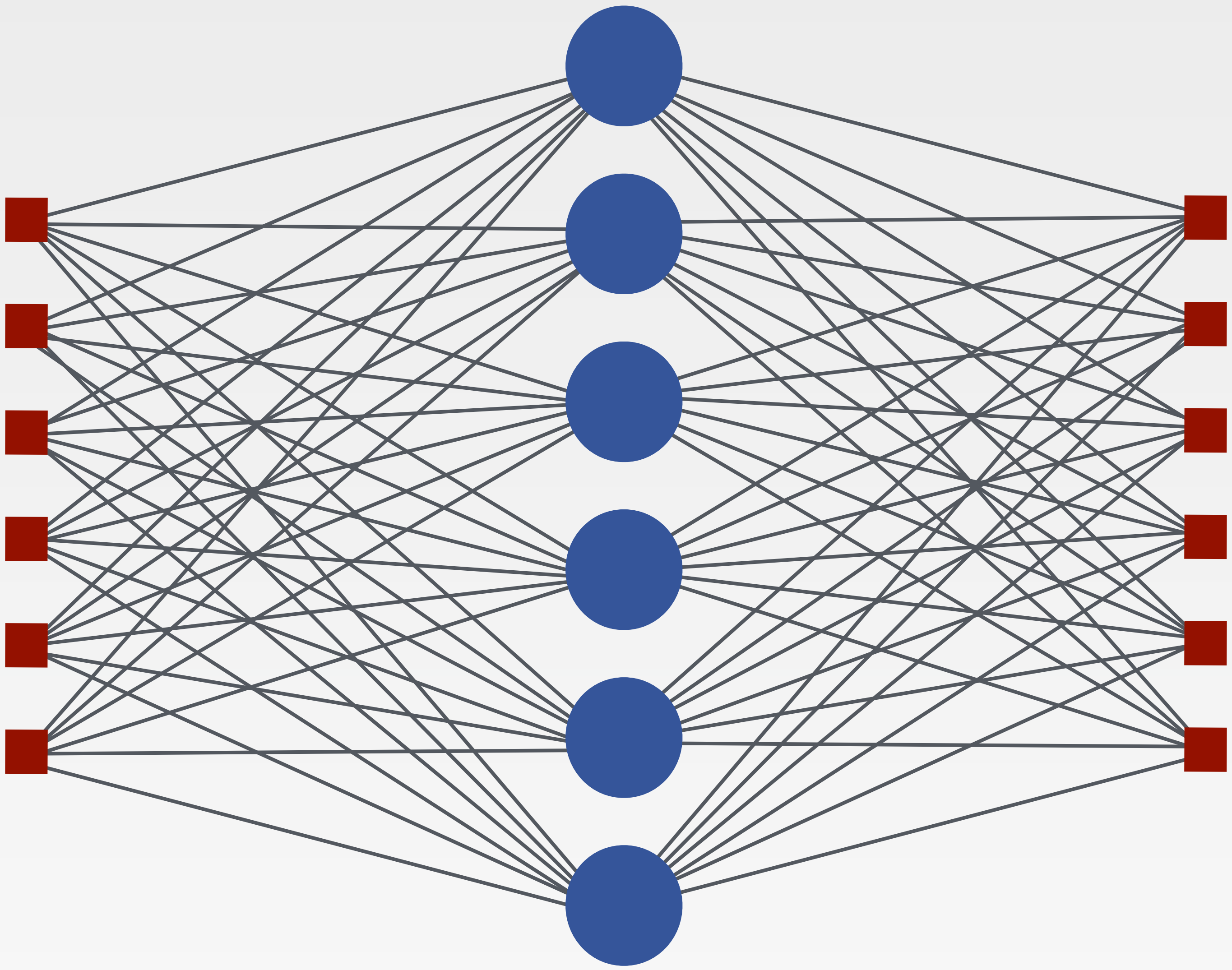


Neural network applications to spectral reconstruction

Antonin Portelli

26th of August 2019

Santa Fe, NM, USA

In collaboration with

Max T. Hansen (CERN) and José Angel Romero (IFT)

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No 757646.

- Motivations
- Spectral reconstruction as an inverse problem
- Case study: the light vector resonance
- Summary & perspectives

Motivations

Motivations

- Traditional lattice analysis: extract energies and matrix elements through fitting the exponential behaviour of correlators.
- **Density of state increases quickly with the volume.** Standard analysis increasingly challenging.
- Non-gapped correlators (QED, scattering, …) **Even ground state challenging at large volume.**
- Reconstruction of **long-distance effects**.
- **How to go beyond exponential fits?**

Spectral reconstruction as an inverse problem

Spectral integral equation

- In the case of 2-point functions
  Källén-Lehmann spectral representation

$$D(q^2) = \int \mathrm{d}^4x \, \langle \phi(x)\phi(0)^\dagger \rangle \, e^{-iq\cdot x} = \int_0^{+\infty} \mathrm{d}s \, \frac{\rho(s)}{q^2+s}$$

- Spectral reconstruction from lattice data:
  Find ρ given D.

- Integral operator $(S\rho)(q^2) = \int \mathrm{d}s \, \frac{\rho(s)}{q^2+s}$.
  Solution "$\rho = S^{-1}D$".
  Linear problem

Naive approach

- Arbitrary discretisation q_j^2 and s_k

$$S_{jk} = \frac{1}{q_j^2 + s_k}$$

- Distinct points: unique solution

$$\det(S) = \frac{\prod_{k>j}(q_j^2 - q_k^2)(s_j - s_k)}{\prod_{j,k}(q_j^2 + s_k)}$$

- In practice: extremely ill-conditioned matrix (typically $\kappa \simeq \mathcal{O}(10^{100})$)

Appropriateness of machine learning

- Machine-learning approach: **fit a parametrisation of the inverse transform** on a relevant functional space.
- Advantage here:
  trivial to construct random problem-solution pairs.
- Neural networks are just a specific parametrisation.
- Existing work: [Kades *et al.*, arXiv:1905.04305]

Challenges

- Determine an appropriate neural architecture.
- Establish a strategy for generating training and validation datasets.
- Design a Monte-Carlo strategy to validate predictions on synthetic lattice data.
- *Investigate generalisation of trained models.*
- *Compare against existing methods.*

Case study: the light vector resonance

Generalities

- Vector-vector spectral representation

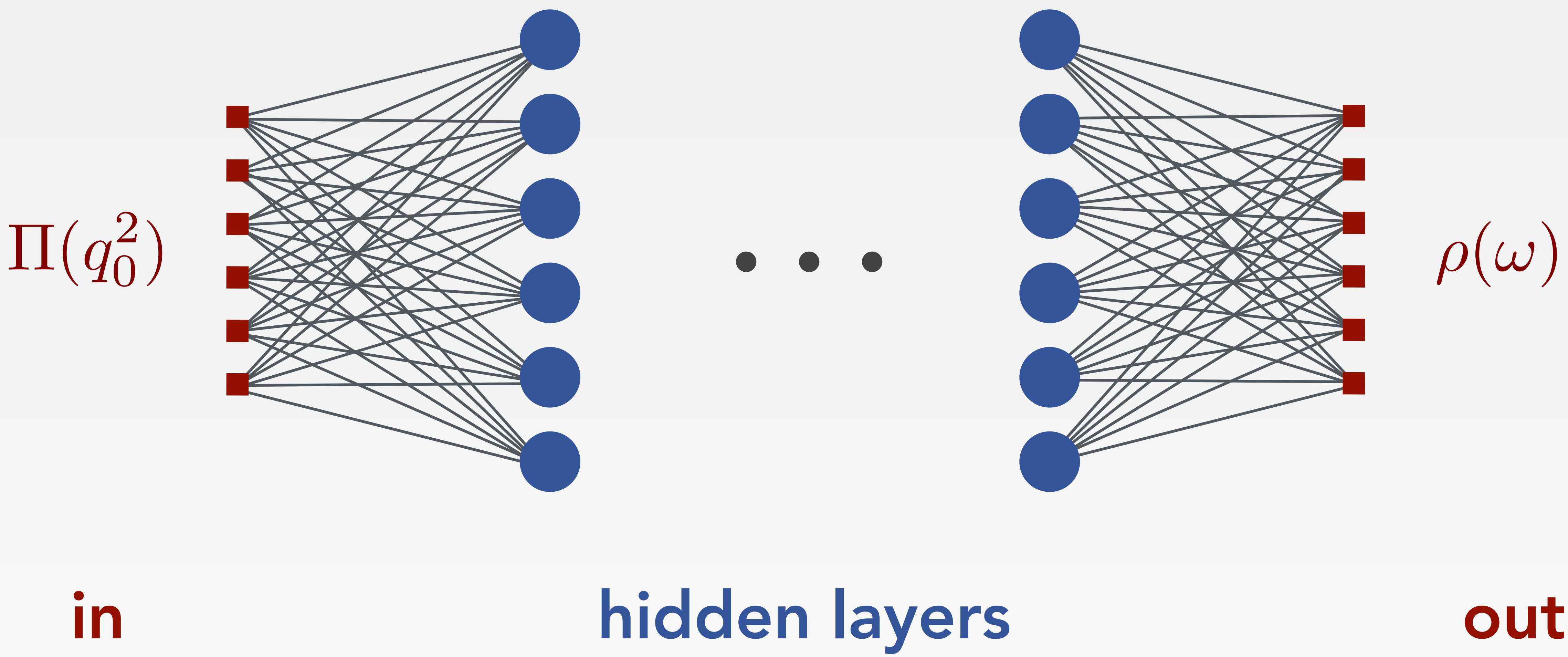
$$C_{\mu\nu}(t) = \int \mathrm{d}^3\mathbf{x}\ \langle 0|\, \mathrm{T}[J_\mu(t,\mathbf{x})J_\nu(0)]\, |0\rangle$$

$$\Pi_{\mu\nu}(q_0) = \int \mathrm{d}t\, C_{\mu\nu}(t)e^{-iq_0 t} = \delta_{\mu\nu}q_0^2\Pi(q_0^2)$$

$$\Pi(q_0^2) = \int_0^{+\infty} \mathrm{d}s\, \frac{\rho(s)}{q_0^2 + s}$$

$$\rho(s) = \frac{1}{3s}\int \mathrm{d}\alpha \textstyle\sum_\mu |\langle 0|\, J_\mu(0)\, |\alpha\rangle|^2 \delta(\omega_\alpha^2 - s)$$

Neural network architecture



Size	$\frac{N_T}{2} - 1$	N_h	N_ω
Step	$\frac{2\pi}{T}$		$\delta\omega$

● : $\tanh(x)$ ■ : x

Data preconditioning

- **Momentum space input layer**
  Reduce the hierarchy of weights, accelerate training.
  Just a change of basis on the input layer.

- **Linear scaling enforcement**
  Always normalise in and out by the input average.
  Quality of prediction insensitive to overall scale.

Training

- Loss function: 2-norm difference

$$L(\Pi, \rho) = \frac{1}{N} \sum_j \|F(\Pi_j) - \rho_j\|_2^2$$

F: neural net

- Training: stochastically minimise L on a training set
  Monitor loss through an independent validation set.

- Mini-batch size 4

- ADAM minimiser with “reduce-on-plateau” adaptive learning rate.

Training/Validation set

- Training set: 5000 random Breit-Wigner examples

$$300\ \mathrm{MeV} \leq M \leq 1000\ \mathrm{MeV}$$
$$120\ \mathrm{MeV} \leq \Gamma \leq 300\ \mathrm{MeV}$$

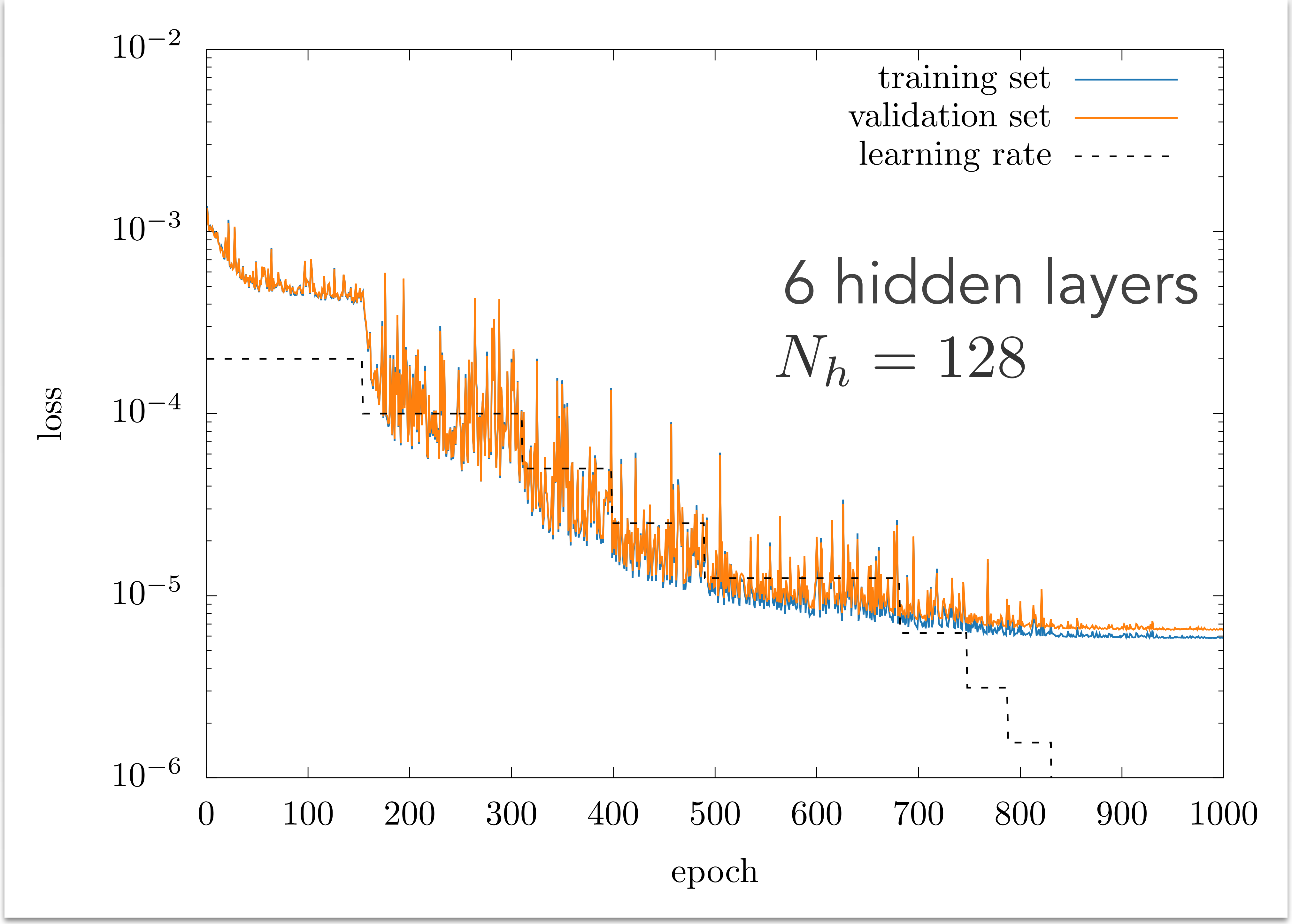
- Validation set: same, with 1000 examples.

- Loss function value not very meaningful.

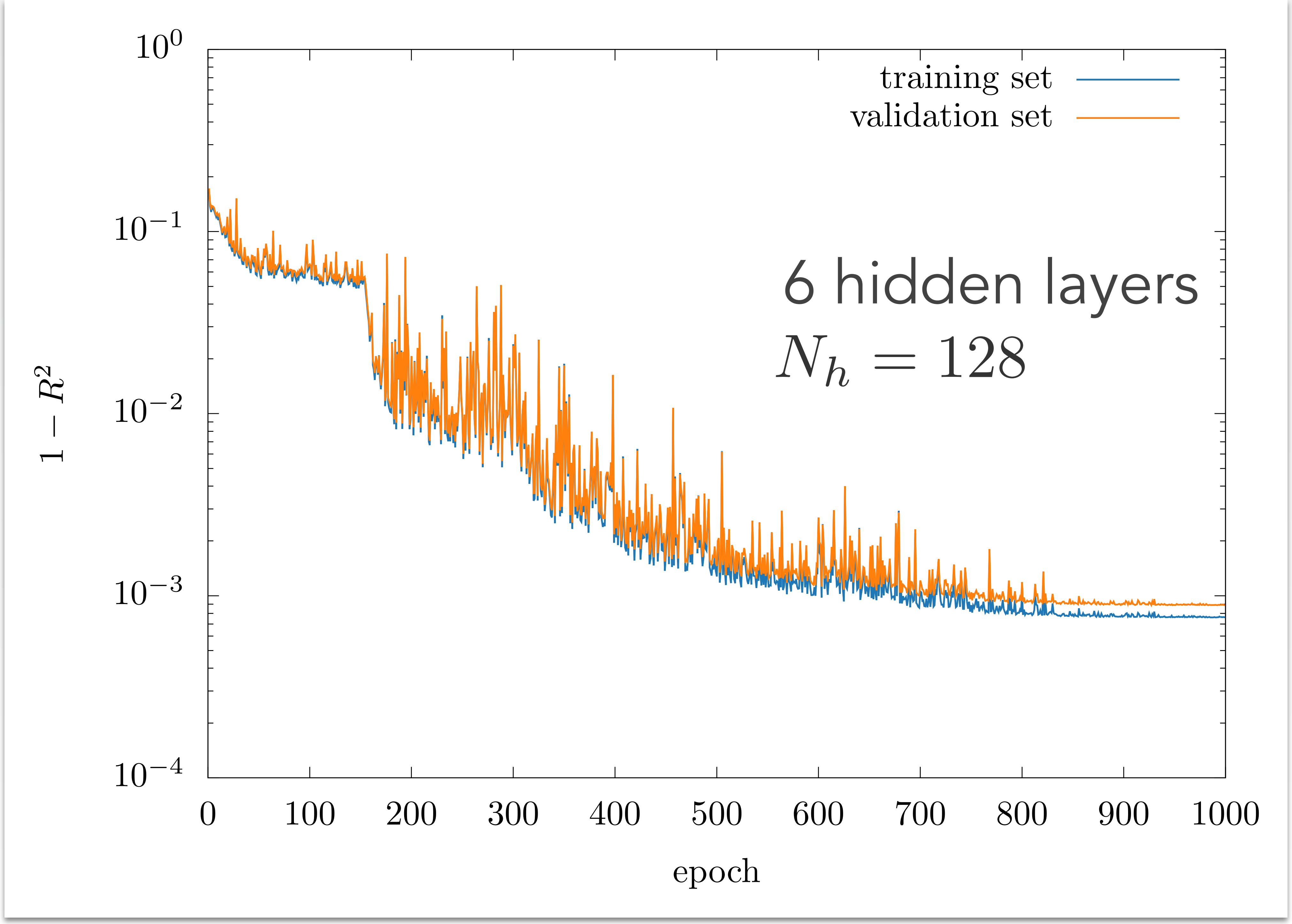
- R^2-score more useful

$$R^2 = 1 - \frac{\|F(\Pi) - \rho\|_2^2}{\mathrm{Var}(\rho)}$$

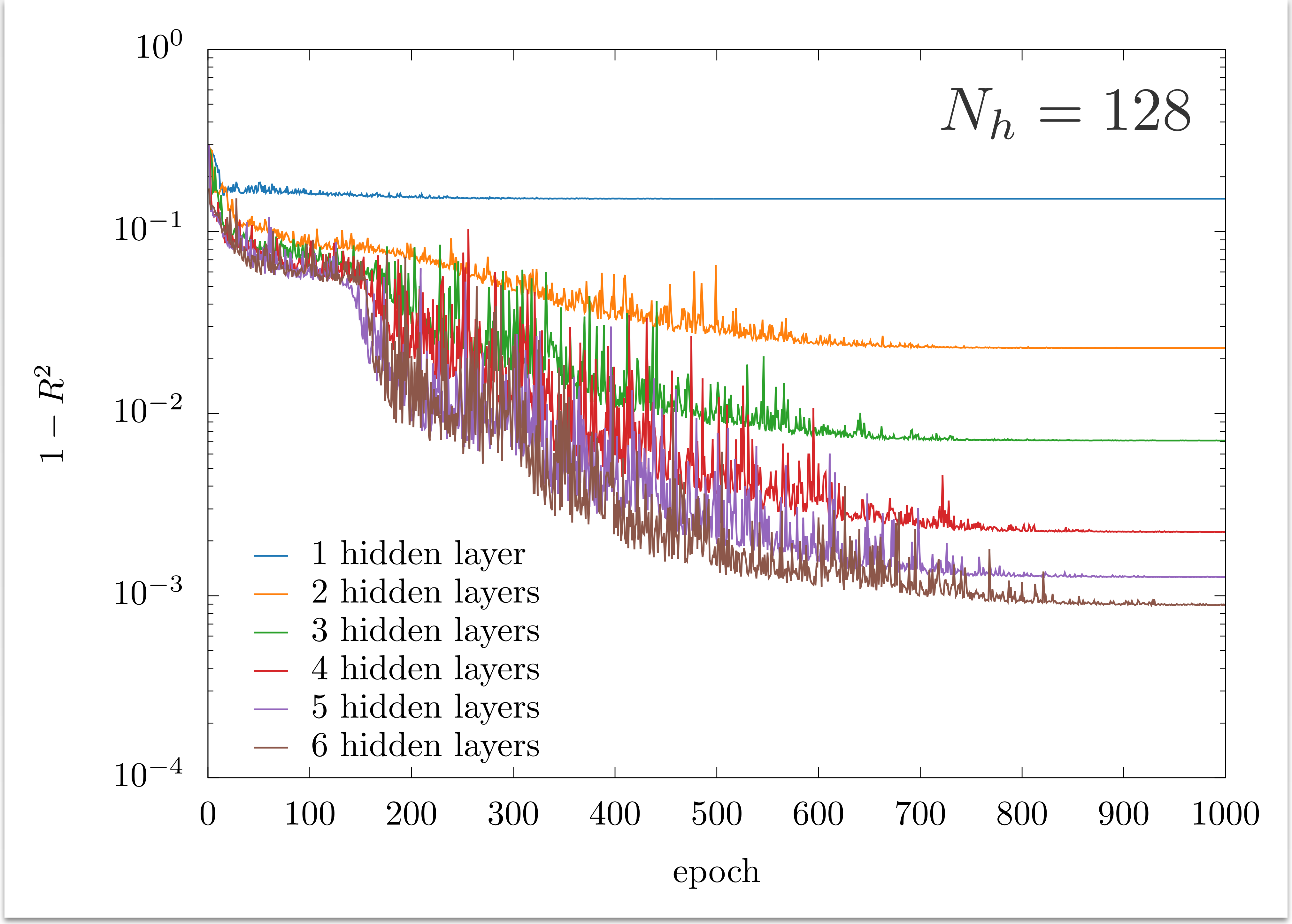
Learning curve example



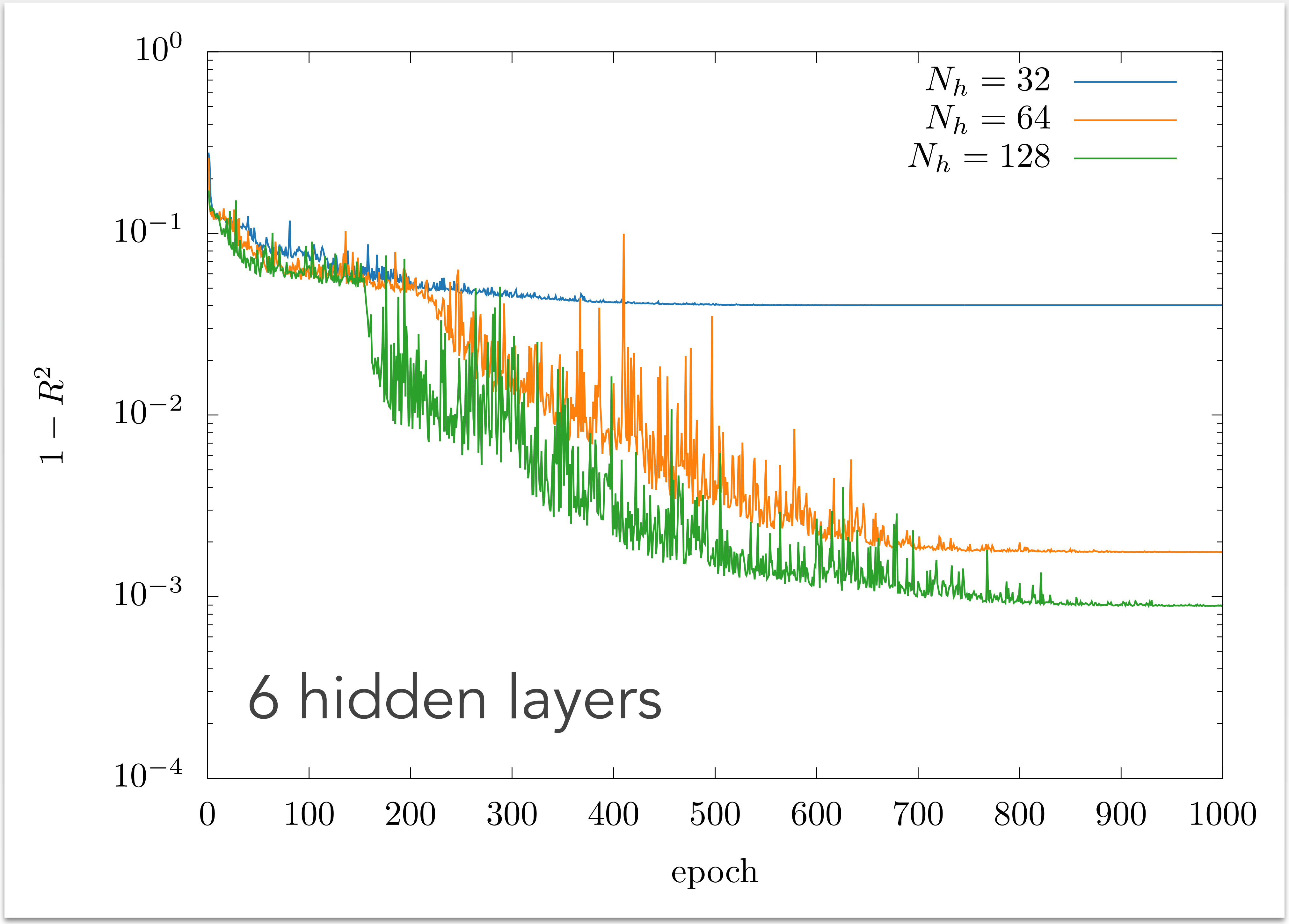
Learning curve example



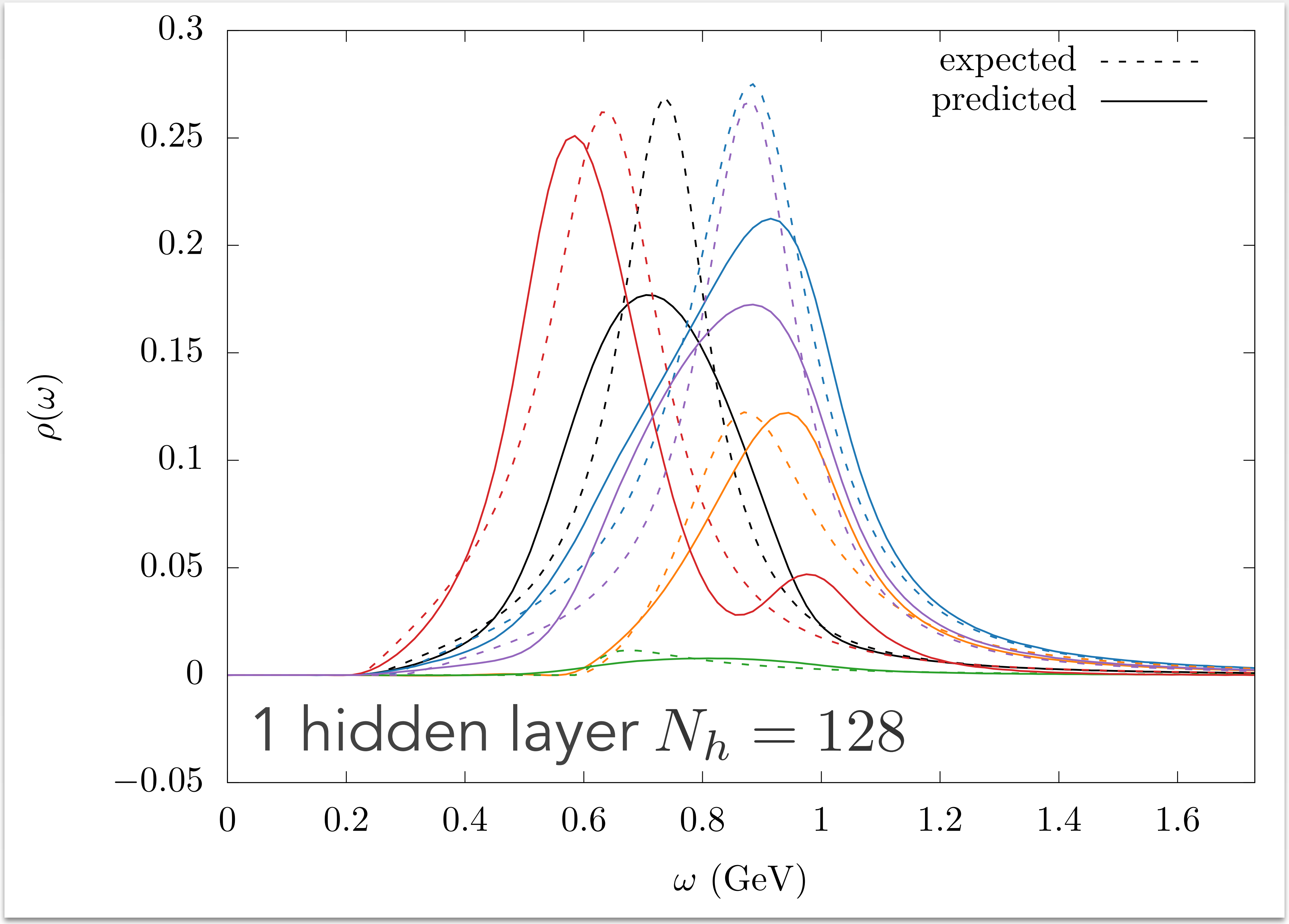
Learning curves vs. hidden layers



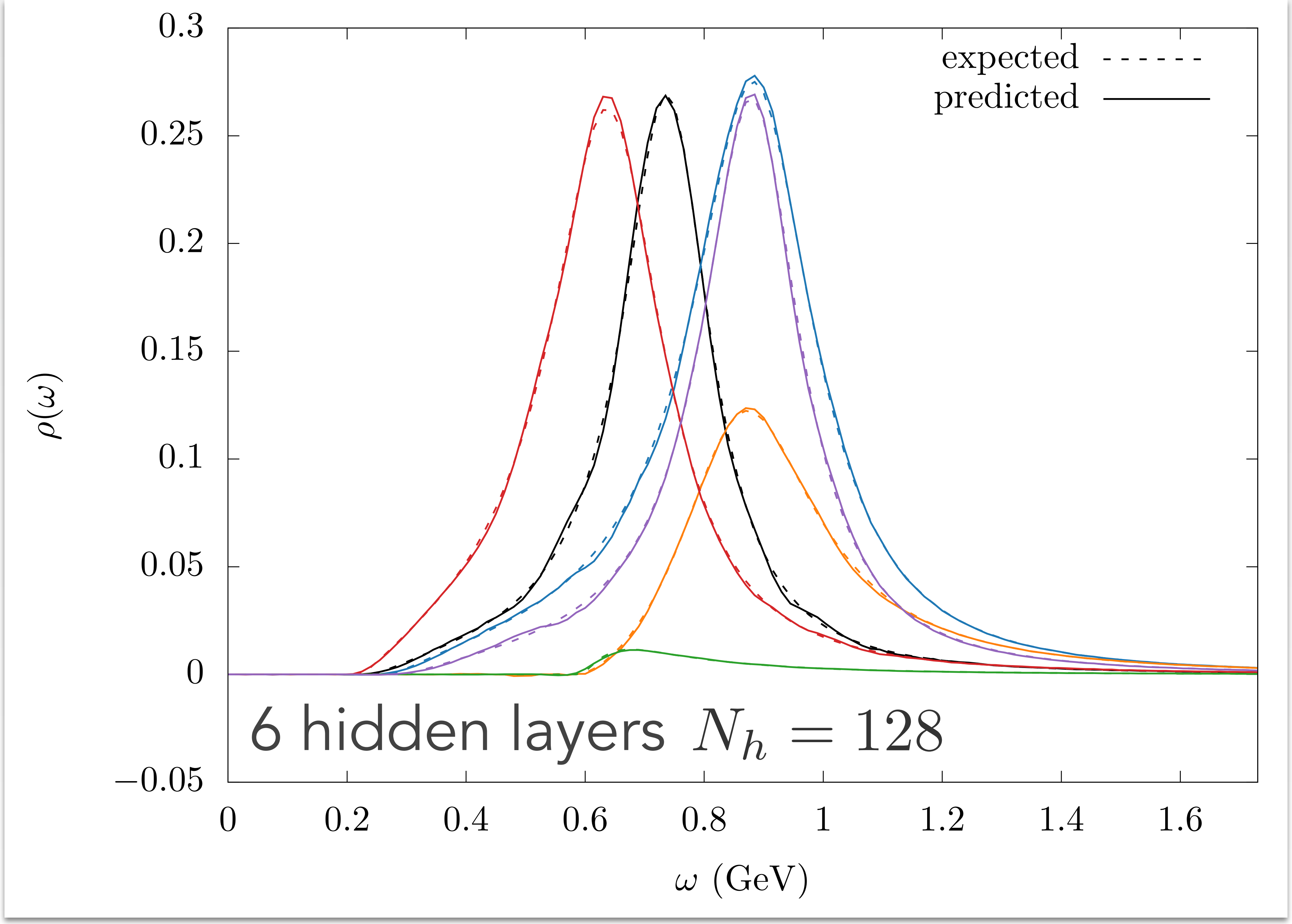
Learning curves vs. hidden layer size



Validation set sample



Validation set sample



Synthetic lattice data

- Correlator based on Gunaris-Sakurai parametrisation with physical parameters.

- Phenomenological variance model (work in progress)

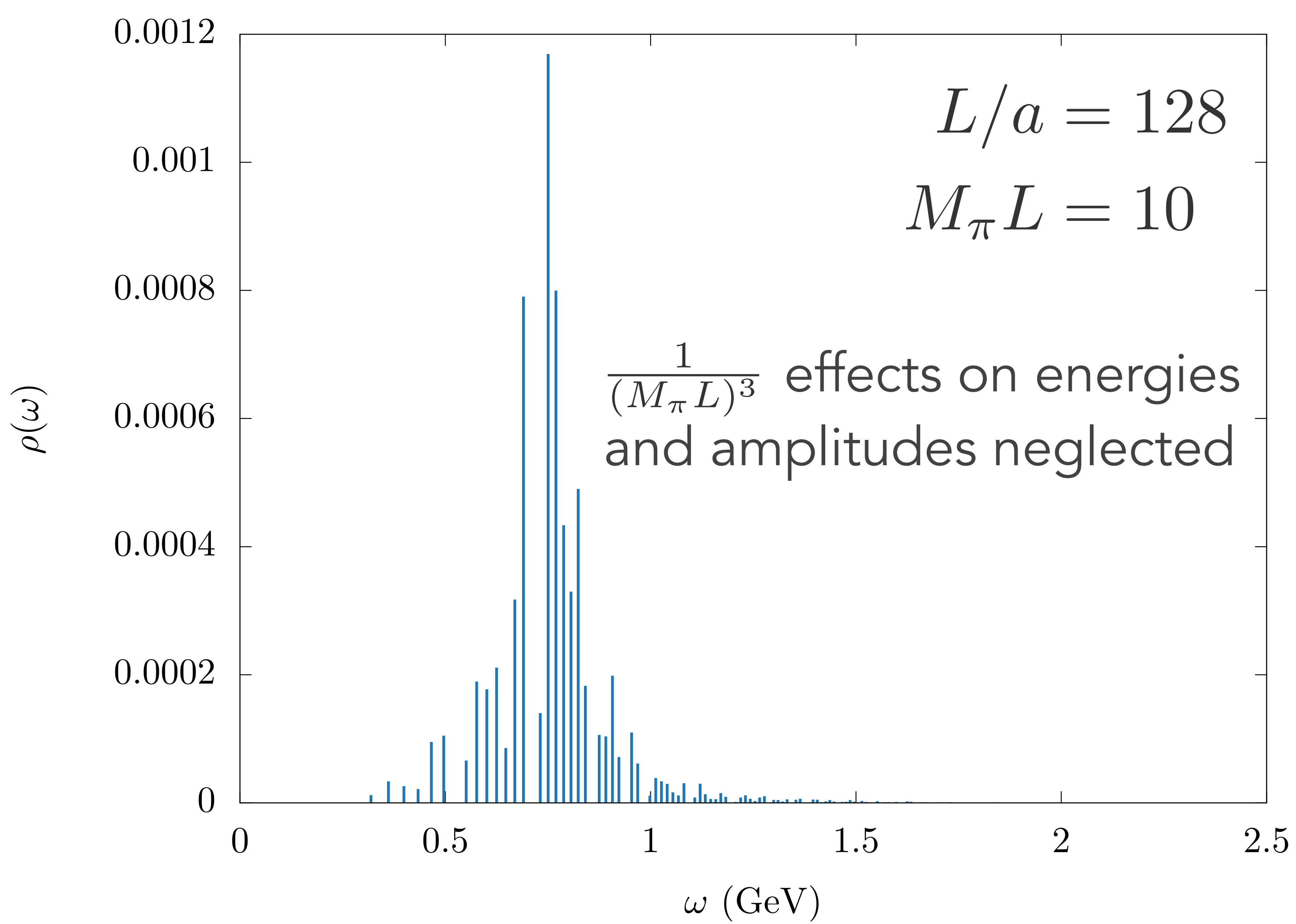
$$C_{jk} = \frac{1}{1 + \frac{1-\alpha}{\alpha}(j-k)^2} \qquad \rho_j = \frac{1}{j^2}\frac{\beta}{1+\gamma^2(j-1)^2}$$

C_{jk} : correlation between $\Pi(q_j^2)$ and $\Pi(q_k^2)$

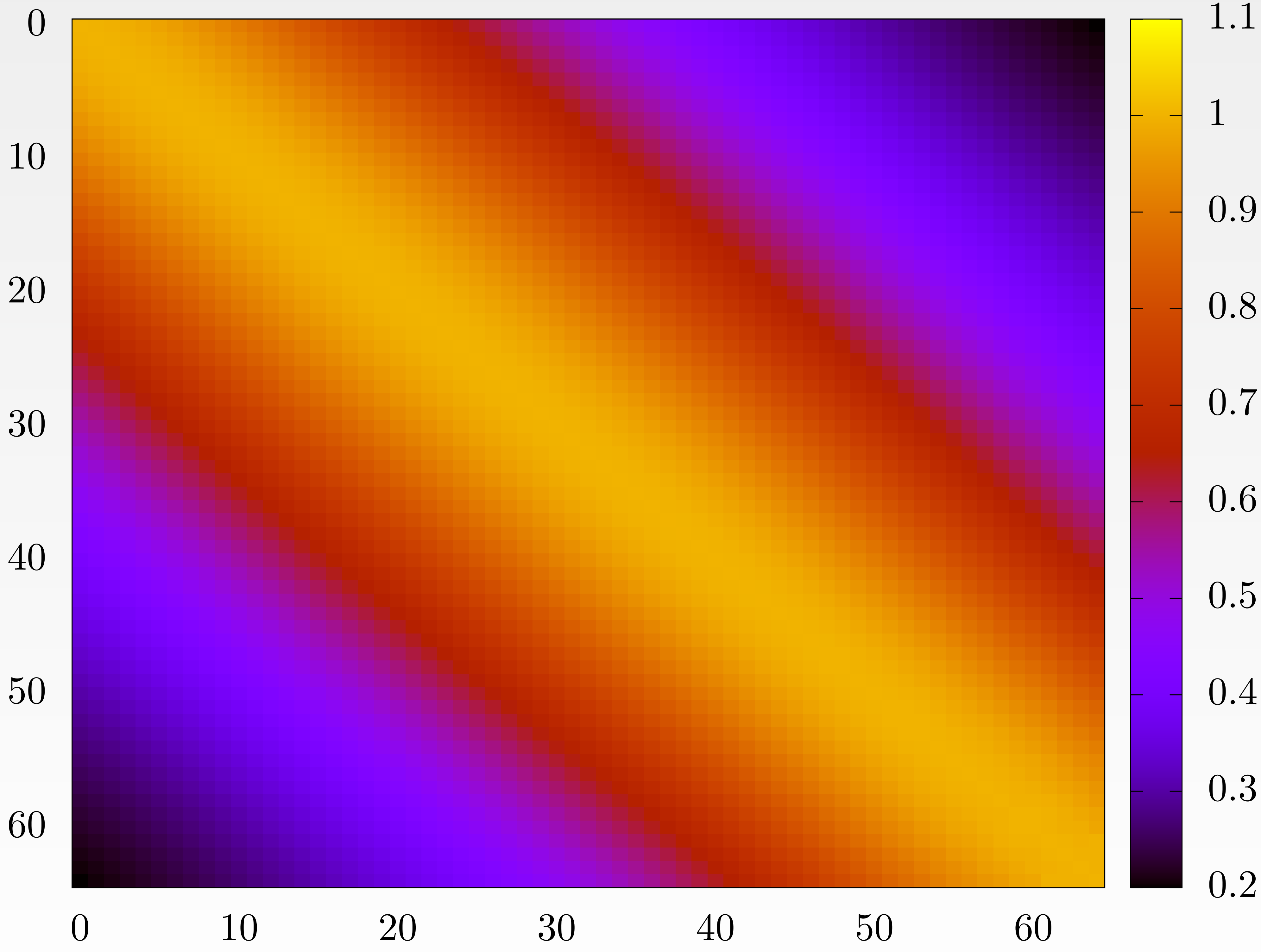
ρ_j : relative error on $\Pi(q_j^2)$

- Realistic for $\alpha = 99.9\%$ $\beta = \mathcal{O}(1\%)$ $\gamma = 0.0039$

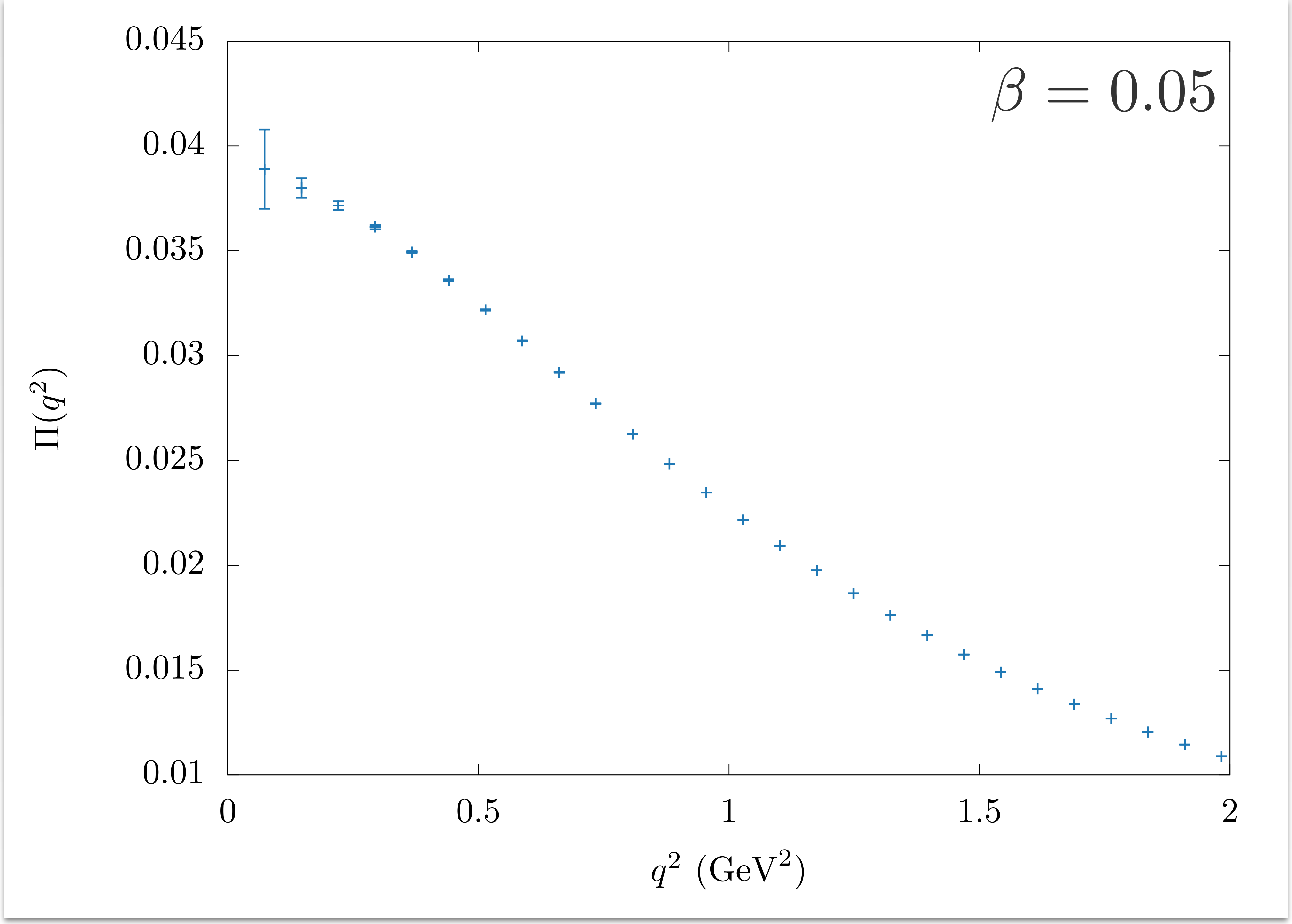
Finite-volume spectrum



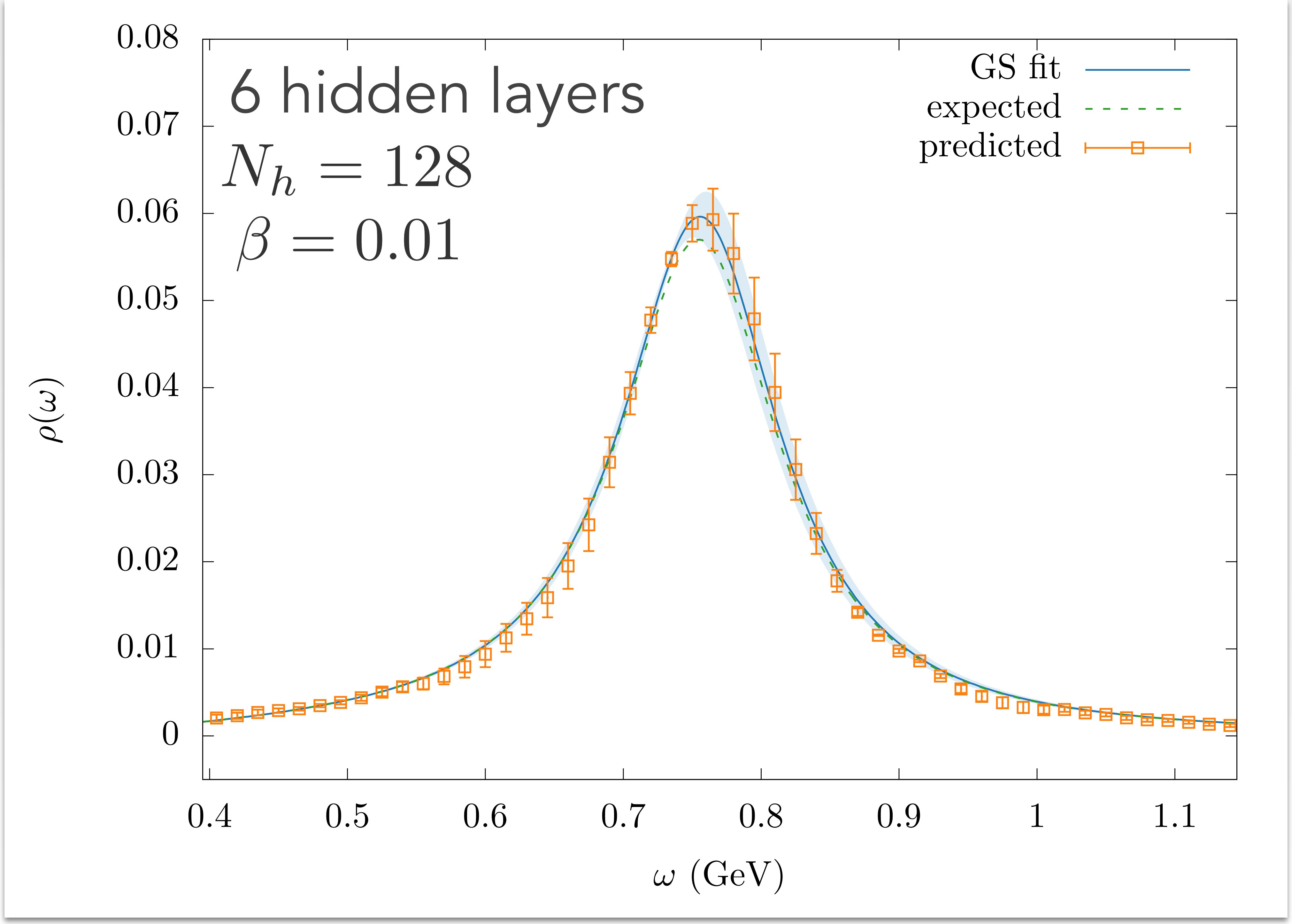
Synthetic lattice data



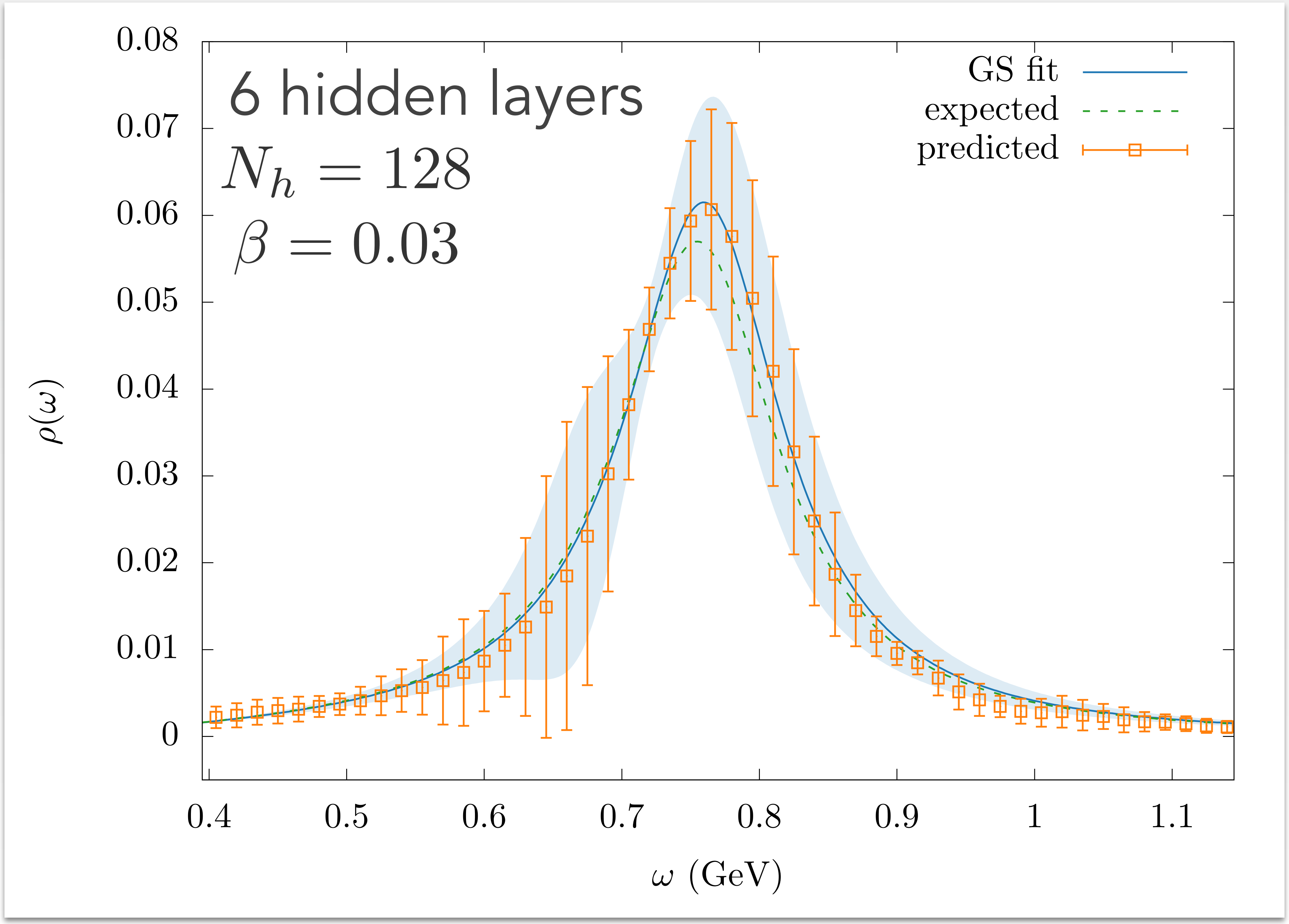
Synthetic lattice data



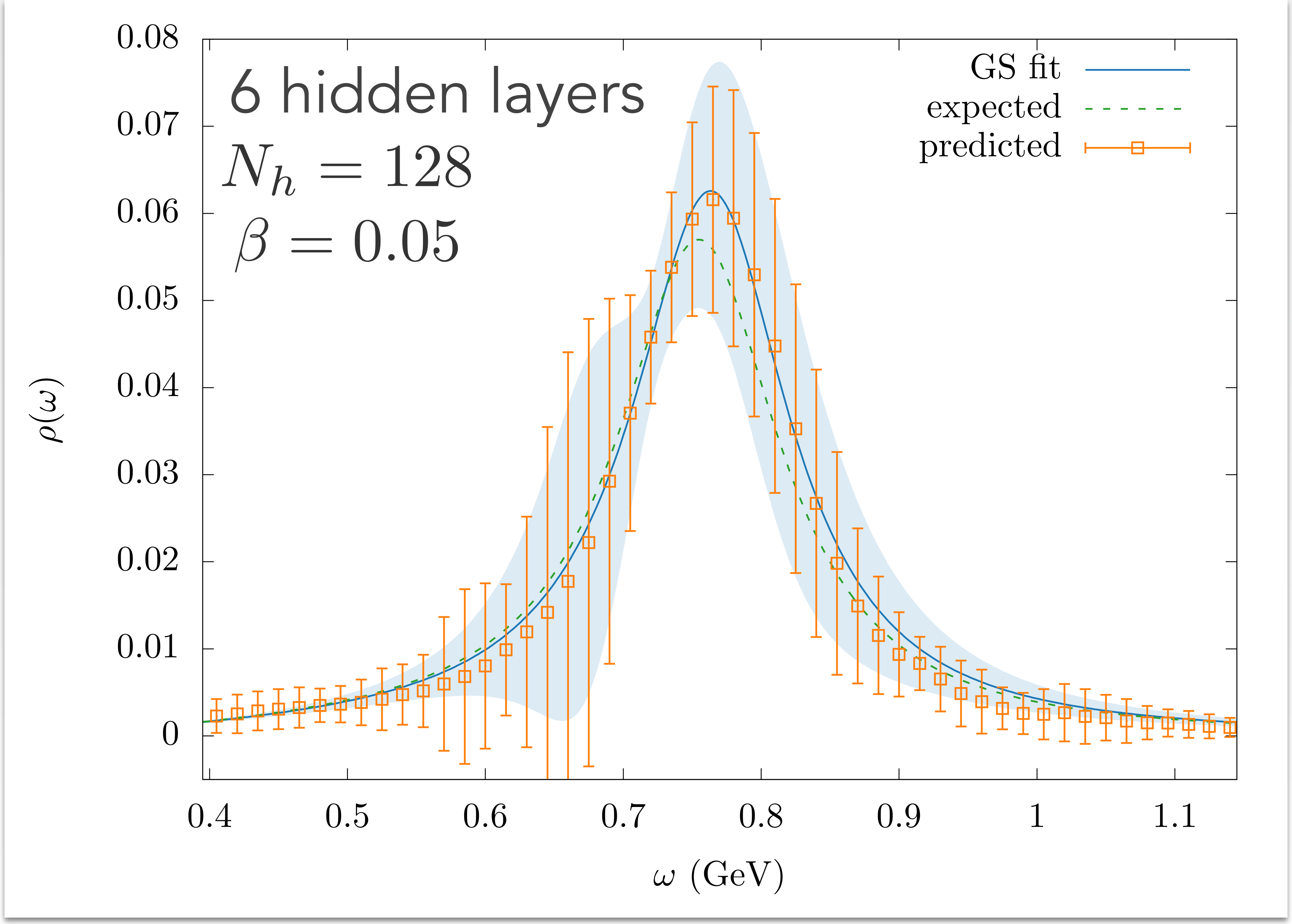
Prediction validation



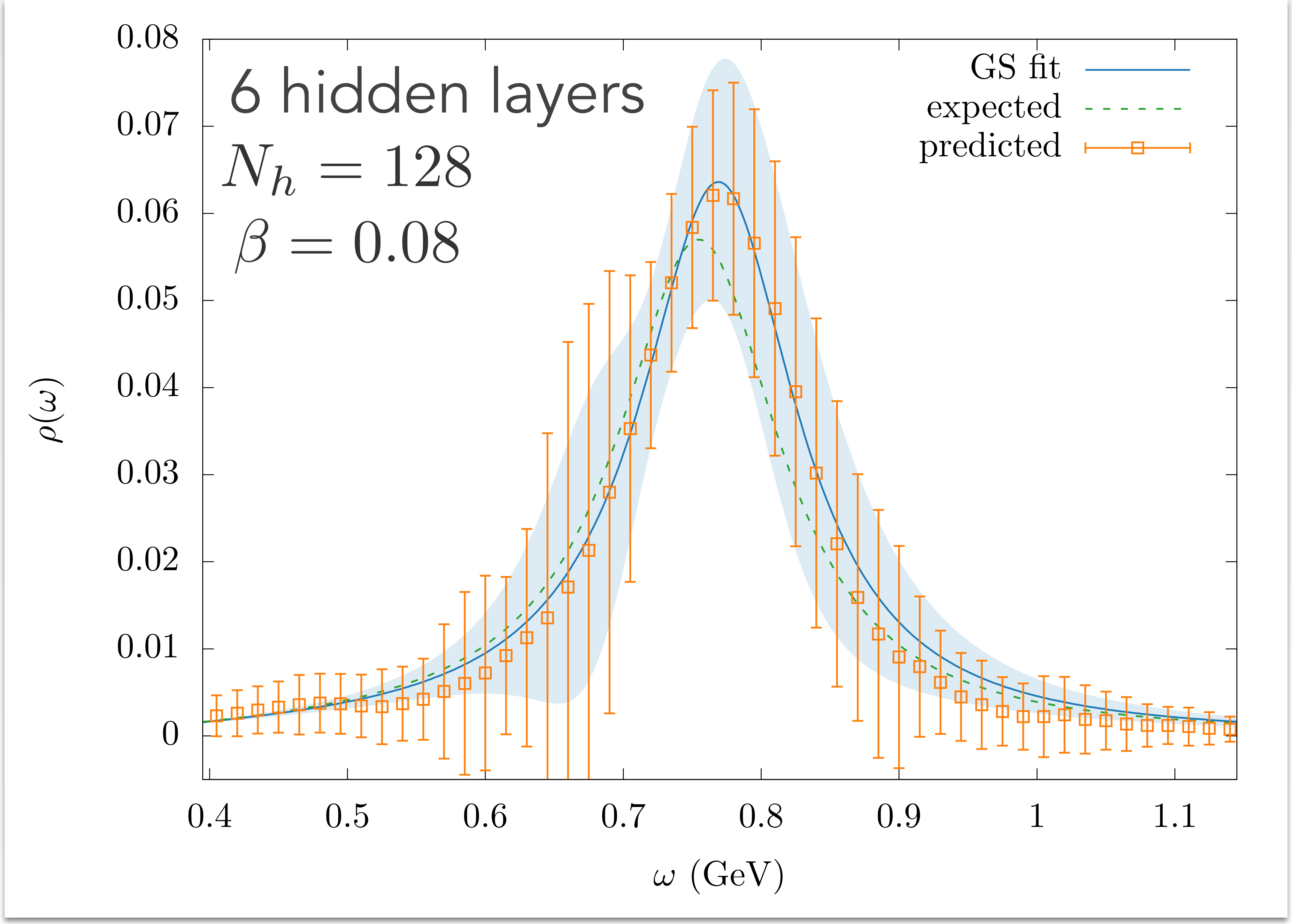
Prediction validation



Prediction validation

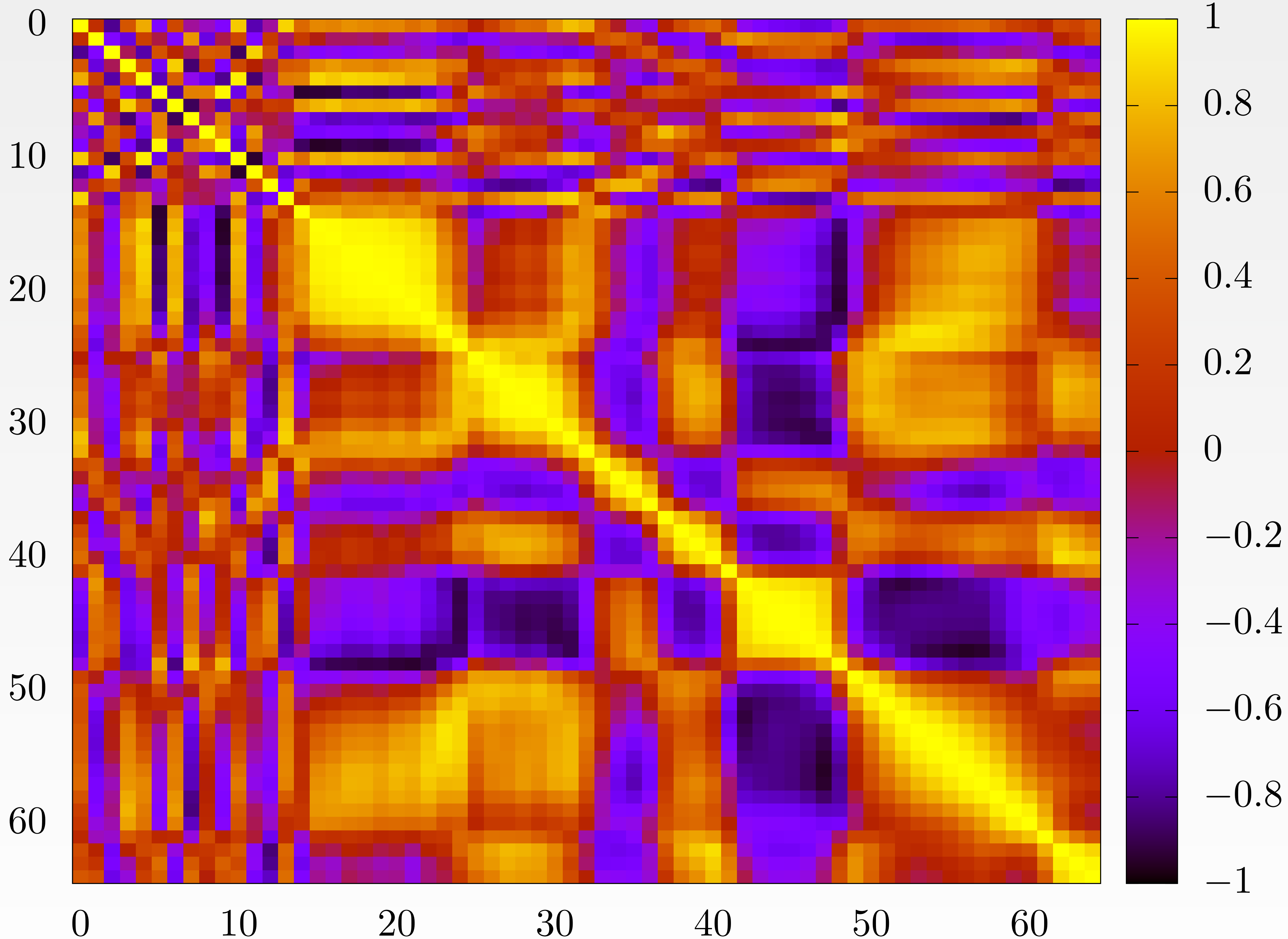


Prediction validation

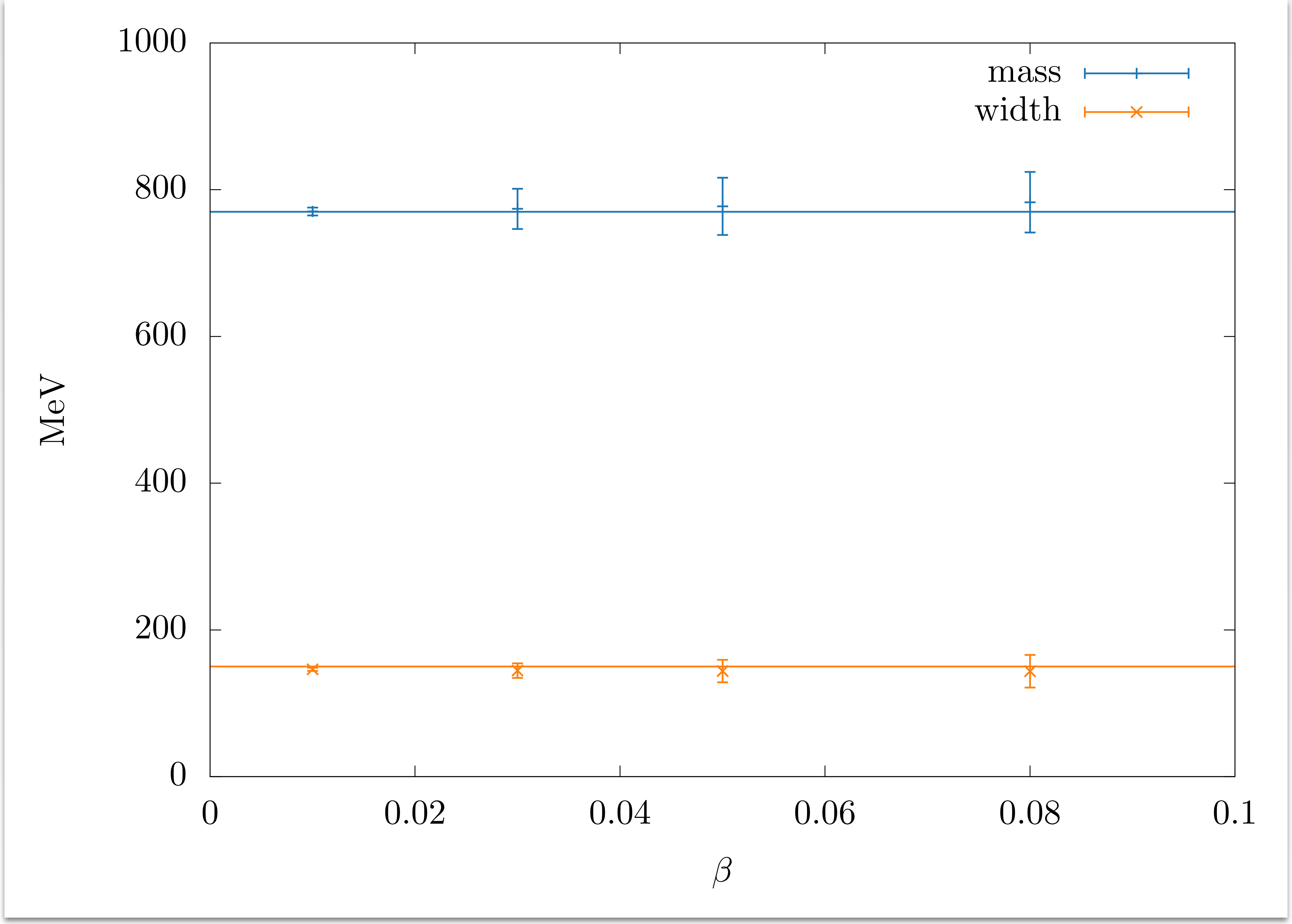


Prediction correlations

?!?



Summary of prediction validations



Summary & perspectives

Summary

- Neural networks are an **attractive approach to inverse problems** for data analysis.
- Neural networks allow to **infer non-linear generalisations** to linear algorithms.
- Spectral reconstruction: possible for unsmeared density using **deep** neural networks.
- **Seems to survive an prediction test** using realistic synthetic Monte-Carlo data.

Perspectives

- Direct comparison to Backus-Gilbert techniques.
- Study of generalisation
  (multi-peaks or even more arbitrary)
- Other approaches *e.g.* classifiers?
- Test on real large-volume lattice data.

Monument Valley 25/08/2019

Thank you!

Infinite-volume 2-pion contribution

- Amplitude and spectral measure separation

$$A(\omega_\alpha^2) = \sum_\mu |\langle 0| J_\mu(0) |\alpha\rangle|^2 \qquad \mu(s) = \int \mathrm{d}\alpha\, \delta(\omega_\alpha^2 - s)$$

$$\rho(s) = \frac{1}{3s} A(s)\mu(s)$$

- 2-pion contribution

$$A_{\pi\pi}(s) = \frac{2}{\sqrt{s}}(s - 4M_\pi^2)|F_\pi(s)|^2 \quad \mu_{\pi\pi}(s) = \frac{1}{32\pi^2}(s - 4M_\pi^2)^{\frac{1}{2}}$$

$$\rho_{\pi\pi}(s) = \frac{1}{48\pi^3}\left(1 - \frac{4M_\pi^2}{s}\right)^{\frac{3}{2}} |F_\pi(s)|^2$$

Finite-volume 2-pion contribution

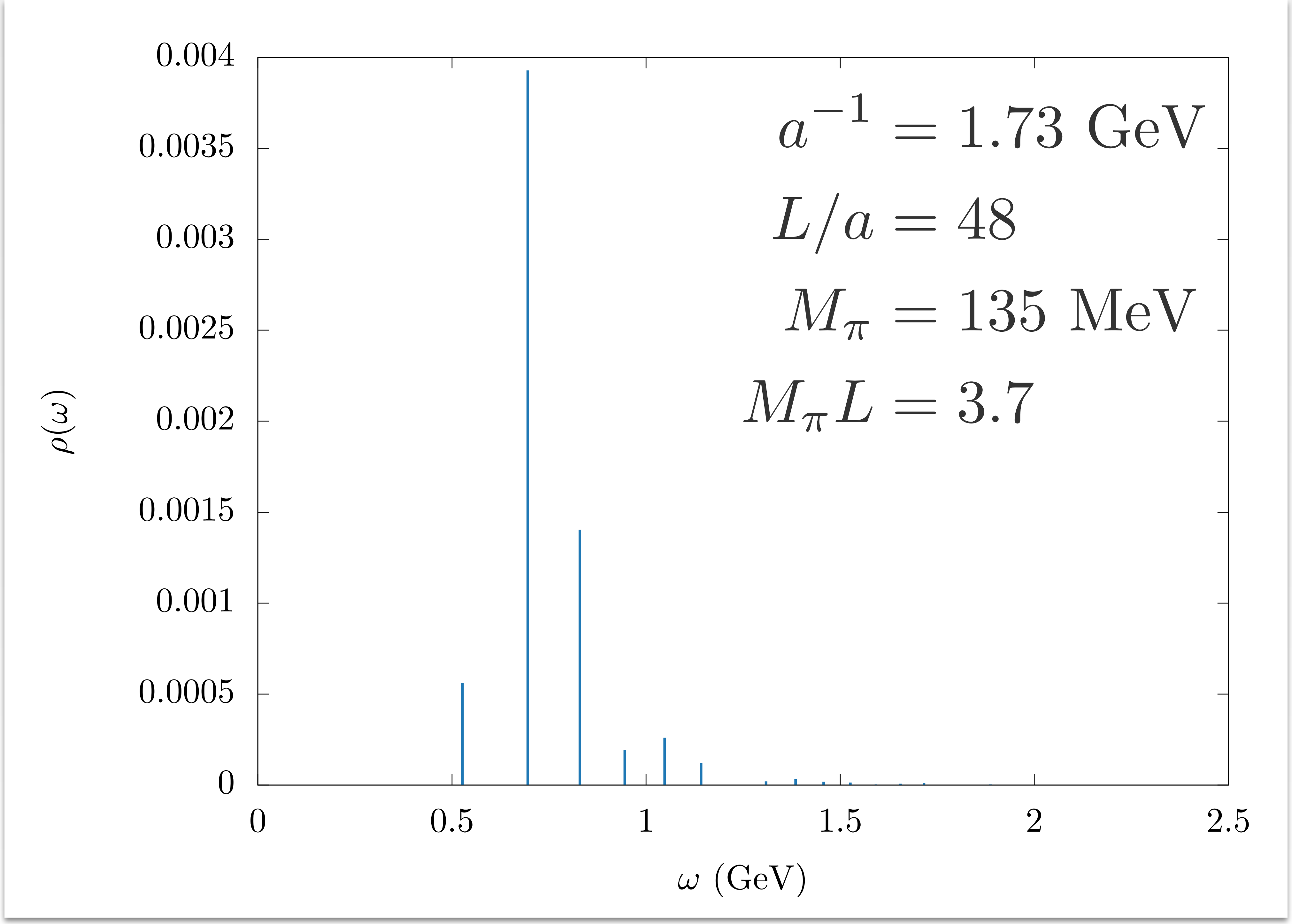
- Two assumptions:
  1) use free pions quantisation
  2) use infinite-volume amplitudes

Only accurate if $(M_\pi L)^{-3}$ **can be neglected**
(typically $M_\pi L \gtrsim 10$)

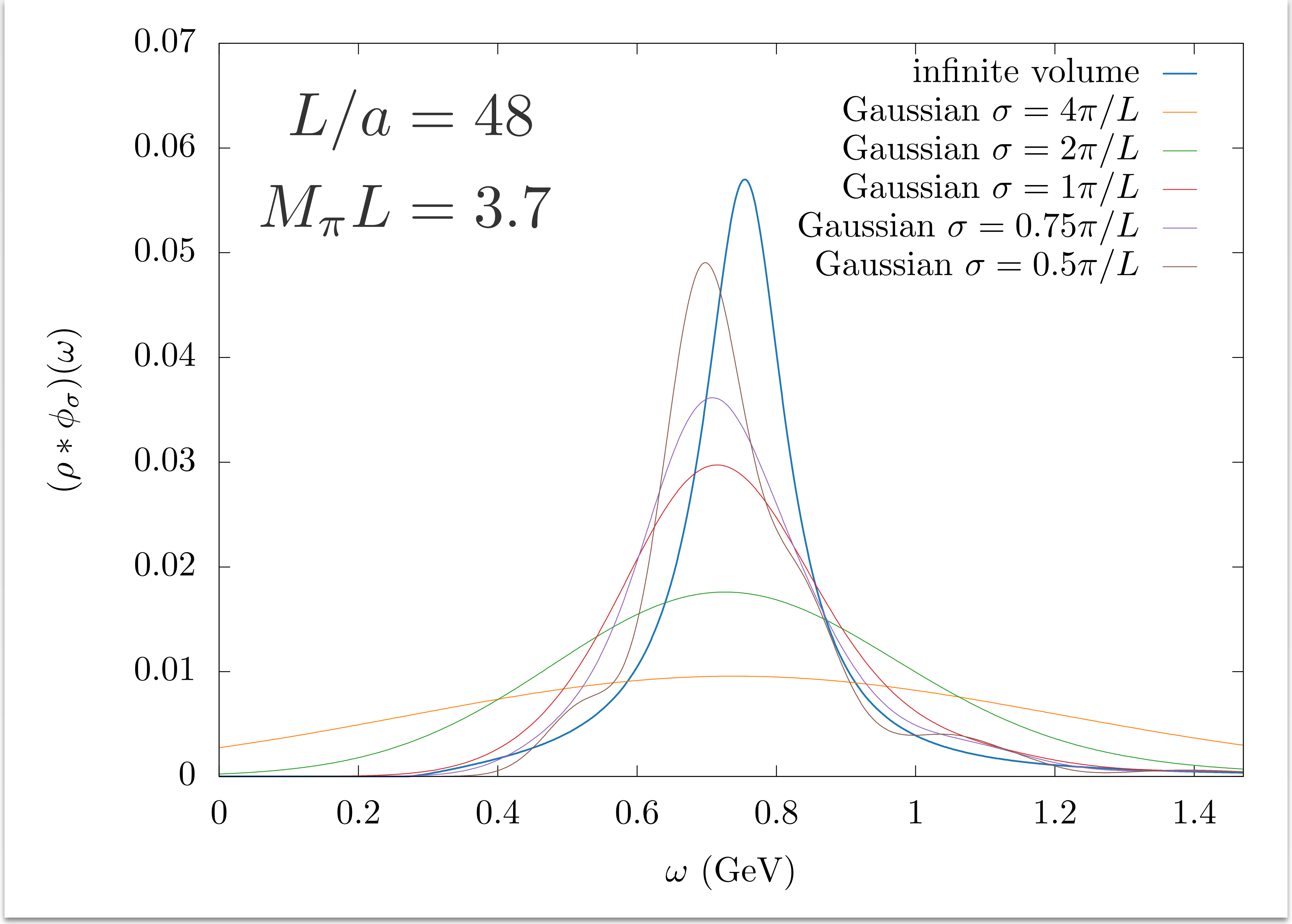
$$\mu_{\pi\pi,\mathrm{FV}}(s) = \frac{1}{L^3}\sum_{n=0}^{+\infty} r_3(n)\delta[s - 4\omega_\pi(\tfrac{4\pi^2}{L^2}n)^2]$$

$$\rho_{\pi\pi,\mathrm{FV}}(s) = \frac{2}{3L^3\sqrt{s}}\left(1 - \frac{4M_\pi^2}{s}\right)|F_\pi(s)|^2\sum_{n=0}^{+\infty} r_3(n)\delta[s - 4\omega_\pi(\tfrac{4\pi^2}{L^2}n)^2]$$

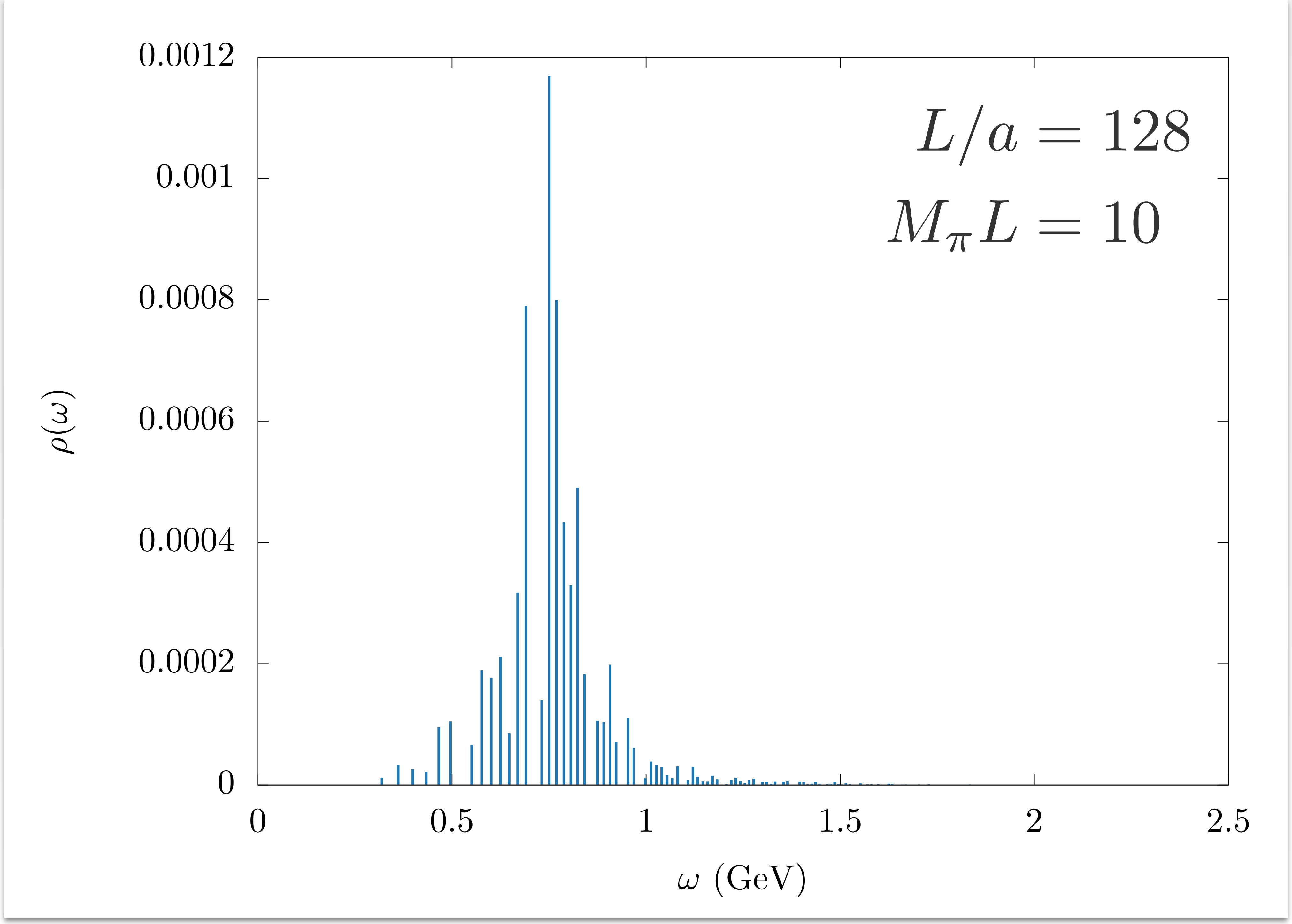
Finite-volume spectrum



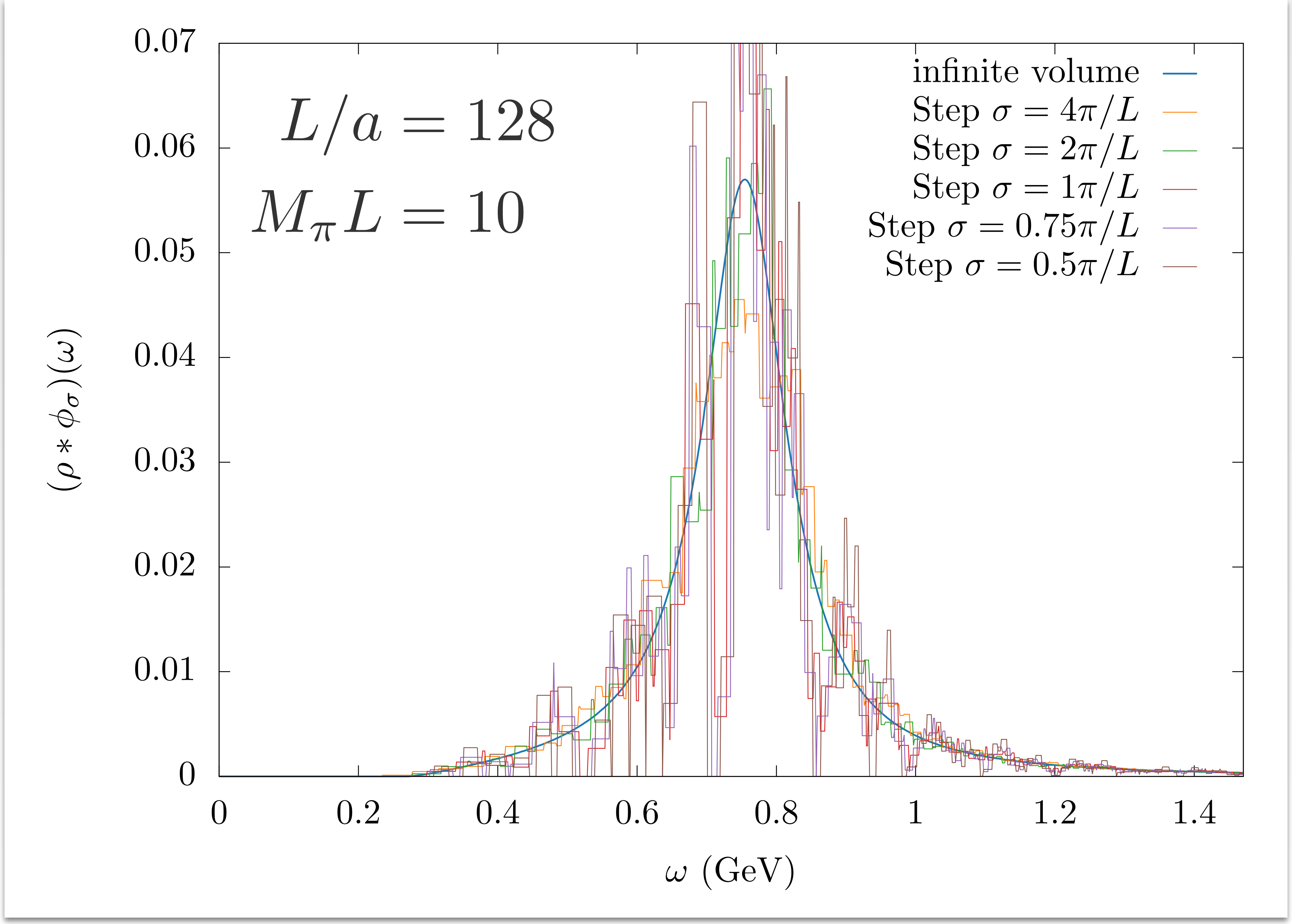
Finite-volume spectrum



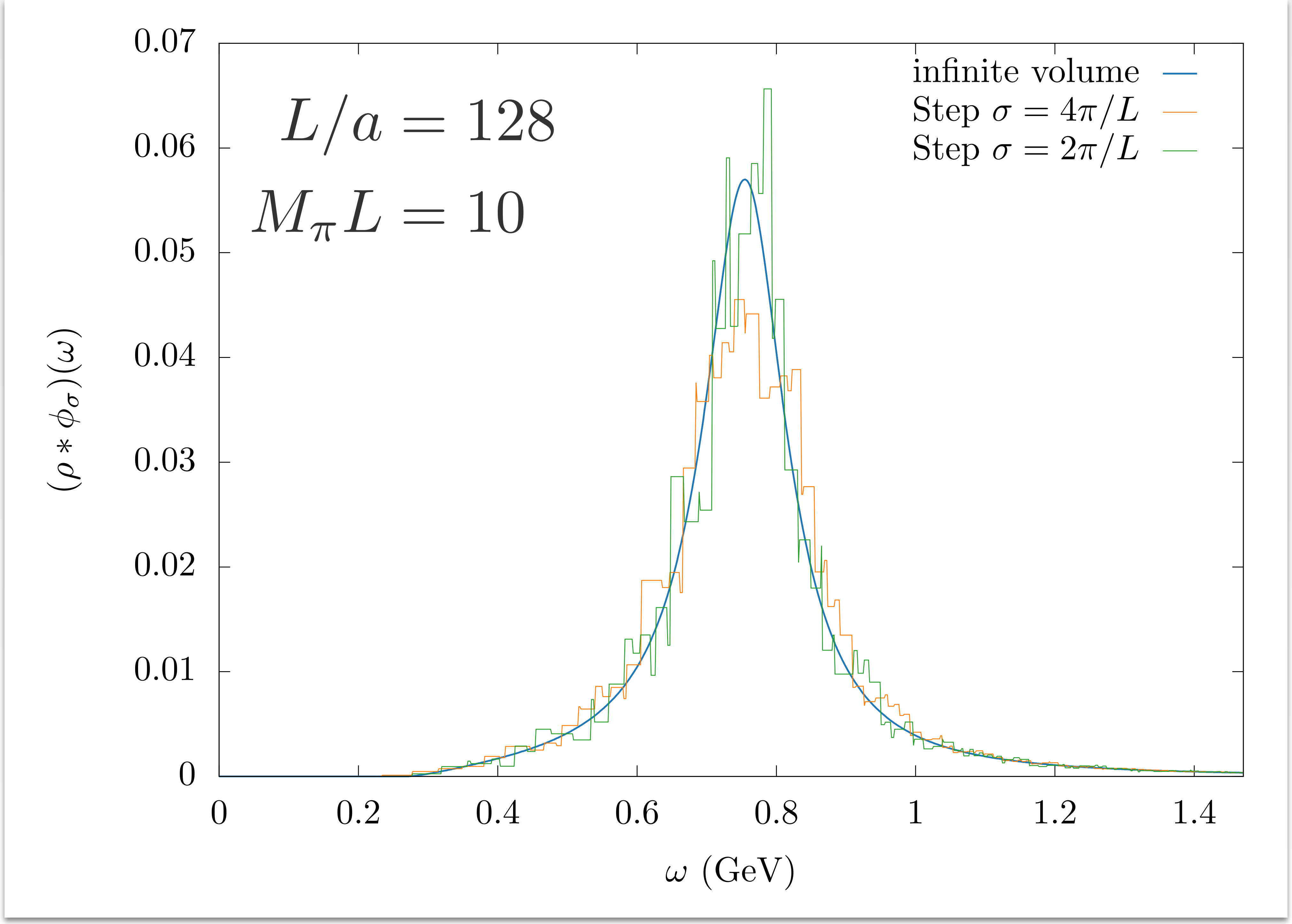
Finite-volume spectrum



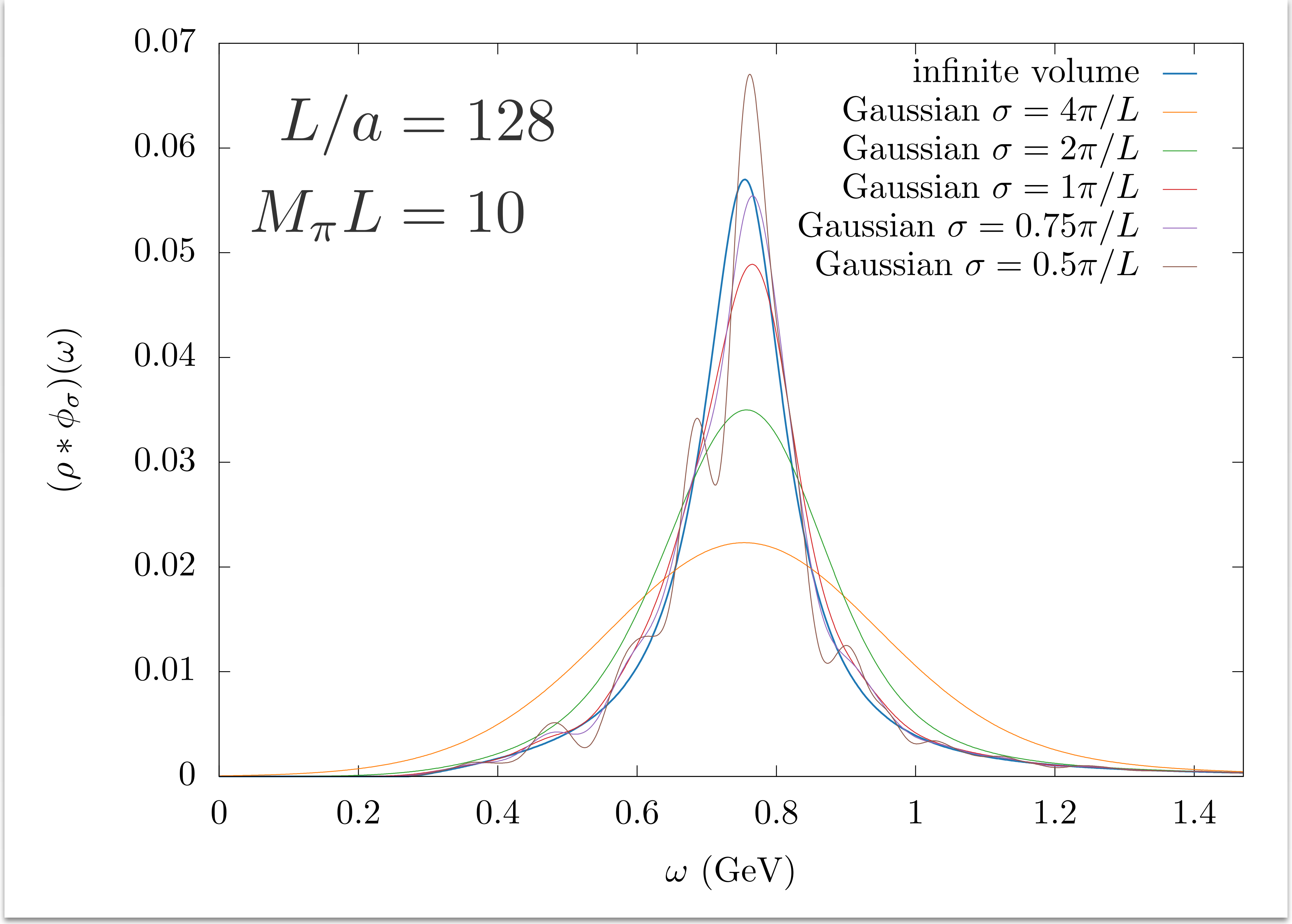
Finite-volume spectrum



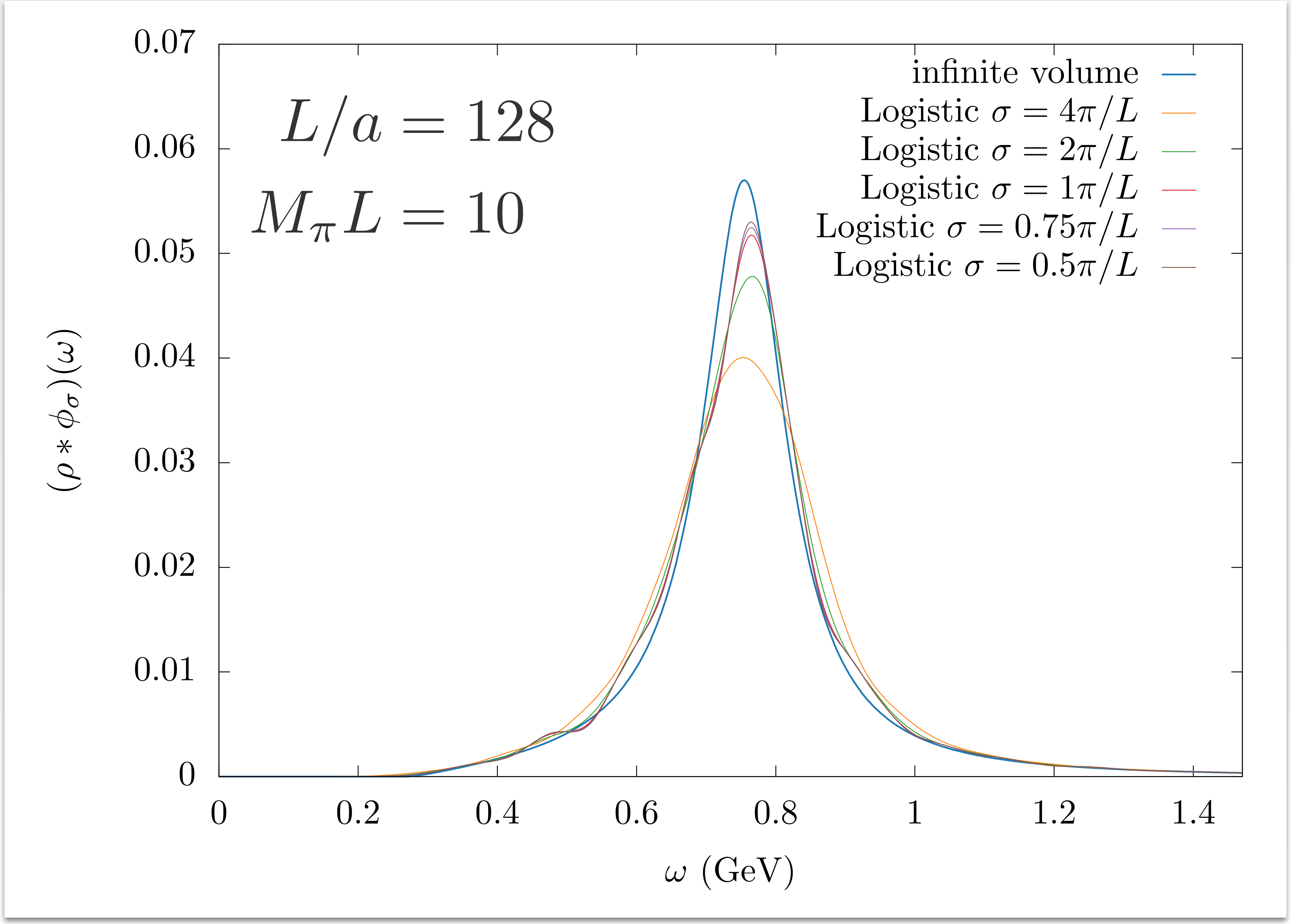
Finite-volume spectrum



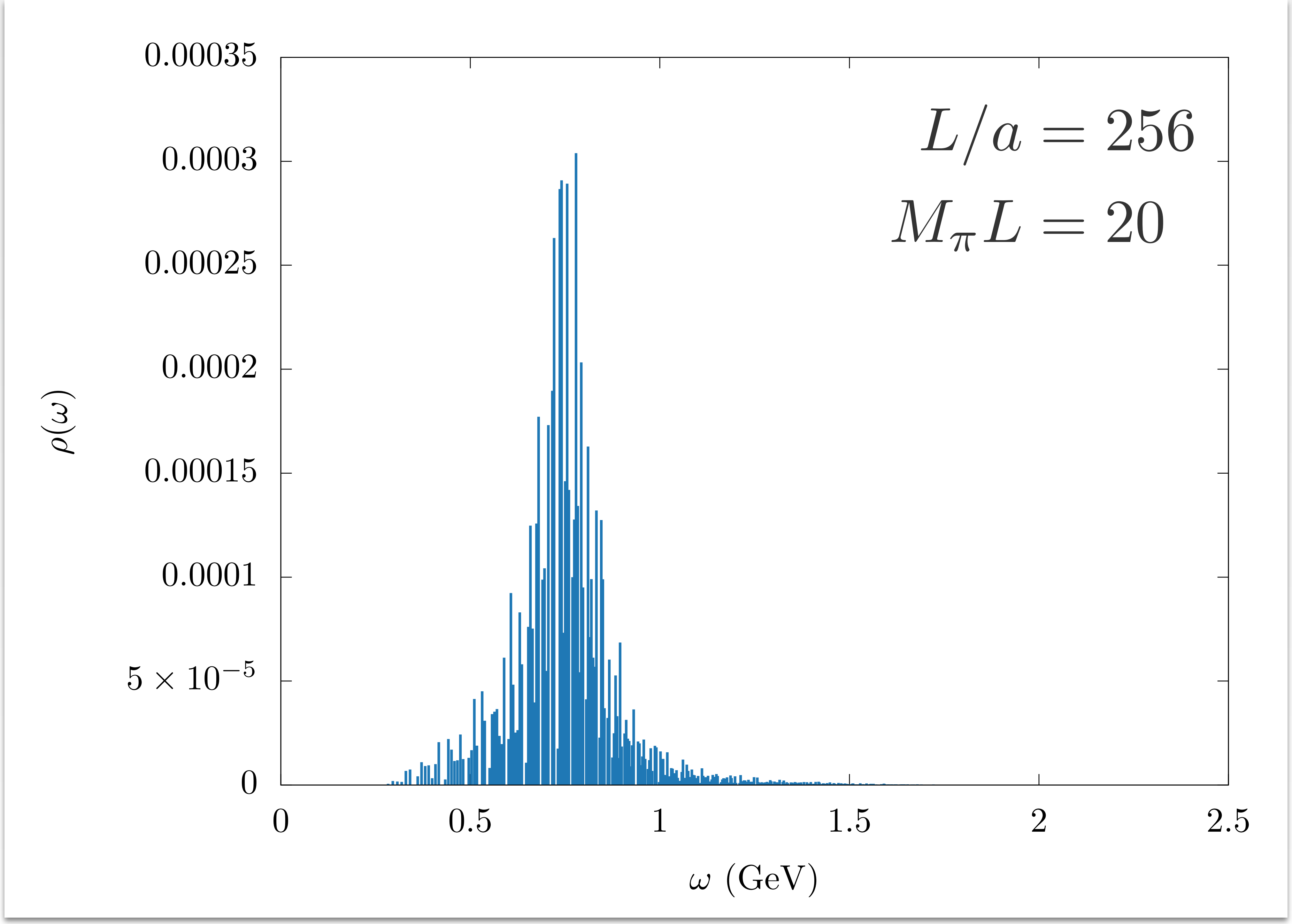
Finite-volume spectrum



Finite-volume spectrum



Finite-volume spectrum



Finite-volume spectrum