Physics of Machine Learning

Gert Aarts

LEVERHULME TRUST

Swansea University Prifysgol Abertawe

THE ROYAL SOCIETY

KEK, Jan 2026

ML and theoretical physics

- commonality between methods used in ML and in theoretical physics
- obvious examples: linear algebra, optimisation, stochastic dynamics, sampling, …
- many more intriguing connections: understand ML algorithms better, contribute new ideas
- theoretical physicists are/should not satisfied with a ‘black-box’ algorithm
- quantum field theory: extensive experience in analytical and computational studies of systems with many fluctuating degrees of freedom → fairly unique perspective

Outline: ML and theoretical physics

three examples:

- **diffusion models** and **stochastic quantisation**
- **stochastic gradient descent** and **random matrix theory**
- **learning capability of neural networks** and **phase diagram of disordered systems**

I. Diffusion models and stochastic quantisation

very enjoyable collaboration with Lingxiao Wang, Kai Zhou

and

Diaa Habibi

Qianteng Zhu, Wei Wang

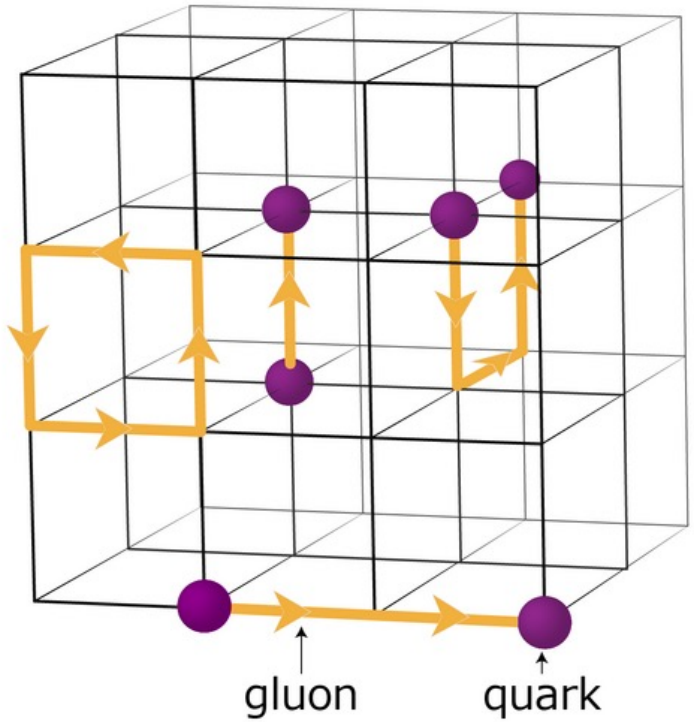
Thomas Ranner, Andreas Ipp, David Müller

Lattice field theory: nonperturbative physics

- simulating strongly interacting matter
- quarks and gluons on a four-dimensional spacetime lattice
- numerical evaluation of the (Euclidean) path integral

$$Z = \int DU\, D\bar{\psi} D\psi\, e^{-S_{\mathrm{YM}}(U)+\bar{\psi}M(U)\psi} = \int DU\, \det M(U) e^{-S_{\mathrm{YM}}(U)}$$

- very high-dimensional integral, importance sampling
- numerically expensive, requires extensive HPC resources



https://www.physik.uni-wuppertal.de/de/startseite-fachgruppe-physik/

Generative AI for lattice field theory (LFT)

- generate ensembles of configurations, varying lattice spacing, volume, temperature, …
- standard algorithm: Hybrid Monte Carlo (HMC) + variations
- can GenAI offer alternatives, complement HMC, or replace bottlenecks?
- pioneered by MIT group: normalising flow, learning an invertible transformation

  MS Albergo, G Kanwar, PE Shanahan, *Phys Rev D* **100** (2019) 3, 034515 [1904.12072 [hep-lat]]
- and by Akio Tomiya (see talk on Monday)
- in this talk: diffusion models, learning from data

References

- LW, GA, KZ, *JHEP* **05** (2024) 060 [2309.17082 [hep-lat]]; *ML4PS NeurIPS* 2023, 2311.03578 [hep-lat]
- + **D Habibi**, *Mach Learn Sci Tech* **6** (2025) 2, 025004, *ML4PS NeurIPS* 2024 [2410.21212 [hep-lat]]
  JHEP **12** (2025) 160 [2510.01328 [hep-lat]]
- + **Q Zhu**, W Wang, 2502.05504; *ML4PS NeurIPS* 2024, 2410.19602 [hep-lat]
- + **T Ranner**, A Ipp, D Müller: in preparation (non-abelian gauge theories)

see also work by

- Y Hirono, A Tanaka, K Fukushima, 2403.11262 [cs.LG]
- K Fukushima, S Kamata, Y Hirono, *J Phys Soc Jap* **94** (2025) 3, 031010 [2411.11297 [hep-lat]]
- O Vega et al, 2510.26081 [hep-lat], 2512.19877 [hep-lat]

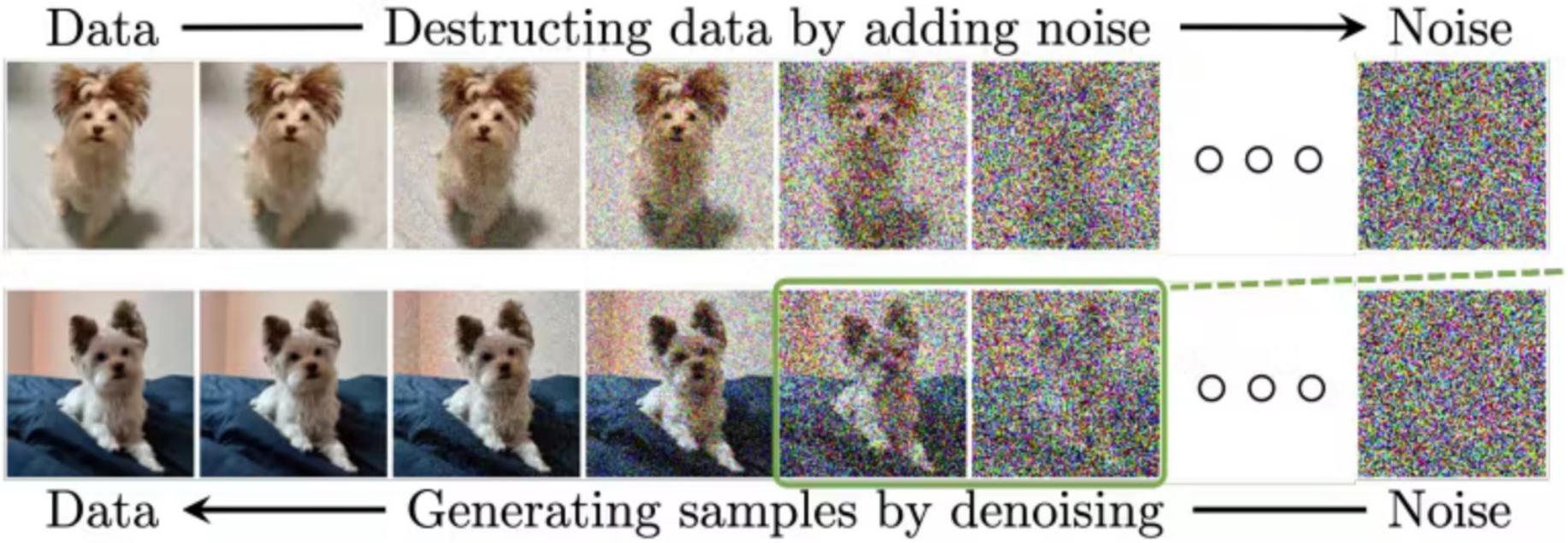
denoising

Generative Modeling by Estimating Gradients of the Data Distribution
Yang Song, Stefano Ermon
1907.05600 [cs.LG]

interpolation

Score-Based Generative Modeling through Stochastic Differential Equations
Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, Ben Poole, 2011.13456 [cs.LG]

Generative AI using diffusion models



465-page review: C-H Lai, Y Song, D Kim, Y Mitsufuji, S Ermon
The Principles of Diffusion Models, 2510.21890 [cs.LG]

https://encord.com/blog/diffusion-models/

Diffusion models: scalar fields

application to LFT straightforward: noise is applied at each lattice point separately

- forward $\partial_t \phi(x,t) = K[\phi(x,t),t] + g(t)\eta(x,t)$
- backward $\partial_\tau \phi(x,\tau) = -K[\phi(x,\tau),T-\tau] + g^2(T-\tau)\nabla_\phi \log P(\phi,T-\tau) + g(T-\tau)\eta(x,\tau)$

  score
- neural network approximates the time-dependent score
- learned by minimising a loss function $s_\theta(\phi,t) \sim \nabla_\phi \log P(\phi,t)$

neural network **score**

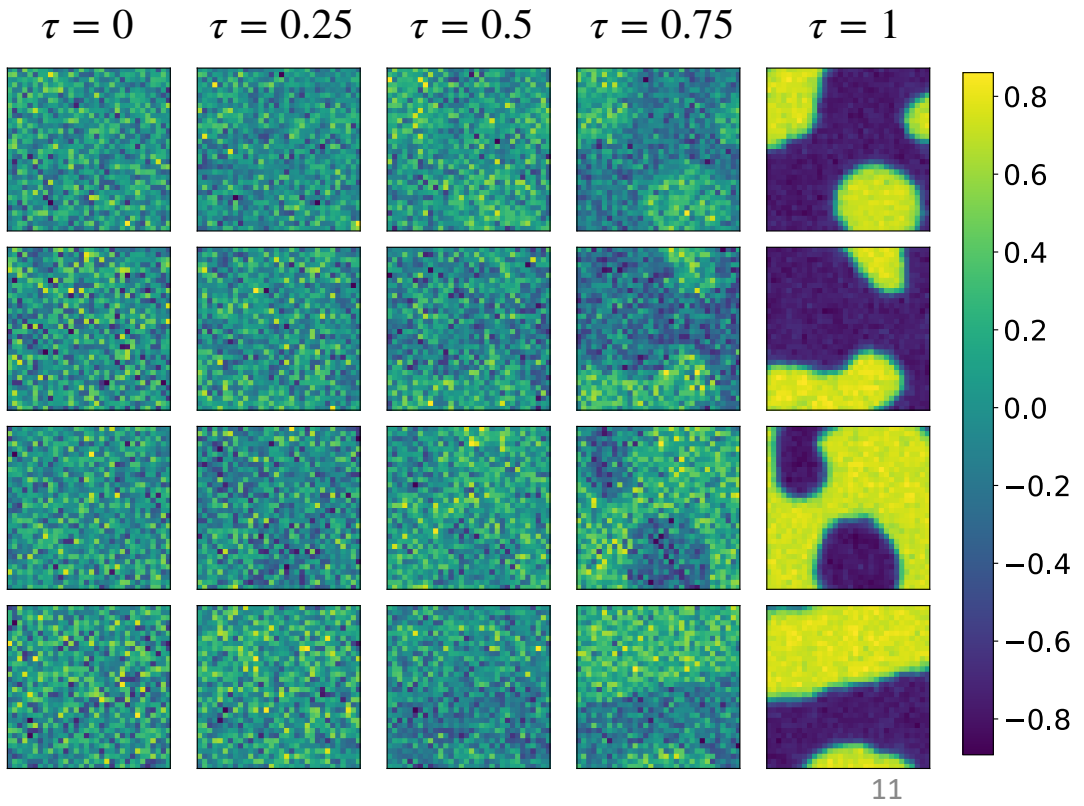
Diffusion model for 2d ϕ^4 lattice scalar theory

- 32^2 lattice, choice of action parameters in symmetric and broken phase
- training data set generated using Hybrid Monte Carlo (HMC)

- first application of diffusion models in lattice field theory

generating configurations:

- broken phase
- "denoising" (backward process)
- large-scale clusters emerge, as expected



L Wang, GA, K Zhou, *JHEP* **05** (2024) 060 [2309.17082 [hep-lat]]

Diffusion models and stochastic quantisation

- backward/denoising process

$$\frac{\partial \phi(x,\tau)}{\partial \tau} = g^2(\tau)\nabla_\phi \log P(\phi;\tau) + g(\tau)\eta(x,\tau)$$

score

- write the “time-dependent” distribution in terms of effective action

$$P(\phi;\tau) = \frac{e^{-S(\phi,\tau)}}{Z}$$

- DMs resemble approach well known in LFT: stochastic quantisation (Parisi and Wu 1981)

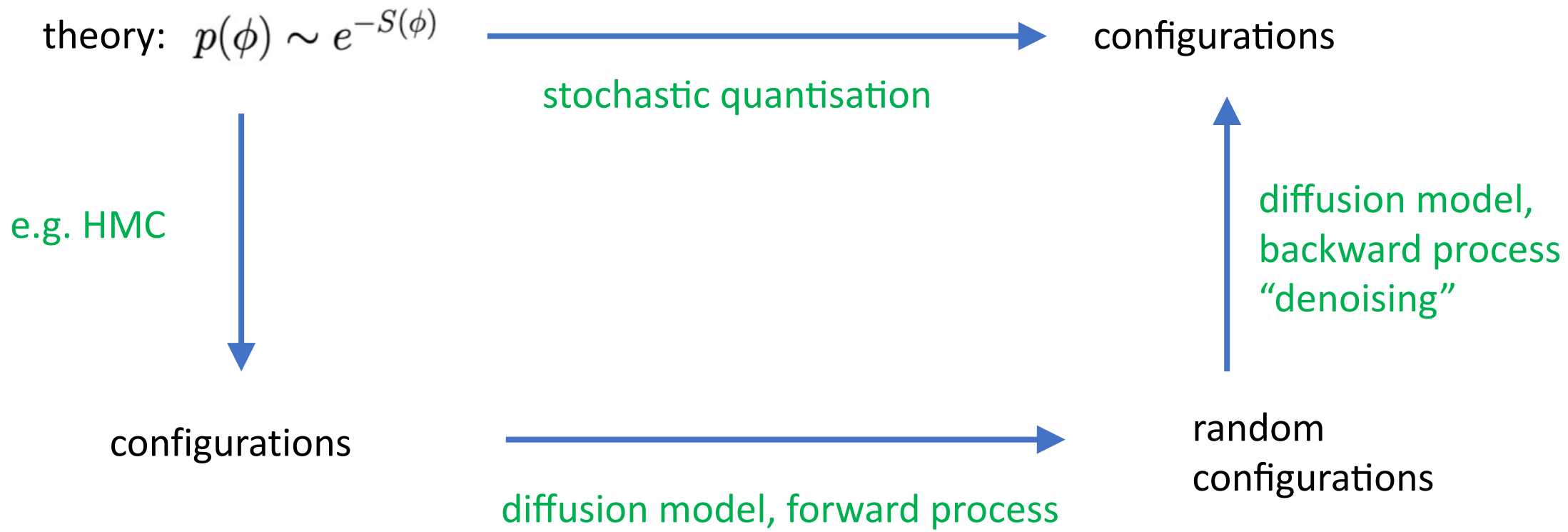
$$\nabla_\phi \log P(\phi,\tau) = -\nabla_\phi S(\phi,\tau)$$

- but with time-dependent drift and different approach to thermalisation

$$\frac{\partial \phi(x,\tau)}{\partial \tau} = -\nabla_\phi S(\phi) + \sqrt{2}\eta(x,\tau)$$

Diffusion models and stochastic quantisation

- diffusion models as an alternative approach to stochastic quantisation



L Wang, GA, K Zhou, *JHEP* **05** (2024) 060 [2309.17082 [hep-lat]]

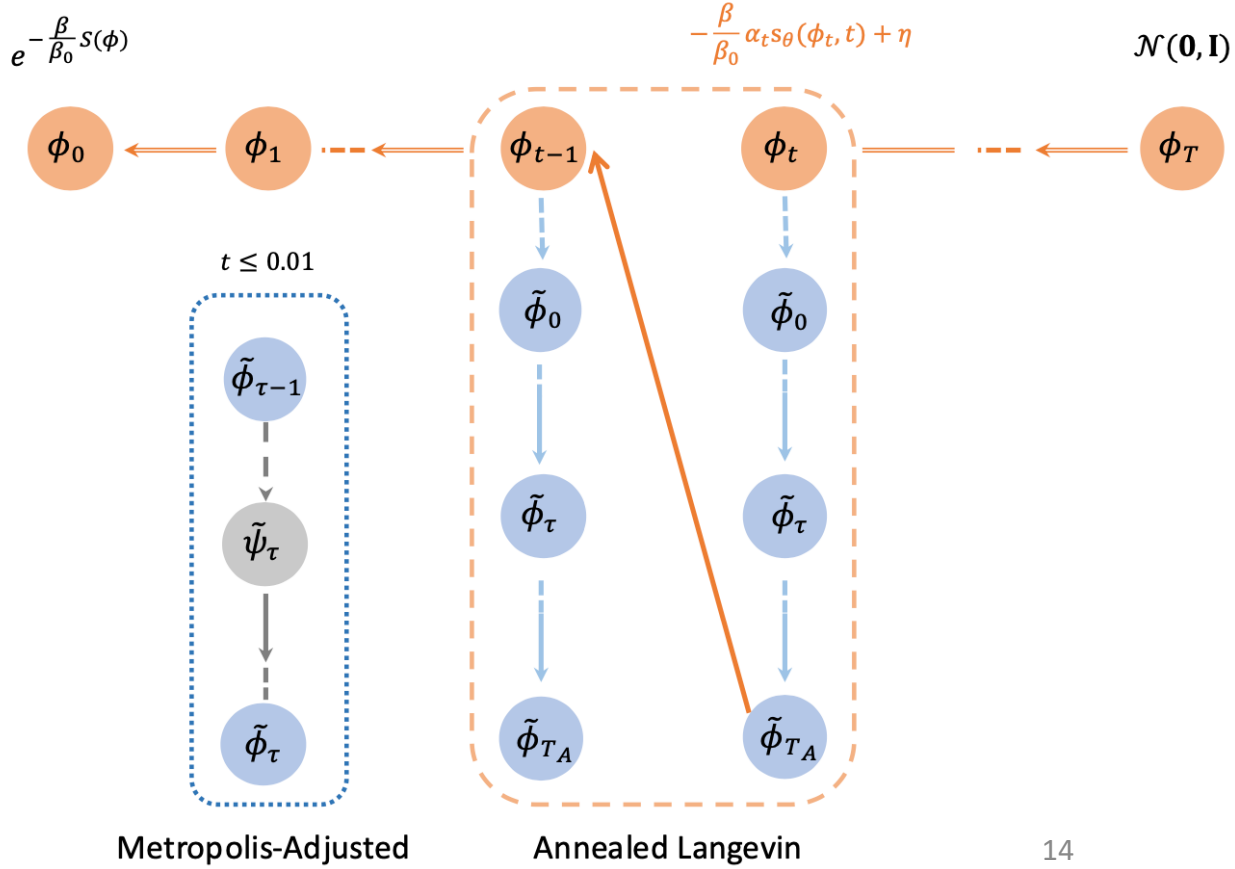
see also Y Hirono, A Tanaka, K Fukushima, 2403.11262

Incorporate (new/old) ideas in DM dynamics

use experience from stochastic quantisation and other well-tested approaches:

backward process (after model has been trained):

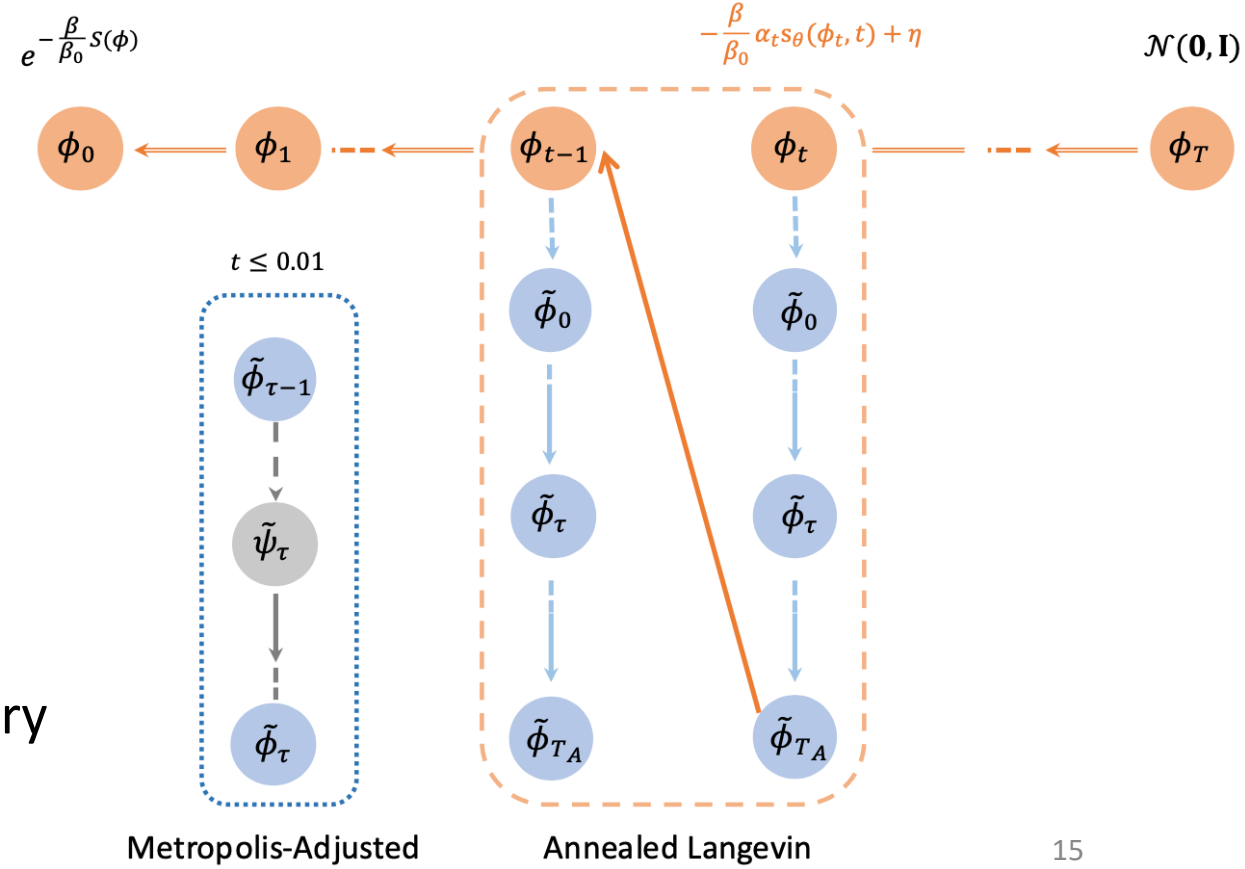
- exactness: accept/reject step
  Metropolis-adjusted Langevin algorithm
- thermalisation: annealing stage
- conditioning: reweighting from β_0 to β



Q Zhu, W Wang, GA, K Zhou, L Wang, 2502.05504 [hep-lat]

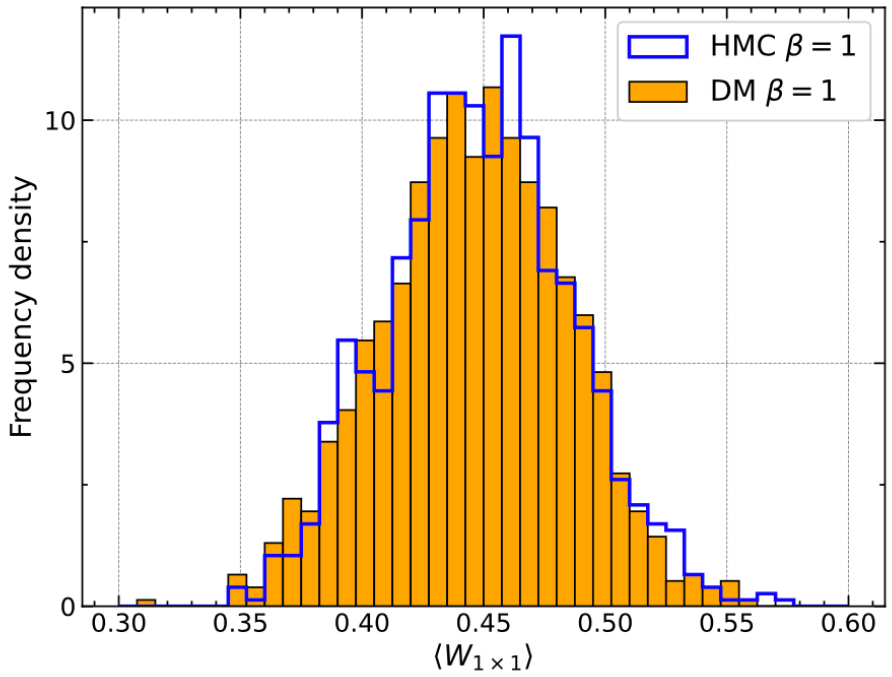
Physics conditioning: gauge theory

- action and drift scale with gauge coupling β
- motivated by stochastic quantisation: score is proportional to β
- train using data generated at β_0
- employ at different β values
- applied to U(1) and U(2), SU(2) gauge theory (latter are in preparation)



2D U(1) gauge theory: vary the volume

- training: 30k configurations at $\beta = 1$ on 16^2 obtained using HMC
- generating: 1024 configs at $\beta = 1, 3, 5, 7, 9, 11$ on $8^2, 16^2, 32^2$



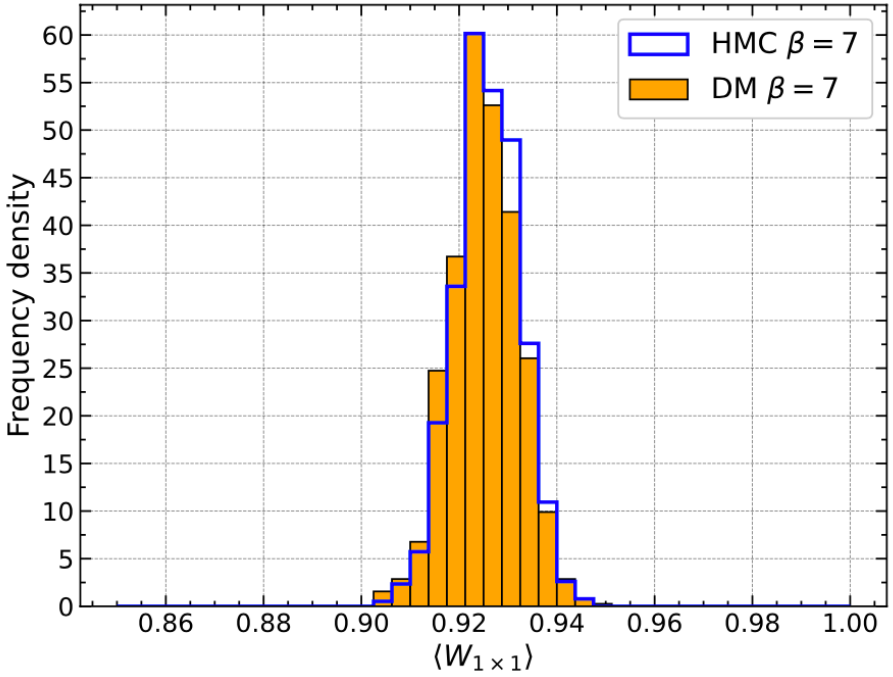
$\beta = 1, L = 16$, HMC vs DM

Lattice Size (L)	1×1 Wilson Loop			
	HMC	DM	Langevin	Exact
8	0.447(72)	0.445(74)	0.443(80)	0.446
16	0.447(37)	0.446(37)	0.444(36)	0.446
32	0.446(18)	0.445(19)	0.445(18)	0.446
64	0.446(9)	0.446(11)	0.445(9)	0.446

increase the volume, after training on $L = 16$

2D U(1) gauge theory: vary the coupling

- training: 30k configurations at $\beta = 1$ on 16^2 obtained using HMC
- generating: 1024 configs at $\beta = 1, 3, 5, 7, 9, 11$ on $8^2, 16^2, 32^2$



$\beta = 7, L = 16$, HMC vs DM

coupling (β)	1×1 Wilson Loop			
	HMC	DM	Langevin	Exact
3	0.811(17)	0.811(17)	0.809(17)	0.810
5	0.894(9)	0.894(9)	0.891(10)	0.894
7	0.926(7)	0.926(7)	0.924(6)	0.926
9	0.944(3)	0.942(4)	0.940(6)	0.942
11	0.954(3)	0.953(4)	0.950(5)	0.953

increase the coupling, after training at $\beta = 1$

Diffusion models: summary

- implementation in LFT is relatively straightforward
- trained on data: need high-quality ensembles
- apply to larger volumes, different couplings
- relation to well-known algorithms in computational physics very useful !

further work:

- theories with a sign problem: GA, D Habibi, L Wang, K Zhou, *JHEP* **12** (2025) 160 [2510.01328 [hep-lat]]
- non-abelian SU(2) and U(2) gauge theories, using gauge equivariant networks (L-CNNs):

  in preparation, with D Habibi, A Ipp, D Müller, T Ranner, L Wang, W Wang, Q Zhu

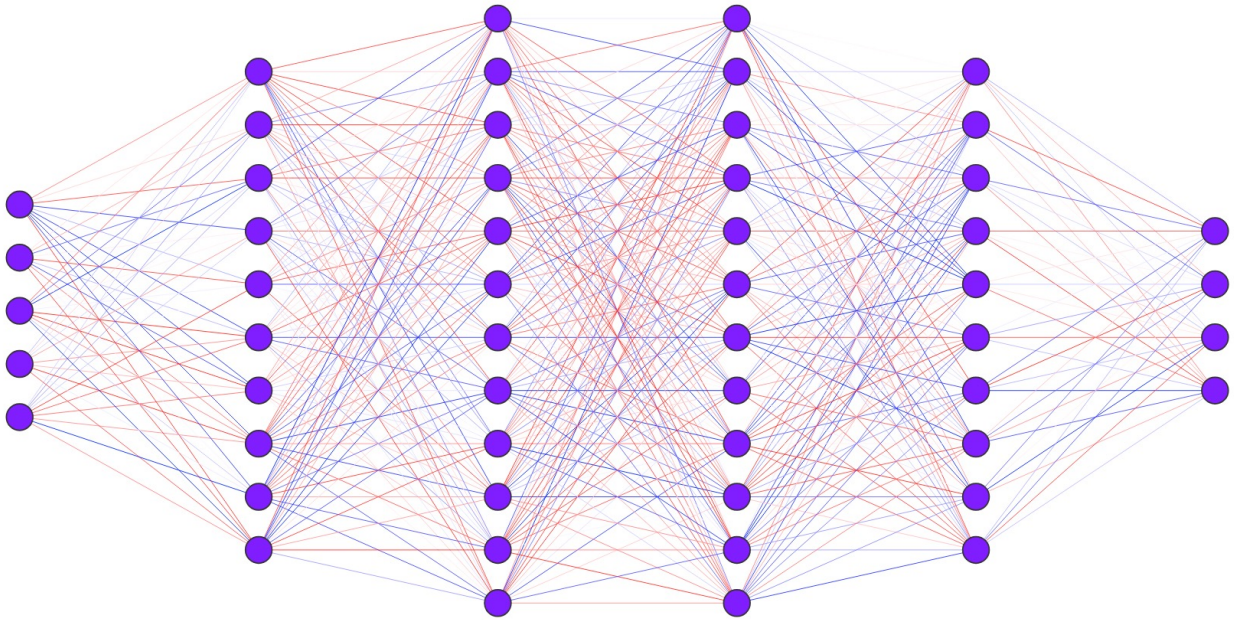
II. Stochastic gradient descent and random matrix theory

- GA, B Lucini, **Chanju Park**, *Phys Rev E* **111** (2025) 1, 015303 [2407.16427 [cond-mat.dis-nn]]
- GA, O Hajizadeh, B Lucini, **Chanju Park**, *ML4PS NeurIPS* 2024, 2411.13512 [cond-mat.dis-nn]

Feed-forward neural network

supervised learning: data set $\mathcal{D} = \{\boldsymbol{x}, \boldsymbol{y}\}$, input $\boldsymbol{x}$ and associated output $\boldsymbol{y}$

input data $\boldsymbol{x}$



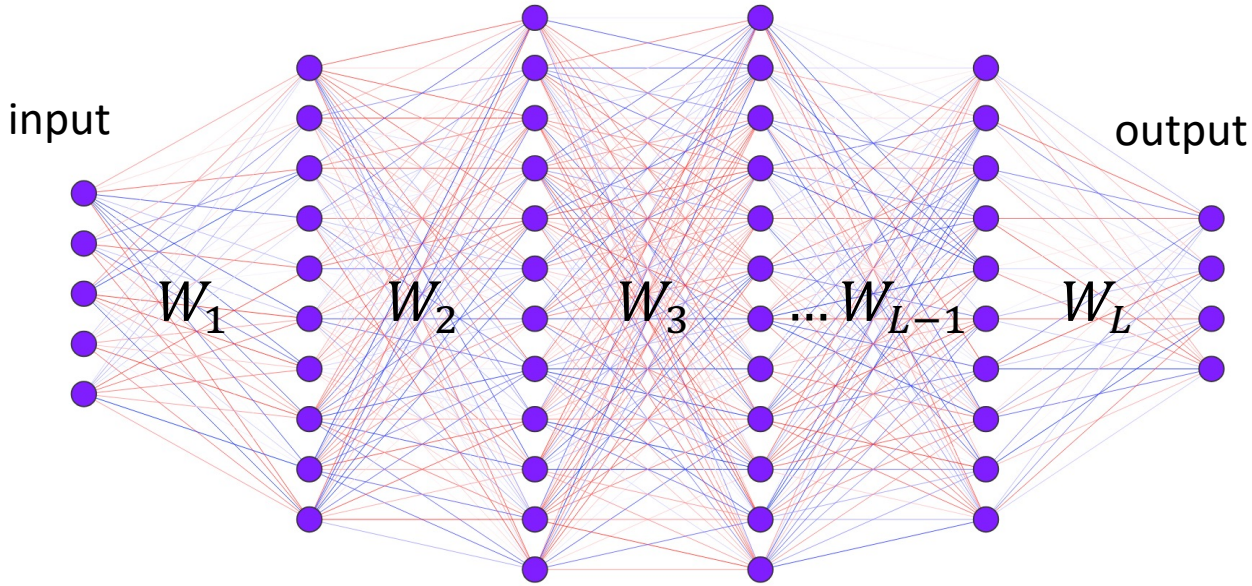
output $\hat{\boldsymbol{y}} = f(\boldsymbol{x})$

NN is a "universal approximator": $\hat{\boldsymbol{y}} \sim \boldsymbol{y}$, should be able to generalise, predict $\boldsymbol{y}$ for unseen data $\boldsymbol{x}$

combination of linear transformations (matrices) and nonlinear activations on the nodes

Stochastic gradient descent

- learn by minimising loss function
- network parameters are contained in (rectangular) weight matrices
- stochastic gradient descent: update weight matrices by averaging gradient of loss over mini-batches of data



$$W'_{ij} = W_{ij} - \alpha \frac{\partial \mathcal{L}(\theta)}{\partial W_{ij}} \qquad \delta W_{\mathcal{B}} = \frac{1}{|\mathcal{B}|} \sum_{b \in \mathcal{B}} \delta W_b \qquad \theta = \{W^{(1)}, \dots, W^{(L)}\}$$

- hyperparameters: learning rate and batch size: $\alpha, |\mathcal{B}|$

Stochastic weight matrix dynamics

- update using stochastic gradient descent: $W \to W' = W + \delta W$ with $\delta W = -\alpha \frac{\delta \mathcal{L}}{\delta W}$
- estimated using a batch $\mathcal{B}$ with batch size: $\delta W_{\mathcal{B}} = \frac{1}{|\mathcal{B}|} \sum_{b \in \mathcal{B}} \delta W_b$
- fluctuations controlled by finite batch size (central limit theorem): $\frac{1}{\sqrt{|\mathcal{B}|}} \sqrt{\mathbb{V}[\delta W)]}$
- update is stochastic: $\delta W = \mathbb{E}[\delta W] + \frac{1}{\sqrt{|\mathcal{B}|}} \sqrt{\mathbb{V}[\delta W)]}\, \eta$ with $\eta_{ij} \sim \mathcal{N}(0,1)$
- in terms of the gradient of the loss function

$$W' = W - \alpha \mathbb{E}\left[\frac{\delta \mathcal{L}}{\delta W}\right] + \frac{\alpha}{\sqrt{|\mathcal{B}|}} \sqrt{\mathbb{V}\left[\frac{\delta \mathcal{L}}{\delta W}\right]}\, \eta$$

mean variance

Stochastic matrix dynamics

- work with symmetric combination $X = W^T W$ (real and positive eigenvalues)
- what is the framework to consider stochastic matrix dynamics?
- goes back to **Wigner** (1955) and **Dyson** (1962): random matrix theory (RMT)
- stochastic matrix dynamics: Dyson Brownian motion (**Dyson**, 1962)
- first applied to nuclear spectra (1950/60s)
- applied in (lattice) QCD to spectrum of Dirac operator (1990-2000s)

Stochastic matrix dynamics: Dyson Brownian motion and the Coulomb gas

- framework to consider stochastic matrix dynamics for symmetric matrix X

- Dyson Brownian motion (in continuous time):

$$\frac{dX_{ij}}{dt} = K_{ij}(X) + \sqrt{A_{ij}}\eta_{ij}$$

- eigenvalues then evolve according to

$$\frac{dx_i}{dt} = K_i(x_i) + \sum_{j \neq i} \frac{g_i^2}{x_i - x_j} + \sqrt{2} g_i \eta_i \qquad \sqrt{A_{ii}} = \sqrt{2} g_i$$

Coulomb repulsion

→ Wigner surmise for level spacing (universal)

Dyson Brownian motion and Coulomb gas

- eigenvalues dynamics $$\frac{dx_i}{dt} = K_i(x_i) + \sum_{j \neq i} \frac{g_i^2}{x_i - x_j} + \sqrt{2} g_i \eta_i$$
- stationary distribution for distribution of eigenvalues (via Fokker-Planck equation)

$$P_s(\{x_i\}) = \frac{1}{Z} \prod_{i<j} |x_i - x_j| \, e^{-\sum_i V_i(x_i)/g_i^2}$$

- with partition function $Z = \int dx_1 \dots dx_N \, P_s(\{x_i\})$ and drift $K_i(x_i) = -\frac{dV_i(x_i)}{dx_i}$

→ Coulomb gas

Back to weight matrix dynamics

- stochastic dynamics $X \to X' = X + \mathbb{E}\left[\delta X\right] + \frac{1}{\sqrt{|\mathcal{B}|}}\sqrt{\mathbb{V}\left[\delta X\right)]}\,\eta$
- what can be carried over from Dyson's matrix dynamics? implications? universality?
- eigenvalue equation, with explicit learning rate and batch size dependence

$$x_i \to x_i' = x_i + \alpha K_i + \frac{\alpha^2}{|\mathcal{B}|}\sum_{j\neq i}\frac{\tilde{g}_i^2}{x_i - x_j} + \frac{\alpha}{\sqrt{|\mathcal{B}|}}\sqrt{2}\tilde{g}_i\eta_i$$

- only continuous time limit (SDE) in some weak sense $\alpha, |\mathcal{B}| \to 0$

$$\frac{dx_i(t)}{dt} = K_i(x,t) + T\sum_{j\neq i}\frac{\tilde{g}_i^2}{x_i - x_j} + \sqrt{2T}\tilde{g}_i\eta_i \qquad \alpha/|\mathcal{B}| \equiv T$$

$$\frac{dx_i(t)}{dt} = K_i(x,t) + T\sum_{j\neq i}\frac{\tilde{g}_i^2}{x_i - x_j} + \sqrt{2T}\tilde{g}_i\eta_i$$

Coulomb gas: effective temperature

- distribution for fixed $\alpha, |\mathcal{B}|$: $P_s(\{x_i\}) = \frac{1}{Z}\prod_{i<j}|x_i - x_j|\, e^{-\sum_i V_i(x_i)/g_i^2}$
- make explicit dependence on learning rate and batch size

$$g_i = \frac{\alpha}{\sqrt{|\mathcal{B}|}}\tilde{g}_i \qquad V_i(x_i) = \alpha\tilde{V}_i(x_i) \qquad \frac{V_i(x_i)}{g_i^2} = \frac{1}{\alpha/|\mathcal{B}|}\frac{\tilde{V}_i(x_i)}{\tilde{g}_i^2}$$

- exponent scales with effective temperature: universal scaling

$$T = \alpha/|\mathcal{B}|$$

- potential itself is problem, i.e. loss function, dependent $K_i = -\frac{dV_i(x_i)}{dx_i} \sim \frac{\partial\mathcal{L}}{\partial W_{ij}}\Big|_{\text{diag}}$

Linear scaling relation

- dependence on $\alpha/|\mathcal{B}|$ in training has been observed before, empirically

✓ P. Goyal, P. Dollár, R.B. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola et al.,
Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour [1706.02677]

✓ S.L. Smith and Q.V. Le,
A Bayesian Perspective on Generalization and Stochastic Gradient Descent [1710.06451]

✓ S.L. Smith, P. Kindermans and Q.V. Le,
Don't Decay the Learning Rate, Increase the Batch Size [1711.00489]

- finds a natural place in the framework of Dyson Brownian motion and Coulomb gas

GA, B Lucini, **Chanju Park**, PRE 111 (2025) 1, 015303 [2407.16427 [cond-mat.dis-nn]]

Manifestations of RMT in weight matrix dynamics

RMT predicts universal behaviour:

- universal distribution of level spacing $S_i = x_{i+1} - x_i$: Wigner surmise $P(S)$
- Coulomb repulsion of eigenvalues
- spectral density is problem-specific $\rho(x) = \left\langle \frac{1}{N} \sum_{i=1}^{N} \delta(x - x_i) \right\rangle$
- universal behaviour has indeed been observed for variety of ML algorithms and data sets

here: explore emergence of temperature further

III. Learning capability of neural networks and phase diagram of disordered systems

- **Chanju Park**, B Lucini, GA, *Mach Learn Sci Tech* **6** (2025) 4, 045048 [2509.01349 [cond-mat.dis-nn]]

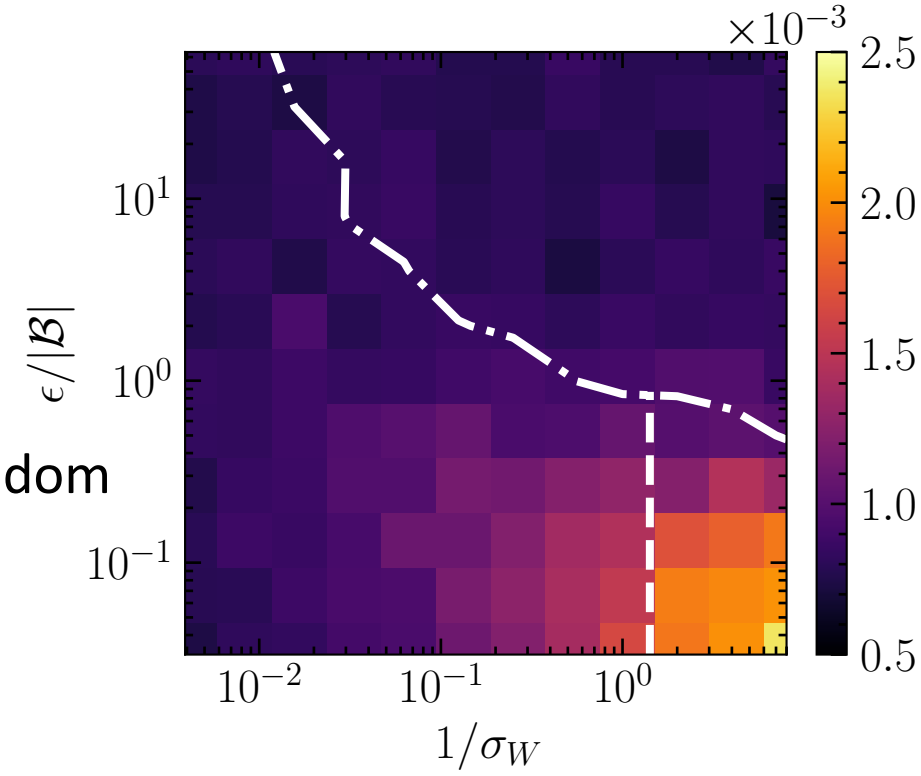
Neural networks as disordered systems

how well do NNs learn?

- dependence on hyperparameters:
  - learning rate
  - batch size
  - weight matrix initialisation

- treat as statistical system with fluctuating degrees of freedom

- already identified temperature $T = \alpha/|\mathcal{B}|$

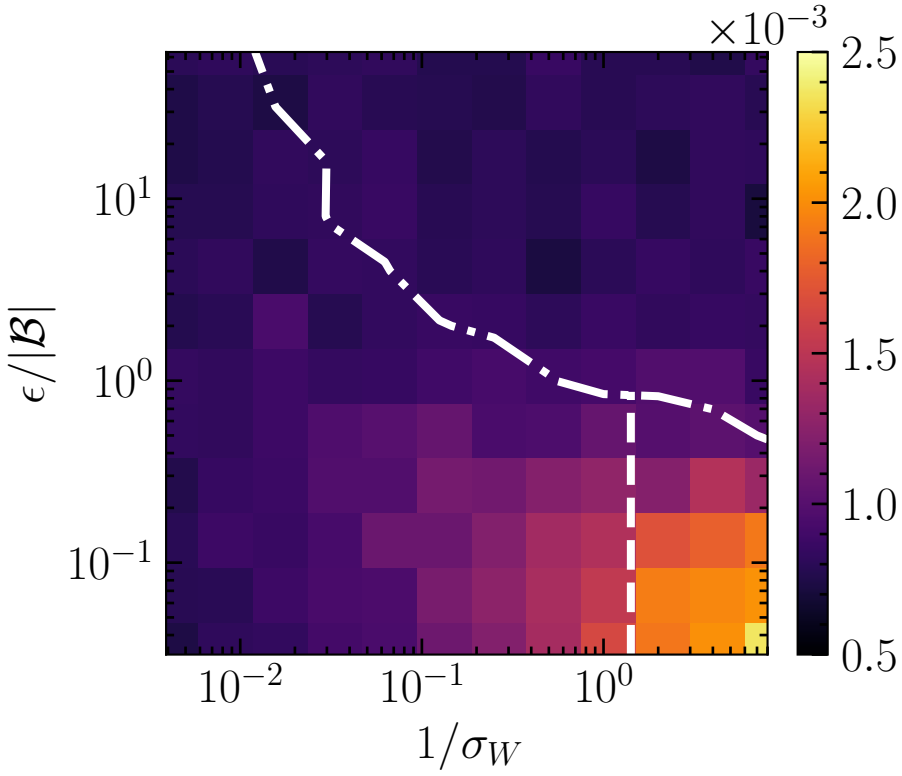


→ arrive at NN phase diagram

note: learning rate denoted as ϵ from now on

Hyperparameters: initialisation

- already discussed $T = \epsilon/|\mathcal{B}|$
- weight matrices need to be initialised
- usual choice $W_{ij} \sim \mathcal{N}(0, \sigma_W^2/n_{l-1})$
- introduces another hyperparameter σ_W



will demonstrate that a NN phase diagram emerges in plane spanned by these

- draw analogy to disordered systems and spin glasses

note: learning rate denoted as ϵ from now on

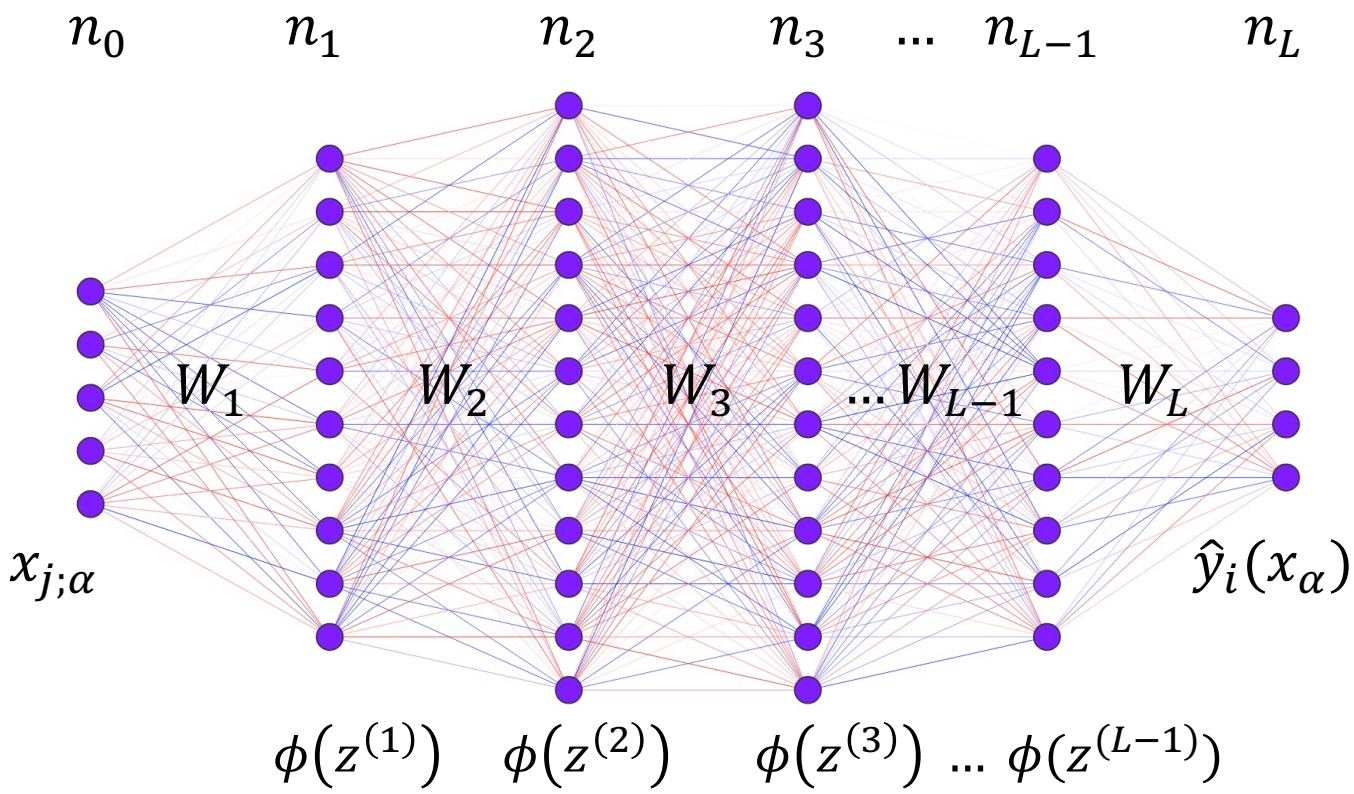
Feed-forward neural network: explicit function

NN function: $$\hat{y}_i(x_\alpha;\theta) \equiv z_i^{(L)}(x_\alpha) = \sum_{j=1}^{n_{L-1}} W_{ij}^{(L)} \phi\left(z_j^{(L-1)}(x_\alpha)\right)$$

pre-activations:

$$z_i^{(l+1)}(x_\alpha) = \sum_{j=1}^{n_l} W_{ij}^{(l+1)} \phi\left(z_j^{(l)}(x_\alpha)\right)$$

$$z_i^{(1)}(x_\alpha) = \sum_{j=1}^{n_0} W_{ij}^{(1)} x_{j;\alpha}$$

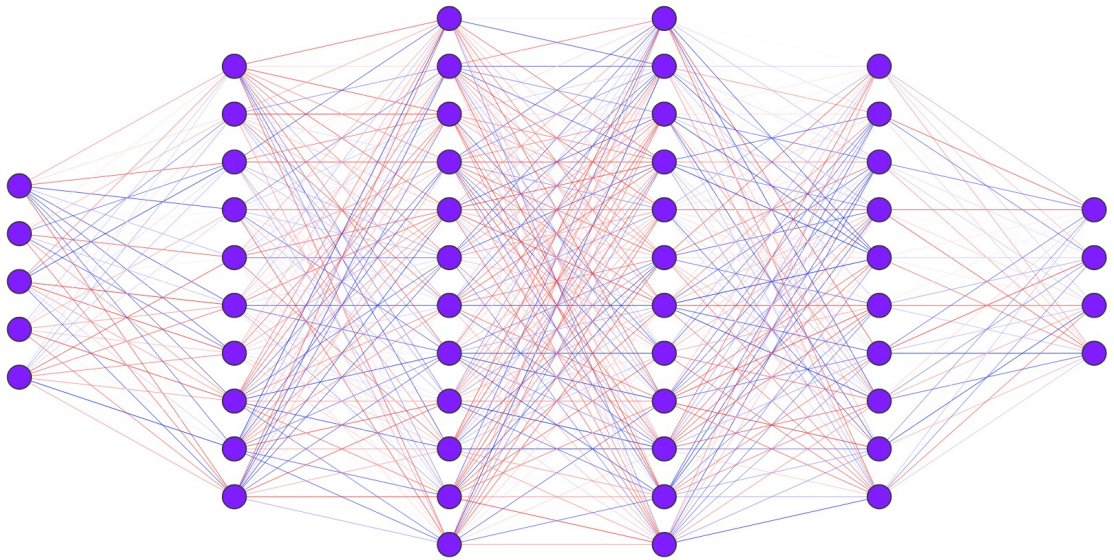


Loss function example: mean squared error

$$\mathcal{L}(\theta) \equiv \frac{1}{|\mathcal{D}|} \sum_{\alpha=1}^{|\mathcal{D}|} \ell\left(y(x_\alpha), \hat{y}(x_\alpha; \theta)\right)$$

$$\ell(y, \hat{y}) \equiv \frac{1}{2} \sum_{i=1}^{n_L} (y_i - \hat{y}_i)^2$$

activations on final hidden layer: features $\phi_{j\alpha} \equiv \phi\left(z_j^{(L-1)}(x_\alpha)\right)$



Loss function example: mean squared error

$$\mathcal{L}(\theta) \equiv \frac{1}{|\mathcal{D}|} \sum_{\alpha=1}^{|\mathcal{D}|} \ell\left(y(x_\alpha), \hat{y}(x_\alpha; \theta)\right) \qquad \ell(y, \hat{y}) \equiv \frac{1}{2} \sum_{i=1}^{n_L} (y_i - \hat{y}_i)^2$$

activations on final hidden layer: features $\phi_{j\alpha} \equiv \phi\left(z_j^{(L-1)}(x_\alpha)\right)$

network prediction is linear combination of features

$$\begin{aligned}
\mathcal{L}(\theta) &= \frac{1}{2|\mathcal{D}|} \sum_{\alpha=1}^{\mathcal{D}} \sum_{i=1}^{n_L} \left(y_{i\alpha} - \sum_{j=1}^{n_{L-1}} W_{ij}^{(L)} \phi_{j\alpha} \right)^2 \\
&= \frac{1}{2|\mathcal{D}|} \sum_{\alpha=1}^{|\mathcal{D}|} \sum_{i=1}^{n_L} \left(\sum_{j,k=1}^{n_{L-1}} W_{ij}^{(L)} W_{ik}^{(L)} \phi_{j\alpha} \phi_{k\alpha} - 2 \sum_{j=1}^{n_{L-1}} y_{i\alpha} W_{ij}^{(L)} \phi_{j\alpha} + y_{i\alpha} y_{i\alpha} \right)
\end{aligned}$$

express loss function as function of features

$$= \frac{1}{2|\mathcal{D}|} \sum_{\alpha=1}^{|\mathcal{D}|} \sum_{i,j=1}^{n_{L-1}} J_{ij} \phi_{i\alpha} \phi_{j\alpha} - \frac{1}{|\mathcal{D}|} \sum_{\alpha=1}^{|\mathcal{D}|} \sum_{j=1}^{n_{L-1}} h_{j\alpha} \phi_{j\alpha} + C$$

Neural network as a disordered system

loss function as a function of features:

$$\mathcal{L}(\theta) = \frac{1}{2|\mathcal{D}|} \sum_{\alpha=1}^{|\mathcal{D}|} \sum_{i,j=1}^{n_{L-1}} J_{ij} \phi_{i\alpha} \phi_{j\alpha} - \frac{1}{|\mathcal{D}|} \sum_{\alpha=1}^{|\mathcal{D}|} \sum_{j=1}^{n_{L-1}} h_{j\alpha} \phi_{j\alpha}$$

with couplings:

$$J_{ij} \equiv \sum_{k=1}^{n_L} W_{ki}^{(L)} W_{kj}^{(L)}, \qquad h_{j\alpha} \equiv \sum_{i=1}^{n_L} y_{i\alpha} W_{ij}^{(L)}$$

resembles disordered “spin” system:
(Ising model with local couplings)

$$\mathcal{H} = -\frac{1}{2} \sum_{ij} J_{ij} s_i s_j + \sum_j h_j s_j$$

Solvable Model of a Spin-Glass

David Sherrington* and Scott Kirkpatrick
IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598
(Received 16 October 1975)

We consider an Ising model in which the spins are coupled by infinite-ranged random interactions independently distributed with a Gaussian probability density. Both "spin-glass" and ferromagnetic phases occur. The competition between the phases and the type of order present in each are studied.

$$\mathcal{H} = -\frac{1}{2}\sum_{ij} J_{ij} s_i s_j + \sum_j h_j s_j$$

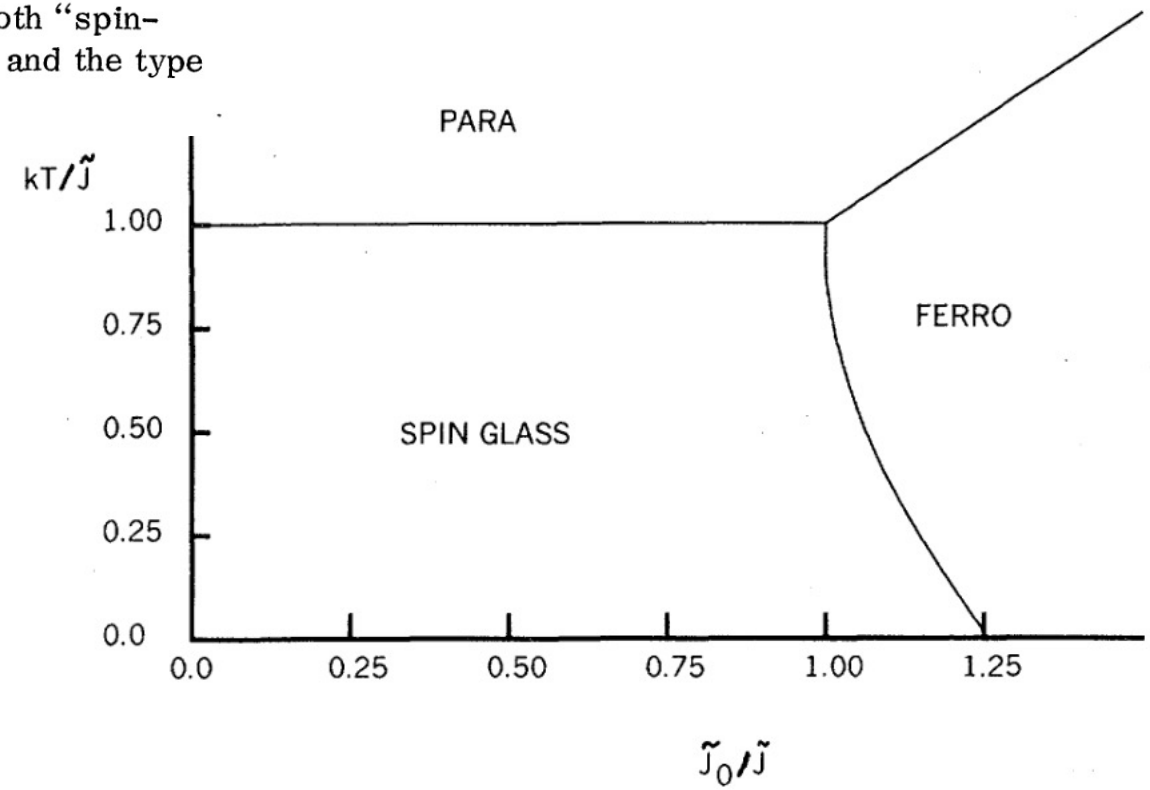


FIG. 1. Phase diagram of spin-glass ferromagnet.

Neural network phase diagram

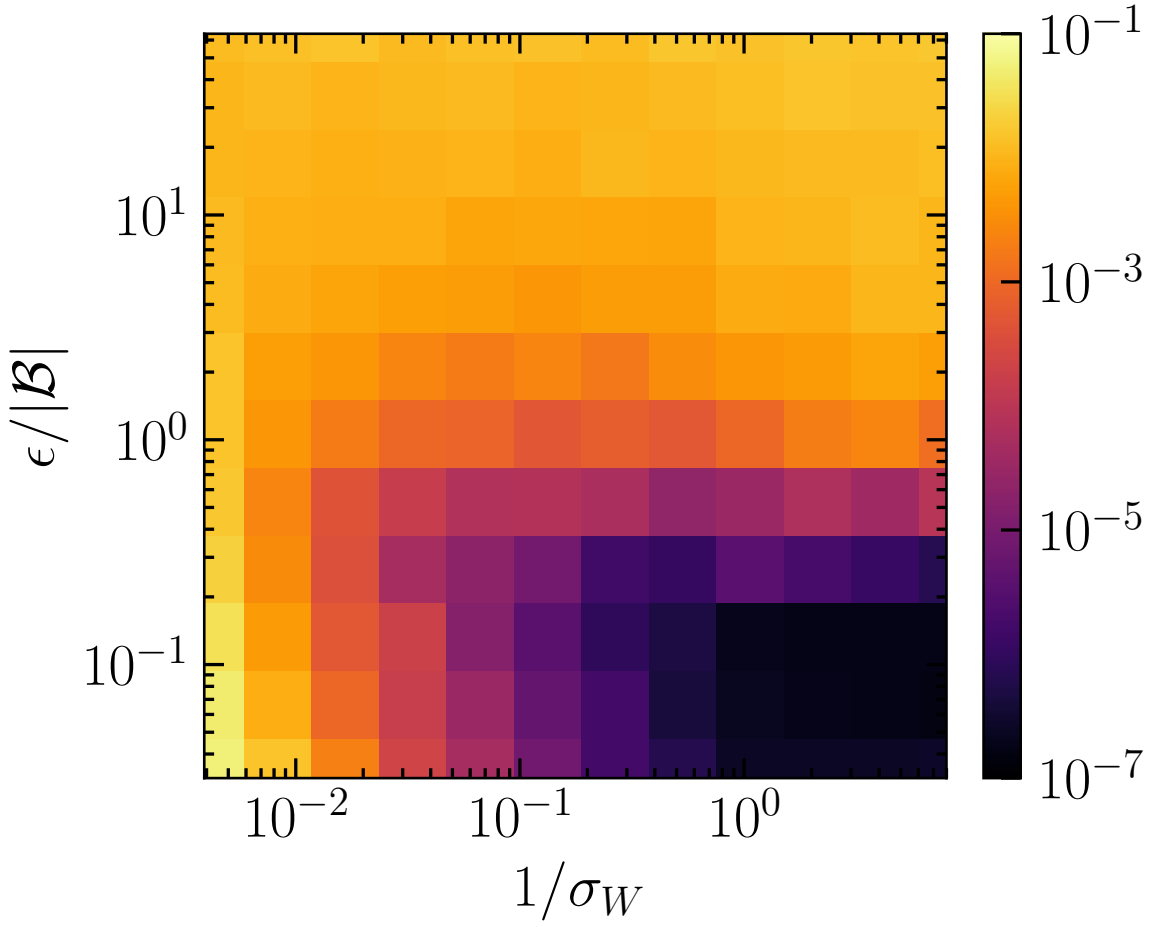
- trained a NN in the teacher-student setup
- teacher network has fixed weight matrices, student has to learn those, or equivalent ones
- two hidden layers [3,32,16,1]
- vary learning rate/batch size $T = \epsilon/|\mathcal{B}|$
- vary initial weight matrix variance $W_{ij}^{(l)} \sim \mathcal{N}\left(0, \sigma_W^2/n_{l-1}\right)$
- 100 runs for each choice of parameter combination
- monitor number of ‘’observables’’, loss, grad loss, feature alignment, …

NN phase diagram: mean test loss

effective temperature
~ learning rate/batch size

$$T = \epsilon/|\mathcal{B}|$$

no learning, jamming
~ spin glass phase



no convergence
(loss is large)
~ paramagnetic phase

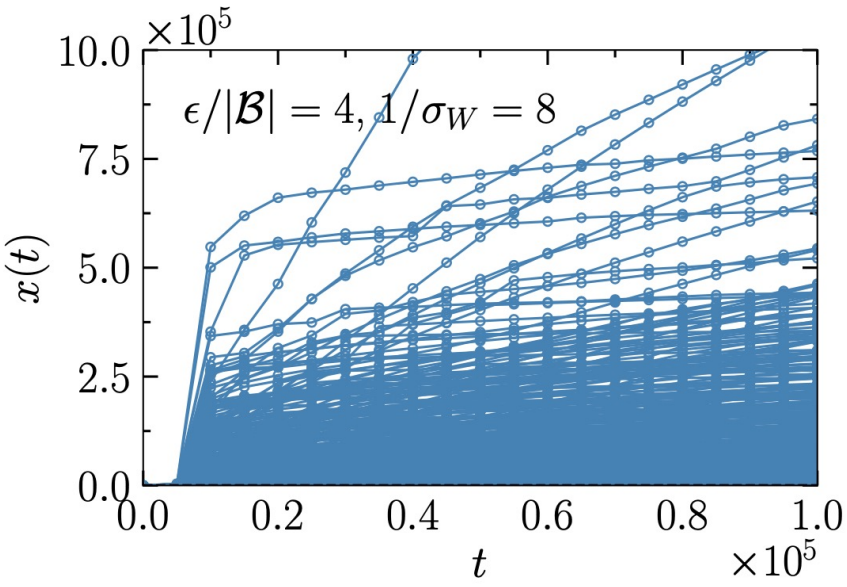
excellent learning
(loss is small)
~ ferromagnetic phase

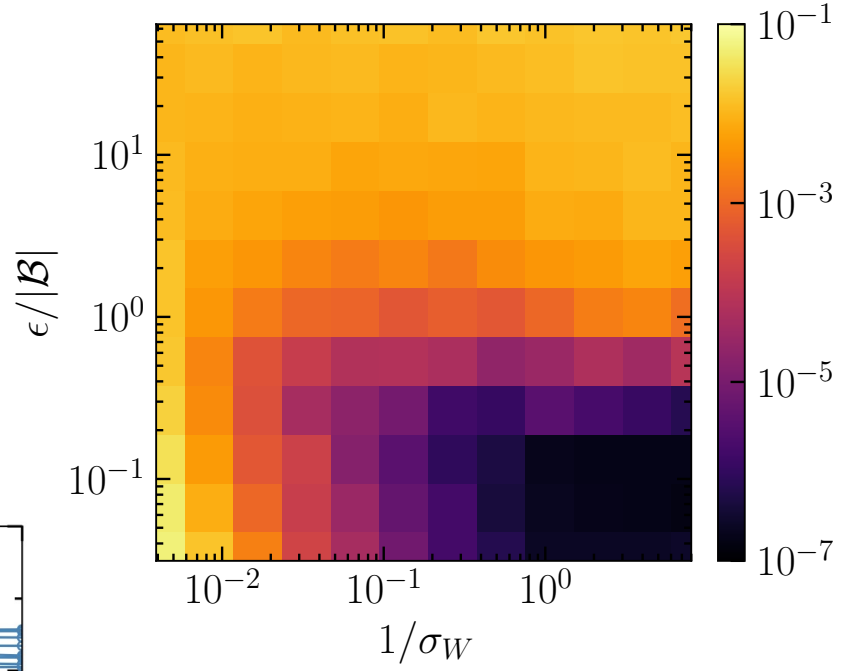
disorder ~ variance of weight matrices upon initialisation

Three distinct phases

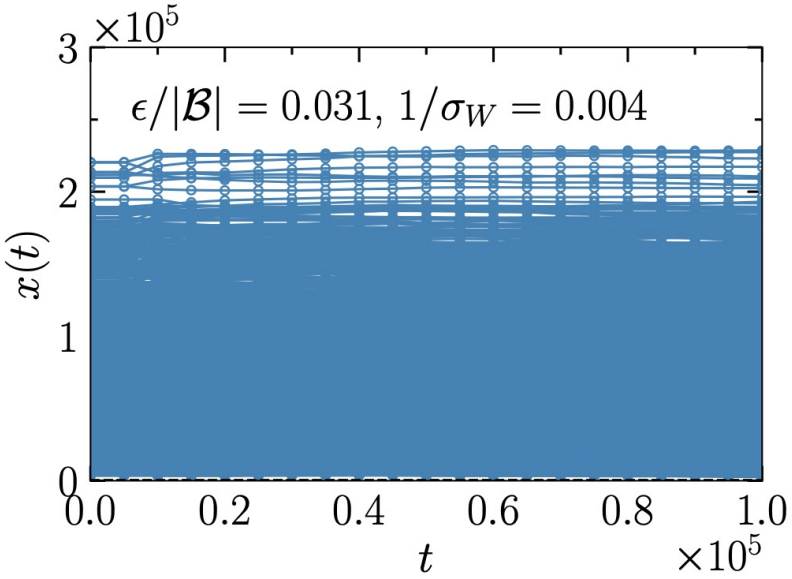
evolution of singular values
of weight matrices

divergence

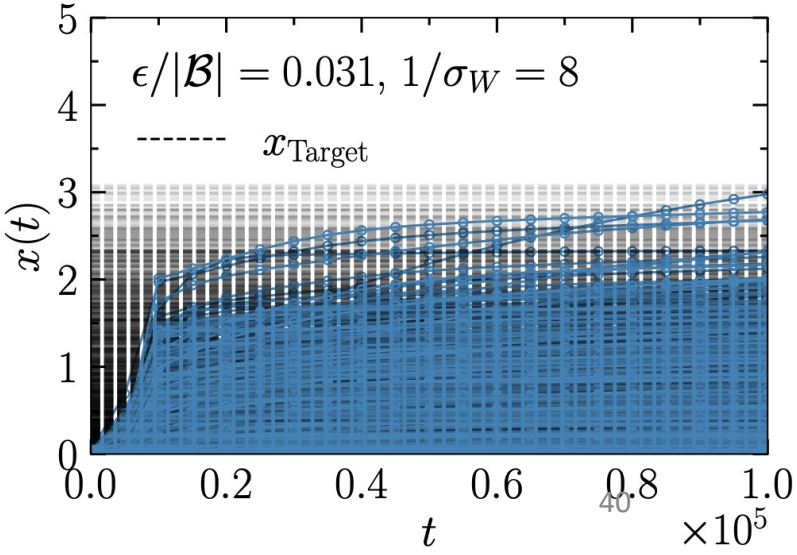




jamming



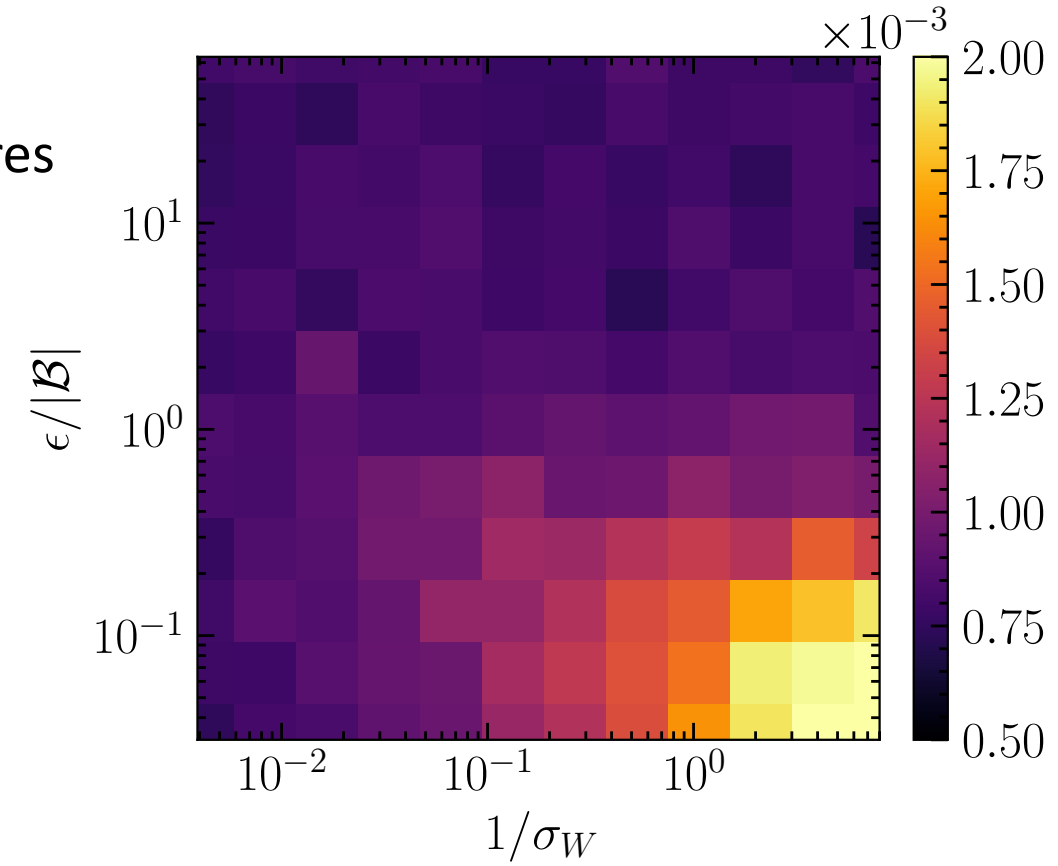
convergence



NN phase diagram: external field alignment

$$\mathcal{L}(\theta) = \frac{1}{2|\mathcal{D}|}\sum_{\alpha=1}^{|\mathcal{D}|}\sum_{i,j=1}^{n_{L-1}} J_{ij}\phi_{i\alpha}\phi_{j\alpha} - \frac{1}{|\mathcal{D}|}\sum_{\alpha=1}^{|\mathcal{D}|}\sum_{j=1}^{n_{L-1}} h_{j\alpha}\phi_{j\alpha}$$

alignment between features and external field



$h \parallel \phi$

most aligned in ferromagnetic phase

Phase boundaries

find semi-analytical expression for the phase boundaries

- between ordered and disordered phase: (in)ability to learn, vanishing gradient problem

  ➔ features as “soft spins”

- between paramagnetic and other phases: strong fluctuations, no convergence

  ➔ signal-to-noise ratio of gradient of loss

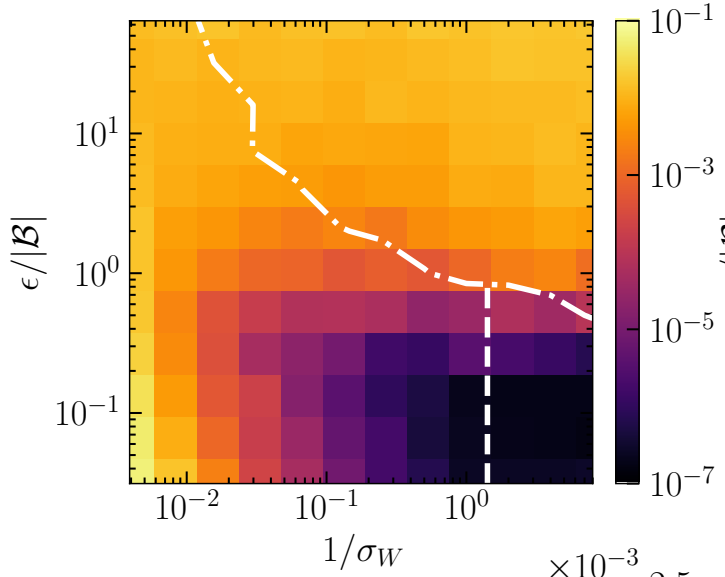
- see paper for details

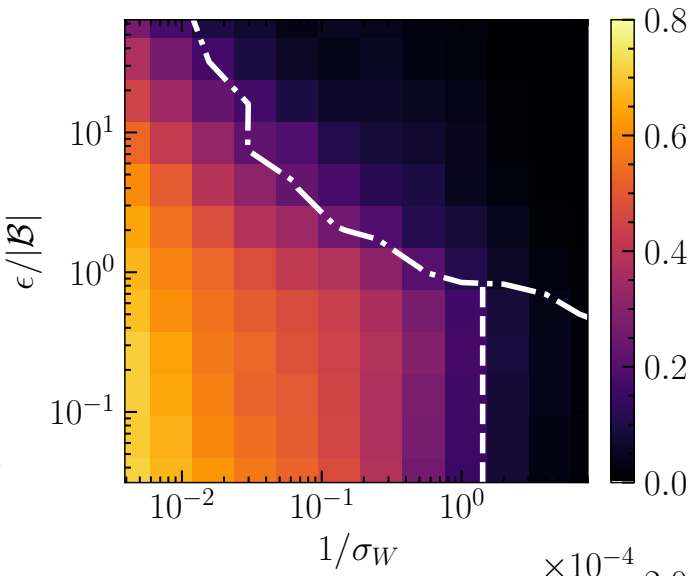
Chanju Park, B Lucini, GA, *Mach Learn Sci Tech* **6** (2025) 4, 045048 [2509.01349 [cond-mat.dis-nn]]

Collection of phase diagrams

different observables probe different aspects of the dynamics

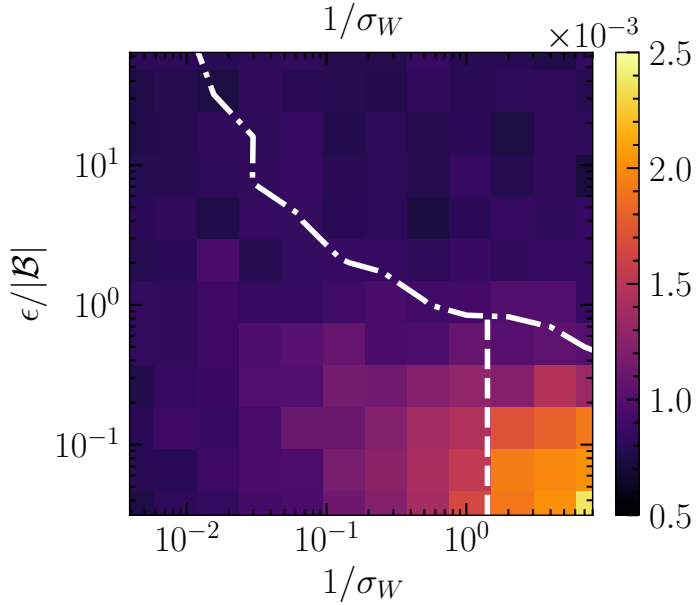
mean test loss

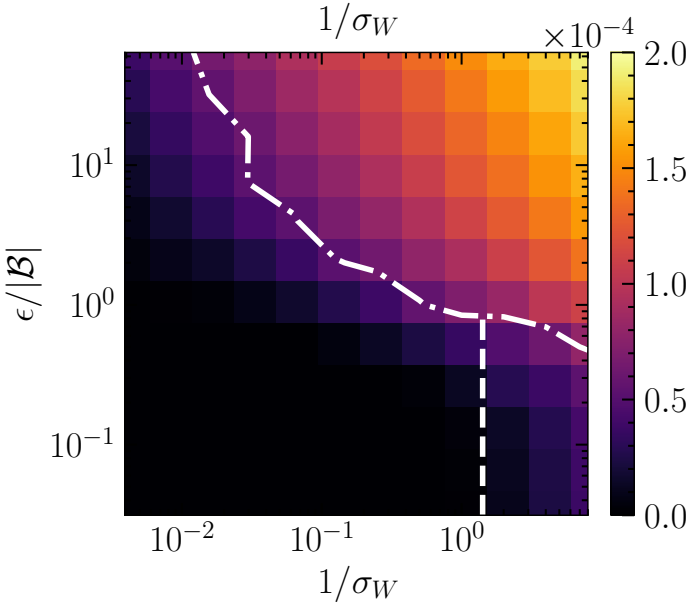




time correlation of features

feature alignment





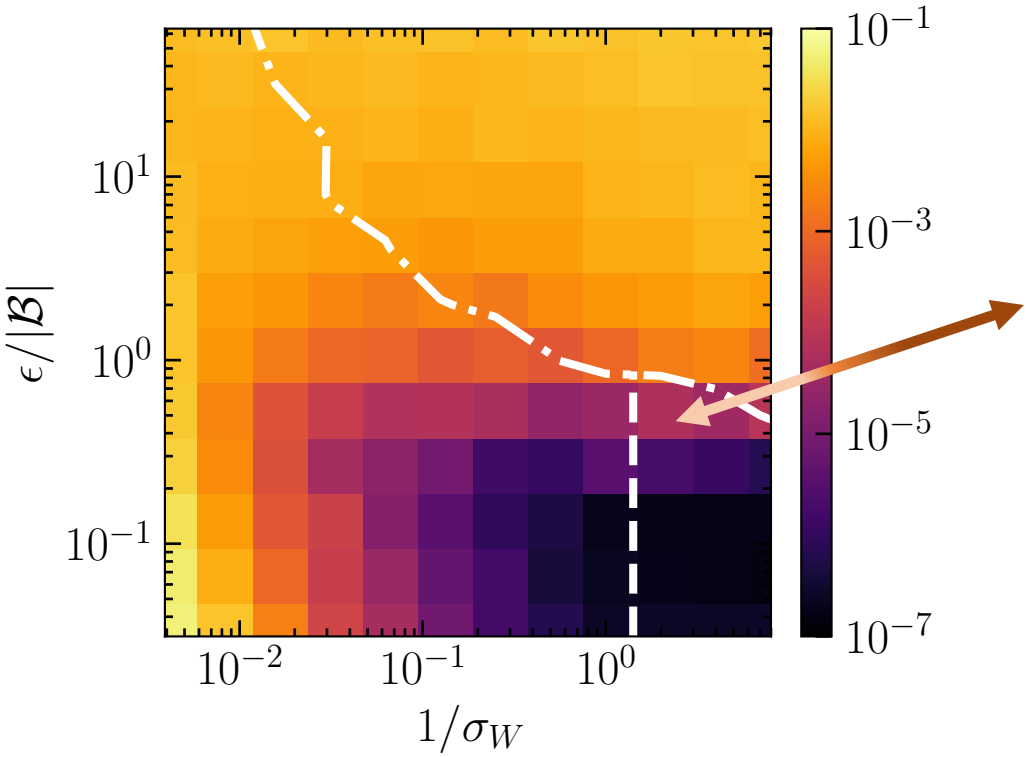
time-averaged rate of average level spacing

$$\overline{\lambda}(t_f) = \frac{1}{t_f} \log \frac{S(t_f)}{S(0)}$$

Practical application: choice of hyperparameters

- identification of ferro, paramagnetic and jammed or spin glass phases
- helps in understanding which choice of hyperparameters is preferred

mean test loss



choose as large $T = \epsilon/|\mathcal{B}|$ as possible with initial variance of weight matrices around 1

Summary: physics of machine learning

- deep connection between paradigms in ML and theoretical physics

three examples:

  - diffusion models and stochastic quantisation
  - stochastic gradient descent and random matrix theory
  - learning efficiency and phase diagrams

- all provide useful insight to understand and improve methods
- many opportunities ahead for theoretical physicists to contribute

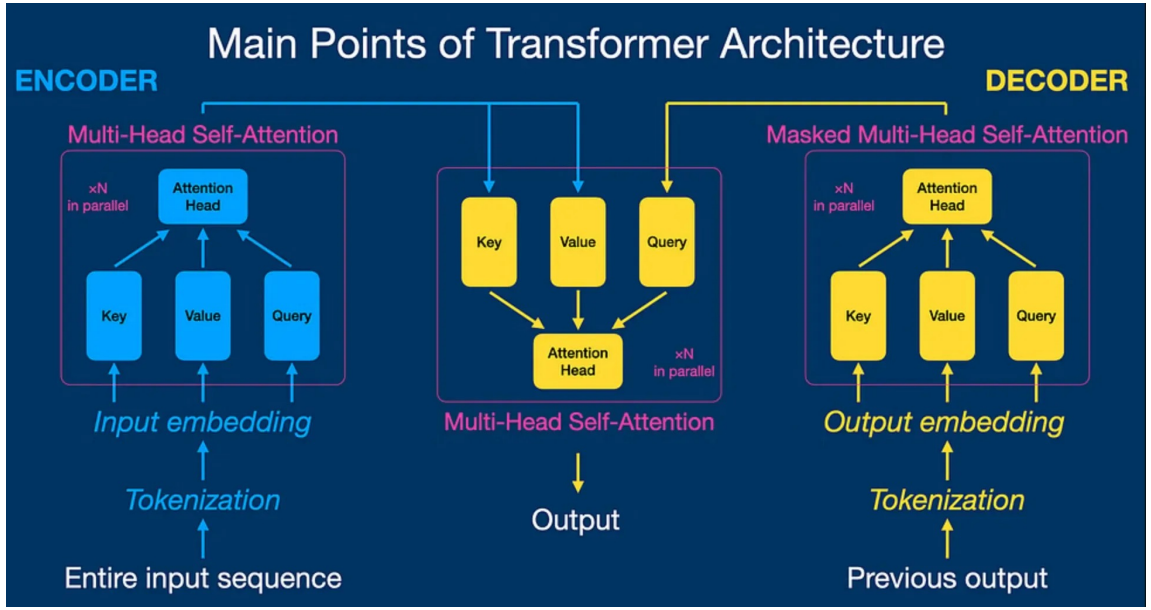
BACKUP SLIDES

Transformer: nano-GPT

- four attention blocks with each four attention heads: many matrices

- each attention head:
  - one key (K) matrix
  - one query (Q) matrix
  - one value (V) matrix

- matrix sizes: $M \times N = 64 \times 16$
- about 2.1×10^5 parameters
- use AdamW optimiser
- trained on opus of Shakespeare

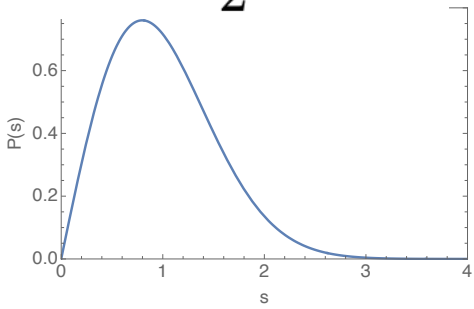


https://botpenguin.com/glossary/transformer-architecture

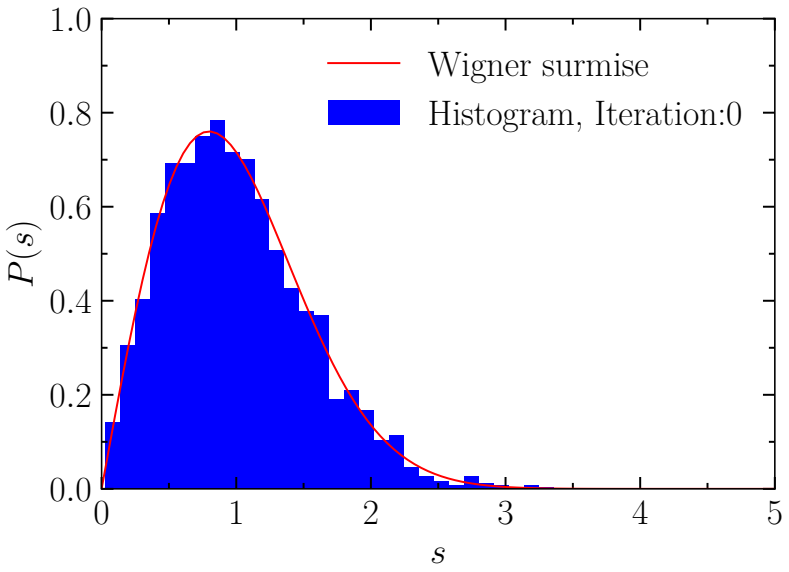
GA, O Hajizadeh, B Lucini, C Park, NeurIPS 2024 workshop *ML and the Physical Sciences*, 2411.13512 [cond-mat.dis-nn]

Transformer: Wigner surmise

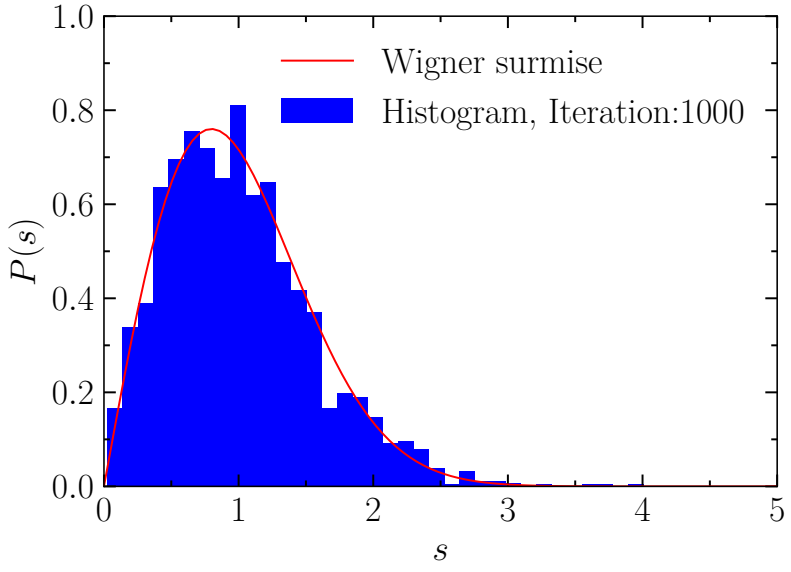
$$P(s) = \frac{\pi}{2} s e^{-\pi s^2/4}$$



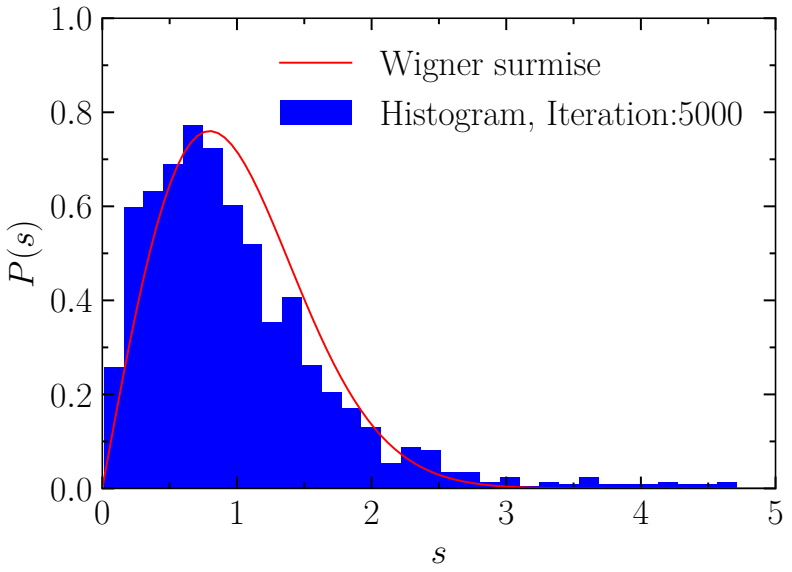
- short-distance fluctuations: level spacing described by Wigner surmise
- remains approximately described by RMT prediction (shown K matrix of layer 1)



iteration 0



iteration 1000



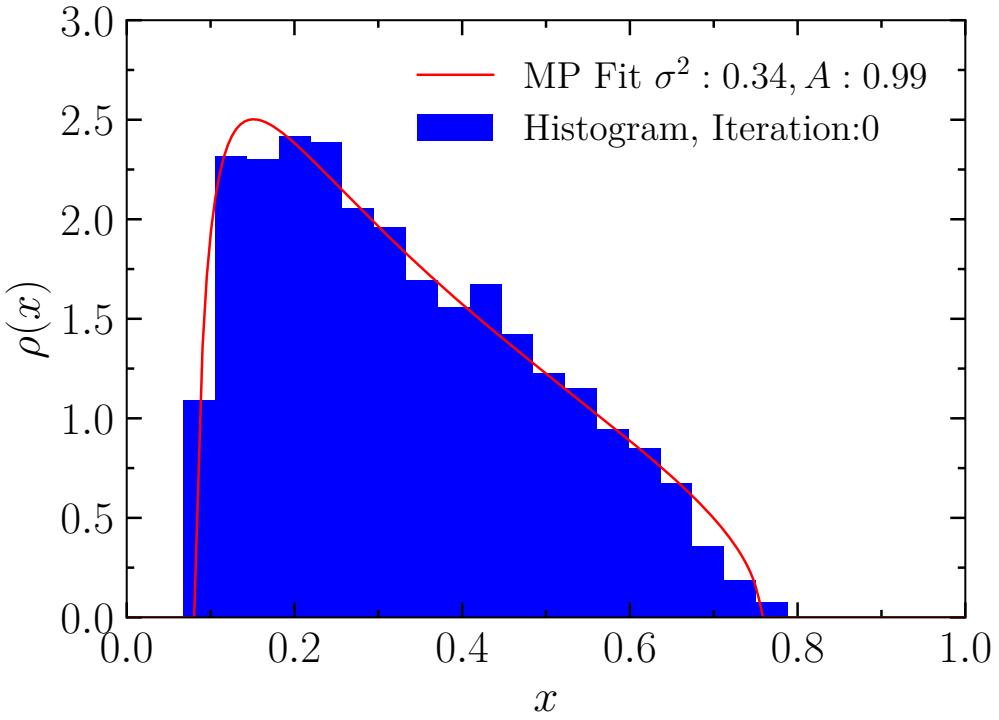
iteration 5000

GA, O Hajizadeh, B Lucini, C Park, NeurIPS 2024 workshop *ML and the Physical Sciences*, 2411.13512 [cond-mat.dis-nn]

Transformer: spectral density

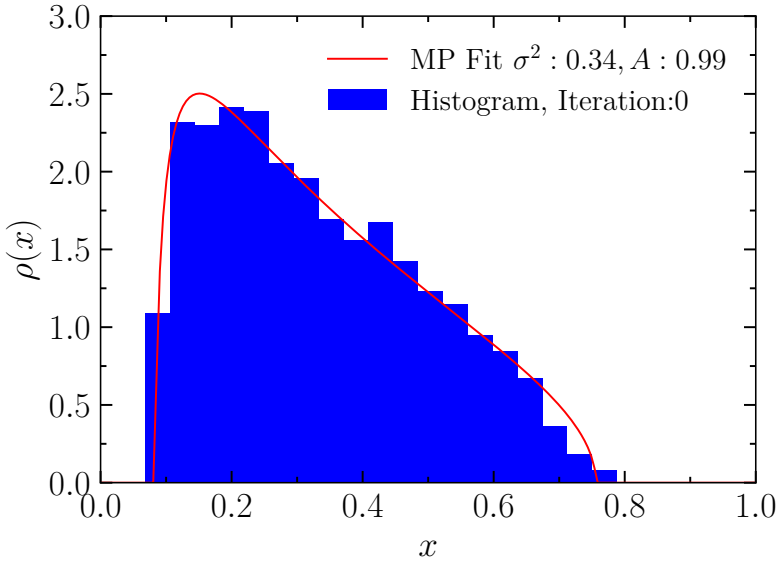
- initialisation: eigenvalues of $X = W^T W$ follow Marchenko-Pastur distribution

$$P_{\mathrm{MP}}(x;\sigma^2,A) = \frac{A}{2\pi\sigma^2 r x}\sqrt{(x_+ - x)(x - x_-)}\,\theta(x_+ - x)\theta(x - x_-)$$

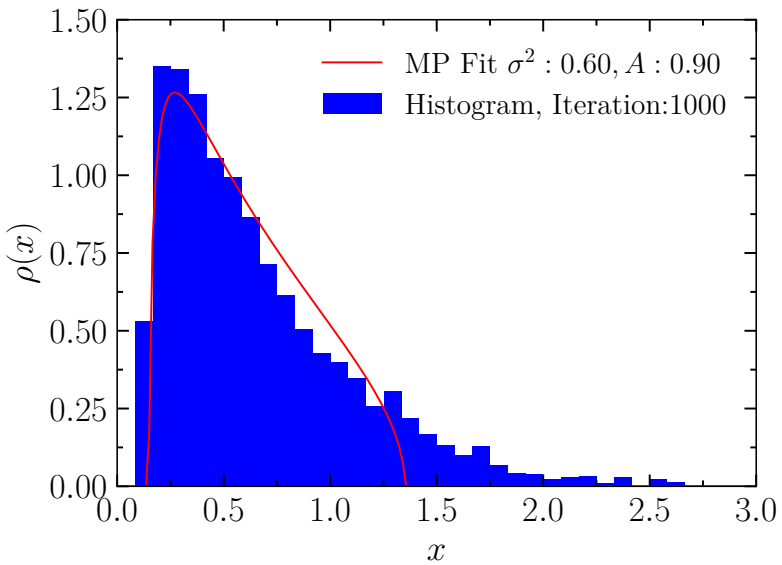


Transformer: spectral density

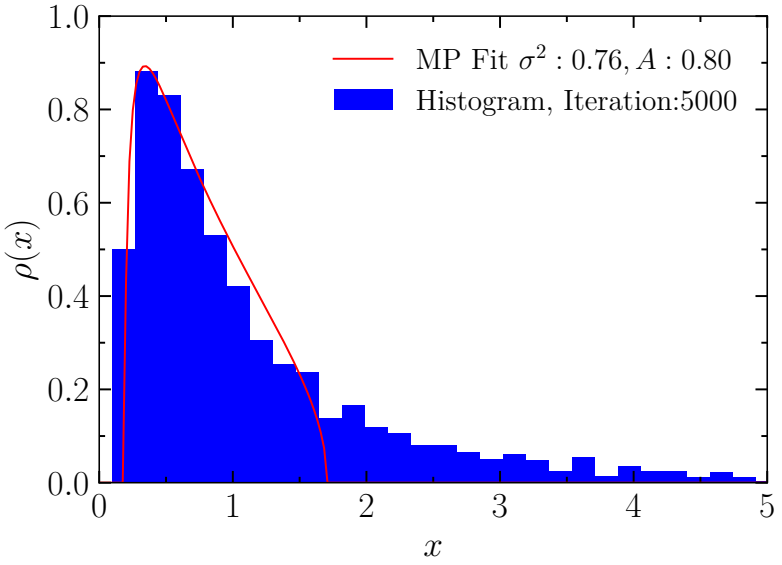
- evolves from initial Marchenko-Pastur distribution to distribution with power decay
- shown K matrix of layer 1



iteration 0



iteration 1000



iteration 5000

Open questions

- Dyson Brownian motion is present at “microscopic” level in weight matrix dynamics
- how does it manifest itself for more advanced architectures?
- is there universality beyond level repulsion (power law tails)?
- phase diagrams of deeper NNs
- practical implications: description of learning, algorithmic advances, …