Extraction of lattice QCD spectral densities from an ensemble of trained machines

M. Buzzicotti,[1] **A. De Santis**[1,*] **and N. Tantalo**[1]

[1]*Dept. of Physics and INFN, University of Rome Tor Vergata, Via della Ricerca Scientifica 1, I-00133 Roma, Italy*

E-mail: alessandro.desantis@roma2.infn.it

In this talk we discuss a novel method, that we have presented in Ref. [1], to extract hadronic spectral densities from lattice correlators by using deep learning techniques. Hadronic spectral densities play a crucial role in the study of the phenomenology of strong-interacting particles and the problem of their extraction from Euclidean lattice correlators has already been approached in the literature by using machine learning techniques. A distinctive feature of our method is a model-independent training strategy that we implement by parametrizing the training sets over a functional space spanned by Chebyshev polynomials. The other distinctive feature is a reliable estimate of the systematic uncertainties that we obtain by introducing an ensemble of machines in order to study numerically the asymptotic limits of infinitely large networks and training sets. The method is validated on a very large set of random mock data and also in the case of lattice QCD data.

The 40th International Symposium on Lattice Field Theory (Lattice 2023)
July 31st - August 4th, 2023
Fermi National Accelerator Laboratory

---

[*]Speaker

1. Introduction

The problem of the extraction of hadronic spectral densities from Euclidean correlators, computed from numerical lattice QCD simulations, has attracted a lot of attention over the years (see Refs. [2–25]). Indeed, from the knowledge of these key objects it is possible to extract all the information needed to study the scattering of hadrons and, more generally, their interactions. The problem of the extraction of spectral densities from lattice correlators is equivalent to that of an inverse Laplace-transform operation and, when dealing with a discrete and finite set of noisy input data, it is numerically ill-posed. Moreover, in the case of lattice field theory correlators further complications arise as lattice simulations have necessarily to be performed on finite volumes where the spectral densities are badly-behaving distributions. A regularization procedure is therefore mandatory and this can be conveniently achieved by targeting the numerical calculation of *smeared* spectral densities.

In Ref. [1] we have recently presented a new method for the extraction of smeared spectral densities from lattice correlators that is based on a supervised deep-learning approach. In order to develop an approach that can be used to obtain trustworthy theoretical predictions we have to address the following two pivotal questions

1. is it possible to devise a *model independent* training strategy?

2. if such a strategy is found, is it then possible to quantify reliably, together with the statistical errors, also the unavoidable *systematic uncertainties*?

The importance of the first question can hardly be underestimated. Under the working assumption, supported by the so-called universal reconstruction theorems (see Refs. [26–28]), that a sufficiently large neural network can perform any task, limiting either the size of the network or the information to which it is exposed during the training process means, in fact, limiting its ability to solve the problem in full generality. Addressing the second question makes the difference between providing a possibly efficient but qualitative solution to the problem and providing a scientific numerical tool to be used in order to derive theoretical predictions for phenomenological analyses.

2. Outline of the strategy and numerical setup

The method that we have proposed in Ref. [1] (see FIG. 1) is built on two pillars

1. the introduction of a *functional-basis* to parametrize the correlators and the smeared spectral densities of the training sets in a model independent way;

2. the introduction of the *ensemble of machines* to estimate the systematic errors.

In order to validate the method we consider both mock and real lattice QCD data and choose the algorithmic parameters in order to be able to extract the hadronic spectral density from the lattice QCD correlator C^{latt}, associated with the so-called R-ratio, calculated on the B64 ensemble generated by the ETMC (see TABLE I in Ref. [29] and Ref. [30, 31]). We then generate the needed mock data by starting from a model unsmeared spectral density $\rho(E)$ and by computing the associated correlator according to

$$C(t) = \int_{E_0}^{\infty} d\omega \frac{\omega^2}{12\pi^2} \left[e^{-t\omega} + e^{-(T-t)\omega} \right] \rho(\omega) , \tag{1}$$

with $T = 64a$ and $a = 0.07957$ fm, and the associated smeared spectral density according to

$$\hat{\rho}_\sigma(E) = \int_{E_0}^{\infty} d\omega \, K_\sigma(E, \omega) \rho(\omega) . \tag{2}$$
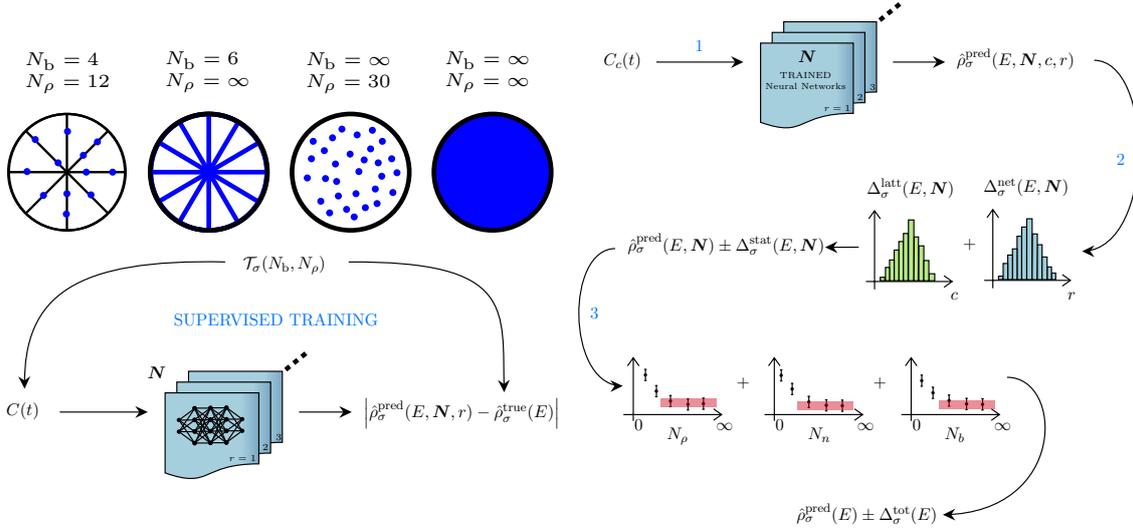
Figure 1: *Top-left*: By introducing a discrete functional-basis, with elements $B_n(E)$, that is dense in the space of square-integrable functions $f(E)$ in the interval $[E_0, \infty)$ with $E_0 > 0$, any such function can exactly be represented as $f(E) = \sum_{n=0}^{\infty} c_n B_n(E)$. With an infinite number of basis functions ($N_b = \infty$) and by randomly selecting an infinite number ($N_\rho = \infty$) of coefficient vectors $\boldsymbol{c} = (c_0, \cdots, c_{N_b})$, one can get any possible spectral density. This is the situation represented by the filled blue disk. If the number of basis functions N_b and the number of randomly extracted spectral densities N_ρ are both finite one has a training set that is finite and that also depends on N_b. This is the situation represented in the first disk on the left. The other two disks schematically represent the situations in which either N_b or N_ρ is infinite. *Bottom-left*: We generate several training sets $\mathcal{T}_\sigma(N_b, N_\rho)$ built by considering randomly chosen spectral densities. These are obtained by choosing N_ρ random coefficients vectors with N_b entries. For each spectral density $\rho(E)$ we build the associated correlator $C(t)$ and smeared spectral density $\hat{\rho}_\sigma(E)$, where σ is the smearing parameter. We then distort the correlator $C(t)$, by using the information provided by the statistical variance of the lattice correlator, and obtain the input-output pair $(C_{\text{noisy}}(t), \hat{\rho}_\sigma(E))$ that we add to $\mathcal{T}_\sigma(N_b, N_\rho)$. We then implement different neural networks with N_n neurons and at fixed $\boldsymbol{N} = (N_n, N_b, N_\rho)$ we introduce an *ensemble of machines* with N_r replicas. Each machine $r = 1, \cdots, N_r$ belonging to the ensemble has the same architecture and, before the training, differs from the other replicas for the initialization parameters. All the replicas are then trained over $\mathcal{T}_\sigma(N_b, N_\rho)$ and, at the end of the training process, each replica will give a different answer depending upon $\boldsymbol{N}$. *Right*: Flowchart illustrating the procedure that we use to extract the final result. Here $C(t)$ represents the input lattice correlator that, coming from a Monte Carlo simulation, is affected by statistical noise. We call $C_c(t)$ the c-th bootstrap sample (or jackknife bin) of the lattice correlator with $c = 1, \cdots, N_c$. In the first step, $C_c(t)$ is fed to all the trained neural networks belonging to the ensemble at fixed N and the corresponding answers $\hat{\rho}_\sigma^{\text{pred}}(E, \boldsymbol{N}, c, r)$ are collected. In the second step, by combining in quadrature the widths of the distributions of the answers as a function of the index c ($\Delta_\sigma^{\text{latt}}(E, \boldsymbol{N})$) and of the index r ($\Delta_\sigma^{\text{net}}(E, \boldsymbol{N})$) we estimate the error $\Delta_\sigma^{\text{stat}}(E, \boldsymbol{N})$) at fixed N. At the end of this step we are left with a collection of results $\hat{\rho}_\sigma^{\text{pred}}(E, \boldsymbol{N}) \pm \Delta_\sigma^{\text{stat}}(E, \boldsymbol{N})$. In the third and last step, the limits $\boldsymbol{N} \mapsto \infty$ are studied numerically and an unbiased estimate of $\hat{\rho}_\sigma^{\text{pred}}(E)$ and of its error $\Delta_\sigma^{\text{tot}}(E)$, with the latter also taking into account the unavoidable systematics associated with these limits, is finally obtained.

We focus on a Gaussian smearing kernel, $K_\sigma(E, \omega) = e^{-\frac{(E-\omega)^2}{2\sigma^2}} / \sqrt{2\pi\sigma^2}$, but any other kernel can easily be implemented within the proposed strategy. Among the many computational paradigms available within the machine-learning framework, we opt for the most direct one and represent both the correlator and the smeared spectral density as finite dimensional vectors, that we use respectively as input and output of our neural networks. More precisely, the dimension of the input correlator vector is fixed to $N_T = 64$, coinciding with the available number of Euclidean lattice times in the case of $\boldsymbol{C}^{\text{latt}}$. The inputs of the neural networks are thus the 64-components vectors $\boldsymbol{C} = \{C(a), C(2a), \cdots, C(64a)\}$. The output vectors are instead given

by $\hat{\boldsymbol{\rho}}_\sigma = \{\hat{\rho}_\sigma(E_{\min}), \cdots, \hat{\rho}_\sigma(E_{\max})\}$. As in Ref. [29], we choose to measure energies in units of the muon mass, $m_\mu = 0.10566$ GeV, and set $E_{\min} = m_\mu$ and $E_{\max} = 24 m_\mu$. The interval $[E_{\min}, E_{\max}]$ is discretized in steps of size $m_\mu/2$. With these choices our output smeared spectral densities are the vectors $\hat{\boldsymbol{\rho}}_\sigma$ with $N_E = 47$ elements corresponding to energies ranging from about 100 MeV to 2.5 GeV.

We treat σ, the width of the smearing Gaussian, as a fixed parameter by including in the corresponding training sets only spectral functions that are smeared with the chosen value of σ. We consider two different values, $\sigma = 0.44$ GeV and $\sigma = 0.63$ GeV, that correspond respectively to the smallest and largest values used in Ref. [29].

3. Model independent training set

In the supervised deep-learning framework a neural network is trained over a training set which is representative of the problem that has to be solved. In our case the inputs to each neural network are the correlators $\boldsymbol{C}$ and the target outputs are the associated smeared spectral densities $\hat{\boldsymbol{\rho}}_\sigma$. As discussed in the introduction, our main goal is that of devising a model-independent training strategy. To this end, the challenge is that of building a training set which contains enough variability so that, once trained, the network is able to provide the answer, correct within the quoted errors, for any possible input correlator. In order to face this challenge we use the algorithmic strategy described in FIG. 1. In our strategy

- the fact that the network cannot be infinitely large is parametrized by the fact that N_n (the number of neurons) is finite;

- the fact that during the training a network cannot be exposed to any possible spectral density is parametrized by the fact that N_b (the number of basis functions) and N_ρ (the number of spectral densities to which a network is exposed during the training) are finite;

- the fact that at fixed

$$\boldsymbol{N} = (N_n, N_b, N_\rho) \tag{3}$$

the answer of a network cannot be exact, and therefore has to be associated with an error, is taken into account by introducing an ensemble of machines, with N_r replicas, and by estimating this error by studying the distribution of the different N_r answers in the $N_r \mapsto \infty$ limit;

- once the network (and statistical) errors at fixed $\boldsymbol{N}$ are given, we can study numerically the $\boldsymbol{N} \mapsto \infty$ limits and also quantify, reliably, the additional systematic errors associated with these unavoidable extrapolations.

For our numerical study we use the Chebyshev polynomials of the first kind as basis functions and generate the unsmeared spectral densities that we use to build our training sets according to

$$\rho(E; N_b) = \theta(E - E_0) \sum_{n=0}^{N_b} c_n \left[T_n(x(E)) - T_n(x(E_0)) \right] , \tag{4}$$

where $x(E) = 1 - 2e^{-E}$ maps the energy domain $E \in [E_0, \infty)$ into $[-1, +1]$ in which the Chebyshev polynomials are naturally defined. The subtraction of the constant term $T_n(x(E_0))$ is introduced in order to include a threshold E_0 below which the spectral density is vanishing. Each $\rho(E; N_b)$ that we use to populate the training sets is obtained by choosing E_0 randomly in the interval $[0.2, 1.3]$ GeV and by inserting in Eq. (4) the coefficients

$$c_0 = r_0 ; \qquad c_n = \frac{r_n}{n^{1+\varepsilon}} , \quad n > 0 , \tag{5}$$

4

Type	Maps	Size	Kernel size	Stride	Activation
Input		64			
Conv1D	2	32x2	3	2	LeakyReLu
Conv1D	4	16x4	3	2	LeakyReLu
Conv1D	8	8x8	3	2	LeakyReLu
Flatten		384			
Fully conn.		256			LeakyReLu
Fully conn.		256			LeakyReLu
Output		47			
Parameters	94651				

Table 1: *arcS*: the smallest neural network architecture used in Ref. [1]. The architecture is of the type feed-forward and the structure can be read from top to bottom of the table. It consist of three 1D convolutional layers with an increasing number of maps followed by two fully connected layers. The column denoted by 'Size' reports the shape of the signal produced by the corresponding layer. The stride of the filters is set to 2 in such a way that the dimension of the signal is halved at each 1D convolutional layer thus favouring the neural network to learn a more abstract, and possibly more effective, representation of the input data. The output layer has no activation function in order not to limit the output range.

where the r_n's are N_b uniformly distributed random numbers in the interval $[-1, 1]$ and ε is a non-negative parameter that we set to 10^{-7}. Notice that with this choice of the coefficients c_n the Chebyshev series of Eq. (4) is convergent in the $N_b \mapsto \infty$ limit and that the resulting spectral densities can be negative. Once E_0 and the coefficients c_n that define $\rho(E; N_b)$ are given, the correlator C and the smeared spectral density $\hat{\rho}_\sigma$ associated with $\rho(E; N_b)$ can be calculated by using Eq. (1) and Eq. (2). We generate N_ρ pairs $(C, \hat{\rho}_\sigma)$ and populate the training sets at fixed N_b and σ, $\mathcal{T}_\sigma(N_b, N_\rho)$, with the pairs $(C_{\text{noisy}}, \hat{\rho}_\sigma)$ where C_{noisy} are random correlators extracted from the multivariate Gaussian distribution

$$\mathbb{G}\left[C, \left(\frac{C(a)}{C_{\text{latt}}(a)} \right)^2 \hat{\Sigma}_{\text{latt}} \right], \tag{6}$$

with center C and covariance $\hat{\Sigma}_{\text{latt}}$, the covariance matrix of the lattice correlator C^{latt} mentioned above.

In order to be able to perform a numerical study of the limits $N \mapsto \infty$, we generate with the procedure just described several training sets, corresponding to $N_b = \{16, 32, 128, 512\}$ and $N_\rho = \{50, 100, 200, 400, 800\} \times 10^3$.

4. Training setup and ensemble of machines

From the machine-learning point of view, we need to implement neural networks to solve a $\mathbb{R}^{N_T} \mapsto \mathbb{R}^{N_E}$ regression problem with $N_T = 64$ and $N_E = 47$. For our analysis, we use feed-forward convolutional neural networks based on the LeNet architecture introduced in Ref. [32]. We study in details the dependence of the output of the neural networks upon their size N_n and, to this end, we implement three architectures that we call *arcS*, *arcM* and *arcL*. The architecture *arcS* is described in full details in TABLE 1. *arcM* and *arcL* differ only for an increasing number of maps in the convolutional layers for a total number of trainable parameters corresponding to 180871 and 371311.

We train each machine with N_n neurons over the training set $\mathcal{T}_\sigma(N_b, N_\rho)$ by minimizing as loss function the Mean Absolute Error (MAE)

$$\ell(\boldsymbol{w}) = \frac{1}{N_\rho} \sum_{i=1}^{N_\rho} \left| \hat{\boldsymbol{\rho}}_\sigma^{\text{pred},i}(\boldsymbol{w}) - \hat{\boldsymbol{\rho}}_\sigma^i \right|, \tag{7}$$

where $\hat{\boldsymbol{\rho}}_\sigma^{\text{pred},i}(\boldsymbol{w})$ is the output of the neural network in correspondence of the input correlator C_{noisy}^i. We implement a Mini-Batch Gradient Descent algorithm, with Batch Size (BS) set to 32, by using the Adam
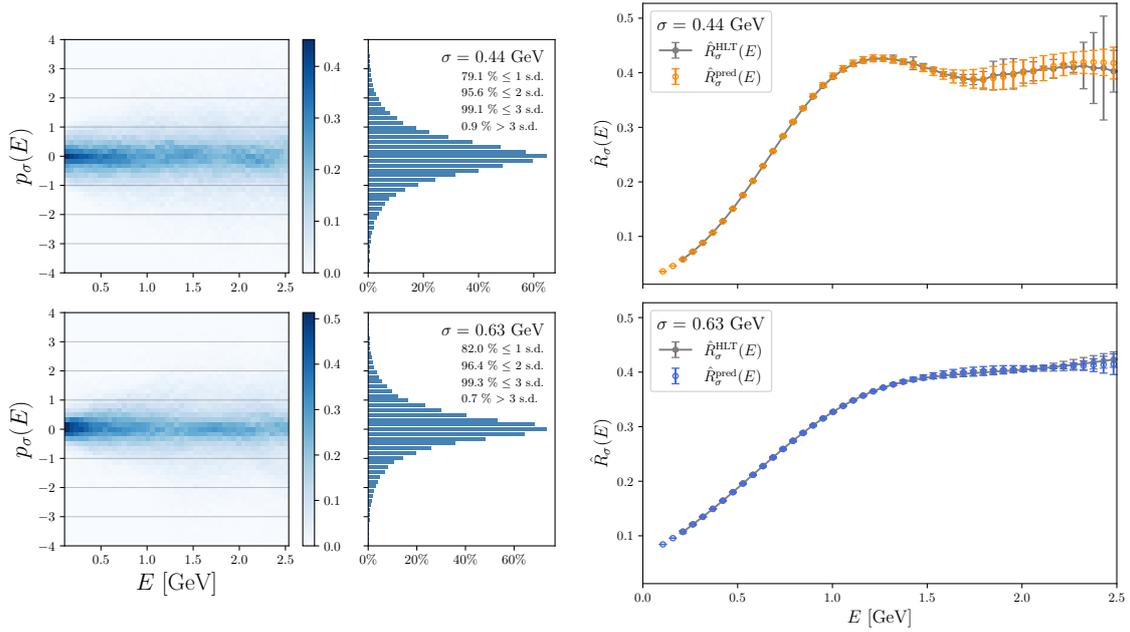
Figure 2: *Left*: Normalized density distribution of the significance defined in Eq. (9) as a function of the energy. The distribution is calculated over 2000 validation samples generated according to Eq. (8). Aside, the cumulative histogram of $p_\sigma(E)$ and related fractions of values that fall within 1, 2 or 3 standard deviations. *Right*: The results obtained with our ensembles of neural networks in the case of a real lattice QCD correlator are compared with those obtained with the HLT method (grey, Ref. [9, 29]) for the two values of σ.

optimizer [33] combined with a decaying learning rate. We adopt the standard early stop criterion to terminate the training by monitoring the validation loss (20% of N_ρ). As already stressed, at fixed N the answer of a neural network cannot be exact. In order to be able to study numerically the $N \mapsto \infty$ limits, the error associated with the limited abilities of the networks at finite N *has* to be quantified. To do this we introduce the *ensemble of machines* by considering, at fixed N, $N_r = 20$ machines with the same architecture. Each machine of the ensemble is trained by using a training set $\mathcal{T}_\sigma(N_b, N_\rho)$ obtained by starting from the same unsmeared spectral densities but with different noisy input correlators C_{noisy}.

5. Validation and application

Once the ensemble of trained machines has been built, the procedure sketched in FIG. 1 and illustrated in much more detail in Ref. [1], is used to predict the smeared spectral density for any new correlator. In order to perform a stringent validation test of the proposed method we consider unsmeared spectral densities not belonging to the space spanned by the Chebyshev basis but generated according to

$$\rho(E) = \sum_{n=1}^{N_{\text{peaks}}} c_n \delta(E - E_n). \tag{8}$$

By plugging Eq. (8) into Eq. (1) and Eq. (2) we calculate the correlator and smeared spectral density associated to each unsmeared $\rho(E)$. We generate 2000 such unsmeared spectral densities by randomly generating c_n and E_n. The trains of isolated Dirac-delta peaks are representative of unsmeared spectral densities that might arise in the study of finite volume lattice correlators and therefore represent a particularly challenging

benchmarking set to validate the procedure. The left panel of FIG 2 shows the pull variable

$$p_\sigma(E) = \frac{\hat{\rho}_\sigma^{\text{pred}}(E) - \hat{\rho}_\sigma^{\text{true}}(E)}{\Delta_\sigma^{\text{tot}}(E)}, \tag{9}$$

which is a quantitative estimate for the deviation of the predicted result from the true one within the total quoted error. $\Delta_\sigma^{\text{tot}}(E)$ includes both statistical and systematic errors combined in quadrature. As it can be seen, the final results are consistent with the true ones at the level of one standard deviation in about 80% of the cases and below three standard deviations in more than 99%. This is a reassuring evidence of the ability of our ensembles of neural networks to generalize very efficiently outside the training set and, more importantly, on the robustness of the procedure that we use to estimate the errors.

As application to real lattice QCD, we consider a lattice correlator, measured by the ETMC, that has already been used in Ref. [29] to extract the so-called strange-strange connected contribution to the smeared R-ratio by using the HLT method of Ref. [9]. The right panel of FIG. 2 shows the comparison of $\hat{R}_\sigma^{\text{pred}}(E)$ and $\hat{R}_\sigma^{\text{HLT}}(E)$. The agreement between the two determinations is remarkably good over all the energy range. We stress that neither the new method proposed in Ref. [1] nor the HLT algorithm assume any prior information to predict the spectral density. The results presented in FIG. 2 are then unbiased determinations from first principles obtained from two completely different methods.

6. Conclusions

In Ref. [1] we have proposed a new method to extract smeared hadronic spectral densities from lattice correlators. The method is based on supervised deep-learning and the distinctive features are the implementation of a model-independent training strategy and a reliable estimation of the systematics achieved by building an *ensemble* of trained machines. We validate the method on a large set of mock data by measuring the failure rate of the ensemble of machines at the end of the procedure. In addition we consider a true lattice QCD correlator obtained from a simulation performed by the ETM Collaboration. We extract the strange-strange connected contribution to the smeared R-ratio and compare the predictions obtained by using our ensembles of trained machines with the ones previously obtained by using the HLT method [9, 29]. The two determinations, obtained by using the two totally unrelated methods, show a remarkably good agreement and a similar precision.

The proposed method may be admittedly computationally demanding. However, on the basis of our experience, we are firmly convinced that a careful study of the different sources of systematic uncertainties is mandatory when dealing with machine-learning if the goal is to compare theoretical predictions with experiments. In fact, the computational cost of the proposed method is the price that we have to pay for the reliability of the results. On the other end, our strategy, devised in the specific case of the extraction of spectral densities, can be fruitfully generalized to other applications and research fields.

References

[1] M. Buzzicotti, A.D. Santis and N. Tantalo, *Teaching to extract spectral densities from lattice correlators to a broad audience of learning-machines*, 2023.

[2] J.C.A. Barata and K. Fredenhagen, *Particle scattering in Euclidean lattice field theories*, *Commun. Math. Phys.* **138** (1991) 507.

[3] M. Jarrell and J.E. Gubernatis, *Bayesian inference and the analytic continuation of imaginary-time quantum Monte Carlo data*, *Phys. Rept.* **269** (1996) 133.

[4] Y. Nakahara, M. Asakawa and T. Hatsuda, *Hadronic spectral functions in lattice QCD*, *Phys. Rev. D* **60** (1999) 091503 [hep-lat/9905034].

[5] M. Asakawa, T. Hatsuda and Y. Nakahara, *Maximum entropy analysis of the spectral functions in lattice QCD*, *Prog. Part. Nucl. Phys.* **46** (2001) 459 [hep-lat/0011040].

[6] Y. Burnier and A. Rothkopf, *Bayesian Approach to Spectral Function Reconstruction for Euclidean Quantum Field Theories*, *Phys. Rev. Lett.* **111** (2013) 182003 [1307.6106].

[7] M.T. Hansen, H.B. Meyer and D. Robaina, *From deep inelastic scattering to heavy-flavor semileptonic decays: Total rates into multihadron final states from lattice QCD*, *Phys. Rev. D* **96** (2017) 094513 [1704.08993].

[8] J. Bulava and M.T. Hansen, *Scattering amplitudes from finite-volume spectral functions*, *Phys. Rev. D* **100** (2019) 034521 [1903.11735].

[9] M. Hansen, A. Lupo and N. Tantalo, *Extraction of spectral densities from lattice correlators*, *Phys. Rev. D* **99** (2019) 094508 [1903.06476].

[10] L. Kades, J.M. Pawlowski, A. Rothkopf, M. Scherzer, J.M. Urban, S.J. Wetzel et al., *Spectral Reconstruction with Deep Neural Networks*, *Phys. Rev. D* **102** (2020) 096001 [1905.04305].

[11] G. Bailas, S. Hashimoto and T. Ishikawa, *Reconstruction of smeared spectral function from Euclidean correlation functions*, *PTEP* **2020** (2020) 043B07 [2001.11779].

[12] P. Gambino and S. Hashimoto, *Inclusive Semileptonic Decays from Lattice QCD*, *Phys. Rev. Lett.* **125** (2020) 032001 [2005.13730].

[13] M. Bruno and M.T. Hansen, *Variations on the Maiani-Testa approach and the inverse problem*, *JHEP* **06** (2021) 043 [2012.11488].

[14] J. Horak, J.M. Pawlowski, J. Rodríguez-Quintero, J. Turnwald, J.M. Urban, N. Wink et al., *Reconstructing QCD spectral functions with Gaussian processes*, *Phys. Rev. D* **105** (2022) 036014 [2107.13464].

[15] J. Bulava, M.T. Hansen, M.W. Hansen, A. Patella and N. Tantalo, *Inclusive rates from smeared spectral densities in the two-dimensional O(3) non-linear σ-model*, *JHEP* **07** (2022) 034 [2111.12774].

[16] S.Y. Chen, H.T. Ding, F.Y. Liu, G. Papp and C.B. Yang, *Machine learning spectral functions in lattice QCD*, 2110.13521.

[17] L. Wang, S. Shi and K. Zhou, *Reconstructing spectral functions via automatic differentiation*, *Phys. Rev. D* **106** (2022) L051502 [2111.14760].

[18] T. Lechien and D. Dudal, *Neural network approach to reconstructing spectral functions and complex poles of confined particles*, *SciPost Phys.* **13** (2022) 097 [2203.03293].

[19] T. Bergamaschi, W.I. Jay and P.R. Oare, *Hadronic Structure, Conformal Maps, and Analytic Continuation*, 2305.16190.

[20] A. Rothkopf, *Inverse problems, real-time dynamics and lattice simulations*, *EPJ Web Conf.* **274** (2022) 01004 [2211.10680].

[21] J. Bulava, *The spectral reconstruction of inclusive rates*, *PoS* **LATTICE2022** (2023) 231 [2301.04072].

[22] J. Karpie, K. Orginos, A. Rothkopf and S. Zafeiropoulos, *Reconstructing parton distribution functions from Ioffe time data: from Bayesian methods to Neural Networks*, *JHEP* **04** (2019) 057 [1901.05408].

[23] M. Zhou, F. Gao, J. Chao, Y.-X. Liu and H. Song, *Application of radial basis functions neutral networks in spectral functions*, *Phys. Rev. D* **104** (2021) 076011 [2106.08168].

[24] D. Boyda et al., *Applications of Machine Learning to Lattice Quantum Field Theory*, in *2022 Snowmass Summer Study*, 2, 2022 [2202.05838].

[25] S. Shi, L. Wang and K. Zhou, *Rethinking the ill-posedness of the spectral function reconstruction — Why is it fundamentally hard and how Artificial Neural Networks can help*, *Comput. Phys. Commun.* **282** (2023) 108547 [2201.02564].

[26] K. Hornik, M. Stinchcombe and H. White, *Multilayer feedforward networks are universal approximators*, *Neural Networks* **2** (1989) 359.

[27] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press (2016).

[28] G. Cybenko, *Approximation by superpositions of a sigmoidal function*, *Mathematics of control, signals and systems* **2** (1989) 303.

[29] EXTENDED TWISTED MASS COLLABORATION (ETMC) collaboration, *Probing the Energy-Smeared R Ratio Using Lattice QCD*, *Phys. Rev. Lett.* **130** (2023) 241901 [2212.08467].

[30] D. Bernecker and H.B. Meyer, *Vector Correlators in Lattice QCD: Methods and applications*, *Eur. Phys. J. A* **47** (2011) 148 [1107.4388].

[31] EXTENDED TWISTED MASS COLLABORATION collaboration, *Lattice calculation of the short and intermediate time-distance hadronic vacuum polarization contributions to the muon magnetic moment using twisted-mass fermions*, *Phys. Rev. D* **107** (2023) 074506.

[32] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, *Gradient-based learning applied to document recognition*, *Proceedings of the IEEE* **86** (1998) 2278.

[33] D.P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, 2017.