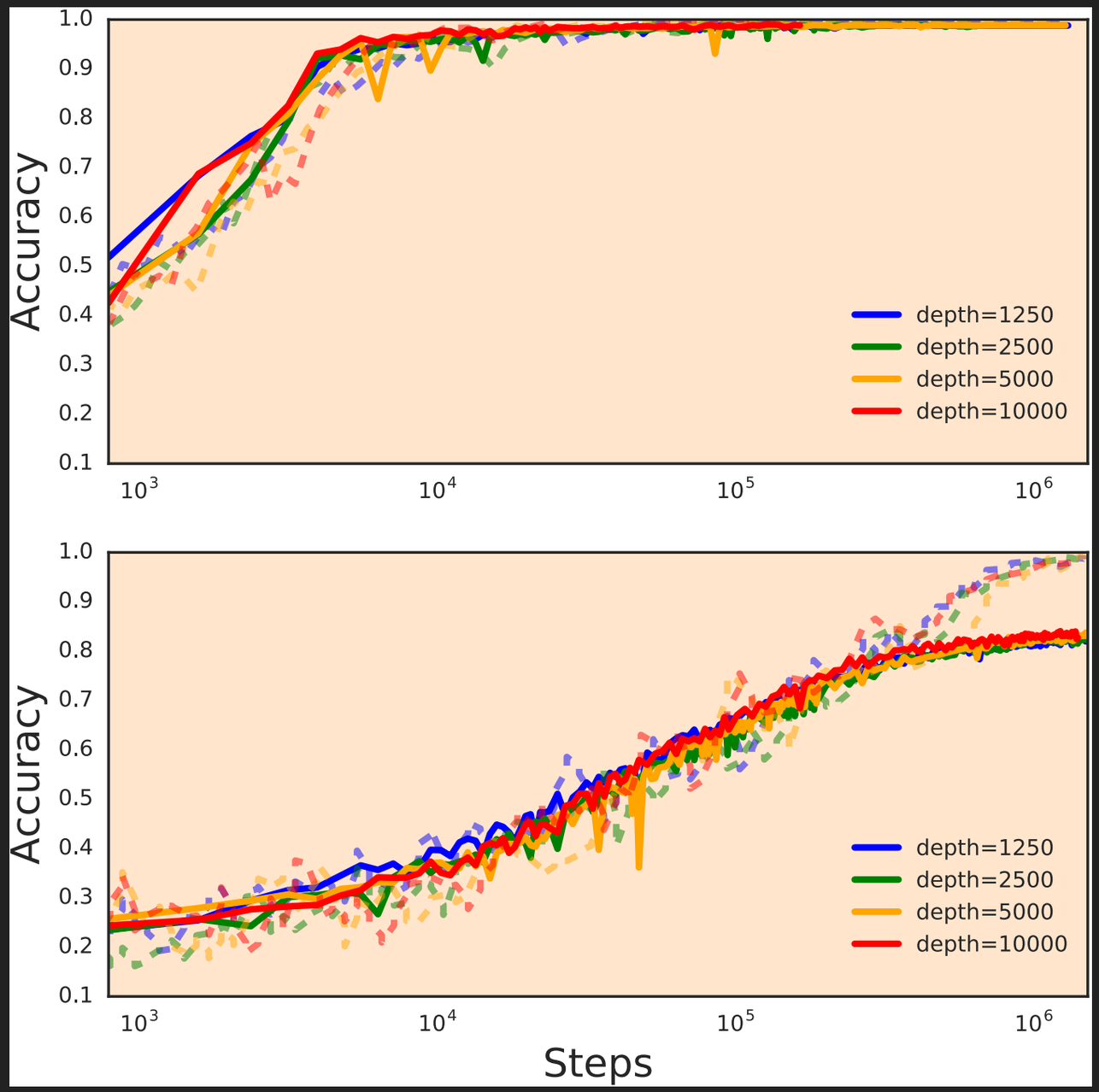
Naive initialization loses spatial information at large depth





CONVOLUTIONAL NETWORKS

Exists initialization that is isometric in all frequencies



CONCLUSIONS

We can say a lot about networks by understanding their priors

- Predict network trainability
- Choose fast initialization schemes

Disentangles trainability from model performance

Many different conditions that must be satisfied

Make deep learning (slightly) more principled

Physicists have a lot of tools at our disposal to do this