The Gaussian Equivalence of generative models for learning with two-layer neural networks

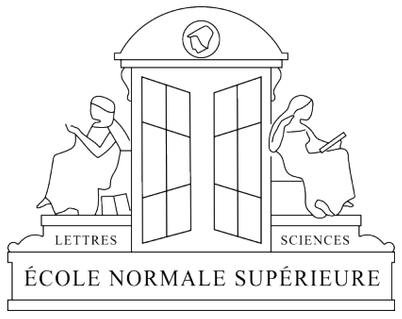
Sebastian Goldt (ENS Paris)

July 3rd, 2020 • Youth in high dimensions

Joint work w/

F. Krzakala (ENS), M. Mézard (ENS),

G. Reeves (Duke), L. Zdeborová (CEA Saclay)



The problem of data structure

The data sets we care about in machine learning contain a lot of structure.

Written text (NLP)

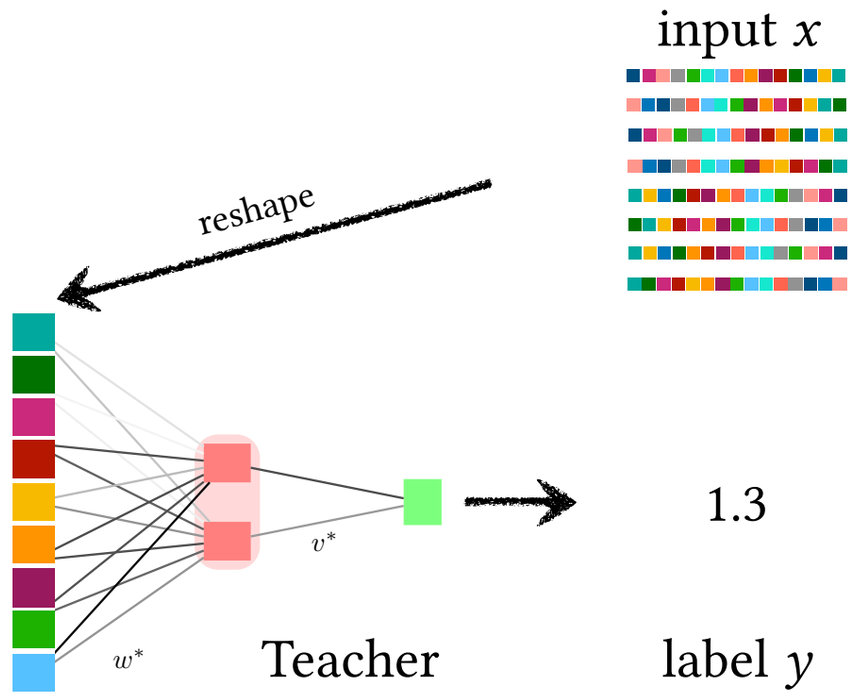
Images

Games of Go

How does data structure impact learning in neural networks?

The vanilla teacher-student setup

Supervised regression task with samples $(x \in \mathbb{R}^N, y \in \mathbb{R})$
drawn from some distribution q(x, y)



Inputs are i.i.d. draws
from the normal distribution:

$$x = (x_i), \quad x_i \sim \mathcal{N}(0, 1)$$

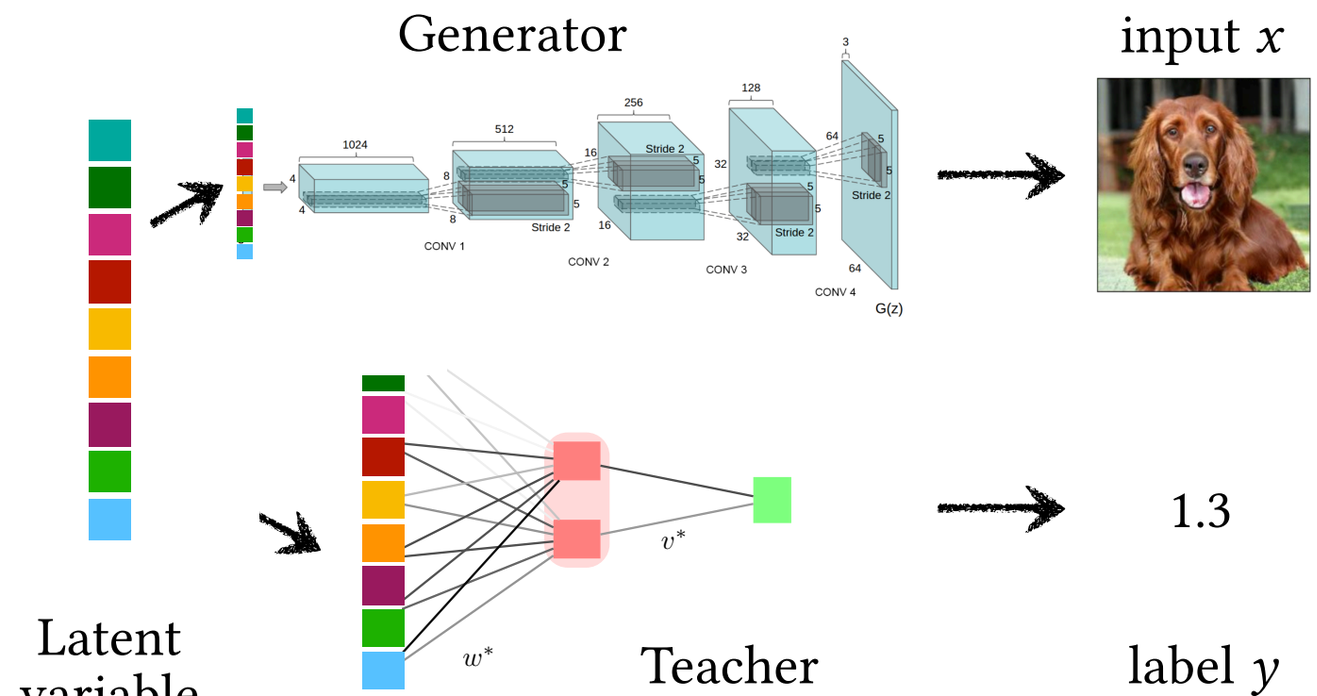
Generate label by applying a
teacher network directly to the input:

$$y = \phi_{\tilde{\theta}}(x) = \sum_{m=1}^{M} v^m \, \tilde{g}\left(\tilde{w}^m x\right)$$

E. Gardner and B. Derrida, J. Phys. A. Math. Gen. **22**, 1983 (1989)
D. Saad and S. A. Solla, Phys. Rev. Lett. **74**, 4337 (1995)

H.S. Seung, H. Sompolinsky and N. Tishby, Phys. Rev. A **45** (1992)
M. Biehl and H. Schwarze, J. Phys. A. Math. Gen. 28, 643 (1995)

The hidden manifold model



Generate an i.i.d. latent variable

$$c = (c_r), \quad c_r \sim \mathcal{N}(0,1)$$

Generate the input using a generative network

$$x = \mathcal{G}(c)$$

Apply a teacher to the latent representation

$$y = \phi_{\tilde{\theta}}(c)$$

SG, M. Mézard, F. Krzakala, L. Zdeborová, arXiv:1909.11500
SG, G. Reeves, M. Mézard, F. Krzakala, L. Zdeborová, arXiv:2006.14709

Our contributions

SG, M. Mézard, F. Krzakala, L. Zdeborová, arXiv:**1909.1150**
SG, G. Reeves, M. Mézard, F. Krzakala, L. Zdeborová, arXiv:**2006.14709**

Gaussian Equivalence Theorem

We give rigorous conditions under which we can analyse learning from data coming from single-layer generators.

Dynamical equations for two-layer students

The equations track the test error of two-layer students trained on deep generative models.

The Gaussian Equivalence Property

Goal: compute the prediction mean-squared error at all times.

For the **vanilla-teacher student** with i.i.d. inputs *x:*

Saad & Solla, (1995)
Biehl & Schwarze (1995)

$$\mathrm{pmse}\left(\theta, \tilde{\theta}\right) = \mathop{\mathbb{E}}_{x}\left(\sum_{k=1}^{K} v^k g\left(w^k x\right) - \sum_{m=1}^{M} \tilde{v}^m \tilde{g}\left(\tilde{w}^m x\right)\right)^2$$

Student network (trying to learn)

Teacher network (creates the data)

The Gaussian Equivalence Property

Goal: compute the prediction mean-squared error at all times.

For the **vanilla-teacher student** with i.i.d. inputs *x:*

Saad & Solla, (1995)
Biehl & Schwarze (1995)

$$\mathrm{pmse}\left(\theta, \tilde{\theta}\right) = \mathop{\mathbb{E}}_{x}\left(\sum_{k=1}^{K} v^k g\left(w^k x\right) - \sum_{m=1}^{M} \tilde{v}^m \tilde{g}\left(\tilde{w}^m x\right)\right)^2$$

Average over the inputs x

$\lambda^k \sim w^k x$

$\nu^m \sim \tilde{w}^m x$

The Gaussian Equivalence Property

Goal: compute the prediction mean-squared error at all times.

For the **vanilla-teacher student** with i.i.d. inputs *x:*

Saad & Solla, (1995)
Biehl & Schwarze (1995)

$$\mathrm{pmse}\left(\theta,\tilde{\theta}\right) = \mathop{\mathbb{E}}_{\lambda,\nu}\left(\sum_{k=1}^{K} v^k g\left(\lambda^k\right) - \sum_{m=1}^{M} \tilde{v}^m \tilde{g}\left(\nu^m\right)\right)^2$$

Average over the local fields (λ, ν)

$\lambda^k \sim w^k x$

$\nu^m \sim \tilde{w}^m x$

Key random variables
for online learning
and replicas (batch)

The Gaussian Equivalence Property

Goal: compute the prediction mean-squared error at all times.

For the **vanilla-teacher student** with i.i.d. inputs *x:*

Saad & Solla, (1995)
Biehl & Schwarze (1995)

$$\mathrm{pmse}\left(\theta, \tilde{\theta}\right) = \mathop{\mathbb{E}}_{\lambda,\nu}\left(\sum_{k=1}^{K} v^k g\left(\lambda^k\right) - \sum_{m=1}^{M} \tilde{v}^m \tilde{g}\left(\nu^m\right)\right)^2$$

$$\mathbb{E}\, x_i x_j = \delta_{ij}$$

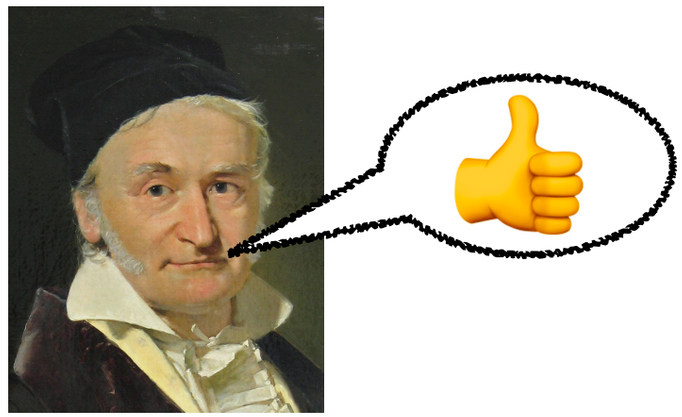
$$\lambda^k \sim \sum_i w_i^k x_i$$

$$\nu^m \sim \sum_i \tilde{w}_i^m x_i$$

Gaussian Equivalence Property:
(λ, ν) are jointly Gaussian

Hence, the *pmse* is a function of only the second moments of (λ, ν):

$$Q^{k\ell} \equiv \mathbb{E}\, \lambda^k \lambda^\ell, \quad R^{km} \equiv \mathbb{E}\, \lambda^k \nu^m, \quad T^{mn} \equiv \mathbb{E}\, \nu^m \nu^n$$



The Gaussian Equivalence Theorem

Setup: Fully connected, single layer generator $\mathcal{G} : \mathbb{R}^D \to \mathbb{R}^N$

$$x_n = \mathcal{G}_n(c) = \sigma(a_n^\top c)$$

with the teacher acting on the latent variable *c:* $y = \phi_{\tilde{\theta}}(c)$

$$\mathbb{E}\, x_i x_j = \Omega_{ij}$$

$$\lambda^k \sim \sum_i w_i^k x_i$$

$$\nu^m \sim \sum_r \tilde{w}_r^m c_r$$

They're still (sometimes) Gaussian!

Theorem: Let *P* be the distribution of the pair (λ, ν) and let $\hat{P}$ be the Gaussian distribution with the same first and second moments. Then…

$$d_{\mathrm{MS}}(P, \hat{P}) = O\left(\left\| \tfrac{1}{\sqrt{N}} W M_1^{1/2} \right\|^2 + \left\| \tfrac{1}{\sqrt{N}} W M_2^{1/2} \right\| + \tfrac{1}{\sqrt{N}} \left\| \tfrac{1}{\sqrt{D}} \tilde{W} A^\top \right\|^2 + \frac{1}{\sqrt{N}} \right)$$

The Gaussian Equivalence Theorem

Theorem: Let *P* be the distribution of the pair (λ, ν) and let $\hat{P}$ be the Gaussian distribution with the same first and second moments. Then…

$$\mathcal{G} : \mathbb{R}^D \to \mathbb{R}^N$$

$$x_n = \mathcal{G}_n(c) = \sigma(a_n^\top c)$$

$$y = \phi_{\tilde{\theta}}(c)$$

Generator weights

Student weights *Teacher weights*

$$d_{\mathrm{MS}}(P, \hat{P}) = O\left(\left\| \tfrac{1}{\sqrt{N}} W M_1^{1/2} \right\|^2 + \left\| \tfrac{1}{\sqrt{N}} W M_2^{1/2} \right\| + \tfrac{1}{\sqrt{N}} \left\| \tfrac{1}{\sqrt{D}} \tilde{W} A^\top \right\|^2 + \frac{1}{\sqrt{N}} \right)$$

Related to input correlations

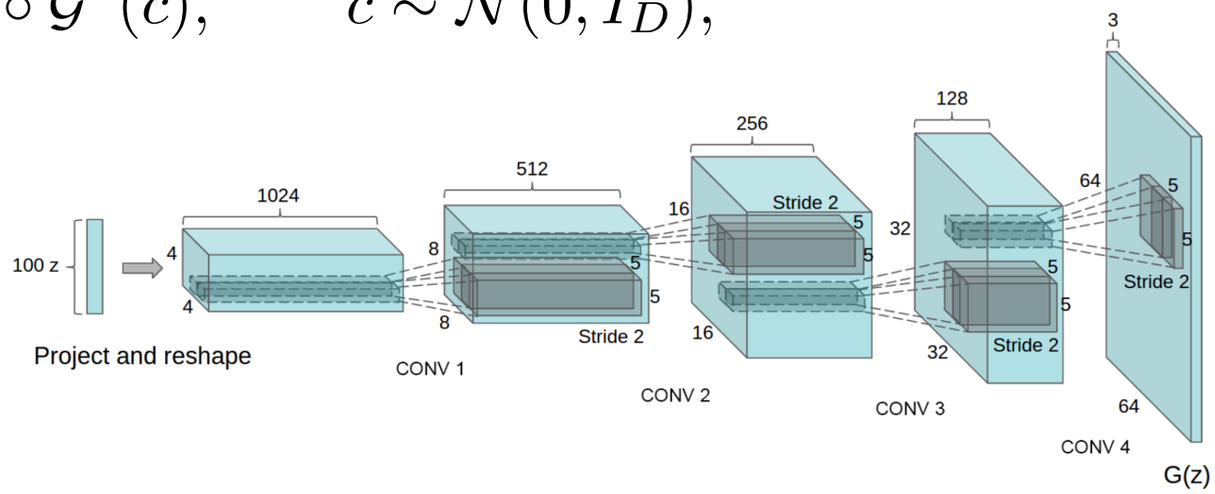
Related work

- Works in wide network limit rely on RMT and thus random weights
- Mei & Montanari; Couillet et al. introduce related equivalent Gaussian models for integrals w.r.t. spectral densities.
- Large body of work on low-dim projections of high-dim data being Gaussian - we quantify how Gaussian they look like.

The deep hidden manifold model

$$x = \mathcal{G}(c) = \mathcal{G}^L \cdots \mathcal{G}^3 \circ \mathcal{G}^2 \circ \mathcal{G}^1(c), \qquad c \sim \mathcal{N}(0, I_D),$$

$$y = \phi_{\tilde{\theta}}(c)$$



Deep Convolutional GAN
Radford et al., ICLR 2016

- Train the student using online SGD:

$$\theta_{\mu+1} = \theta_\mu - \eta \nabla_\theta E(\theta)|_{\theta_\mu, x_\mu, y^*_\mu}$$

Goal: Derive a closed set of equations for the order parameters

$$Q^{k\ell} \equiv \mathbb{E}\, \lambda^k \lambda^\ell, \qquad R^{km} \equiv \mathbb{E}\, \lambda^k \nu^m$$

that track the dynamics of a two-layer student
trained using online SGD on the deep hidden manifold.

Dynamical equations for two-layer students

Statement:

$$Q^{k\ell} \equiv \mathbb{E}\,\lambda^k \lambda^\ell, \quad R^{km} \equiv \mathbb{E}\,\lambda^k \nu^m$$

$$Q^{k\ell} = \int \mathrm{d}\mu_\Omega(\rho)\; \rho\; q^{k\ell}(\rho)$$

Spectral density of input-input covariance

$$\frac{\partial q^{k\ell}(\rho)}{\partial t} = -\eta \left(\rho \sum_{j\neq k}^{K} \left[v^k v^j q^{k\ell}(\rho) h^{kj}_{(1)}(Q) + v^k v^j q^{j\ell}(\rho) h^{kj}_{(2)}(Q) \right] + \rho v^k v^k q^{k\ell}(\rho) h^{k}_{(3)}(Q) \right.$$
$$- v^k \sum_{n}^{M} \left[\rho \tilde{v}^n q^{k\ell}(\rho) h^{kn}_{(4)}(Q,R,T) + \frac{1}{\sqrt{\delta}} \tilde{v}^n r^{\ell n}(\rho) h^{kn}_{(5)}(Q,R,T) \right]$$
$$\left. + \text{all of the above with } \ell \to k, k \to \ell \right) + \eta^2 \gamma v^k v^\ell h^{k\ell}_{(6)}(Q,R,T,v,\tilde{v}).$$

$$R^{km} = \frac{1}{\sqrt{\delta}} \int \mathrm{d}\mu_\Omega(\rho)\; r^{km}(\rho)$$

$$\frac{\partial r^{km}(\rho)}{\partial t} = -\eta v^k \left(\rho \sum_{j\neq k}^{K} \left[v^j r^{km}(\rho) h^{kj}_{(1)}(Q) + v^j \rho r^{jm}(\rho) h^{kj}_{(2)}(Q) \right] + v^k \rho r^{km}(\rho) h^{k}_{(3)}(Q) \right.$$
$$\left. - \sum_{n}^{M} \left[\rho \tilde{v}^n r^{km}(\rho) h^{kn}_{(4)}(Q,R,T) + \frac{1}{\sqrt{\delta}} \tilde{v}^n h^{kn}_{(5)}(Q,R,T) \right] \right).$$

Dynamical equations for two-layer students

Statement:

$$Q^{k\ell} \equiv \mathbb{E}\,\lambda^k \lambda^\ell, \quad R^{km} \equiv \mathbb{E}\,\lambda^k \nu^m$$

$$Q^{k\ell} = \int \mathrm{d}\mu_\Omega(\rho)\; \rho\; q^{k\ell}(\rho) \qquad R^{km} = \frac{1}{\sqrt{\delta}} \int \mathrm{d}\mu_\Omega(\rho)\; r^{km}(\rho)$$

Remarkably, the generator only appears via two covariance matrices:

$$\Omega_{ij} = \mathbb{E}\; x_i x_j$$

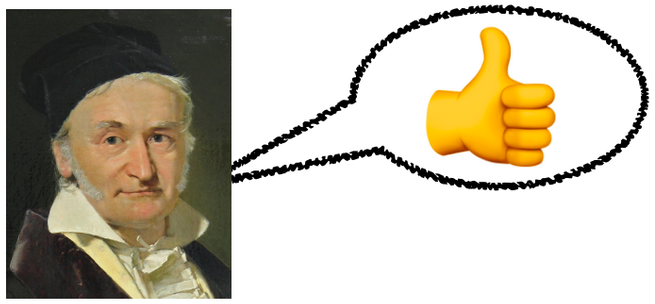
Input-input correlations

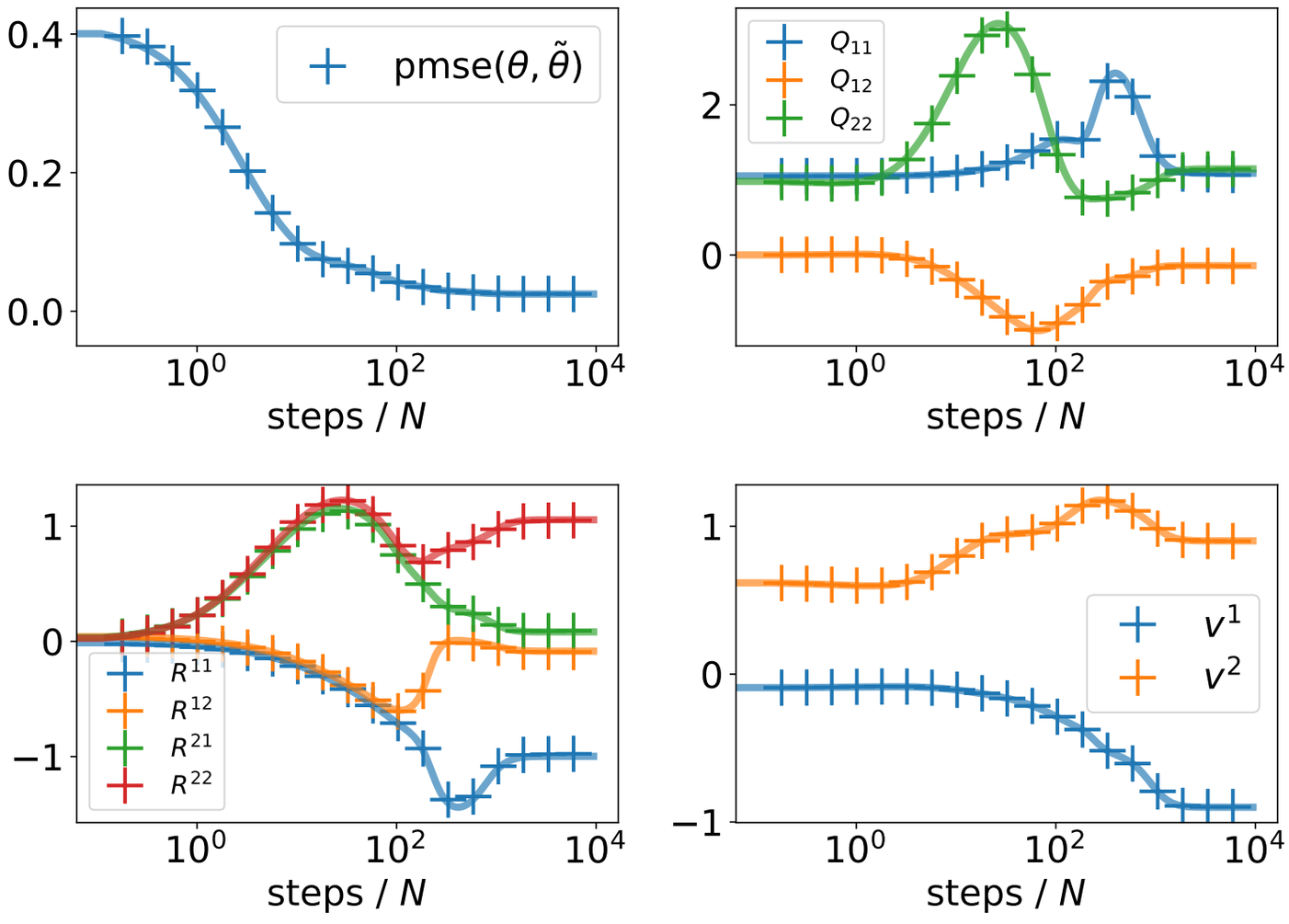
$$\Phi_{ir} = \mathbb{E}\; x_i c_r$$

Input-latent correlations

Single-layer fully connected generator

$$x_n = \mathcal{G}_n(c) = \sigma(a_n^\top c) \qquad y = \phi_{\tilde{\theta}}(c)$$



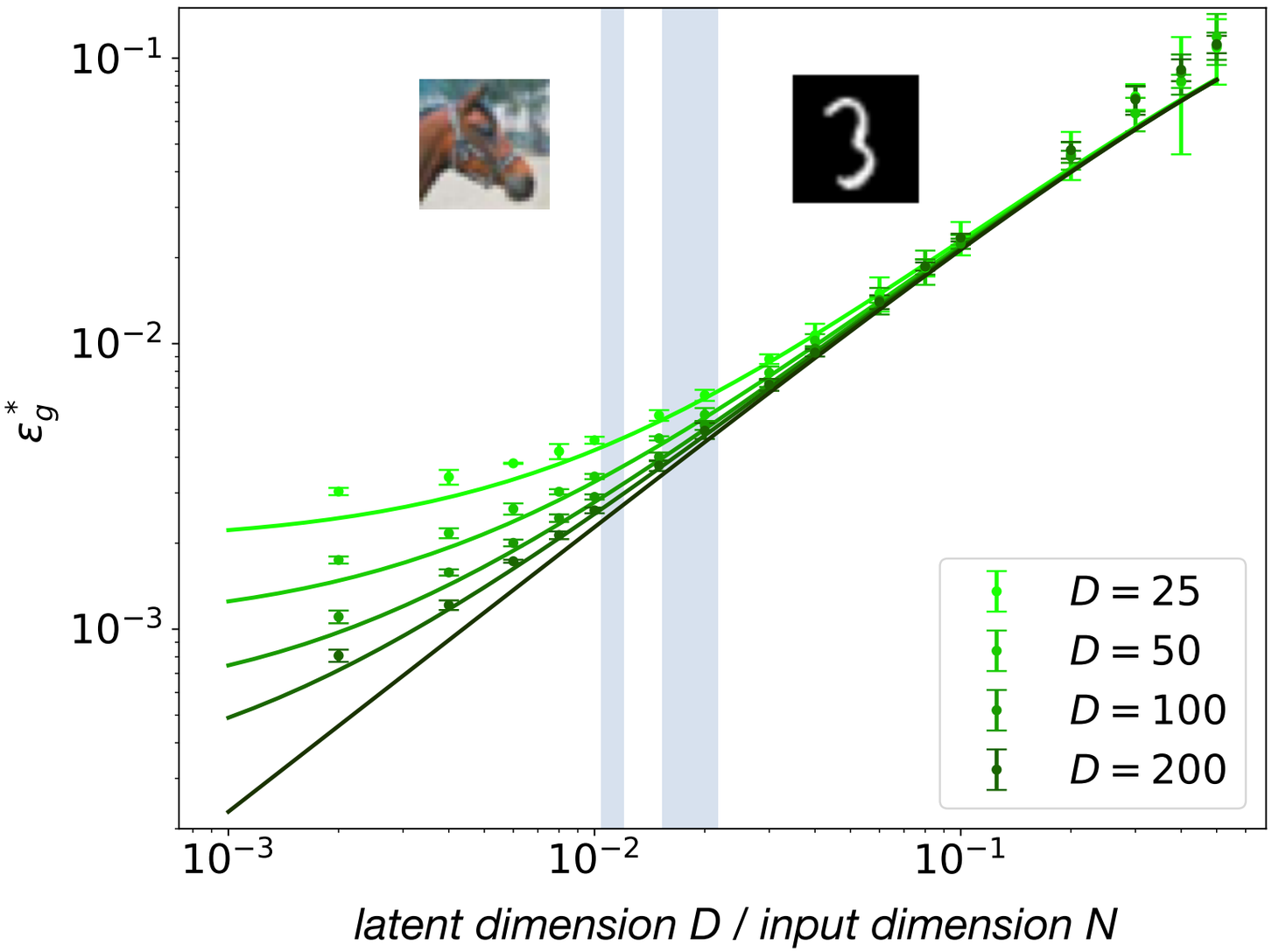


$$\sigma(x) = \mathrm{sgn}(x), \quad g(x) = \mathrm{erf}(x/\sqrt{2})$$

M=K=2, $\eta = 0.2$, D=800, N=8000

The ODEs predict performance of two-layer NN

$$x_n = \mathcal{G}_n(c) = \sigma(a_n^\top c) \qquad y = \phi_{\tilde{\theta}}(c)$$



$$\sigma(x) = \mathrm{sgn}(x), \quad g(x) = \mathrm{erf}(x/\sqrt{2})$$

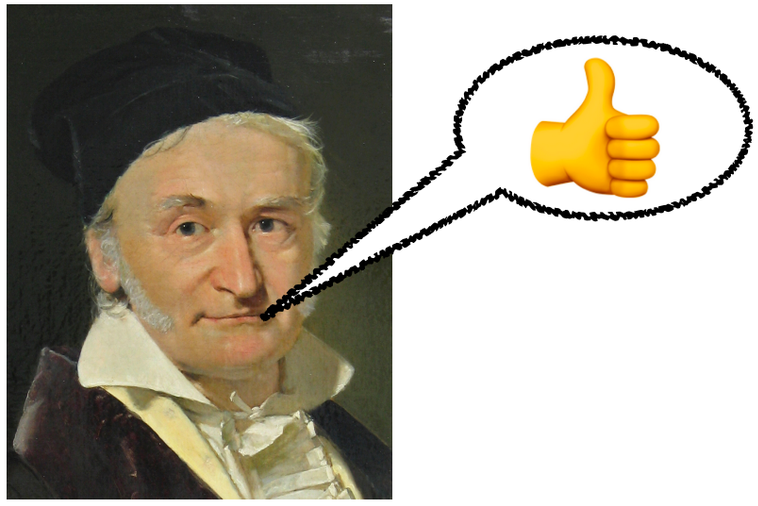
M=K=2, $\eta = 0.2$

The realNVP: normalising flows

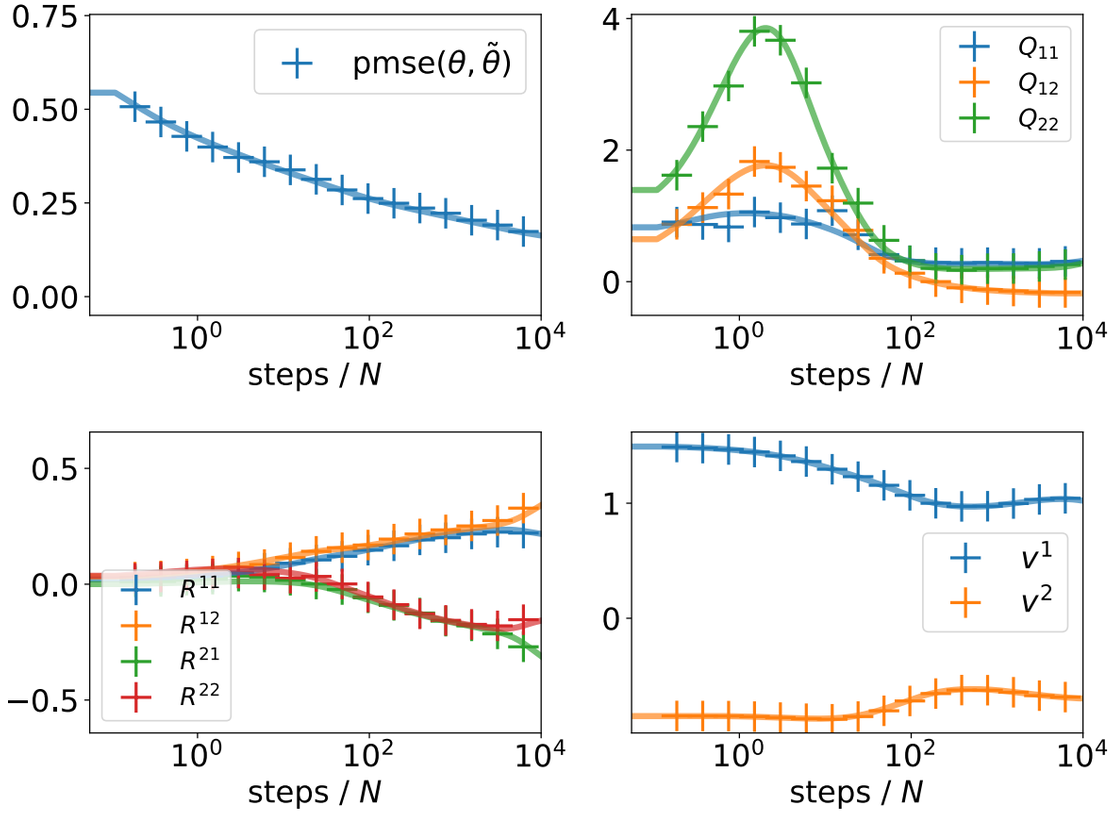
L. Dinh, J. Sohl-Dickstein, S. Bengio (ICLR 2017)

- Generate images using a series of invertible transformations, e.g. convolutions etc.

- Trained realNVP (>6M parameters) on CIFAR10



Top half: CIFAR10 images
Bottom half: Samples from realNVP trained on CIFAR10

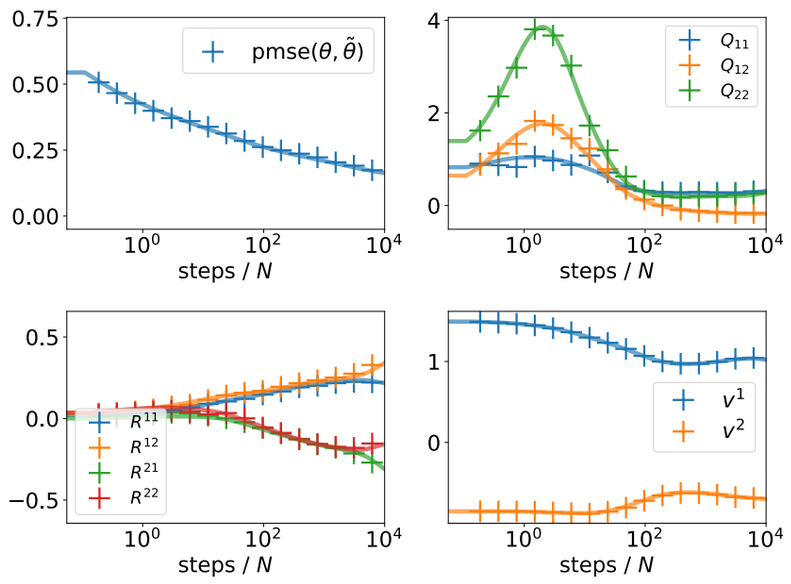


M=K=2, $\eta = 0.2$, D=3072, N=3072

Conclusion

- We study can study the impact of data on learning by analysing NN trained on data from generative models

- The GEP gives analytical control over the learning dynamics, and is rigorous for single-layer generators.



References

- **arXiv:1909.11500**
  w/ M. Mézard, F. Krzakala, L. Zdeborová
- **arXiv:2006.14709**
  w/ G. Reeves, M. Mézard, F. Krzakala, L. Zdeborová

https://github.com/sgoldt