Buongiorno a lor signori
Ve lo dico tosto,
Mi presento a voi dottori:
Son Ludovico Ariosto.

Maledetta fu la mano
Che muto lo appellativo,
Io son Sebastiano
Ma sarò anche comprensivo.

Good morning to you gentlemen
I tell you quickly,
I introduce myself to you doctors:
I am Ludovico Ariosto.

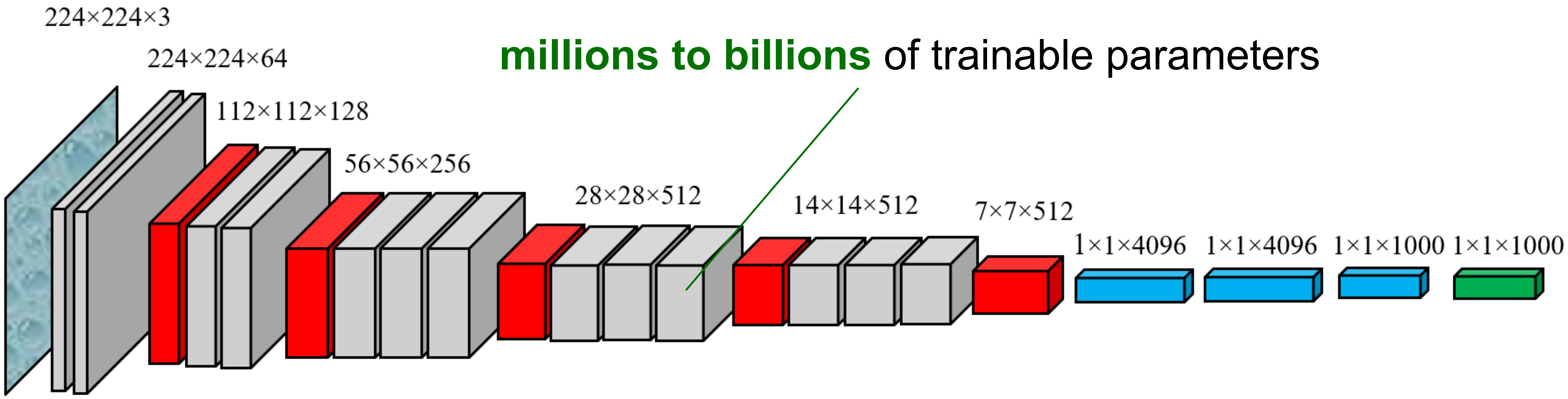
Cursed was the hand
That mute the appellation,
I am Sebastiano
But I will also be sympathetic.

UNIVERSAL MEAN FIELD UPPER BOUND FOR THE GENERALISATION GAP OF DEEP NEURAL NETWORKS

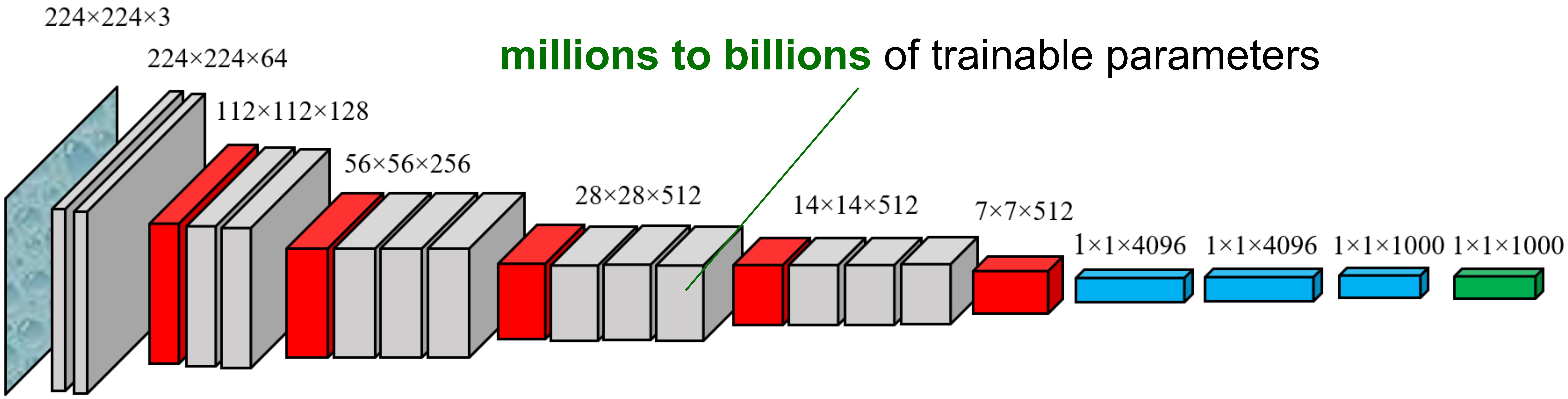
Sebastiano Ariosto — University of Insubria

[SA, R. Pacelli, F. Ginelli, M. Gherardi, P. Rotondo; Phys. Rev. E **105**, 064309]

OVERPARAMETRISATION IN DEEP NETS: A BLESS FOR PRACTITIONERS, A CURSE FOR THEORISTS



OVERPARAMETRISATION IN DEEP NETS: A BLESS FOR PRACTITIONERS, A CURSE FOR THEORISTS



overparametrised regime

number of trainable parameters $\gg$ **size of the training set**

STATISTICAL LEARNING THEORY IN A NUTSHELL: MAIN INGREDIENTS

- $P_{\mathcal{X},\mathcal{Y}}(X,Y)$ **input/output joint probability distribution**

  $\mathcal{X}$ input $\mathcal{Y} = \{+1, -1\}$ output

STATISTICAL LEARNING THEORY IN A NUTSHELL: MAIN INGREDIENTS

- $P_{\mathcal{X},\mathcal{Y}}(X,Y)$ **input/output joint probability distribution**

  $\mathcal{X}$ input $\mathcal{Y}=\{+1,-1\}$ output

  $\mathcal{T}_P=\{X^\mu,Y^\mu\}_{\mu=1,\ldots,P}$ **training set**

STATISTICAL LEARNING THEORY IN A NUTSHELL: MAIN INGREDIENTS

- $P_{\mathcal{X},\mathcal{Y}}(X,Y)$ **input/output joint probability distribution**

  $\mathcal{X}$ input $\mathcal{Y}=\{+1,-1\}$ output

  $\mathcal{T}_P=\{X^\mu,Y^\mu\}_{\mu=1,\ldots,P}$ **training set**

  $f\in\mathcal{F}$ **hypothesis space**

STATISTICAL LEARNING THEORY IN A NUTSHELL: MAIN INGREDIENTS

- $P_{\mathcal{X},\mathcal{Y}}(X,Y)$ **input/output joint probability distribution**

  $\mathcal{X}$ input $\quad \mathcal{Y} = \{+1,-1\}$ output

  $\mathcal{T}_P = \{X^\mu, Y^\mu\}_{\mu=1,\ldots,P}$ **training set**

  $f \in \mathcal{F}$ **hypothesis space**

- $$\epsilon_{\rm g}(f) = \left\langle \mathbf{1}_{f(X)\neq Y} \right\rangle_{P_{\mathcal{X},\mathcal{Y}}}$$

  generalisation error (true risk)

  $$\epsilon_{\rm t}(f) = \frac{1}{P}\sum_{\mu=1}^{P} \mathbf{1}_{f(X^\mu)\neq Y^\mu}$$

  training error (empirical risk)

STATISTICAL LEARNING THEORY IN A NUTSHELL: (ONE) MAIN THEOREM

$$\Delta\epsilon(f) = \epsilon_{\mathrm{g}}(f) - \epsilon_{\mathrm{t}}(f)$$

generalisation gap

STATISTICAL LEARNING THEORY IN A NUTSHELL: (ONE) MAIN THEOREM

$$\Delta\epsilon(f) = \epsilon_{\mathrm{g}}(f) - \epsilon_{\mathrm{t}}(f)$$

generalisation gap

Theorem [Vapnik-Chervonenkis]

For any $\delta > 0$*, with probability at least* $1 - \delta$

$$\forall f \in \mathcal{F}\,, \quad \Delta\epsilon(f) \leq 2\sqrt{2\frac{d_{\mathrm{VC}} \log \frac{eP}{d_{\mathrm{VC}}} + \log\frac{2}{\delta}}{P}}$$

STATISTICAL LEARNING THEORY IN A NUTSHELL: (ONE) MAIN THEOREM

$$\Delta\epsilon(f) = \epsilon_{\mathrm{g}}(f) - \epsilon_{\mathrm{t}}(f)$$ **generalisation gap**

VC dimension: a compact measure of the expressivity of a model $\mathcal{F}$

Theorem [Vapnik-Chervonenkis]

For any $\delta > 0$, *with probability at least* $1 - \delta$

$$\forall f \in \mathcal{F}\,, \quad \Delta\epsilon(f) \leq 2\sqrt{2\frac{d_{\mathrm{VC}}\log\frac{eP}{d_{\mathrm{VC}}} + \log\frac{2}{\delta}}{P}}$$

STATISTICAL LEARNING THEORY IN A NUTSHELL: (ONE) MAIN THEOREM

$$\Delta\epsilon(f) = \epsilon_{\mathrm{g}}(f) - \epsilon_{\mathrm{t}}(f)$$ **generalisation gap**

VC dimension: a compact measure of the expressivity of a model $\mathcal{F}$

Theorem [Vapnik-Chervonenkis]

For any $\delta > 0$*, with probability at least* $1-\delta$

$$\forall f \in \mathcal{F}, \quad \Delta\epsilon(f) \leq 2\sqrt{2\frac{d_{\mathrm{VC}}\log\frac{eP}{d_{\mathrm{VC}}} + \log\frac{2}{\delta}}{P}}$$

- **uniform** in the functions of the model and **data-independent**

STATISTICAL LEARNING THEORY: MAIN LIMITATIONS

$$\Delta\epsilon \lesssim \sqrt{\frac{d_{\mathrm{VC}}}{P}}$$

the VC dimension of a DNN is (very) roughly proportional to the number of trainable parameters $\sim 10^6 - 10^9$

the typical size of a training dataset in a supervised learning problem is of order $\sim 10^4 - 10^6$

STATISTICAL LEARNING THEORY: MAIN LIMITATIONS

$$\Delta\epsilon \lesssim \sqrt{\frac{d_{\mathrm{VC}}}{P}}$$

the VC dimension of a DNN is (very) roughly proportional to the number of trainable parameters $\sim 10^6 - 10^9$

the typical size of a training dataset in a supervised learning problem is of order $\sim 10^4 - 10^6$

"[…] Their derivation reveals many possible causes for their poor quantitative performance:

(i) Practical data distributions may lead to smaller deviations (between the expected and empirical classification error) than the worst possible data distribution.

(ii) Uniform bounds hold for all possible classification functions. Better bounds may hold when one restricts the analysis to functions that perform well on plausible training sets.

(from L. Bottou, "Making Vapnik-Chervonenkis bounds accurate")

MAIN GOAL: Improve this bound with Statistical Physics

THE OTHER MAJOR FRAMEWORK TO INVESTIGATE GENERALISATION: THE TEACHER-STUDENT SCENARIO

$$P(\mathbf{x}, y) = \rho(\mathbf{x})\delta\left(y - f_{\mathrm{T}}(\mathbf{x})\right)$$

input density distribution

$$f_{\mathrm{T}}(\mathbf{x}) = \frac{1}{\sqrt{N_{\mathrm{T}}}} \sum_{\alpha=1}^{N_{\mathrm{T}}} t_\alpha \phi_\alpha^{(\mathrm{T})}(\mathbf{x})$$

a **teacher** provides the ground truth (the label)

$$f_{\mathrm{S}}(\mathbf{x}) = \frac{1}{\sqrt{N_{\mathrm{S}}}} \sum_{\alpha=1}^{N_{\mathrm{S}}} v_\alpha \phi_\alpha^{(\mathrm{S})}(\mathbf{x})$$

a **student** optimises its weights to match the ground truth by minimising a **loss function**

$$\mathscr{L}(v, \mathbf{x}) = \sum_{\mu}^{P} \left(f_T(x^\mu) - f_S(v, x^\mu) \right)^2$$

TEACHER STUDENT SCENARIO: COMPUTATION

Goal: compute the optimal generalisation and training errors for large N_S and P

$$Z(\mathbf{x}) = \int d^{N_s}v \; e^{-\frac{\beta}{2}\mathscr{L}(v,\mathbf{x}) - \frac{\beta\lambda}{2}||v||^2}$$

Partition function

$$\langle \log Z \rangle_{\rho(x)} = \lim_{m\to\infty} \frac{\langle Z^m \rangle_{\rho(x)} - 1}{m}$$

Replica trick

$$\langle Z^m \rangle_{\rho(x)} = \prod_a^m \int d^{N_s} v^a \int d^D x \, \rho(x) \; e^{-\frac{\beta}{2}\mathscr{L}(v^a,\mathbf{x}) - \frac{\beta\lambda}{2}||v^a||^2}$$

Quenched partition function

THE OTHER MAJOR FRAMEWORK TO INVESTIGATE GENERALISATION: THE TEACHER-STUDENT SCENARIO

$$\langle Z^m \rangle_{\rho(x)} = \ldots \simeq \int d\mathbf{Q}\; e^{-\frac{mP}{2} S_\beta(\mathbf{Q})} \to \lim_{m\to\infty} \frac{\langle Z^m \rangle_{\rho(x)} - 1}{m}$$

Boring calculation

$$\partial_{\mathbf{Q}} S_\beta(\mathbf{Q}) = 0 \to \mathbf{Q}^* \to S_\beta(\mathbf{Q}^*)$$

Saddle Point Equation

$$\epsilon_t = \langle \mathscr{L} \rangle = -\lim_{\beta\to\infty} \frac{1}{\beta} S_\beta(\mathbf{Q}^*)$$

Averages of observables

THE OTHER MAJOR FRAMEWORK TO INVESTIGATE GENERALISATION: THE TEACHER-STUDENT SCENARIO

$$\langle Z^m \rangle_{\rho(x)} = \ldots \simeq \int d\mathbf{Q}\ e^{-\frac{mP}{2} S_\beta(\mathbf{Q})} \rightarrow \lim_{m\rightarrow\infty} \frac{\langle Z^m \rangle_{\rho(x)} - 1}{m}$$

Boring calculation

$$\partial_{\mathbf{Q}} S_\beta(\mathbf{Q}) = 0 \rightarrow \mathbf{Q}^* \ \rightarrow S_\beta(\mathbf{Q}^*)$$

Saddle Point Equation

$$\epsilon_t = \langle \mathscr{L} \rangle = - \lim_{\beta\rightarrow\infty} \frac{1}{\beta} S_\beta(\mathbf{Q}^*)$$

Averages of observables

- **linear** teacher, **linear** student, **factorised** input density is a textbook exercise

THE OTHER MAJOR FRAMEWORK TO INVESTIGATE GENERALISATION: THE TEACHER-STUDENT SCENARIO

$$\langle Z^m \rangle_{\rho(x)} = \ldots \simeq \int d\mathbf{Q}\, e^{-\frac{mP}{2} S_\beta(\mathbf{Q})} \to \lim_{m\to\infty} \frac{\langle Z^m \rangle_{\rho(x)} - 1}{m}$$

Boring calculation

$$\partial_{\mathbf{Q}} S_\beta(\mathbf{Q}) = 0 \to \mathbf{Q}^* \to S_\beta(\mathbf{Q}^*)$$

Saddle Point Equation

$$\epsilon_t = \langle \mathscr{L} \rangle = -\lim_{\beta\to\infty} \frac{1}{\beta} S_\beta(\mathbf{Q}^*)$$

Averages of observables

- **linear** teacher, **linear** student, **factorised** input density is a textbook exercise
- **polynomial** teacher, **polynomial** student, **factorised** input density

[R. Dietrich, M. Opper, H. Sompolinsky; PRL (1999)]

RECENT RESULTS: GENERALISATION AND TRAINING ERRORS FOR GENERIC KERNELS/GENERIC (QUENCHED) FEATURES

[A. Canatar, B. Bordelon, C. Pehlevan; Nat. Comm. (2021)]
[B. Loureiro, C. Gerbelot, H. Cui, S. Goldt, F. Krzakala, M. Mézard, L. Zdeborová; NeurIPS (2021)]

RECENT RESULTS: GENERALISATION AND TRAINING ERRORS FOR GENERIC KERNELS/GENERIC (QUENCHED) FEATURES

[A. Canatar, B. Bordelon, C. Pehlevan; Nat. Comm. (2021)]
[B. Loureiro, C. Gerbelot, H. Cui, S. Goldt, F. Krzakala, M. Mézard, L. Zdeborová; NeurIPS (2021)]

- Exact **formulas for the generalisation and training errors**.
Particularly simple for regression problems and quadratic loss function

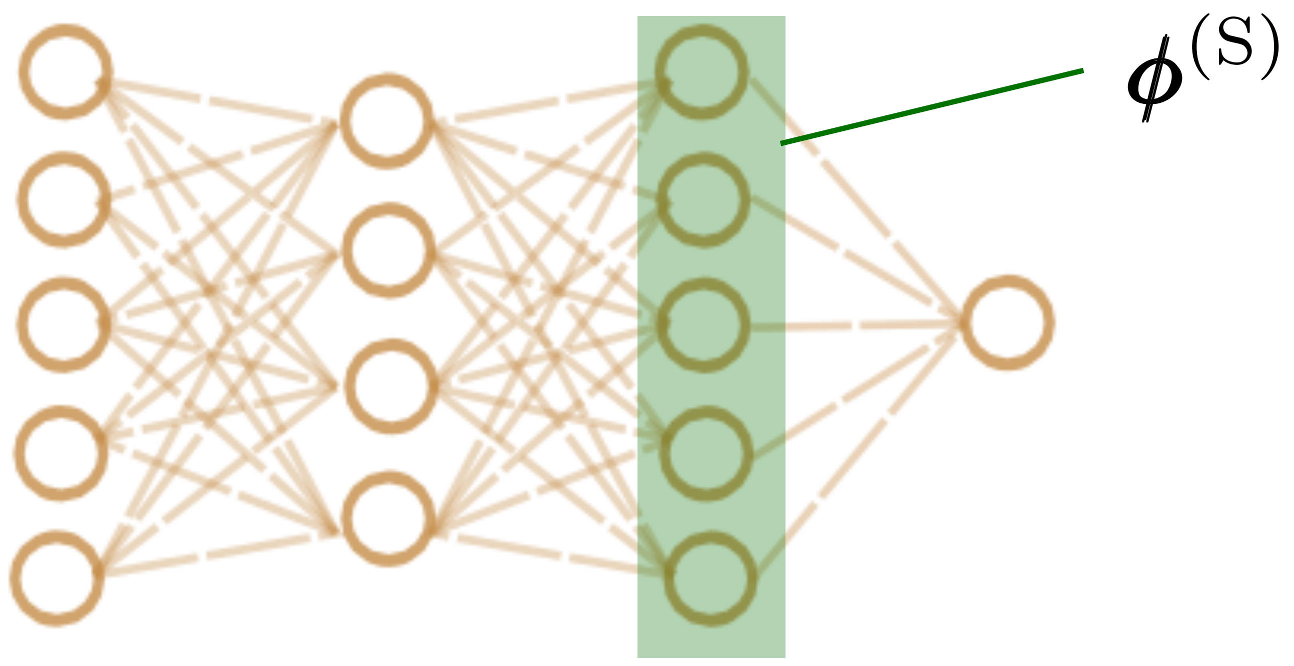
$$\epsilon_{\mathrm{g/t}}\left(N_{\mathrm{S}}, P, \boldsymbol{\phi}^{(\mathrm{T})}, \boldsymbol{\phi}^{(\mathrm{S})}\right)$$

RECENT RESULTS: GENERALISATION AND TRAINING ERRORS FOR GENERIC KERNELS/GENERIC (QUENCHED) FEATURES

[A. Canatar, B. Bordelon, C. Pehlevan; Nat. Comm. (2021)]
[B. Loureiro, C. Gerbelot, H. Cui, S. Goldt, F. Krzakala, M. Mézard, L. Zdeborová; NeurIPS (2021)]

- Exact **formulas for the generalisation and training errors**.
  Particularly simple for regression problems and quadratic loss function

$$\epsilon_{\mathrm{g/t}}\left(N_{\mathrm{S}}, P, \boldsymbol{\phi}^{(\mathrm{T})}, \boldsymbol{\phi}^{(\mathrm{S})}\right)$$



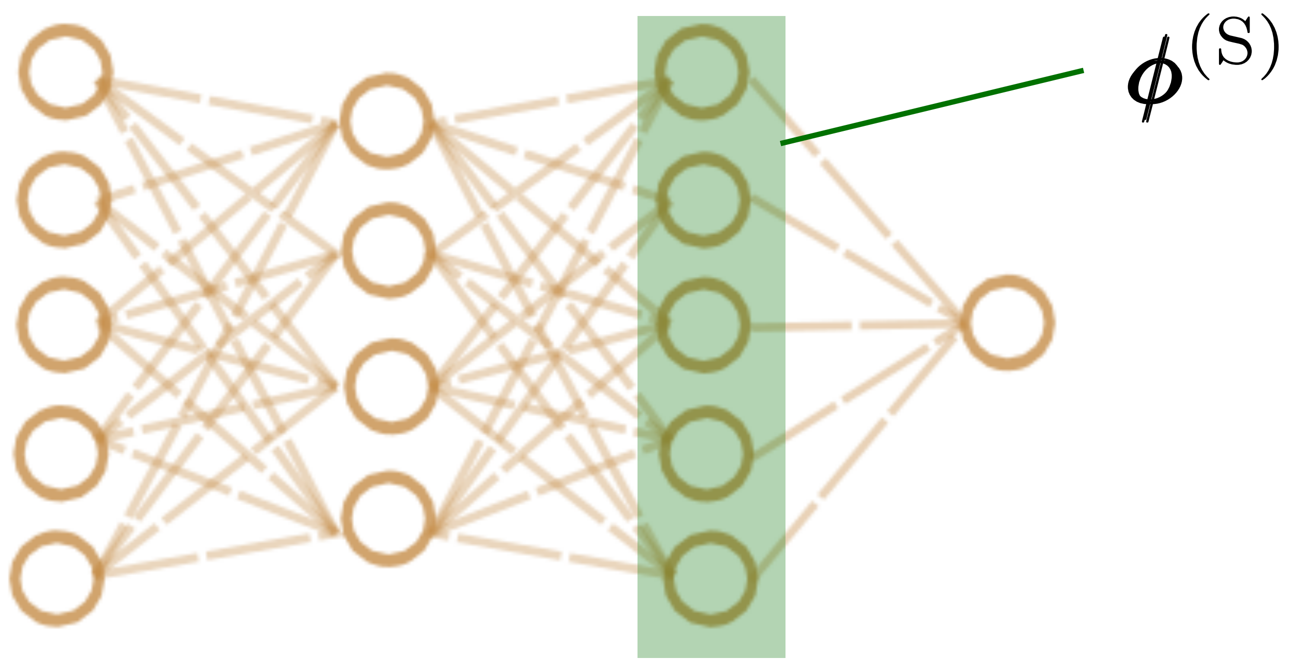
- These formulas **capture the learning curves of multilayer neural networks** if we consider as features those obtained by pre-trained networks on realistic datasets!

RECENT RESULTS: GENERALISATION AND TRAINING ERRORS FOR GENERIC KERNELS/GENERIC (QUENCHED) FEATURES

[A. Canatar, B. Bordelon, C. Pehlevan; Nat. Comm. (2021)]
[B. Loureiro, C. Gerbelot, H. Cui, S. Goldt, F. Krzakala, M. Mézard, L. Zdeborová; NeurIPS (2021)]

- Exact **formulas for the generalisation and training errors**.
  Particularly simple for regression problems and quadratic loss function

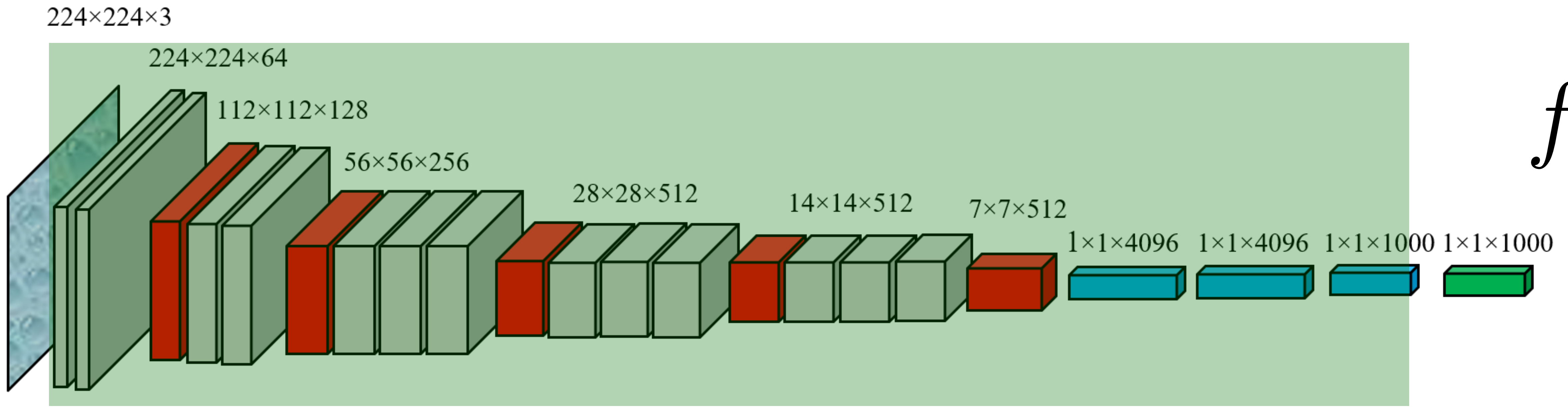
$$\epsilon_{\mathrm{g/t}}\left(N_{\mathrm{S}}, P, \boldsymbol{\phi}^{(\mathrm{T})}, \boldsymbol{\phi}^{(\mathrm{S})}\right)$$



- These formulas **capture the learning curves of multilayer neural networks** if we consider as features those obtained by pre-trained networks on realistic datasets!

- Based on a conjecture: the **Gaussian Equivalence Principle**

RESULTS: FROM QUENCHED FEATURES TO A UNIVERSAL MEAN FIELD UPPER BOUND FOR THE GENERALISATION GAP OF DNNS



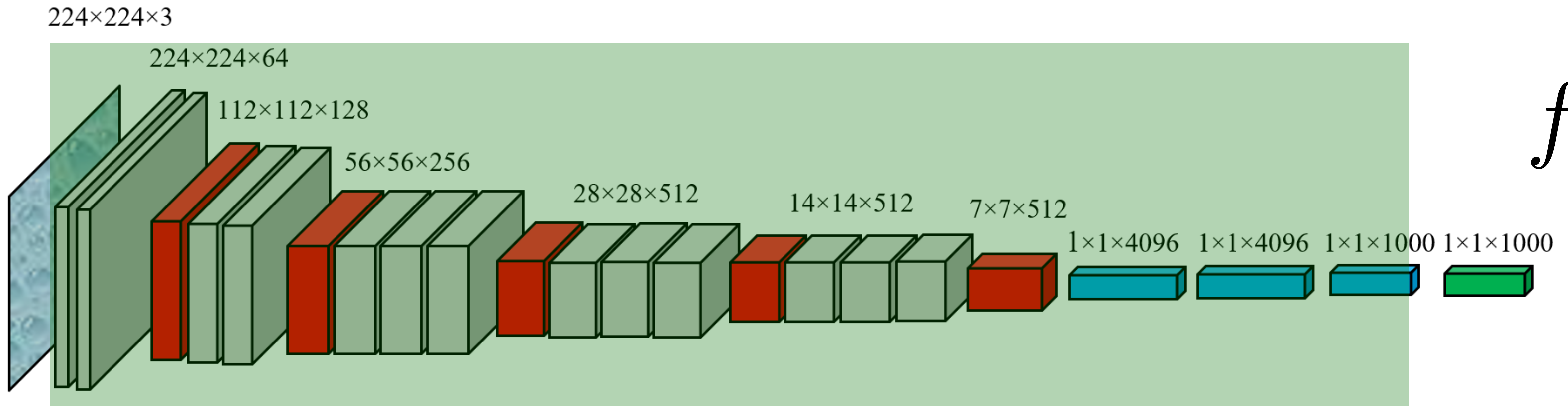
$$\phi_\alpha^{\mathrm{DNN}}(\mathbf{x}, \mathcal{W})$$

$$f_{\mathrm{DNN}}(\mathbf{x}, \theta) = \frac{1}{N_{\mathrm{out}}} \sum_{\alpha=1}^{N_{\mathrm{out}}} v_\alpha \phi_\alpha^{\mathrm{DNN}}(\mathbf{x}, \mathcal{W})$$

$$\theta = \{\mathcal{W}, \mathbf{v}\}$$

number of weights in the last layer

RESULTS: FROM QUENCHED FEATURES TO A UNIVERSAL MEAN FIELD UPPER BOUND FOR THE GENERALISATION GAP OF DNNS



$\phi_\alpha^{\mathrm{DNN}}(\mathbf{x}, \mathcal{W})$

$$f_{\mathrm{DNN}}(\mathbf{x}, \theta) = \frac{1}{N_{\mathrm{out}}} \sum_{\alpha=1}^{N_{\mathrm{out}}} v_\alpha \phi_\alpha^{\mathrm{DNN}}(\mathbf{x}, \mathcal{W})$$

$$\theta = \{\mathcal{W}, \mathbf{v}\}$$

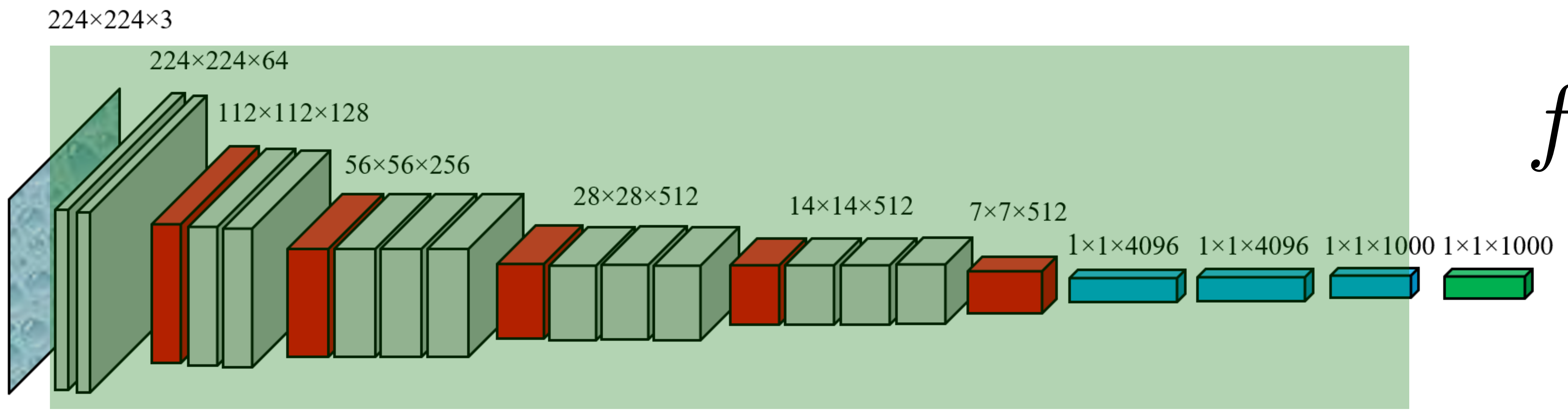
N_{out}: number of weights in the last layer

$$N_{\mathrm{out}} \ll P$$

N_{out}: $10^2 - 10^3$

P: $10^4 - 10^5$

RESULTS: FROM QUENCHED FEATURES TO A UNIVERSAL MEAN FIELD UPPER BOUND FOR THE GENERALISATION GAP OF DNNS



$$\phi_\alpha^{\mathrm{DNN}}(\mathbf{x}, \mathcal{W})$$

$$f_{\mathrm{DNN}}(\mathbf{x}, \theta) = \frac{1}{N_{\mathrm{out}}} \sum_{\alpha=1}^{N_{\mathrm{out}}} v_\alpha \phi_\alpha^{\mathrm{DNN}}(\mathbf{x}, \mathcal{W})$$

$$\theta = \{\mathcal{W}, \mathbf{v}\}$$

N_{out}: number of weights in the last layer

$$N_{\mathrm{out}} \ll P$$

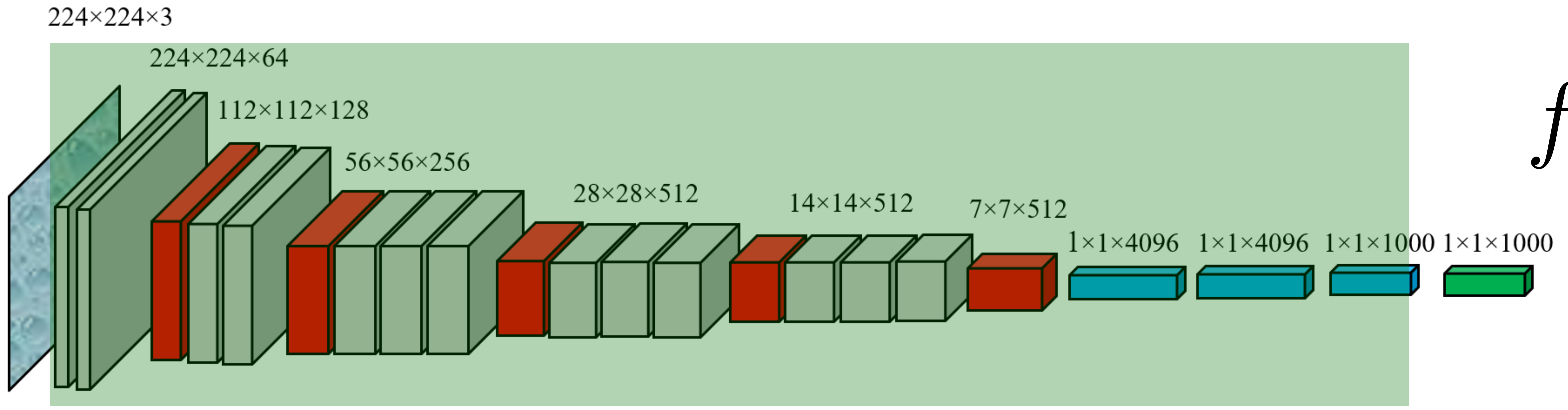
N_{out}: $10^2 - 10^3$

P: $10^4 - 10^5$

the equation holds for each realisation of the weights $\mathcal{W}$ and it assumes perfect training over the last layer

$$\Delta\epsilon(\mathcal{W}) \simeq 2\epsilon_{\mathrm{g}}^{\mathrm{R}}(\mathcal{W}) \frac{N_{\mathrm{out}}}{P}$$

RESULTS: FROM QUENCHED FEATURES TO A UNIVERSAL MEAN FIELD UPPER BOUND FOR THE GENERALISATION GAP OF DNNS



$\phi_\alpha^{\rm DNN}(\mathbf{x}, \mathcal{W})$

$$f_{\rm DNN}(\mathbf{x}, \theta) = \frac{1}{N_{\rm out}} \sum_{\alpha=1}^{N_{\rm out}} v_\alpha \phi_\alpha^{\rm DNN}(\mathbf{x}, \mathcal{W})$$

$$\theta = \{\mathcal{W}, \mathbf{v}\}$$

$N_{\rm out}$: number of weights in the last layer

$$N_{\rm out} \ll P$$

$N_{\rm out}$: $10^2 - 10^3$

P: $10^4 - 10^5$

the equation holds for each realisation of the weights $\mathcal{W}$ and it assumes perfect training over the last layer

$$\Delta\epsilon(\mathcal{W}) \simeq 2\epsilon_{\rm g}^{\rm R}(\mathcal{W}) \frac{N_{\rm out}}{P}$$

$$0 \leq \epsilon_{\rm g}^{\rm R}(\mathcal{W}) \leq T$$

RESULTS: FROM QUENCHED FEATURES TO A UNIVERSAL MEAN FIELD UPPER BOUND FOR THE GENERALISATION GAP OF DNNS

$$\Delta\tilde{\epsilon}(\mathcal{W}) = \frac{\Delta\epsilon(\mathcal{W})}{T} \leq \frac{2N_{\mathrm{out}}}{P}$$

RESULTS: FROM QUENCHED FEATURES TO A UNIVERSAL MEAN FIELD UPPER BOUND FOR THE GENERALISATION GAP OF DNNS

$$\Delta\tilde{\epsilon}(\mathcal{W}) = \frac{\Delta\epsilon(\mathcal{W})}{T} \leq \frac{2N_{\text{out}}}{P}$$

valid for any local minimum of the loss function of the DNN

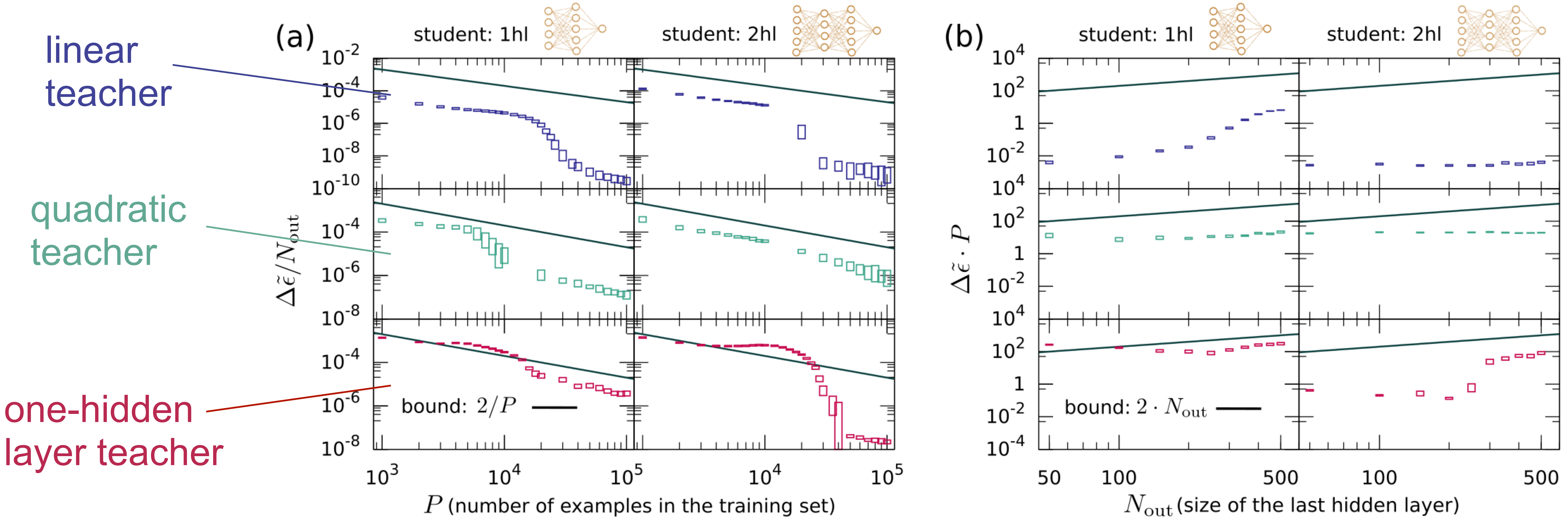
RESULTS: FROM QUENCHED FEATURES TO A UNIVERSAL MEAN FIELD UPPER BOUND FOR THE GENERALISATION GAP OF DNNS

$$\Delta\tilde{\epsilon}(\mathcal{W}) = \frac{\Delta\epsilon(\mathcal{W})}{T} \leq \frac{2N_{\mathrm{out}}}{P}$$

valid for any local minimum of the loss function of the DNN

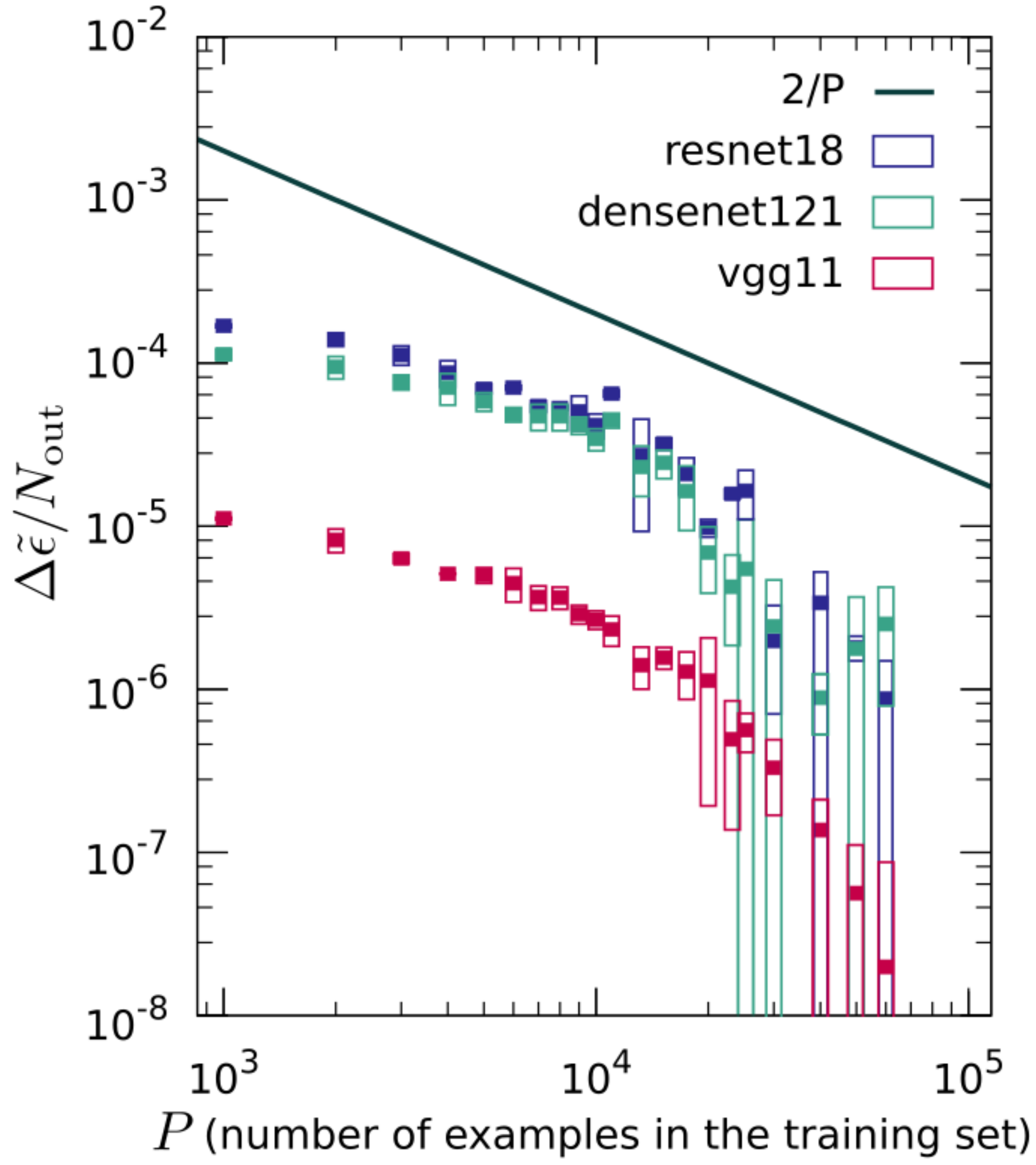
- the gap of fully-trained DNNs should **decrease at least as 1/P asymptotically**
- the **degradation** of the generalisation performance should be **at most linear as the size of the last layer is increased**
- the bound **rules out** any asymptotic **linear or sub-linear dependence on the size of the hidden layers**

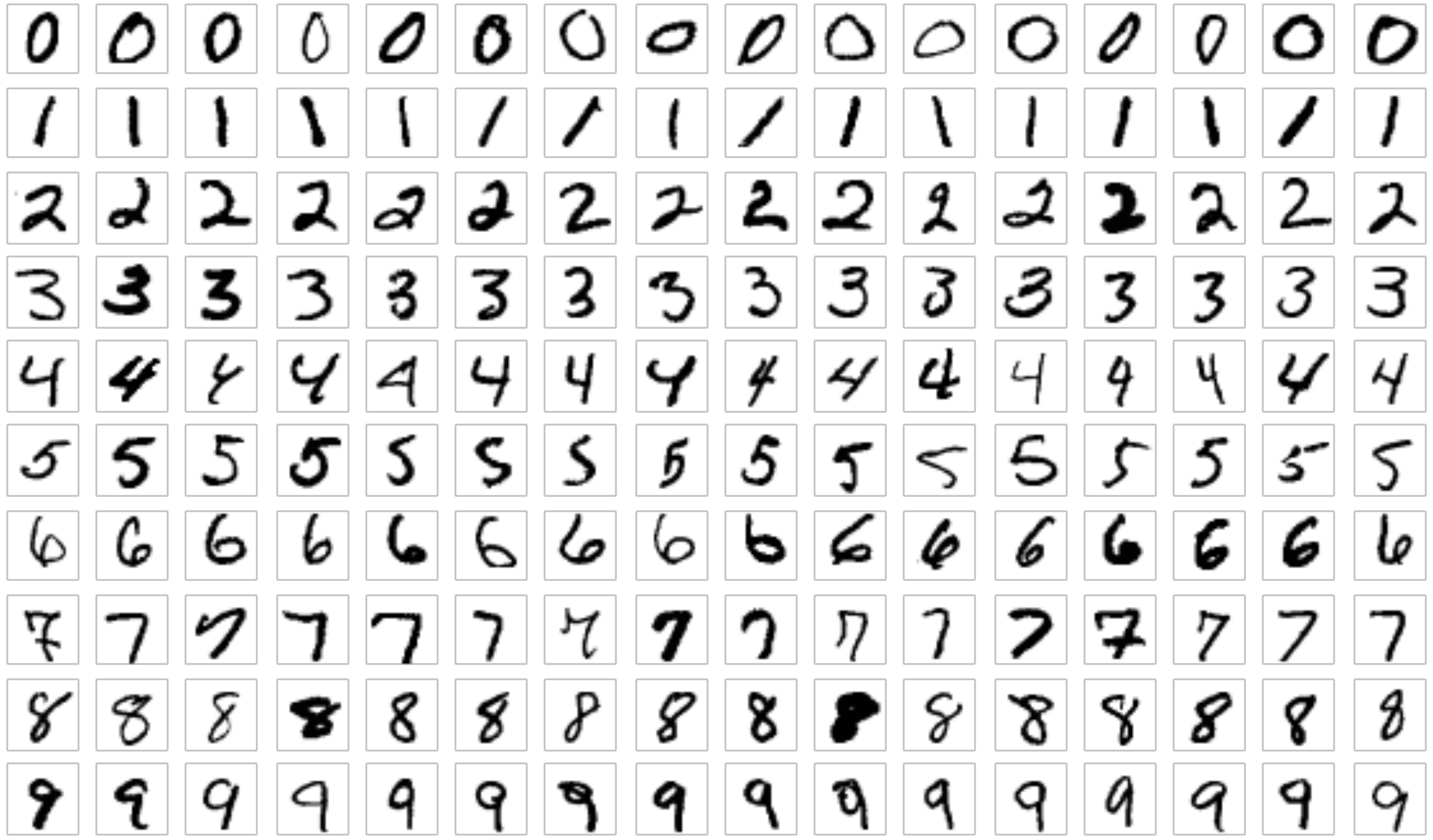
RESULTS: GENERALISATION GAP FOR TOY FULLY CONNECTED STUDENTS TRAINED ON SYNTHETIC DATASETS



In all these experiments the input density is factorised over its coordinates

RESULTS: GENERALISATION GAP FOR STATE-OF-THE-ART ARCHITECTURES TRAINED ON MNIST

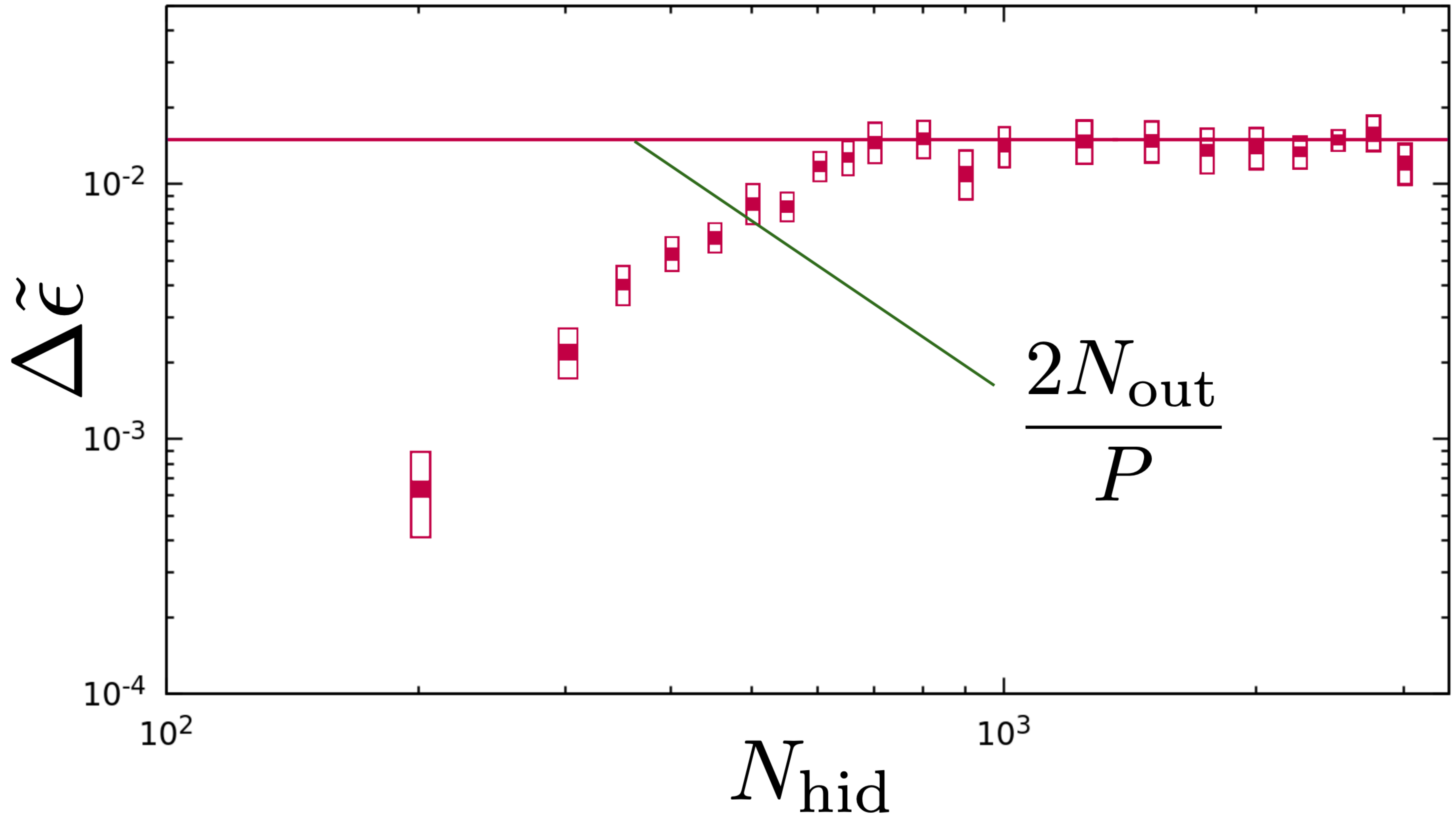


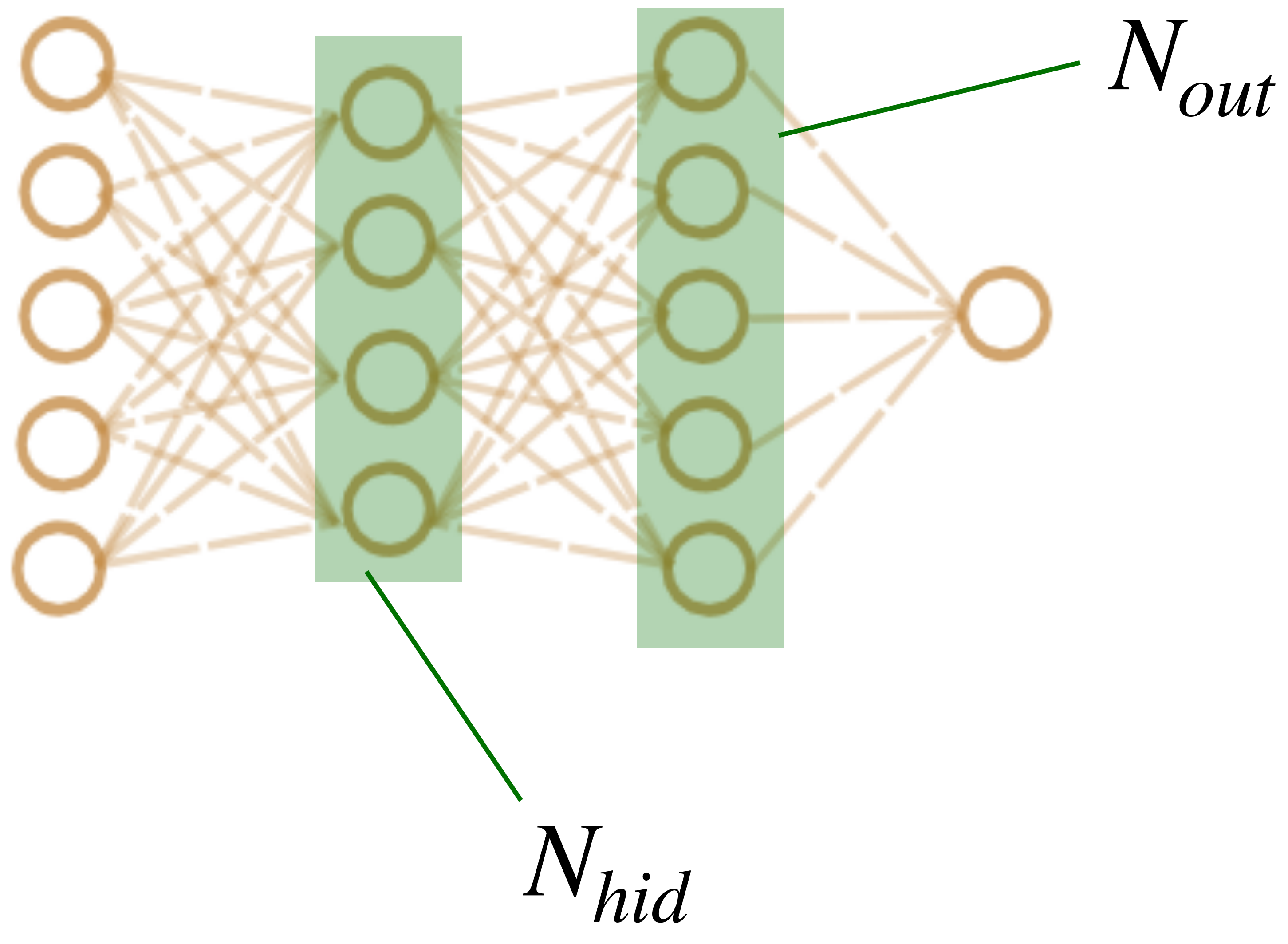


Remark: the generalisation gap is defined for regression, **not** for classification

RESULTS: ANY LINEAR OR SUBLINEAR DEPENDENCE ON THE SIZE OF THE HIDDEN LAYER IS RULED OUT

two hidden layer student learns a one hidden layer teacher





CONCLUSIONS AND FUTURE PERSPECTIVES

- A **more stringent** and **universal** asymptotic upper bound for the generalisation gap of DNNs

CONCLUSIONS AND FUTURE PERSPECTIVES

- A **more stringent** and **universal** asymptotic upper bound for the generalisation gap of DNNs

- One (out of many) possible next step: find a way to treat a DNN analytically, with a fixed training test instance.

[AS, R. Pacelli, M. Pastore, F. Ginelli, M. Gherardi, P. Rotondo; arXiv: 2209.04882 (2022)]

CONCLUSIONS AND FUTURE PERSPECTIVES

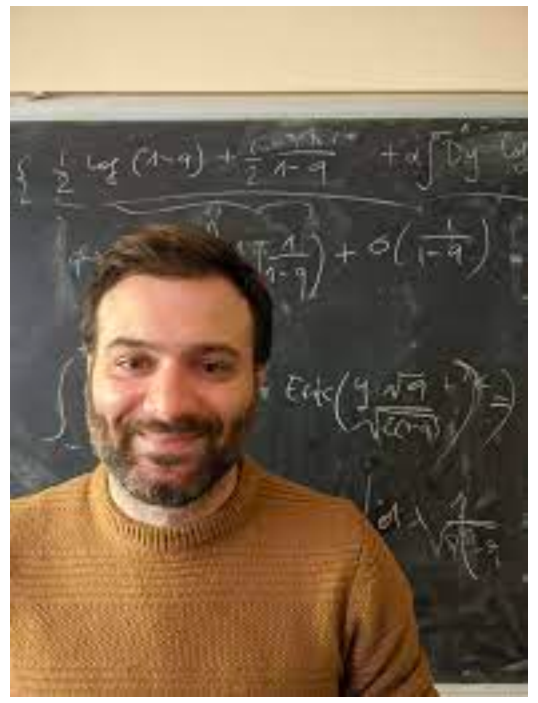
- A **more stringent** and **universal** asymptotic upper bound for the generalisation gap of DNNs

- One (out of many) possible next step: find a way to treat a DNN analytically, with a fixed training test instance.

  [AS, R. Pacelli, M. Pastore, F. Ginelli, M. Gherardi, P. Rotondo; arXiv: 2209.04882 (2022)]

- A second possible next step: find where and when the Gaussian Equivalence Principle holds.

THANKS!



Pietro Rotondo

Rosalba Pacelli

Francesco Ginelli

Marco Gherardi

[SA, R. Pacelli, F. Ginelli, M. Gherardi, P. Rotondo; Phys. Rev. E **105**, 064309]