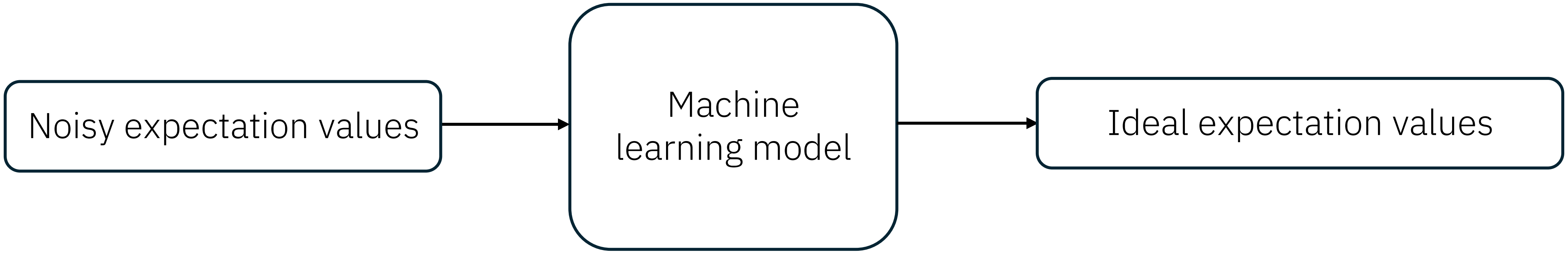
Deep learning for quantum error mitigation

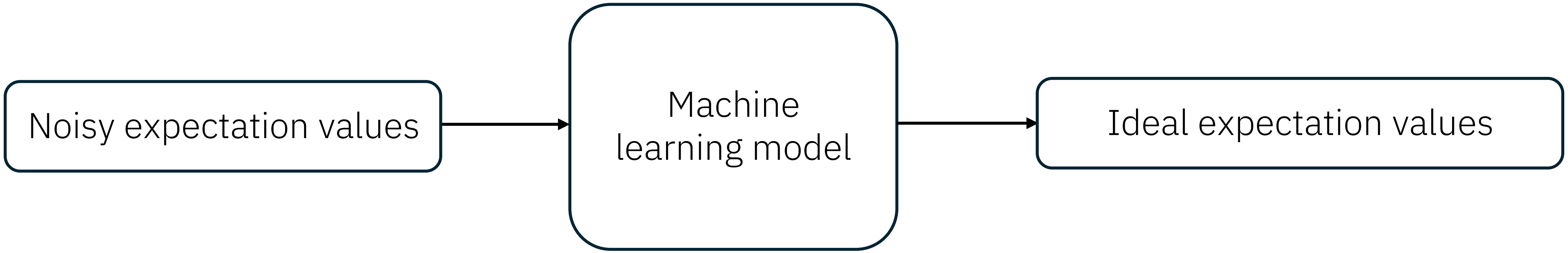
Simone Cantori
University of Camerino - School of Science and Technology
PhD student in Quantum Technologies

simone.cantori@unicam.it

Core idea:


Noisy expectation values
Machine learning model
Ideal expectation values

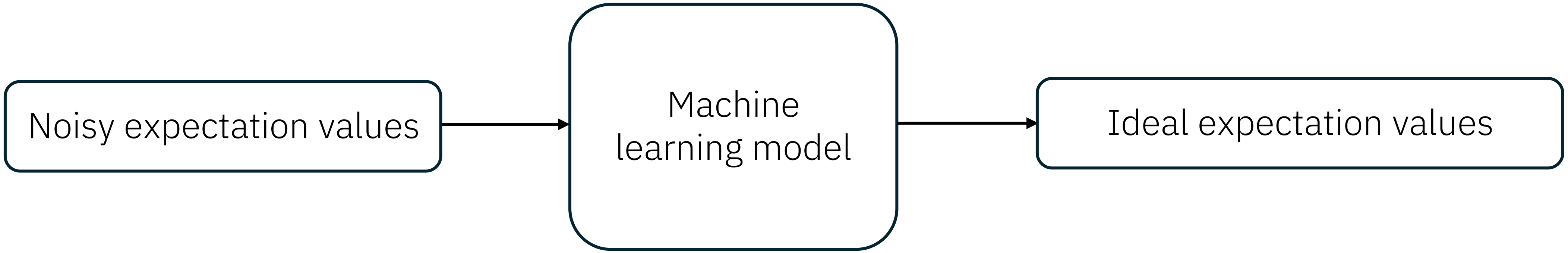
Core idea:



Problem:

- The exact output of a generic large quantum circuit U can't be computed with classical simulation methods.

Core idea:



Problem:

- The exact output of a generic large quantum circuit U can't be computed with classical simulation methods.

Solution:

- Train a model using quantum circuits that are both classically simulable and structurally similar to U.
- Use the model to mitigate the errors in the output of U.

Proposed training-sets:

- Near Clifford quantum circuits [1]
- Product states [2]
- Small quantum circuits and scalable neural networks [3]

[1] Piotr Czarnik, Andrew Arrasmith, Patrick J. Coles, and Lukasz Cincio, Error mitigation with Clifford quantum-circuit data, Quantum, **5**, 592 (2021)

[2] Stefan H. Sack, and Daniel J. Egger, Large-scale quantum approximate optimization on nonplanar graphs with machine learning noise mitigation, Phys. Rev. Research **6**, 013223, (2024)

[3] S. Cantori, A. Mari, D. Vitali, and S. Pilati, Synergy between noisy quantum computers and scalable classical deep learning for quantum error mitigation, EPJ Quantum Technol. **11**, 45 (2024)

Our approach:

- Circuit knitting with small sampling overhead O for VQE processes [1]

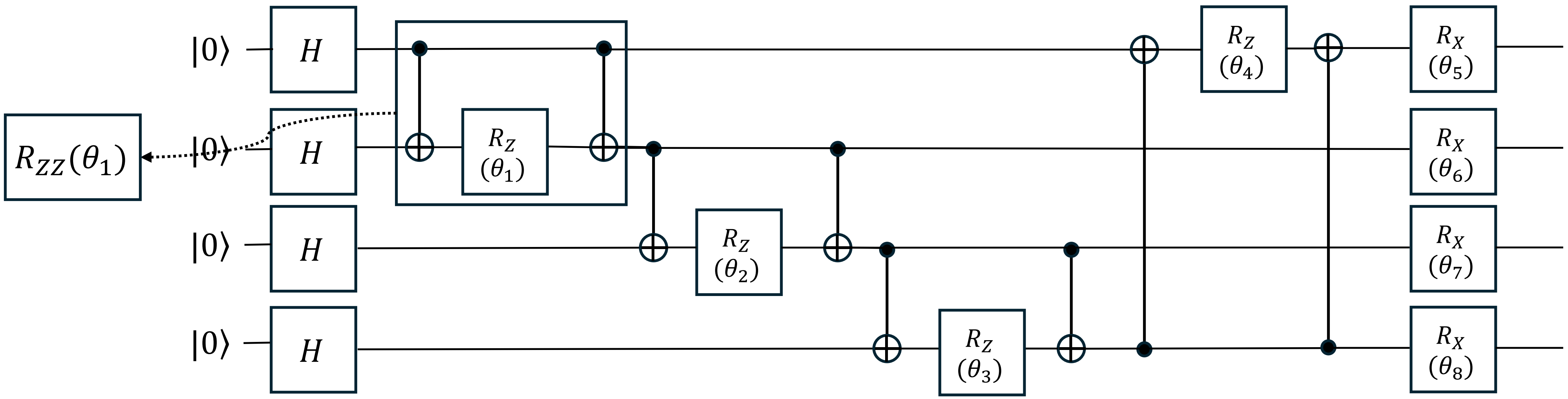
$$O = \prod_{\theta_i \in \mathbb{K}} (1 + 2|\sin\theta_i|)^2 , \qquad \mathbb{K} = \text{set of connecting gates}$$

- We increase the similarity between training circuits and testing circuits

[1] S. Cantori, A. Mari, D. Vitali, and S. Pilati, Deep-learned error mitigation via partially knitted circuits for the variational quantum eigensolver, arXiv:2506.04146 (2025).

Ansatz for the VQE process

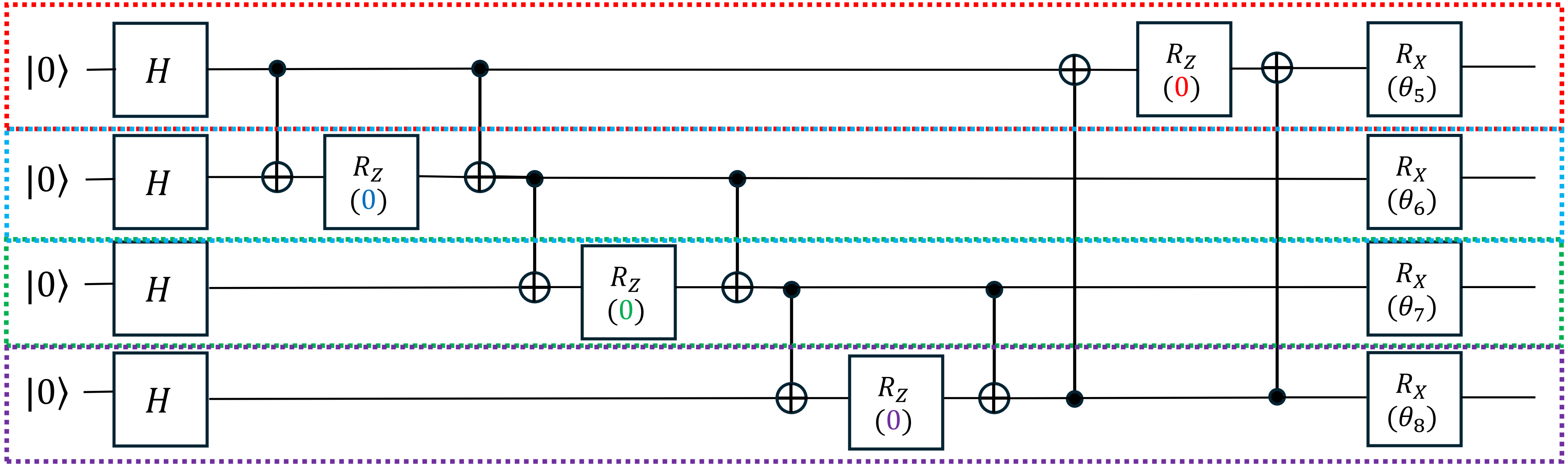
- We use $P = 8$ repetitions of the following block



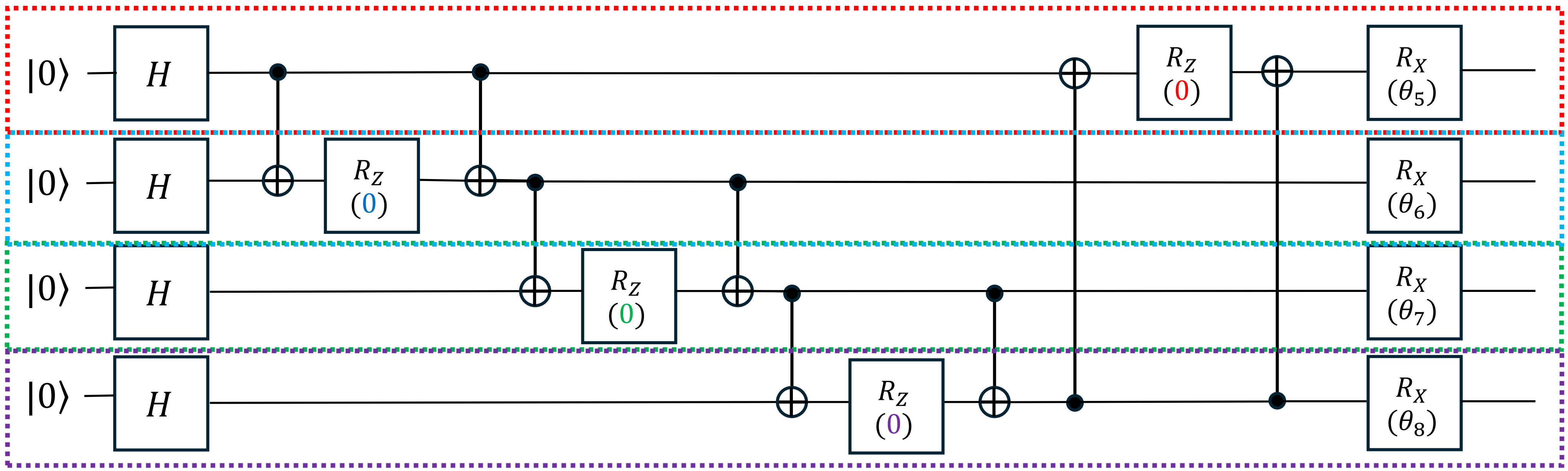
$$H = \sum_i -J_i \, \sigma_i^Z \sigma_{i+1}^Z - h \sum_i \sigma_i^X$$

Wen Wei Ho, Timothy H. Hsieh, Efficient variational simulation of non-trivial quantum states. *SciPost Phys.* 6, 029 (2019)

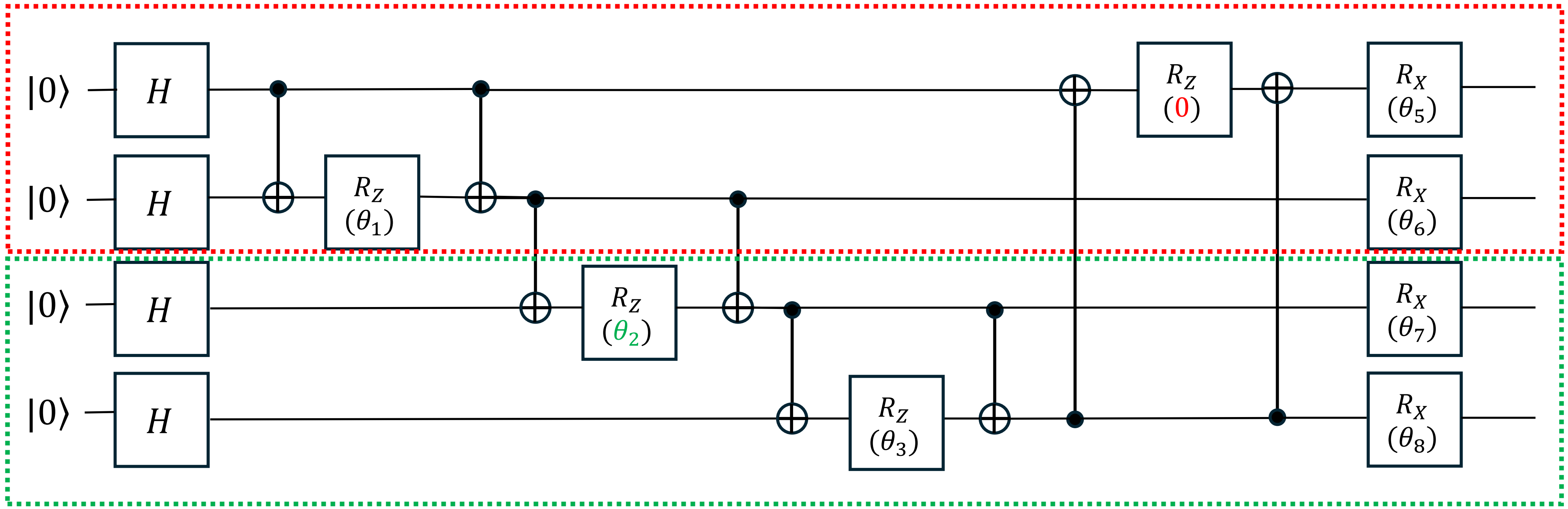
Product states



Product states

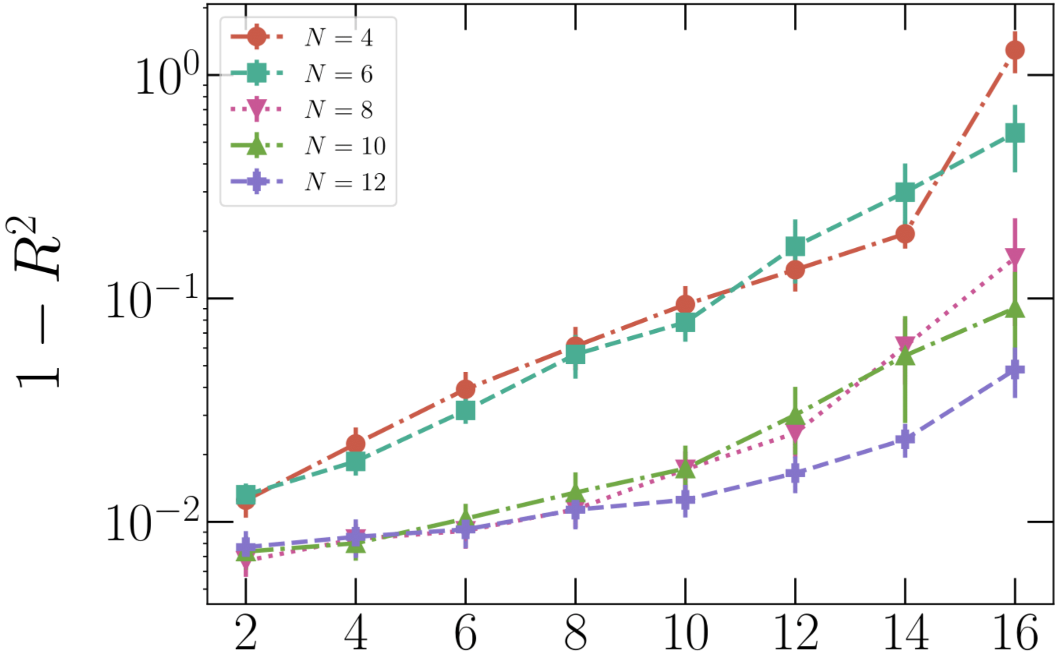


Partially knitted



Importance of similarity between training circuits and testing circuits

$$1 - R^2 = \frac{\sum_{i=1}^{K_{test}} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{K_{test}} (y_i - \bar{y})^2}$$

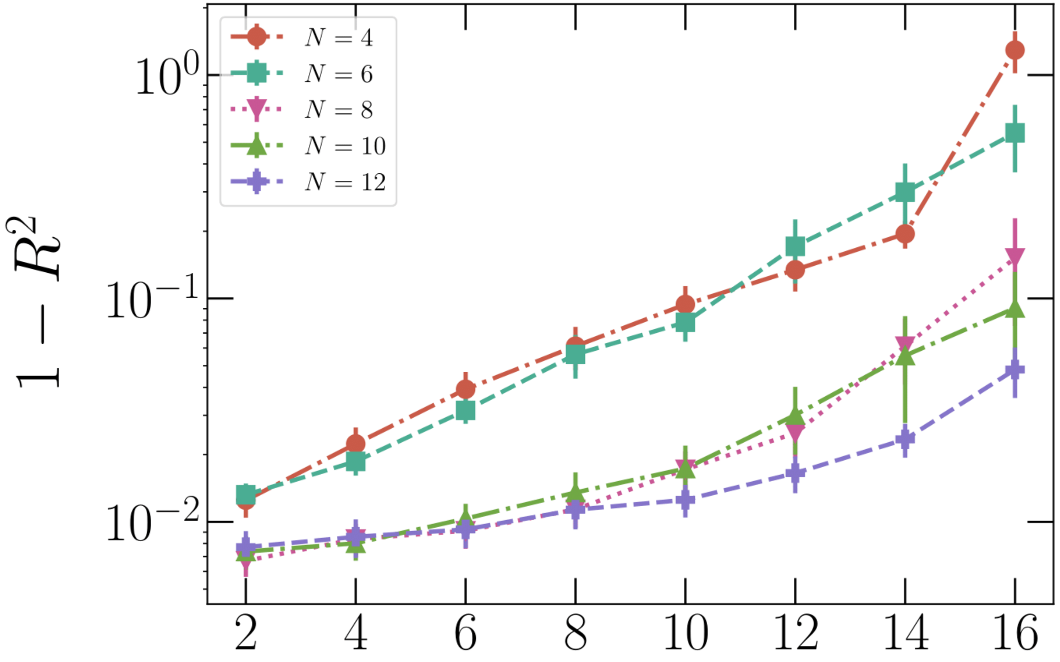


C ⟶ Number of R_{ZZ} gates that have been cut in the training circuits

Importance of similarity between training circuits and testing circuits

- $C = 2 \times N \times P = 48 \longrightarrow$ product state; $C = 16 \longrightarrow$ near-Clifford circuit with 83% non-Clifford gates.

$$1 - R^2 = \frac{\sum_{i=1}^{K_{test}} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{K_{test}} (y_i - \bar{y})^2}$$

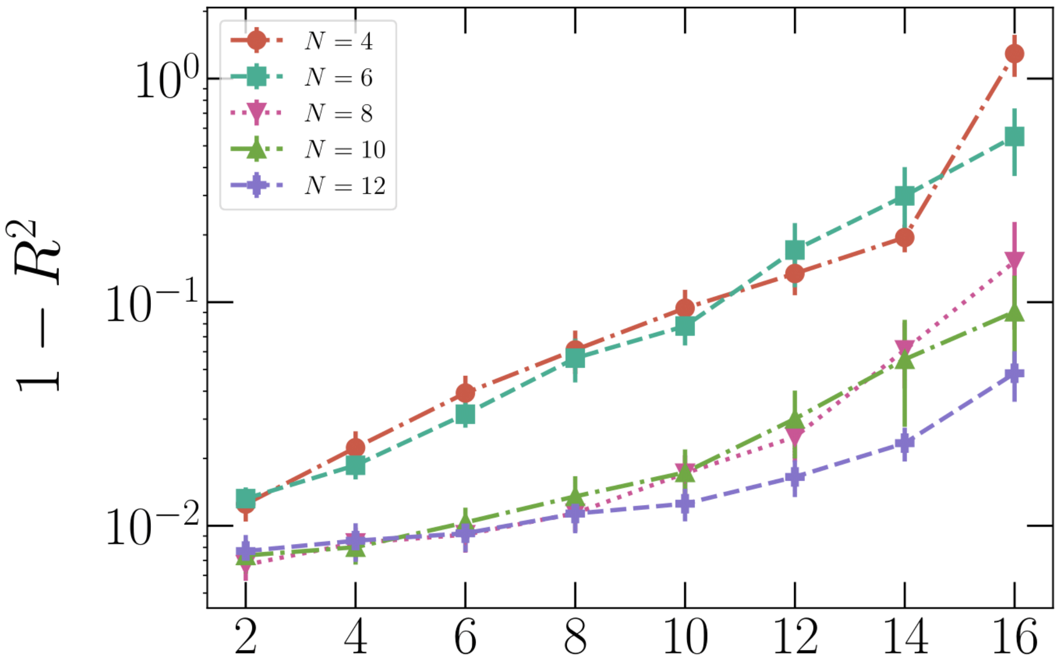


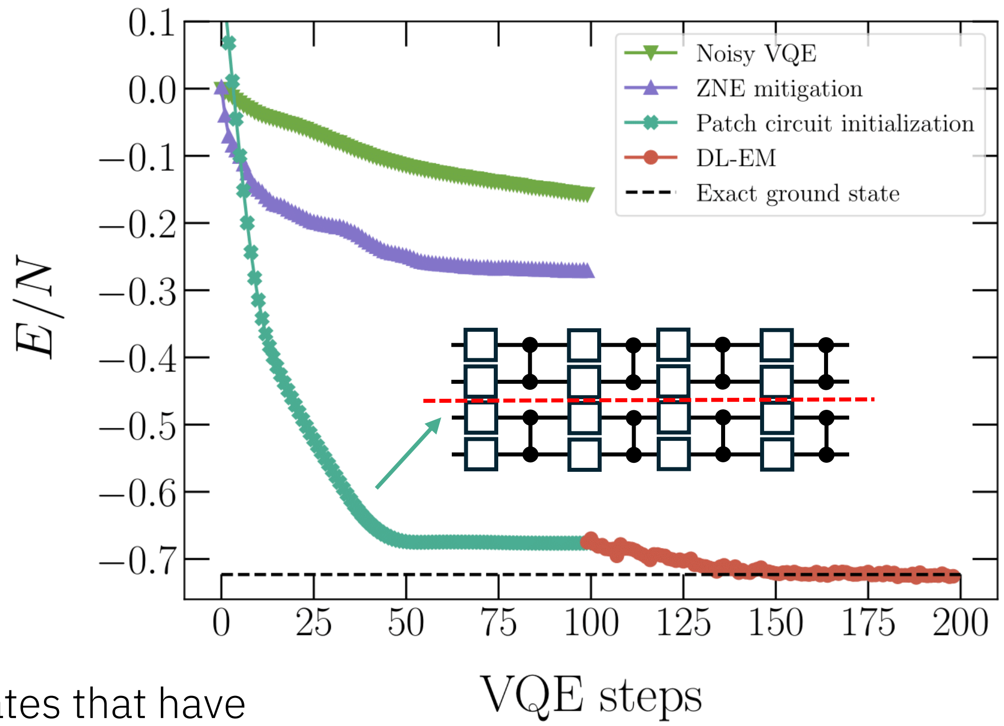
$C \longrightarrow$ Number of R_{ZZ} gates that have been cut in the training circuits

Importance of similarity between training circuits and testing circuits

- $C = 2 \times N \times P = 48 \longrightarrow$ product state; $C = 16 \longrightarrow$ near-Clifford circuit with 83% non-Clifford gates.

$$1 - R^2 = \frac{\sum_{i=1}^{K_{test}} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{K_{test}} (y_i - \bar{y})^2}$$





$C \longrightarrow$ Number of R_{ZZ} gates that have been cut in the training circuits

Future work and open questions:

- Application on real quantum devices.

Future work and open questions:

- Application on real quantum devices.
- Is there a sweet spot where ML-EM can provide “useful” results “efficiently”?

Acknowledgement

- Co-authors: Dr. Andrea Mari, Prof. David Vitali and Prof. Sebastiano Pilati
- Complex quantum matter group

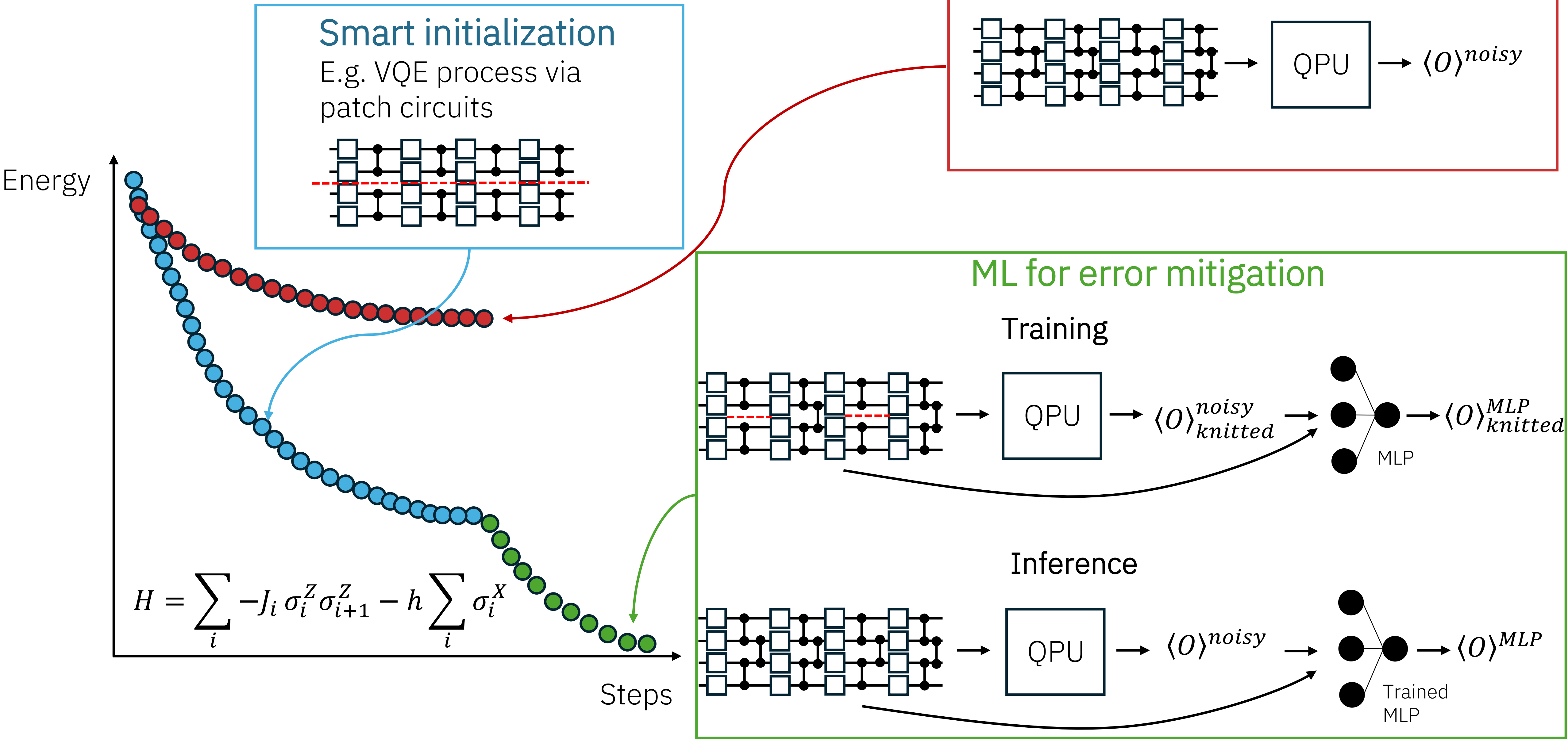
NQSTI
National Quantum Science and Technology Institute

UNICAM
Università di Camerino
1336

INFN

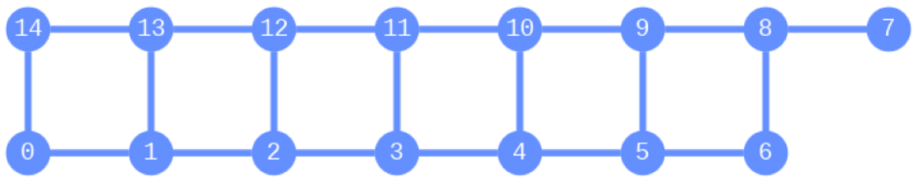
Appendix

Overview (2)


Smart initialization
E.g. VQE process via patch circuits
Noisy VQE
QPU
$\langle O \rangle^{noisy}$
Energy
Steps
$H = \sum_i -J_i \sigma_i^Z \sigma_{i+1}^Z - h \sum_i \sigma_i^X$
ML for error mitigation
Training
QPU
$\langle O \rangle^{noisy}_{knitted}$
MLP
$\langle O \rangle^{MLP}_{knitted}$
Inference
QPU
$\langle O \rangle^{noisy}$
Trained MLP
$\langle O \rangle^{MLP}$

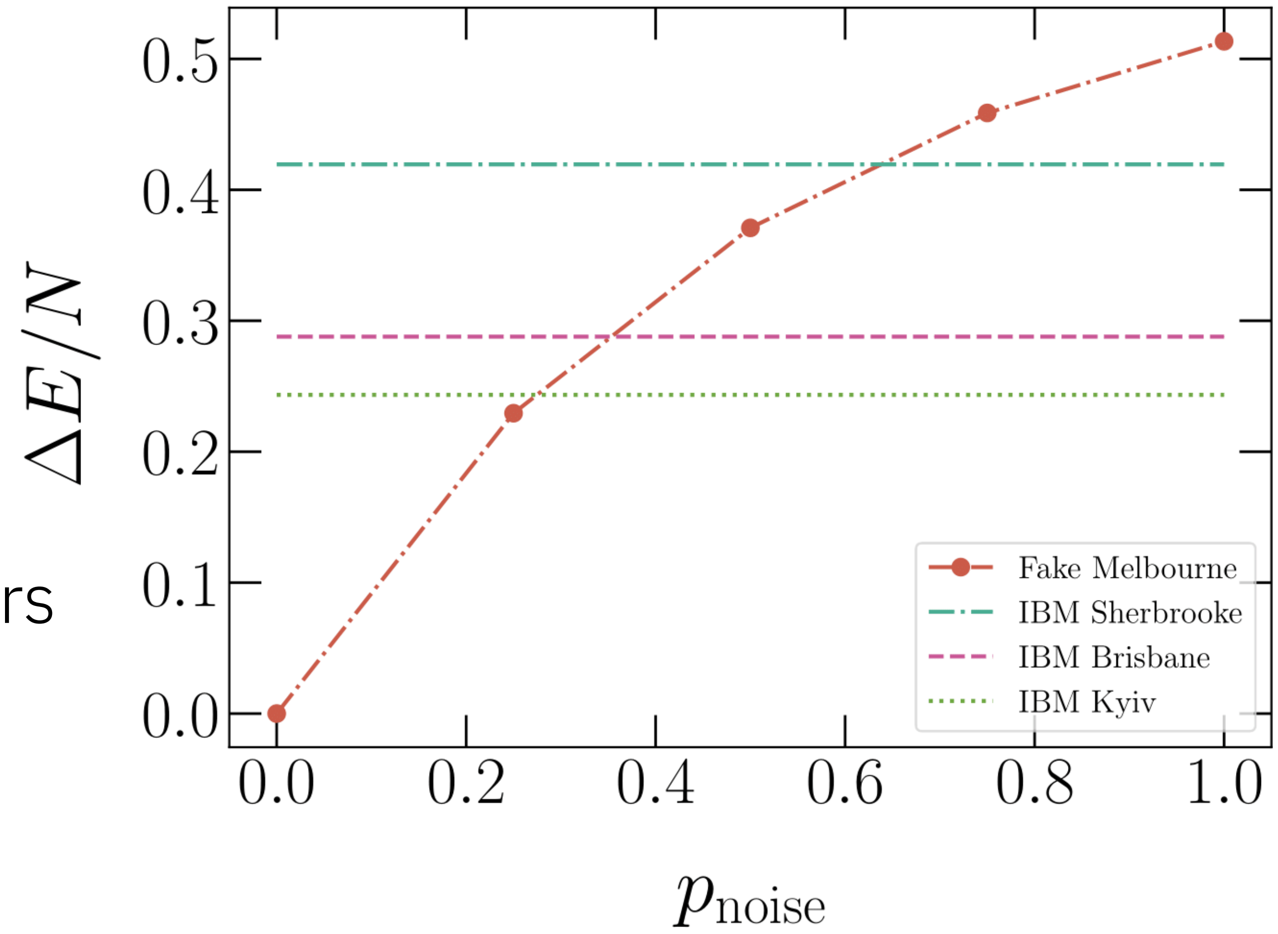
Noise model and comparison with real quantum computers

Fake IBM Melbourne



$p_{\text{noise}} = 0 \implies$ No readout and gate errors

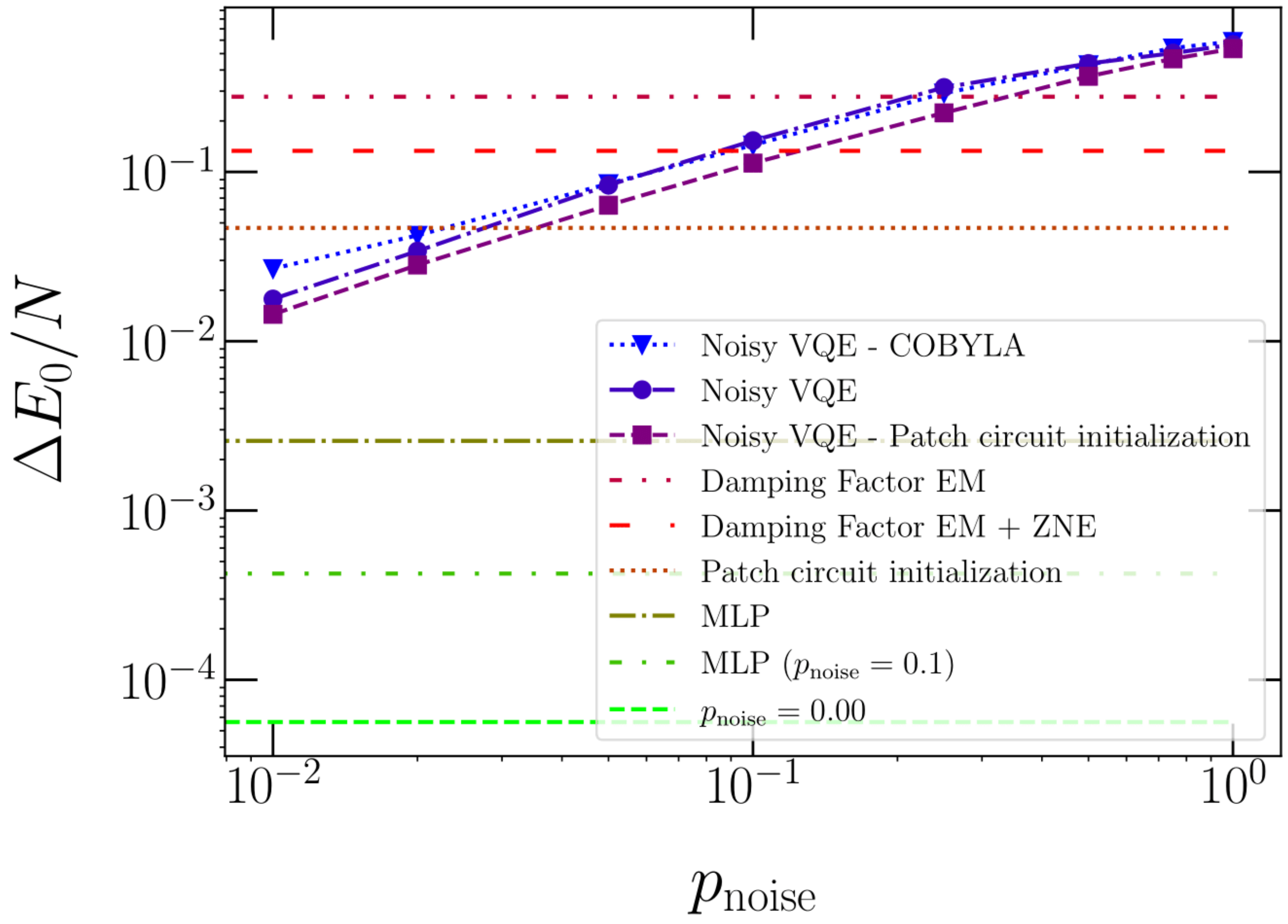
$p_{\text{noise}} = 1 \implies$ Original fake backend

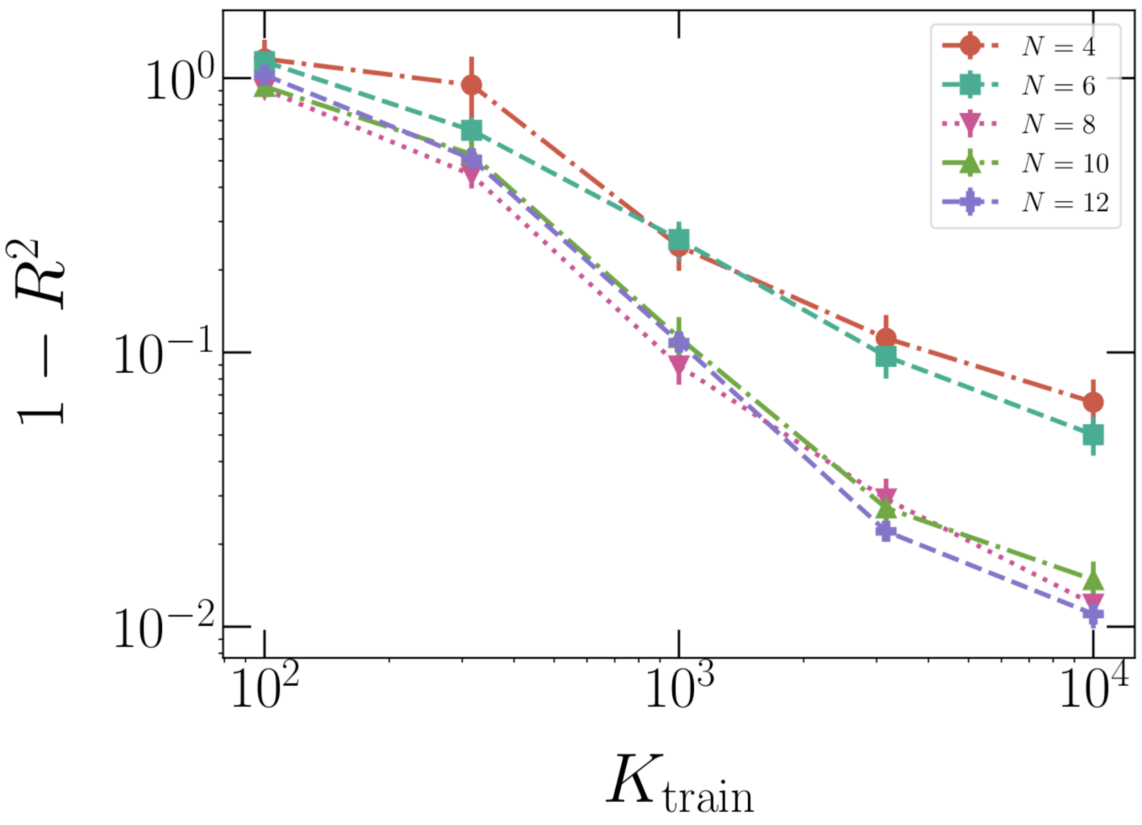


Comparison with other approaches

- Gradient-based (ADAM) and gradient-free (COBYLA) VQE processes;

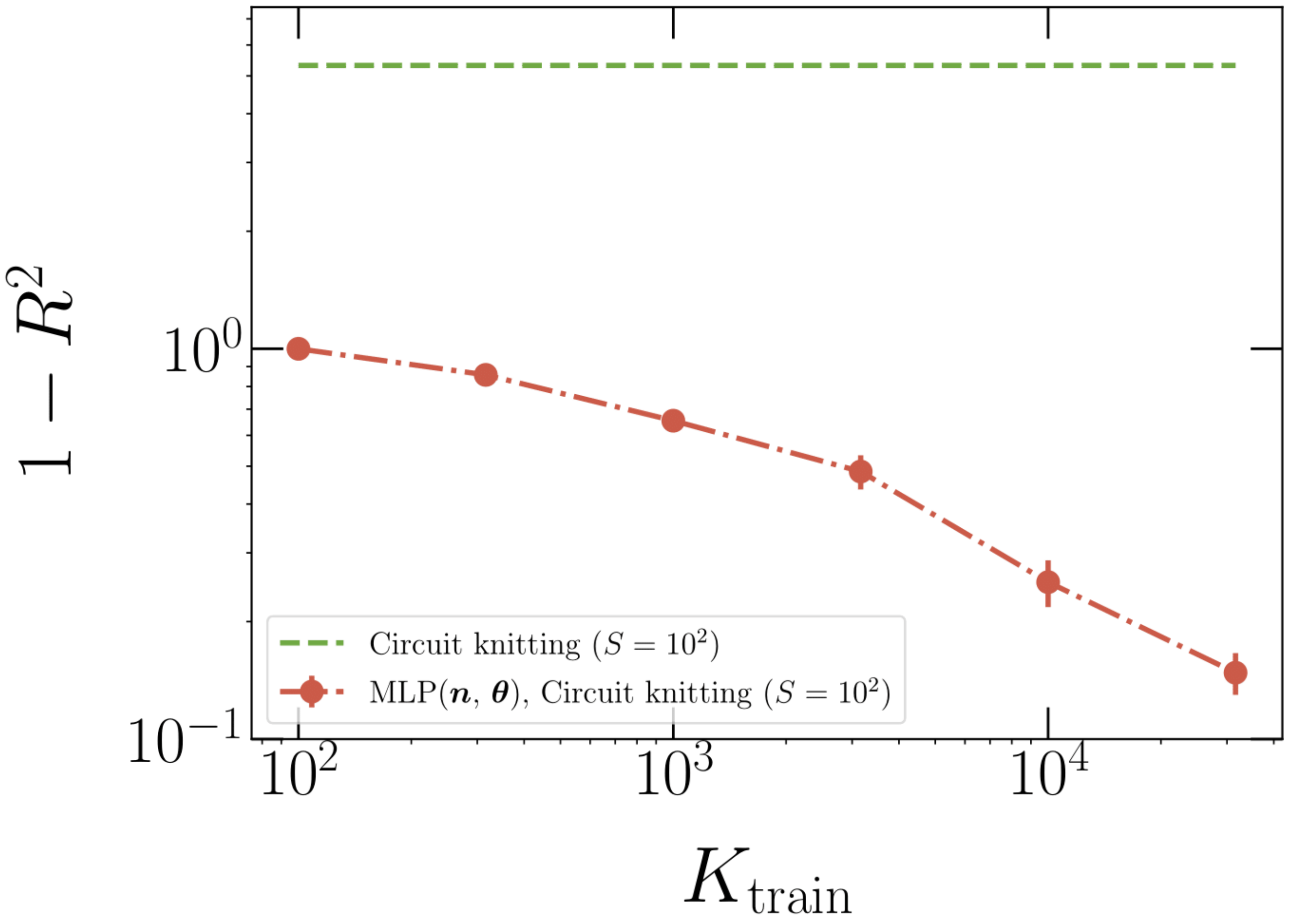
- Damping factor EM: $\langle \mathcal{O} \rangle_{\mathbf{noisy}} = D\langle \mathcal{O} \rangle$, where D can be obtained with reference states;

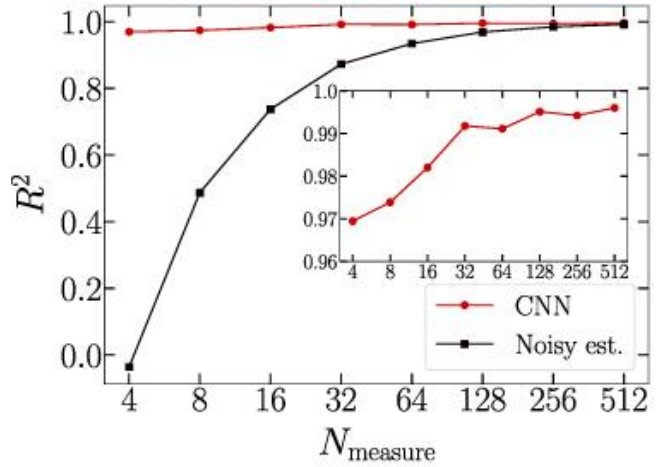




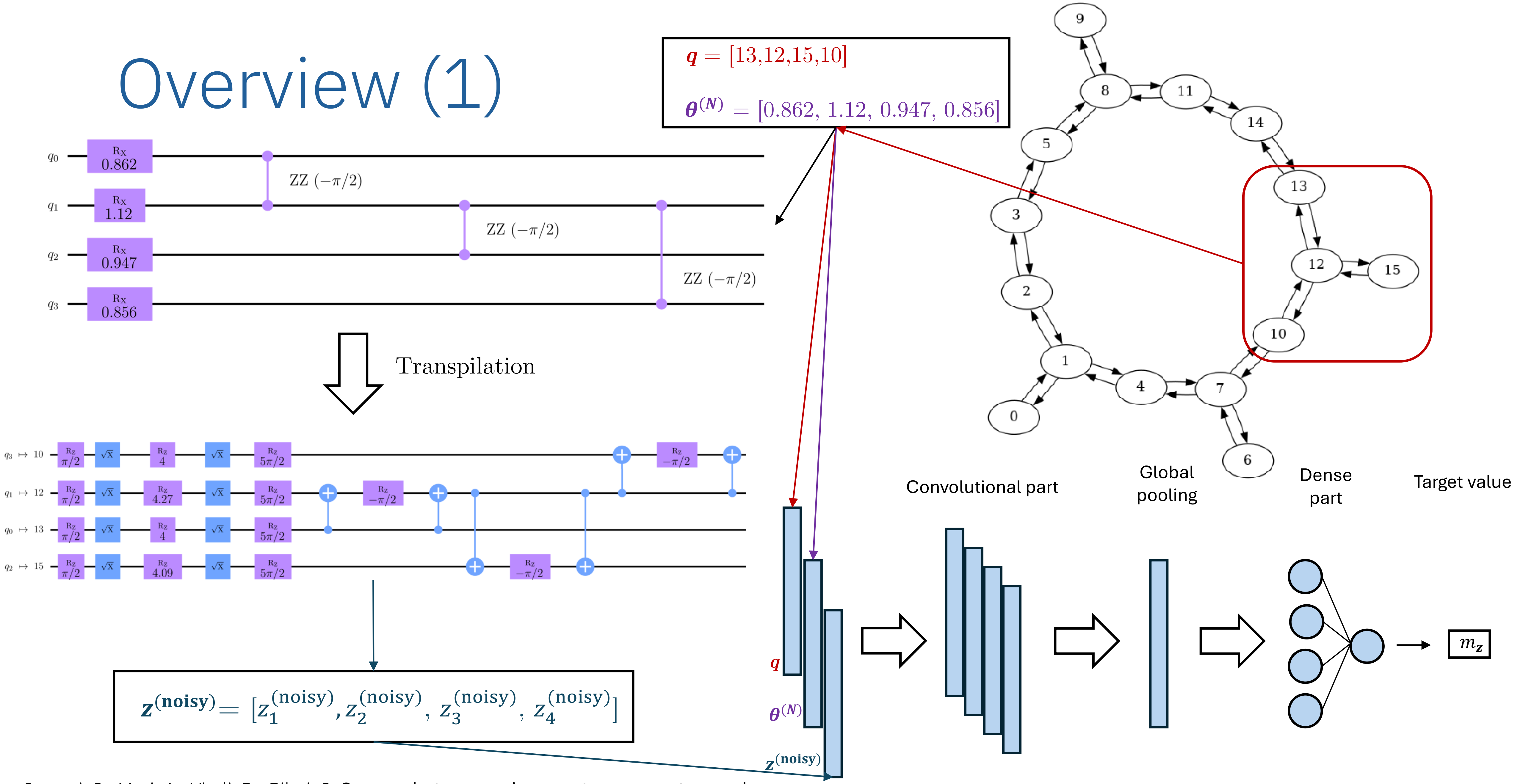
$N = 4$
$N = 6$
$N = 8$
$N = 10$
$N = 12$
$1 - R^2$
10^0
10^{-1}
10^{-2}
10^2
10^3
10^4
K_{train}

Filtering the shot noise of target values





Overview (1)

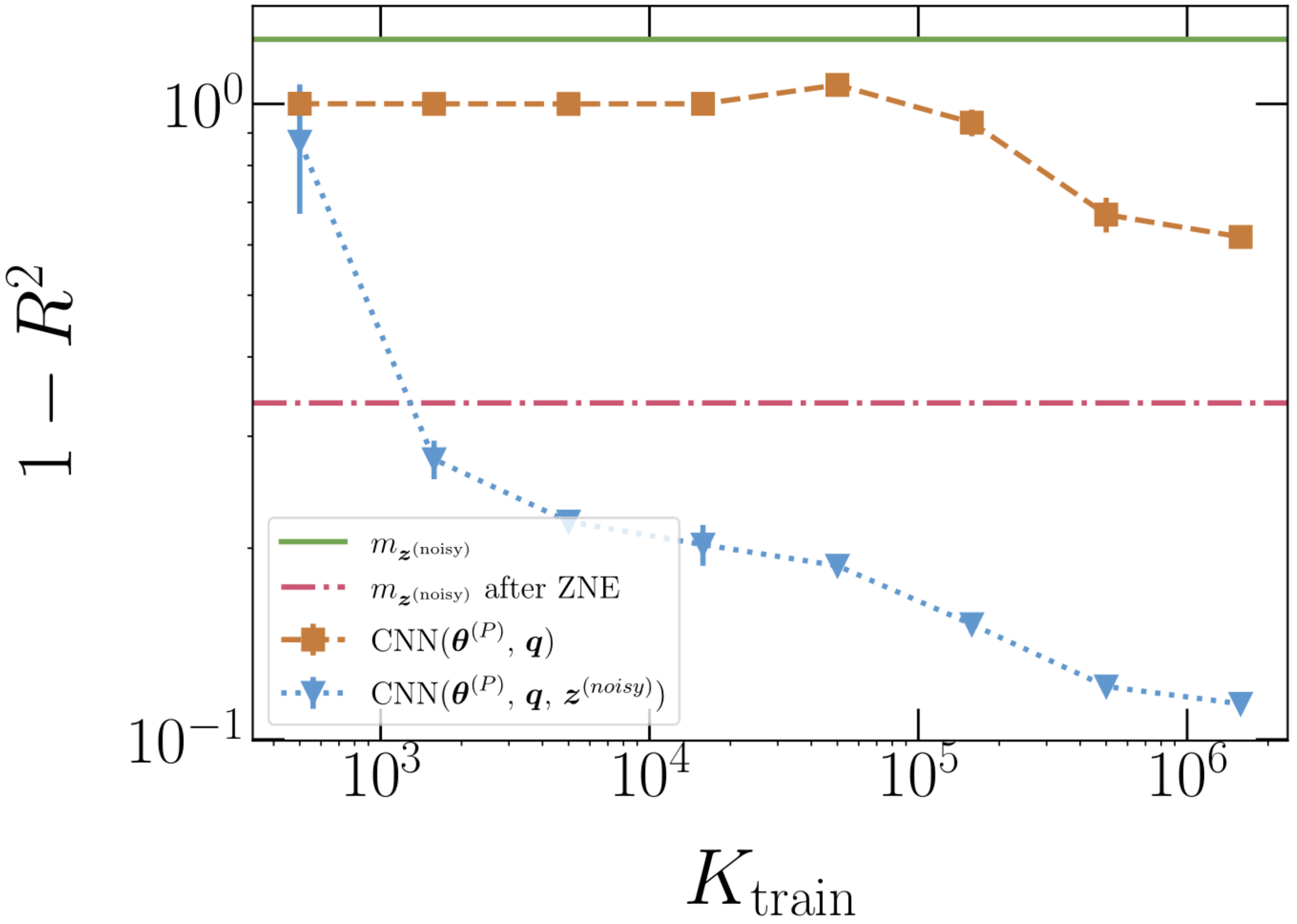


Convolutional neural network (CNN)

Cantori, S., Mari, A., Vitali, D., Pilati, S. **Synergy between noisy quantum computers and scalable classical deep learning for quantum error mitigation.** *EPJ Quantum Technol.* 11, 45 (2024)

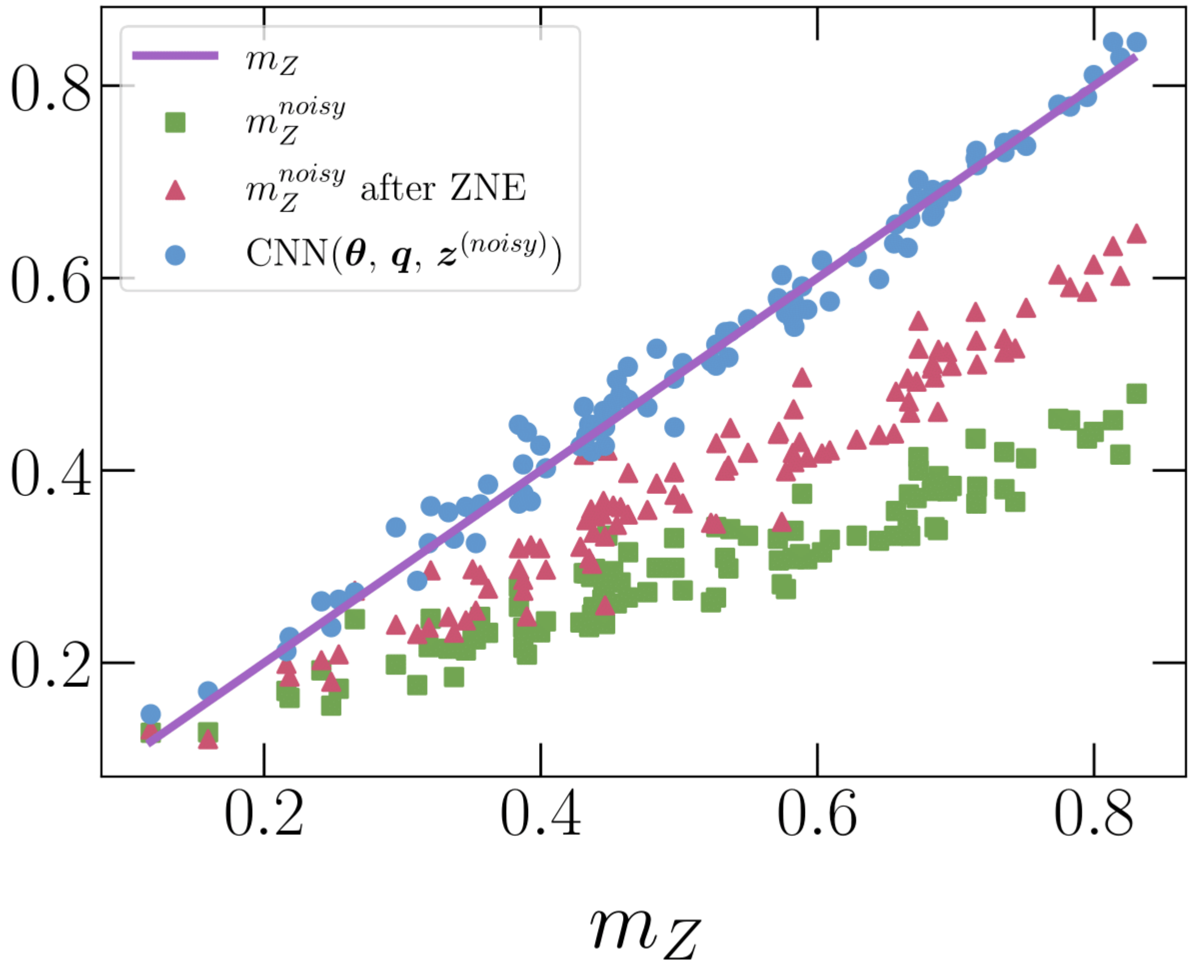
Impact of the training-set size

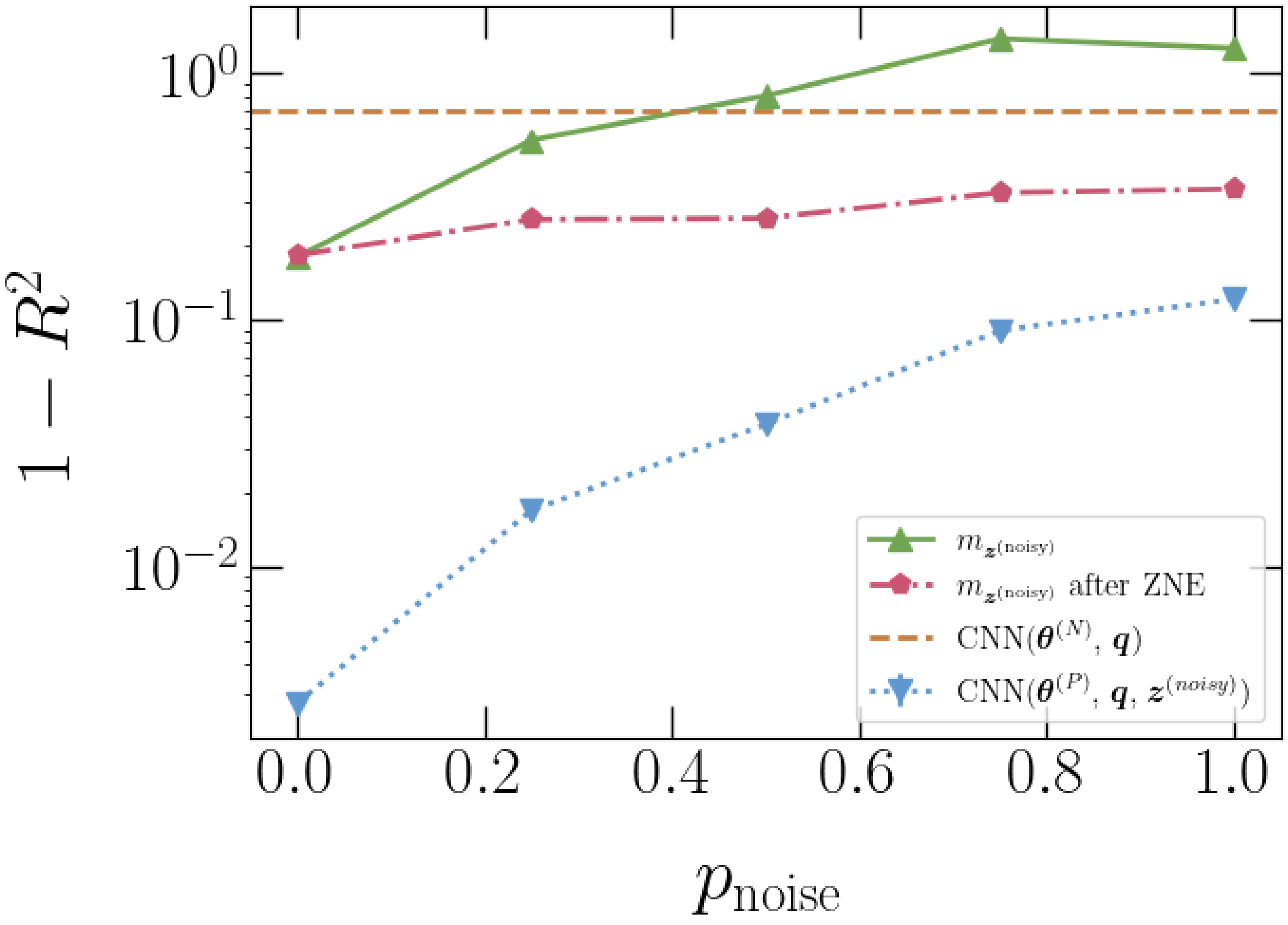
- $1 - R^2 = \frac{\sum_{i=1}^{K_{test}} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{K_{test}} (y_i - \bar{y})^2}$.
- Training on quantum circuits with $N \leq 10$ qubits and testing on quantum circuits with $N = 16$ qubits.
- $P = 20$ layers of gates.



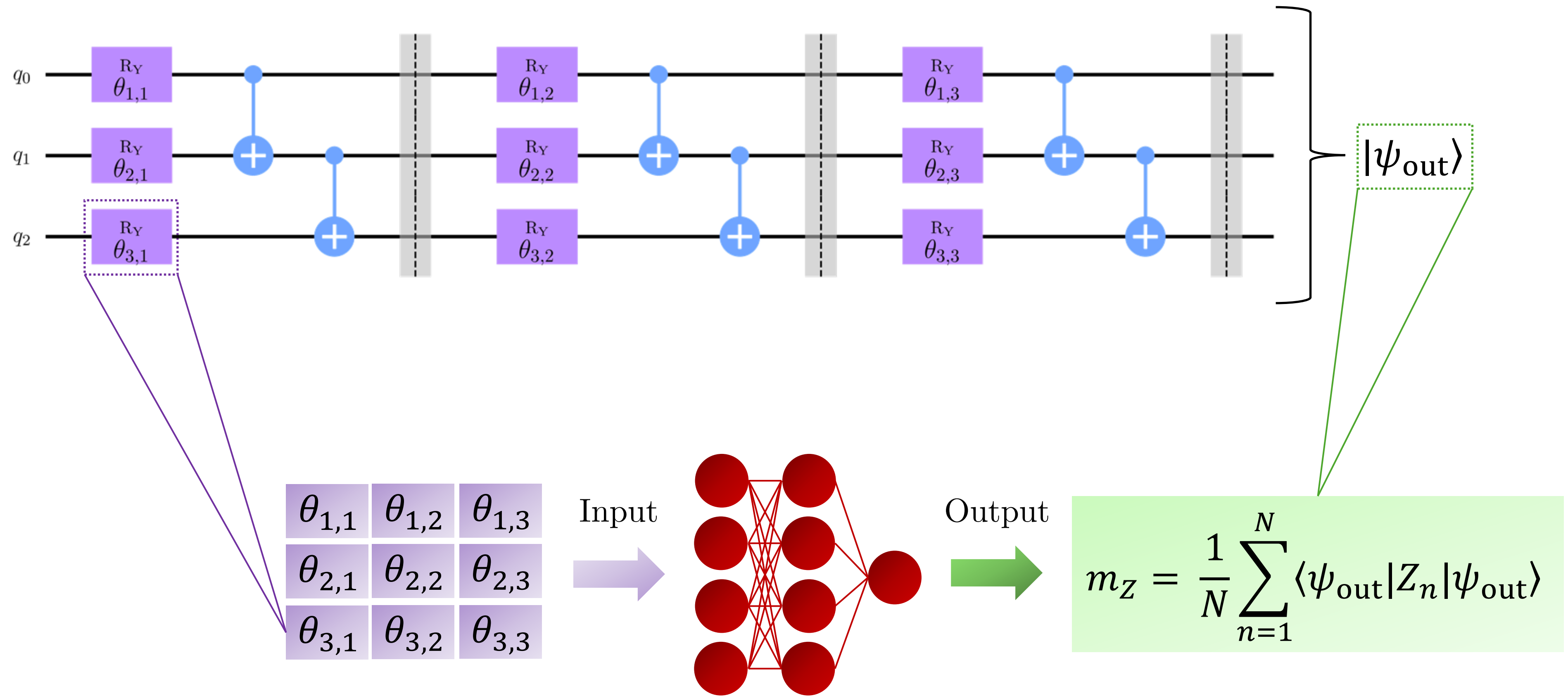
Visualization of the improvement

- Training on quantum circuits with $N \leq 10$ qubits and testing on quantum circuits with $N = 16$ qubits.
- $P = 20$ layers of gates.

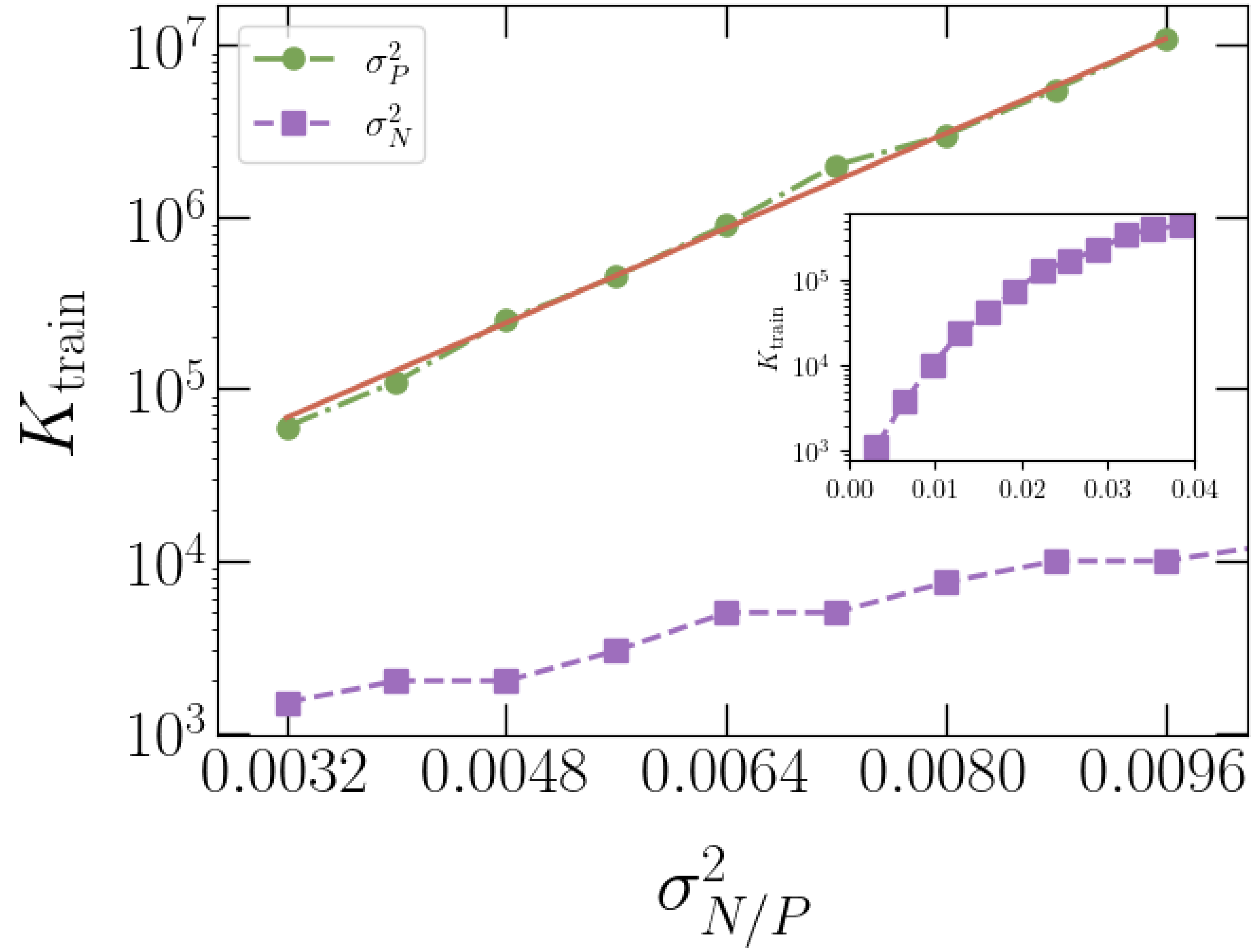




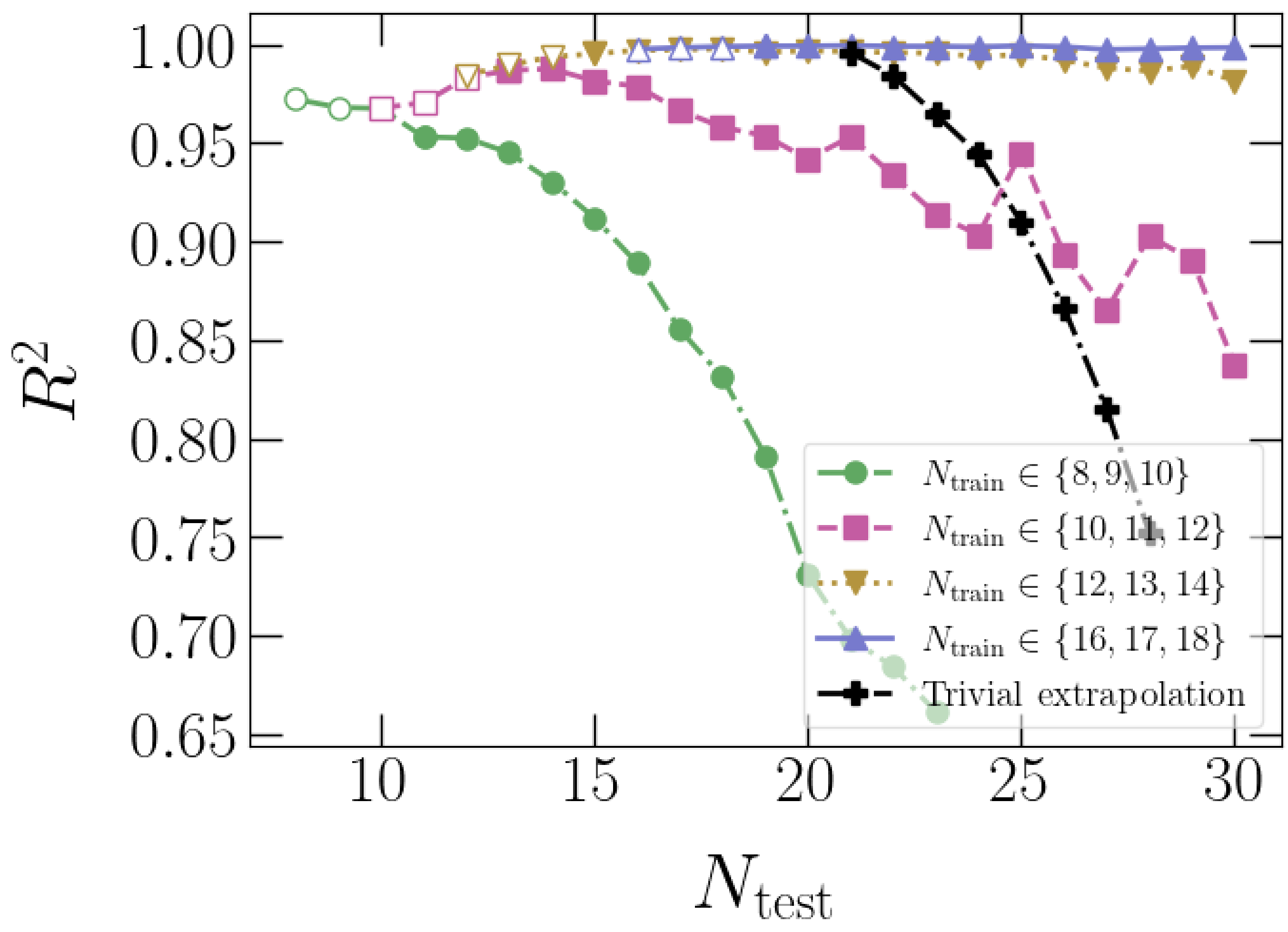
10^{0}
10^{-1}
10^{-2}
$1 - R^2$
0.0
0.2
0.4
0.6
0.8
1.0
p_{noise}
$m_{\boldsymbol{z}^{(\text{noisy})}}$
$m_{\boldsymbol{z}^{(\text{noisy})}}$ after ZNE
CNN($\boldsymbol{\theta}^{(N)}$, $\boldsymbol{q}$)
CNN($\boldsymbol{\theta}^{(P)}$, $\boldsymbol{q}$, $\boldsymbol{z}^{(noisy)}$)



q_0
q_1
q_2
R_Y $\theta_{1,1}$
R_Y $\theta_{2,1}$
R_Y $\theta_{3,1}$
R_Y $\theta_{1,2}$
R_Y $\theta_{2,2}$
R_Y $\theta_{3,2}$
R_Y $\theta_{1,3}$
R_Y $\theta_{2,3}$
R_Y $\theta_{3,3}$
$|\psi_{\text{out}}\rangle$
$\theta_{1,1}$ $\theta_{1,2}$ $\theta_{1,3}$
$\theta_{2,1}$ $\theta_{2,2}$ $\theta_{2,3}$
$\theta_{3,1}$ $\theta_{3,2}$ $\theta_{3,3}$
Input
Output
$m_Z = \frac{1}{N}\sum_{n=1}^{N}\langle\psi_{\text{out}}|Z_n|\psi_{\text{out}}\rangle$



10^7
10^6
10^5
10^4
10^3
K_{train}
σ_P^2
σ_N^2
0.0032
0.0048
0.0064
0.0080
0.0096
$\sigma_{N/P}^2$
0.00
0.01
0.02
0.03
0.04



1.00
0.95
0.90
0.85
0.80
0.75
0.70
0.65
R^2
10
15
20
25
30
N_{test}
$N_{\text{train}} \in \{8, 9, 10\}$
$N_{\text{train}} \in \{10, 11, 12\}$
$N_{\text{train}} \in \{12, 13, 14\}$
$N_{\text{train}} \in \{16, 17, 18\}$
Trivial extrapolation