Entropy and mutual information in models of deep neural networks

NeurIPS 2018 - Thursday Dec 06th - Spotlight

Poster @ Room 210 & 230 AB #110

Marylou Gabrié (LPS ENS), Andre Manoel (INRIA Saclay, Owkin),
Clément Luneau, Jean Barbier, Nicolas Macris (EPFL),
Lenka Zdeborová (CEA Saclay), Florent Krzakala (LPS ENS)

Motivations

Information theoretic arguments to deep learning

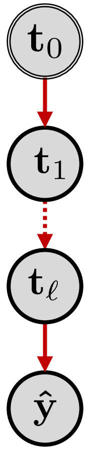
- Theory of generalization: Shwartz-Ziv et al. 2017, Saxe et al. 2018, Goldfeld et al. 2018 etc.
- New regularizer: Chalk et al. 2016, Alemi et al. 2017, Kolchinsky et al. 2017, Belghazi et al. 2017, Zhao et al. 2018, Achille et al. 2018, Hjelm et al. 2018, etc.

Motivations

Information theoretic arguments to deep learning

- Theory of generalization: Shwartz-Ziv et al. 2017, Saxe et al. 2018, Goldfeld et al. 2018 etc.
- New regularizer: Chalk et al. 2016, Alemi et al. 2017, Kolchinsky et al. 2017, Belghazi et al. 2017, Zhao et al. 2018, Achille et al. 2018, Hjelm et al. 2018, etc.

Computing mutual informations in neural networks …

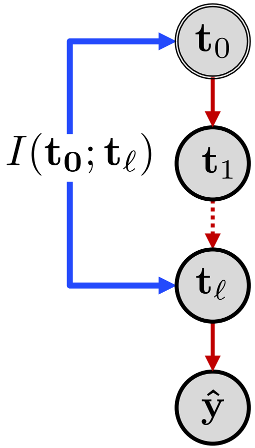


Motivations

Information theoretic arguments to deep learning

- Theory of generalization: Shwartz-Ziv et al. 2017, Saxe et al. 2018, Goldfeld et al. 2018 etc.
- New regularizer: Chalk et al. 2016, Alemi et al. 2017, Kolchinsky et al. 2017, Belghazi et al. 2017, Zhao et al. 2018, Achille et al. 2018, Hjelm et al. 2018, etc.

Computing mutual informations in neural networks …

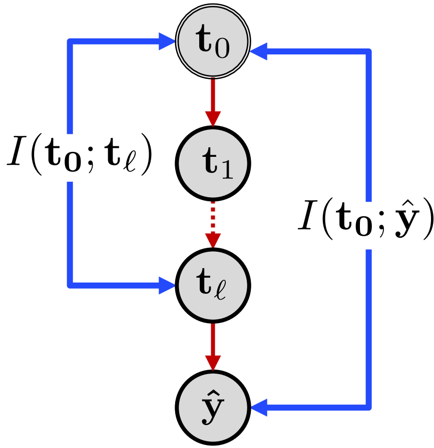


Motivations

Information theoretic arguments to deep learning

- Theory of generalization: Shwartz-Ziv et al. 2017, Saxe et al. 2018, Goldfeld et al. 2018 etc.
- New regularizer: Chalk et al. 2016, Alemi et al. 2017, Kolchinsky et al. 2017, Belghazi et al. 2017, Zhao et al. 2018, Achille et al. 2018, Hjelm et al. 2018, etc.

Computing mutual informations in neural networks …



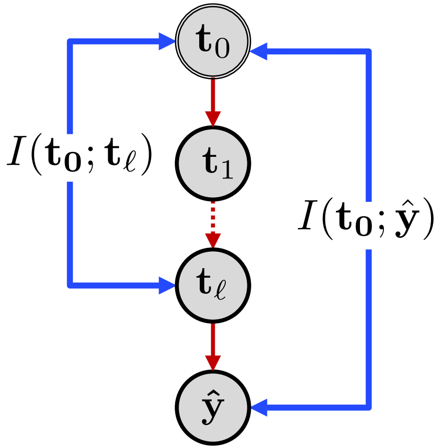
Motivations

Information theoretic arguments to deep learning

- Theory of generalization: Shwartz-Ziv et al. 2017, Saxe et al. 2018, Goldfeld et al. 2018 etc.
- New regularizer: Chalk et al. 2016, Alemi et al. 2017, Kolchinsky et al. 2017, Belghazi et al. 2017, Zhao et al. 2018, Achille et al. 2018, Hjelm et al. 2018, etc.

Computing mutual informations in neural networks … is tricky

- Intractable $p(\mathbf{t}_\ell) = \int \mathrm{d}\mathbf{t}_0 \mathrm{d}\mathbf{t}_1 \cdots \mathrm{d}\mathbf{t}_{\ell-1}\, p(\mathbf{t}_\ell|\mathbf{t}_{\ell-1}) \cdots p(\mathbf{t}_1|\mathbf{t}_0) p(\mathbf{t}_0)$



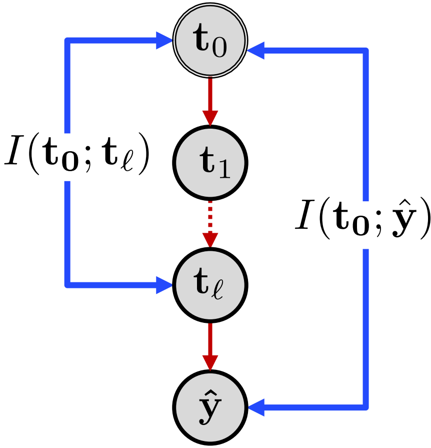
Motivations

Information theoretic arguments to deep learning

- Theory of generalization: Shwartz-Ziv et al. 2017, Saxe et al. 2018, Goldfeld et al. 2018 etc.
- New regularizer: Chalk et al. 2016, Alemi et al. 2017, Kolchinsky et al. 2017, Belghazi et al. 2017, Zhao et al. 2018, Achille et al. 2018, Hjelm et al. 2018, etc.

Computing mutual informations in neural networks … is tricky

- Intractable $p(\mathbf{t}_\ell) = \int \mathrm{d}\mathbf{t}_0 \mathrm{d}\mathbf{t}_1 \cdots \mathrm{d}\mathbf{t}_{\ell-1}\, p(\mathbf{t}_\ell|\mathbf{t}_{\ell-1}) \cdots p(\mathbf{t}_1|\mathbf{t}_0) p(\mathbf{t}_0)$
- Computing $I(\mathbf{t}_0;\mathbf{t}_\ell)$ or $H(\mathbf{t}_\ell) = -\int \mathrm{d}\mathbf{t}_\ell\, p(\mathbf{t}_\ell) \log p(\mathbf{t}_\ell)$ yet more involved



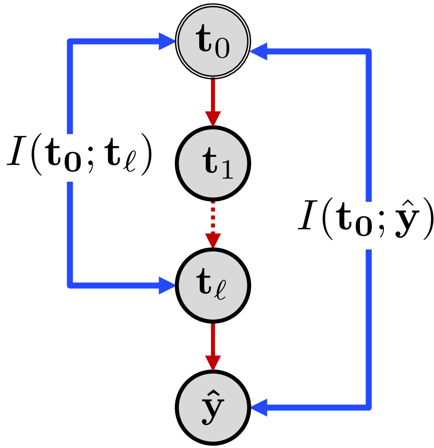
Motivations

Information theoretic arguments to deep learning

- Theory of generalization: Shwartz-Ziv et al. 2017, Saxe et al. 2018, Goldfeld et al. 2018 etc.
- New regularizer: Chalk et al. 2016, Alemi et al. 2017, Kolchinsky et al. 2017, Belghazi et al. 2017, Zhao et al. 2018, Achille et al. 2018, Hjelm et al. 2018, etc.

Computing mutual informations in neural networks … is tricky

- Intractable $p(\mathbf{t}_\ell) = \int \mathrm{d}\mathbf{t}_0 \mathrm{d}\mathbf{t}_1 \cdots \mathrm{d}\mathbf{t}_{\ell-1}\, p(\mathbf{t}_\ell|\mathbf{t}_{\ell-1}) \cdots p(\mathbf{t}_1|\mathbf{t}_0) p(\mathbf{t}_0)$
- Computing $I(\mathbf{t}_0; \mathbf{t}_\ell)$ or $H(\mathbf{t}_\ell) = -\int \mathrm{d}\mathbf{t}_\ell\, p(\mathbf{t}_\ell) \log p(\mathbf{t}_\ell)$ yet more involved



Sampled based estimations (non-parametric, variational methods)

Kraskov et al. 2014, Kolchinsky et al. 2017, Belghazi et al. 2017, Goldfeld et al. 2018 etc.

➔ Less and less reliable as networks get large

➔ Need for benchmark controlled case

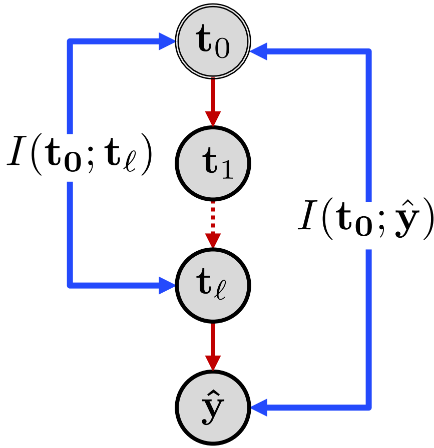
Motivations

Information theoretic arguments to deep learning

- Theory of generalization: Shwartz-Ziv et al. 2017, Saxe et al. 2018, Goldfeld et al. 2018 etc.
- New regularizer: Chalk et al. 2016, Alemi et al. 2017, Kolchinsky et al. 2017, Belghazi et al. 2017, Zhao et al. 2018, Achille et al. 2018, Hjelm et al. 2018, etc.

Computing mutual informations in neural networks … is tricky

- Intractable $p(\mathbf{t}_\ell) = \int \mathrm{d}\mathbf{t}_0 \mathrm{d}\mathbf{t}_1 \cdots \mathrm{d}\mathbf{t}_{\ell-1}\, p(\mathbf{t}_\ell|\mathbf{t}_{\ell-1}) \cdots p(\mathbf{t}_1|\mathbf{t}_0) p(\mathbf{t}_0)$
- Computing $I(\mathbf{t}_0; \mathbf{t}_\ell)$ or $H(\mathbf{t}_\ell) = -\int \mathrm{d}\mathbf{t}_\ell\, p(\mathbf{t}_\ell) \log p(\mathbf{t}_\ell)$ yet more involved



Sampled based estimations (non-parametric, variational methods)

Kraskov et al. 2014, Kolchinsky et al. 2017, Belghazi et al. 2017, Goldfeld et al. 2018 etc.

➔ Less and less reliable as networks get large

➔ Need for benchmark controlled case

Here 1) Restrict to **models** of DNNS, 2) Leverage statistical physics **replica method**

Model: fully connected feed forward neural network with stochastic activations

$$\overset{\mathbb{R}^{N_\ell}}{\mathbf{t}_\ell} = f_\epsilon(\mathbf{W}^{(\ell)} f_\epsilon(\mathbf{W}^{(\ell-1)} \cdots f_\epsilon(\mathbf{W}^{(1)} \overset{\mathbb{R}^{N_0}}{\mathbf{t}_0}) \cdots))$$

Model: fully connected feed forward neural network with stochastic activations

$$\underset{\mathbb{R}^{N_\ell}}{\mathbf{t}_\ell} = f_\epsilon(\mathbf{W}^{(\ell)} f_\epsilon(\mathbf{W}^{(\ell-1)} \cdots f_\epsilon(\mathbf{W}^{(1)} \overset{\mathbb{R}^{N_0}}{\mathbf{t}_0}) \cdots))$$

With :

Model: fully connected feed forward neural network with stochastic activations

$$\overset{\mathbb{R}^{N_\ell}}{\mathbf{t}_\ell} = f_\epsilon(\mathbf{W}^{(\ell)} f_\epsilon(\mathbf{W}^{(\ell-1)} \cdots f_\epsilon(\mathbf{W}^{(1)} \overset{\mathbb{R}^{N_0}}{\mathbf{t}_0}) \cdots))$$

With :

1) known factorized input distribution $P_0(\mathbf{t_0}) = \prod_{i=1}^{N_0} P_0(t_{0i})$

Model: fully connected feed forward neural network with stochastic activations

$$\overset{\mathbb{R}^{N_\ell}}{\mathbf{t}_\ell} = f_\epsilon(\mathbf{W}^{(\ell)} f_\epsilon(\mathbf{W}^{(\ell-1)} \cdots f_\epsilon(\mathbf{W}^{(1)} \overset{\mathbb{R}^{N_0}}{\mathbf{t}_0}) \cdots))$$

With :

1) known factorized input distribution $P_0(\mathbf{t_0}) = \prod_{i=1}^{N_0} P_0(t_{0i})$

2) weight matrices from class of random matrix class

Model: fully connected feed forward neural network with stochastic activations

$$\overset{\mathbb{R}^{N_\ell}}{\mathbf{t}_\ell} = f_\epsilon(\mathbf{W}^{(\ell)} f_\epsilon(\mathbf{W}^{(\ell-1)} \cdots f_\epsilon(\mathbf{W}^{(1)} \overset{\mathbb{R}^{N_0}}{\mathbf{t}_0}) \cdots))$$

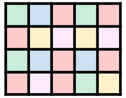
With :

1) known factorized input distribution $P_0(\mathbf{t_0}) = \prod_{i=1}^{N_0} P_0(t_{0i})$

2) weight matrices from class of random matrix class

i.i.d Gaussian

$W^{(\ell)}$



Model: fully connected feed forward neural network with stochastic activations

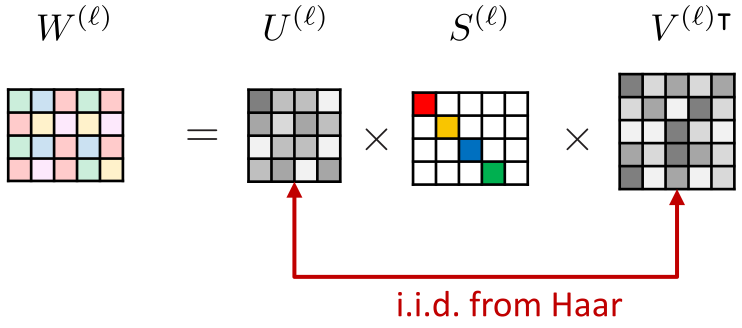
$$\underset{\mathbb{R}^{N_\ell}}{\mathbf{t}_\ell} = f_\epsilon(\mathbf{W}^{(\ell)} f_\epsilon(\mathbf{W}^{(\ell-1)} \cdots f_\epsilon(\mathbf{W}^{(1)} \underset{\mathbb{R}^{N_0}}{\mathbf{t}_0}) \cdots))$$

With :

1) known factorized input distribution $P_0(\mathbf{t_0}) = \prod_{i=1}^{N_0} P_0(t_{0i})$

2) weight matrices from class of random matrix class

i.i.d Gaussian ➔ Orthogonally invariant



Model: fully connected feed forward neural network with stochastic activations

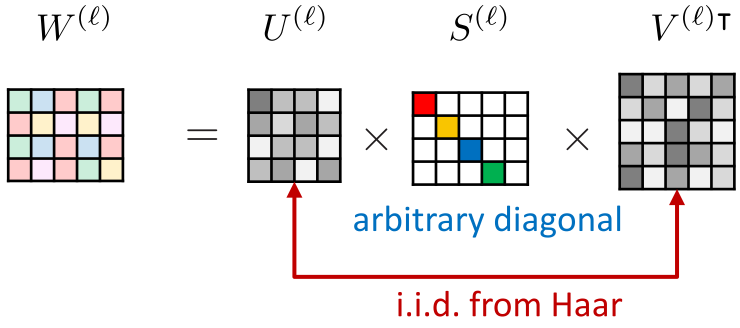
$$\underset{\mathbb{R}^{N_\ell}}{\mathbf{t}_\ell} = f_\epsilon(\mathbf{W}^{(\ell)} f_\epsilon(\mathbf{W}^{(\ell-1)} \cdots f_\epsilon(\mathbf{W}^{(1)} \underset{\mathbb{R}^{N_0}}{\mathbf{t}_0}) \cdots))$$

With :

1) known factorized input distribution $P_0(\mathbf{t_0}) = \prod_{i=1}^{N_0} P_0(t_{0i})$

2) weight matrices from class of random matrix class

i.i.d Gaussian ➔ Orthogonally invariant



Main results

1) Formula: $\lim_{N_0\to\infty} \frac{1}{N_0} H(\mathbf{t}_\ell) = \min_{\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}} \operatorname{extr} \phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}})$

Main results

1) Formula: $\lim_{N_0 \to \infty} \frac{1}{N_0} H(\mathbf{t}_\ell) = \min \operatorname*{extr}_{\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}} \phi_\ell(\mathbf{A}, \mathbf{V}, \tilde{\mathbf{A}}, \tilde{\mathbf{V}})$

Main results

1) Formula: $$\lim_{N_0\to\infty} \frac{1}{N_0} H(\mathbf{t}_\ell) = \min_{\mathbf{A},\mathbf{V}} \operatorname*{extr}_{\tilde{\mathbf{A}},\tilde{\mathbf{V}}} \phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}})$$

potential: $$\phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}) = I\left(t_0; t_0 + \frac{\xi_0}{\sqrt{\tilde{A}_1}}\right) - \frac{1}{2}\sum_{k=1}^{\ell} \tilde{\alpha}_{k-1}\left[\tilde{A}_k V_k + \alpha_k A_k \tilde{V}_k - F_{W_k}(A_k V_k)\right]$$
$$+ \sum_{k=1}^{\ell-1} \tilde{\alpha}_k \left[H(t_k|\xi_k; \tilde{A}_{k+1}, \tilde{V}_k, \tilde{\rho}_k) - \frac{1}{2}\log(2\pi e \tilde{A}_{k+1}^{-1})\right] + \tilde{\alpha}_\ell H(t_\ell|\xi_\ell; \tilde{V}_\ell, \tilde{\rho}_\ell)$$

Main results

1) Formula: $$\lim_{N_0\to\infty} \frac{1}{N_0} H(\mathbf{t}_\ell) = \min \operatorname*{extr}_{\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}} \phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}})$$

potential: $$\phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}) = I\left(t_0; t_0 + \frac{\xi_0}{\sqrt{\tilde{A}_1}}\right) - \frac{1}{2}\sum_{k=1}^{\ell} \tilde{\alpha}_{k-1}\left[\tilde{A}_k V_k + \alpha_k A_k \tilde{V}_k - F_{W_k}(A_k V_k)\right]$$
$$+ \sum_{k=1}^{\ell-1} \tilde{\alpha}_k \left[H(t_k|\xi_k; \tilde{A}_{k+1}, \tilde{V}_k, \tilde{\rho}_k) - \frac{1}{2}\log(2\pi e \tilde{A}_{k+1}^{-1})\right] + \tilde{\alpha}_\ell H(t_\ell|\xi_\ell; \tilde{V}_\ell, \tilde{\rho}_\ell)$$

Main results

1) Formula: $$\lim_{N_0\to\infty} \frac{1}{N_0} H(\mathbf{t}_\ell) = \min \operatorname*{extr}_{\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}} \phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}})$$

potential: $$\phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}) = I\left(t_0; t_0 + \frac{\xi_0}{\sqrt{\tilde{A}_1}}\right) - \frac{1}{2}\sum_{k=1}^{\ell} \tilde{\alpha}_{k-1}\left[\tilde{A}_k V_k + \alpha_k A_k \tilde{V}_k - F_{W_k}(A_k V_k)\right]$$
$$+ \sum_{k=1}^{\ell-1} \tilde{\alpha}_k \left[H(t_k|\xi_k; \tilde{A}_{k+1}, \tilde{V}_k, \tilde{\rho}_k) - \frac{1}{2}\log(2\pi e \tilde{A}_{k+1}^{-1})\right] + \tilde{\alpha}_\ell H(t_\ell|\xi_\ell; \tilde{V}_\ell, \tilde{\rho}_\ell)$$

Main results

1) Formula: $$\lim_{N_0\to\infty} \frac{1}{N_0} H(\mathbf{t}_\ell) = \min \operatorname*{extr}_{\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}} \phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}})$$

potential: $$\phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}) = I\left(t_0; t_0 + \frac{\xi_0}{\sqrt{\tilde{A}_1}}\right) - \frac{1}{2}\sum_{k=1}^{\ell} \tilde{\alpha}_{k-1}\left[\tilde{A}_k V_k + \alpha_k A_k \tilde{V}_k - F_{W_k}(A_k V_k)\right]$$
$$+ \sum_{k=1}^{\ell-1} \tilde{\alpha}_k \left[H(t_k|\xi_k; \tilde{A}_{k+1}, \tilde{V}_k, \tilde{\rho}_k) - \frac{1}{2}\log(2\pi e \tilde{A}_{k+1}^{-1})\right] + \tilde{\alpha}_\ell H(t_\ell|\xi_\ell; \tilde{V}_\ell, \tilde{\rho}_\ell)$$

efficient implementation of extremization: sphinxteam / **dnner**

Main results

1) Formula:

$$\lim_{N_0\to\infty} \frac{1}{N_0} H(\mathbf{t}_\ell) = \min \operatorname*{extr}_{\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}} \phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}})$$

potential: $$\phi_\ell(\mathbf{A},\mathbf{V},\tilde{\mathbf{A}},\tilde{\mathbf{V}}) = I\left(t_0; t_0 + \frac{\xi_0}{\sqrt{\tilde{A}_1}}\right) - \frac{1}{2}\sum_{k=1}^{\ell} \tilde{\alpha}_{k-1}\left[\tilde{A}_k V_k + \alpha_k A_k \tilde{V}_k - F_{W_k}(A_k V_k)\right] + \sum_{k=1}^{\ell-1} \tilde{\alpha}_k \left[H(t_k|\xi_k; \tilde{A}_{k+1}, \tilde{V}_k, \tilde{\rho}_k) - \frac{1}{2}\log(2\pi e \tilde{A}_{k+1}^{-1})\right] + \tilde{\alpha}_\ell H(t_\ell|\xi_\ell; \tilde{V}_\ell, \tilde{\rho}_\ell)$$

efficient implementation of extremization: sphinxteam / **dnner**

2) Theorem:

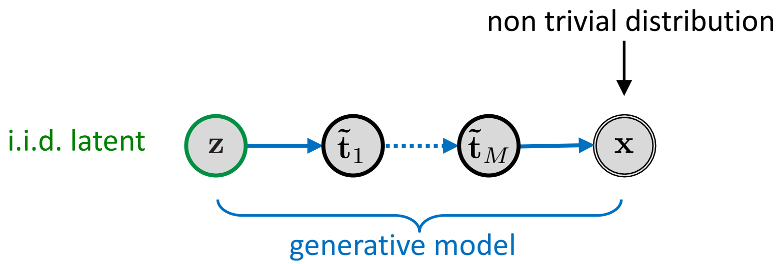
2-layer $\mathbf{t}_2 = f_\epsilon(\mathbf{W}^{(2)} f_\epsilon(\mathbf{W}^{(1)} \mathbf{t}_0))$
Gaussian random weight matrices

$\Rightarrow$ replica prediction exact in limit of large networks ✓

APPLICATION:
Follow mutual information during SGD learning

Applying the replica formula to learning experiments

1) Synthetic data sets for student

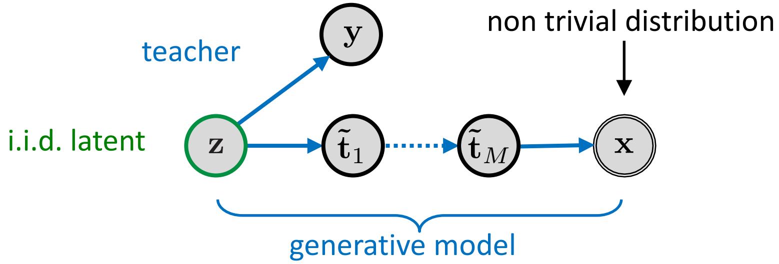


Keras implementation

marylou-gabrie / **learning-synthetic-data**

Applying the replica formula to learning experiments

1) Synthetic data sets for student

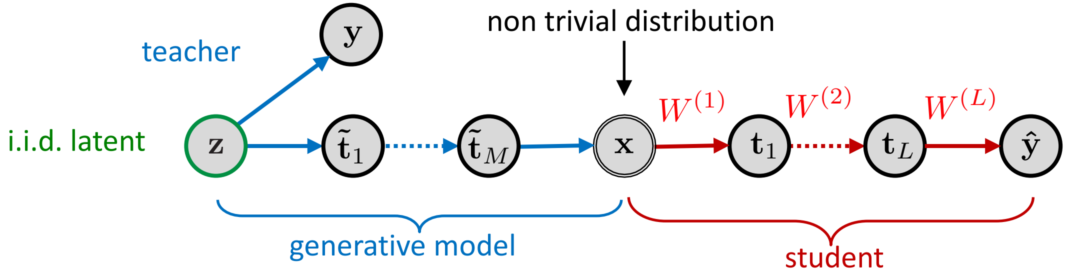


Keras implementation

marylou-gabrie / **learning-synthetic-data**

Applying the replica formula to learning experiments

1) Synthetic data sets for student

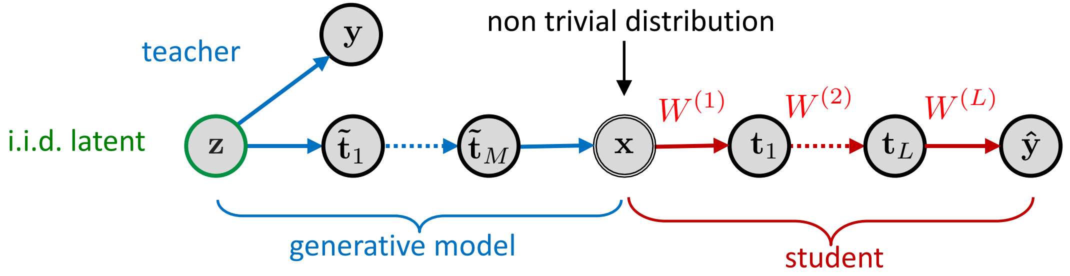


Keras implementation

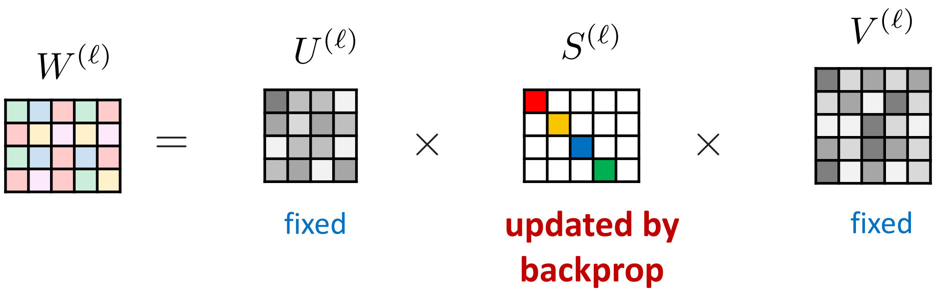
marylou-gabrie / **learning-synthetic-data**

Applying the replica formula to learning experiments

1) Synthetic data sets for student



2) Orthogonally invariant weight matrices during learning for the student

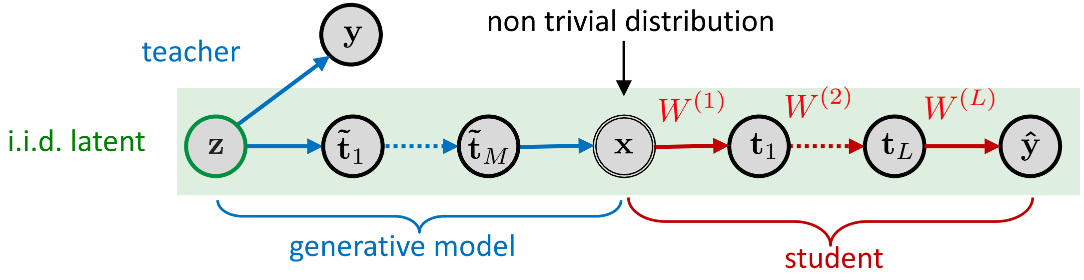


Keras implementation

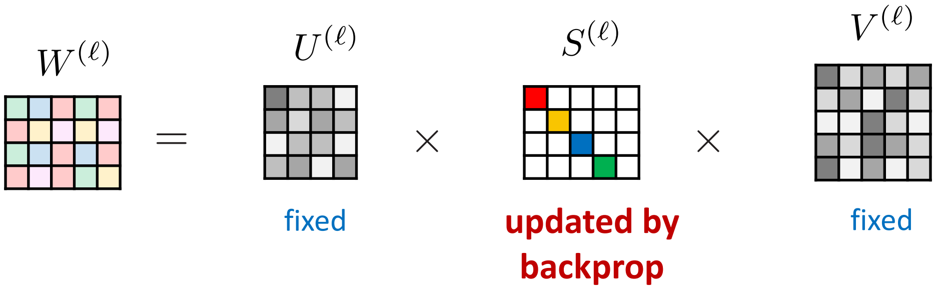
marylou-gabrie / **learning-synthetic-data**

Applying the replica formula to learning experiments

1) Synthetic data sets for student



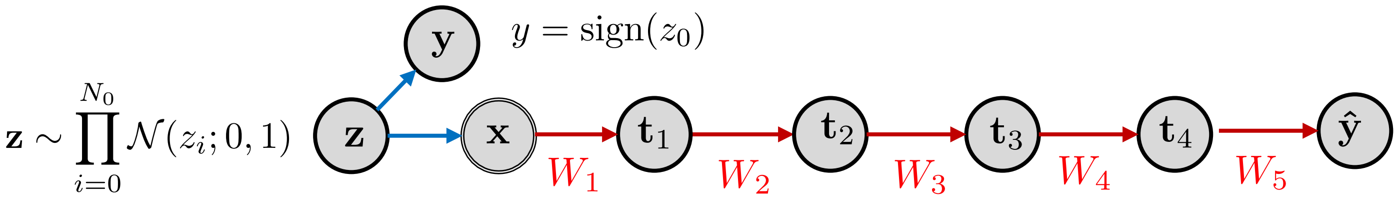
2) Orthogonally invariant weight matrices during learning for the student



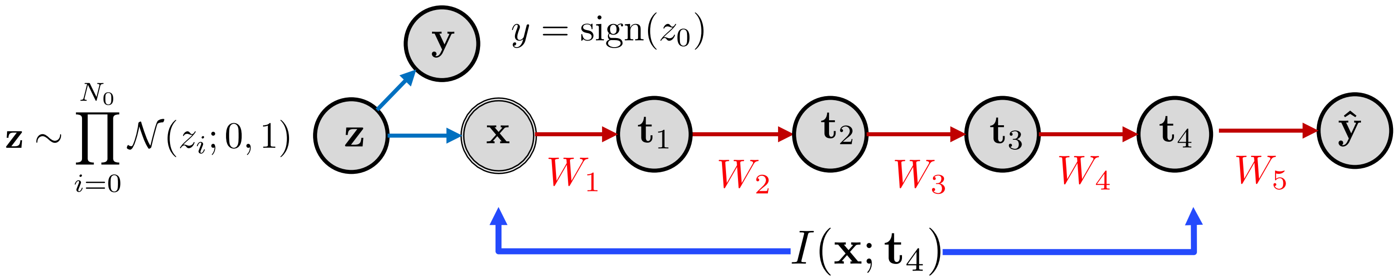
Keras implementation

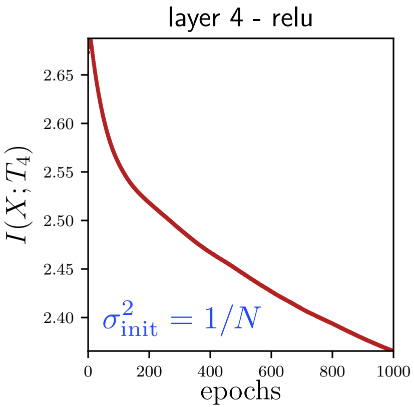
marylou-gabrie / **learning-synthetic-data**

Exploratory experiment: A binary classification example



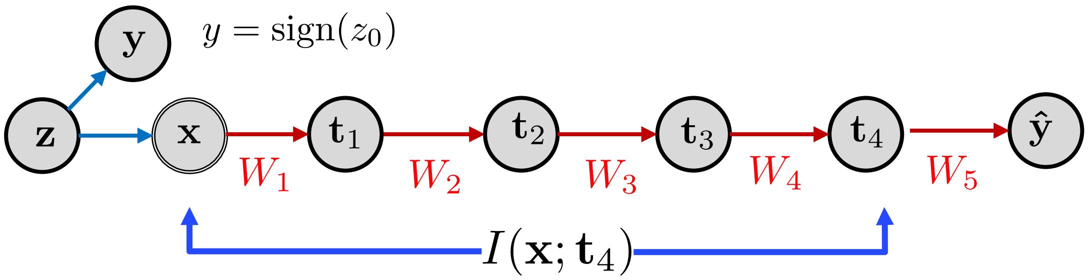
Exploratory experiment: A binary classification example



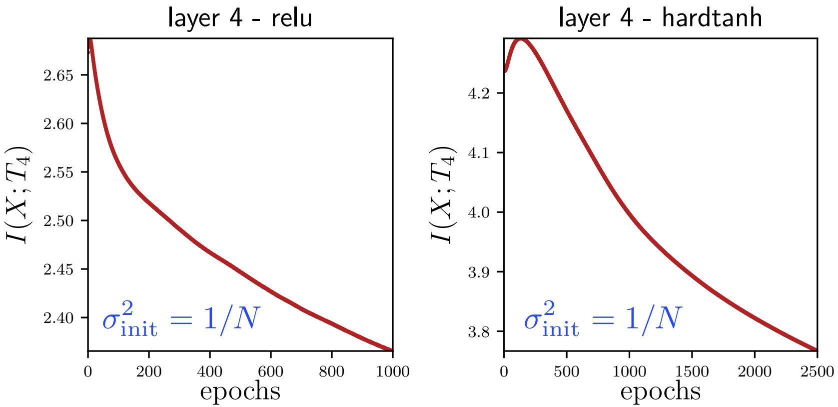


Exploratory experiment: A binary classification example

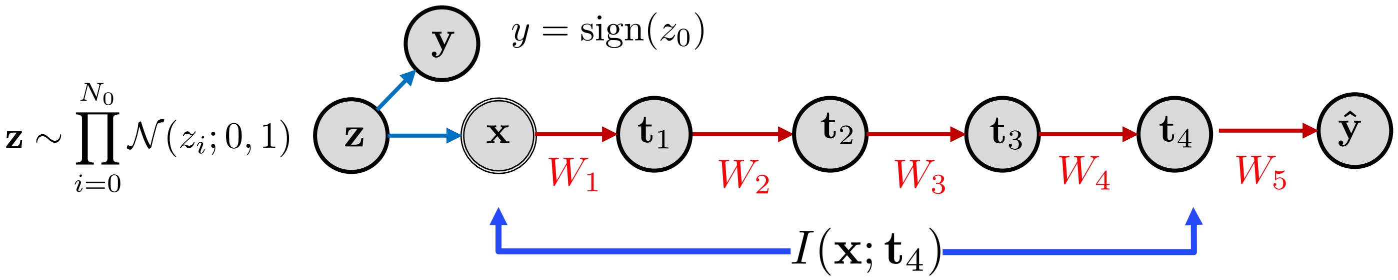
$$\mathbf{z} \sim \prod_{i=0}^{N_0} \mathcal{N}(z_i; 0, 1)$$



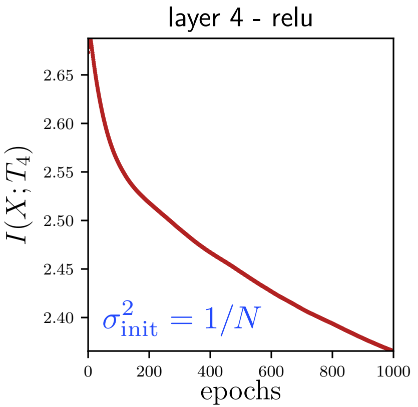
Different non-linearities



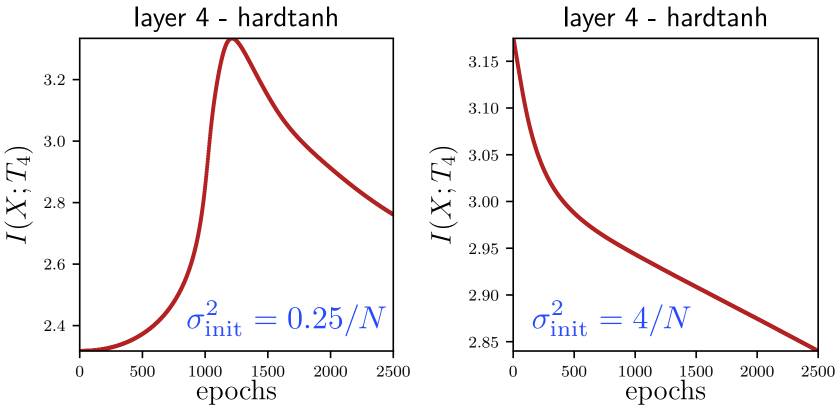
Exploratory experiment: A binary classification example



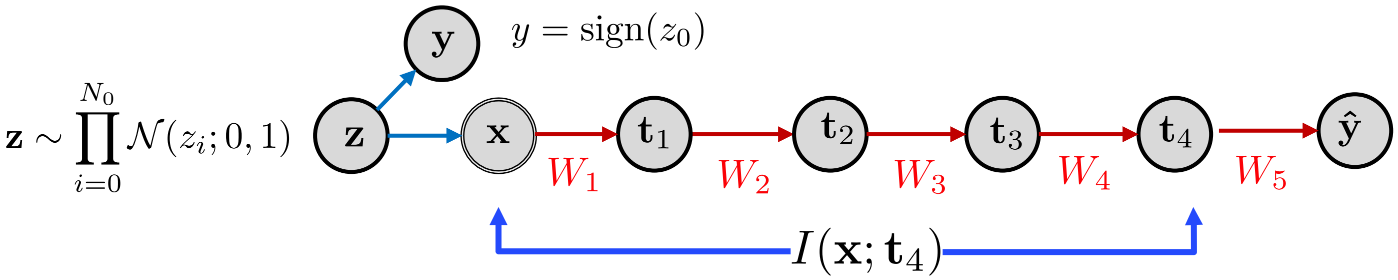
Different non-linearities



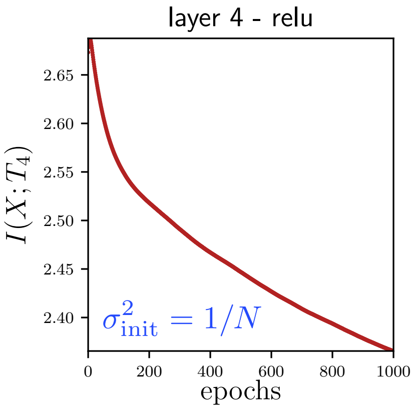
Different initializations



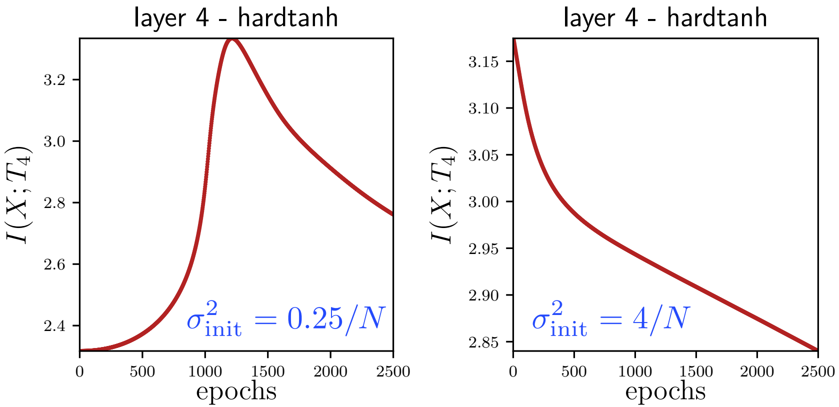
Exploratory experiment: A binary classification example



Different non-linearities



Different initializations



Poster @ Room 210 & 230 AB #110