# Applying machine learning methods to prediction problems of lattice observables

Maxim N. Chernodub, N V Gerasimeniuk, V A Goy, D L Boyda, S D Liubimov, A V Molochkov

▶ **To cite this version:**

Maxim N. Chernodub, N V Gerasimeniuk, V A Goy, D L Boyda, S D Liubimov, et al.. Applying machine learning methods to prediction problems of lattice observables. XXXIII International (ONLINE) Workshop on High Energy Physics "Hard Problems of Hadron Physics: Non-Perturbative QCD & Related Quests", Nov 2021, Protvino, Russia. pp.10005, 10.1051/epjconf/202225810005 . hal-03472367

**HAL Id: hal-03472367**
**https://hal.science/hal-03472367**

Submitted on 9 Dec 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Applying machine learning methods to prediction problems of lattice observables

N.V. Gerasimeniuk[1,⋆], M.N. Chernodub [1,2], V.A. Goy [1,2], D.L. Boyda [3], S.D. Liubimov [1] and A.V. Molochkov[1]

**1** Pacific Quantum Center, Far Eastern Federal University, 690950, Vladivostok, Russia
**2** Institute Denis Poisson CNRS/UMR 7013, Universite' de Tours, 37200 France
**3** Argonne Leadership Computing Facility, Argonne National Laboratory, Lemont IL-60439, USA
* gerasimenyuk_nv@dvfu.ru

December 9, 2021

*XXXIII International (ONLINE) Workshop on High Energy Physics*
*"Hard Problems of Hadron Physics: Non-Perturbative QCD & Related Quests"*
*November 8-12, 2021*
doi:10.21468/SciPostPhysProc.?

## Abstract

**We discuss the prediction of critical behavior of lattice observables in SU(2) and SU(3) gauge theories. We show that feed-forward neural network, trained on the lattice configurations of gauge fields as input data, finds correlations with the target observable, which is also true in the critical region where the neural network has not been trained. We have verified that the neural network constructs a gauge-invariant function and this property does not change over the entire range of the parameter space.**

## 1 Introduction

The low-energy physics of strong interactions cannot be addressed analytically because of the strong coupling, which makes perturbative approaches, usually used in the high-energy region, unreliable. For that reason, all existing calculations in the non-perturbative domain are based on effective low-energy models or sophisticated numerical methods involving the Monte Carlo (MC) algorithms. The MC simulations are reasonably reliable to address various thermodynamic properties of quantum chromodynamics (QCD) from the first principles. In the physically relevant domain of parameters, the numerical simulations are very computationally expensive and thus require powerful supercomputers.

In addition, the Monte Carlo methods cannot be applied to the interesting region of the QCD phase diagram at finite baryon chemical potential, thus calling for the development of alternative approaches aimed, in particular, at the investigation of the quark-gluon plasma at finite density. A promising way to extend the MC techniques involves the machine learning (ML) methods used nowadays to address various problems in physics [1, 2].

Our work discusses an example of the potentially helpful combination of the machine learning techniques with the standard MC methods. In the context of lattice field theory, the synergy of these two approaches remains largely unexplored. The use of machine learning methods is mainly reduced to (i) the investigation of the ability of neural networks to predict lattice observables in non-perturbative domains of parameters and (ii) generate lattice field configurations as an alternative to the generally accepted Monte Carlo approach. In our paper, we

address the question of the critical behavior of lattice observables in SU(2) and SU(3) gauge theories with the ML techniques that respect the gauge-invariant structure of the theory.

## 2 Machine learning

The use of machine learning methods in lattice QCD is reduced to solving several problems: regression problem, classification problem and simulation problem.

Simulations of configurations in lattice QCD are often computationally expensive, that complicates the process of accumulating statistical data. Modern machine learning techniques can provide opportunities to improve simulation speed. One can build neural network to simulate lattice configurations and after training this approach require less computer power and time than common methods to simulate lattice configurations [3,4].

In case of searching for new physics we try to solve regression or classification problem where neural network trains to reconstruct certain observables from a given information of Monte Carlo configurations corresponding to some set of lattice parameters. A well-trained neural network is subsequently able to predict the value of the observable from data that was previously unknown to it [5,6]

But, generally accepted ML methods aimed to solving various problems in the field of computer vision are not suitable for solving problems in the field of gluodynamics, since lattice data is another kind of data that is fundamentally different from classical images. In order to use machine learning in LQCD, it is necessary either to take into account the properties of lattice data within the neural networks itself or transform the lattice data to a more convenient form. The construction of neural networks consistent with the local symmetry and the matrix origin of the lattice data is still an active topic of discussion in the current literature [7,8].

## 3 Lattice Simulations

We consider lattice SU(N) gauge theories with $N = 2$ and $N = 3$ colors at finite temperature. The lattice theory is formulated via the Euclidean path integral

$$Z = \int \left(\prod_l U_l\right) e^{-S[U]}, \tag{1}$$

where the integration over the lattice gauge fields $U_l$ that belong to the SU(N) gauge group.

The Wilson action of the lattice SU(N) Yang-Mills theory,

$$S[U] = \beta \sum_P \left(1 - \frac{1}{N} \mathrm{Re}\left[\,\mathrm{Tr}\, U_P\right]\right). \tag{2}$$

is formulated in the Euclidean spacetime on the lattice with the volume $N_s^3 \times N_t$ with periodic boundary conditions in all dimensions. The sum runs over the lattice plaquettes $P = \{x, \mu\nu\}$ described by the position of a plaquette corner $x$ and the plane orientation with directions $\mu \neq \nu$. The non-Abelian plaquette field $U_P$ is given by the ordered product of the non-Abelian link fields $U_l$ along the perimeter of the plaquette: $U_P = \prod_{l \in \partial P} U_l$.

The Yang-Mills theories possess the confining phase at low $\beta$ values and the deconfinement phase at high values. The phase transition in the simplest two-color ($N = 2$) gluodynamics is of the second order while the theories with the $N \geqslant 3$ colors possess the stronger, first-order phase transition.

The well-known order parameter of the deconfinement phase transition is the Polyakov loop. In the lattice calculations, it is convenient to identify the bulk Polyakov loop:

$$L = \frac{1}{V}\left\langle \left| \sum_x L_x \right| \right\rangle, \tag{3}$$

where the sum goes over all spatial sites $x$ of the lattice and $V = N_s^3$ is the spatial volume. The local Polyakov loop,

$$L_x = \frac{1}{N}\mathrm{Tr} \prod_{t=0}^{N_t-1} U_{x,t;4}, \tag{4}$$

is given by the ordered product of the lattice $U_{x,\mu}$ matrices along the temporal direction $\mu = 4$.

## 4 Neural network architecture and training process

We are trying to find such an architecture of a neural network that could catch correlations with a targeted observable (Polykov loop) and display its properties. In this section, we describe the machine-learning algorithm which includes building of the architecture and training of the neural network. The training points for SU(2) and SU(3) gauge theories are set at the lattice coupling constants $\beta = \beta_{SU(2)} = 4$ and $\beta = \beta_{SU(3)} = 10$, respectively. Both these points correspond to a deep weak-coupling regime.

The training points $\beta_{SU(2)}$ and $\beta_{SU(3)}$ do not correspond to physically viable realizations of the continuum SU(2) and SU(3) Yang-Mills theories in their thermodynamic limits. These points are selected to represent an uninteresting unphysical region of the theory at which, however, the explicit calculations may be performed with the help of a Monte Carlo technique. We will show that the information coming from the MC configurations are enough for the ML algorithm to learn about the order parameter.

To build a machine-learning algorithm that can analyze lattice data of non-Abelian theory, we need to construct a multidimensional dataset from a lattice configuration that is a matrices dataset. To this end, we use the following vector representation for the SU(2) matrices:

$$U = \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix} \equiv \begin{pmatrix} a_1 + ia_2 & a_3 + ia_4 \\ -a_3 + ia_4 & a_1 - ia_2 \end{pmatrix} \rightarrow \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{pmatrix}, \tag{5}$$

where $a_1 = \mathrm{Re}(u_{11})$, $a_2 = \mathrm{Im}(u_{11})$, $a_3 = \mathrm{Re}(u_{12})$, and $a_4 = \mathrm{Im}(u_{12})$.

After the matrix dimension's flattening, an array with shape $[N_t, N_s, N_s, N_s, Dim, 4]$ represents the lattice configuration. The last dimension corresponds to the matrix element numbering discussed above, and $Dim$ is the direction $\mu$ of the matrix $U_\mu(x)$ at every lattice site [Nt, Ns, Ns, Ns]. We use 3D convolutional layers and reshape lattice configuration as a 4D array (3 dimensions for spatial coordinates and one for channels) due to technical reasons. Since we build a neural network that searches correlations between any two matrices $U_\mu(x)$ and $U_\nu(y)$ at the points $x$ and $y$ closed to each other, we merge the last two dimensions of the array. Other two dimensions could be also merged by cost of locality - array $M[y][x]$ can be presented as an array $M[y * N_y + x]$. The arichitecture of the neural network are the following sequence of layers: convolution, relu activation, average pooling, flatten and dense layer. Using different sizes of lattice data requires new architectures to be built, as it turned out, an increase in the temporal direction in the input data requires an additional convolution-relu sequence. The final arichitecture of neural networks are presented in the Table 1. Also more detailed information about the architecture is described in the paper [9].

| | |
|---|---|
| InputData($N_t = 2, N_s^2, N_s, U_\mu$) | InputData($N_t = 4, N_s^2, N_s, U_\mu$) |
| Conv3D + ReLU | Conv3D + ReLU + Conv3D + ReLU |
| AveragePooling3D + Flatten | AveragePooling3D + Flatten |
| Dense | Dense |

Table 1: Neural network architectures for various size of input configurations.

For training in SU(2) and SU(3) case, we generate 9000 lattice configurations at the one point of the lattice coupling $\beta$ for lattices with the spatial sizes $N_s = 8, 16, 32$ and the temporal sizes $N_t = 2, 4$. We also generate 100 configurations for a number of points at lower values of $\beta$, that the neural network does not use for training but rather for prediction.

Although a study of confinement-deconfinement phase transition does not require configurations from all possible vacuum sectors, we found it essential to have high quality data generated from different vacuum sectors to train a neural network.

We train the neural network on the lattice configurations generated in the deconfinement phase at the point $\beta_{SU(2)}$ and $\beta_{SU(3)}$ that is far from the phase transition point. The neural network is trained to predict correctly the value of the Polyakov loop that is already known from the Monte Carlo simulations. It is also important to note that in SU(3) case the Polyakov loop becomes complex number, so we predict the real and imaginary parts separately. Also we use the mean squared error (MSE) as a loss function and the Adam algorithm as the neural network parameters' optimization method. The training is done in batches of size 10 - 50 configurations. The training is halted when the loss function reached a plateau so that the neural network gained the maximal possible – for the given architecture – knowledge how to reconstruct the order parameter from the lattice configurations.

After the training phase, the neural network was able to build a trace of the gauge group matrices product along a closed loop in the time direction. In Figure 1 we show how the neural network trained at one (unphysical) value of the lattice coupling $\beta$ is effectively able to predict the order parameter in the whole region of the $\beta$ values with a good precision.

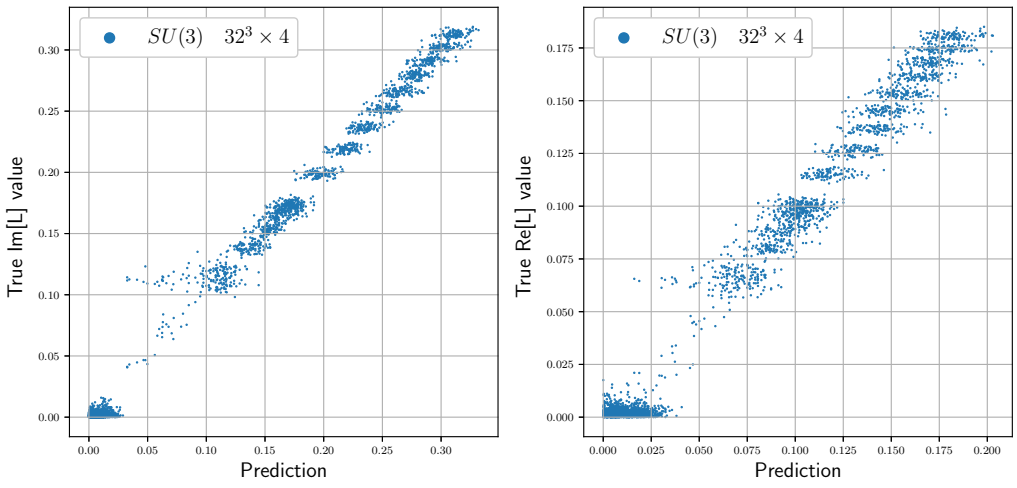


Figure 1: Prediction of neural network on a sample by sample basis of SU(3) $32^3 \times 4$ configurations.

Another important aspect of this prediction is the verification of the invariance of a given observable with respect to gauge transformations. For this check, we completely change the configuration using a set of different uniformly distributed SU(2) or SU(3) matrices respectively. We do several changes and make a prediction for each step for the already changed

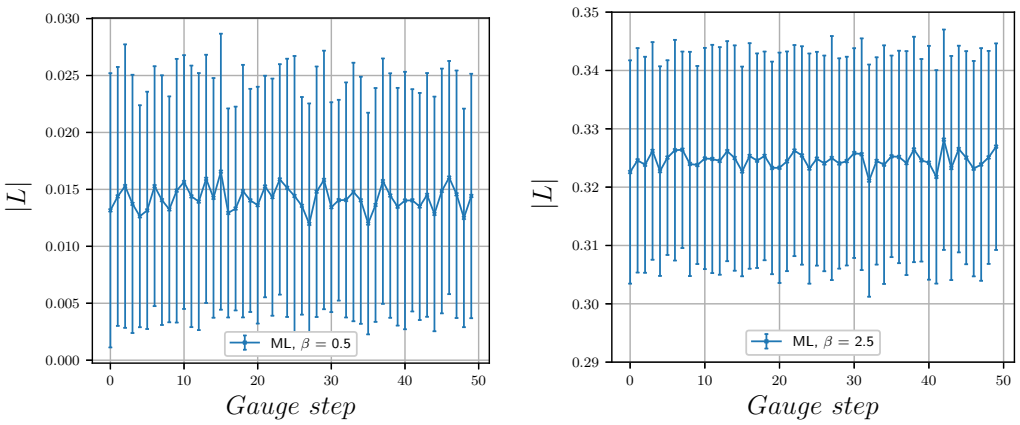


Figure 2: Gauge invariant behavior of numerically constructed Polykov loop at different phases of SU(2) theory as function of uniformly distributed global random gauge transformation step.

configuration. In Figure 2, we demonstrate the result for such a test for the case of the SU(2) theory and lattice size equal to $N_t = 4, N_s = 16$.

## 5 Conclusion

We demonstrate that the machine-learning algorithms allow us to restore, using the data from an unphysical point of the lattice parameter space, the gauge-invariant order parameter applicable to the whole physical critical region of the theory. In other words, our neural network is able to the physical order parameter relevant to the numerically costly critical regime of the model after a training procedure at a set of lattice field configurations that were generated by fast Monte Carlo methods at a single unphysical point outside of the continuum limit of the lattice model.

We also demonstrated that the classical feed-forward neural network could be used to restore simple observables and predict their properties in the critical region of the theory. Our work potentially implies that the ML techniques can predict other, more complex observables and thus be applied to the regions which are unreachable to the standard MC methods. In addition, we demonstrated that gauge-invariant architectures can be built on the basis of feed-forward neural networks.

## Acknowledgements

The work of M.N.C, N.V.G, V.A.G, S.D.L, and A.V.M was supported by the grant of the Russian Foundation for Basic Research No. 18-02-40121 mega. The numerical simulations of Monte Carlo data were performed at the computing cluster Vostok-1 of Far Eastern Federal University. Scientific calculations and analysis by Denis Boyda were supported by the Argonne Leadership Computing Facility, which is a U.S. Department of Energy Office of Science User Facility operated under contract DE-AC02-06CH11357.

## References


[1] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, L. Zdeborová, "Machine learning and the physical sciences", Rev. Mod. Phys. **91**, 045002 (2019)

[2] P. Bedaque, A. Boehnlein, M. Cromaz, et al., "A.I. for nuclear physics", Eur. Phys. J. A **57**, 100 (2021)

[3] M. S. Albergo, G. Kanwar and P. E. Shanahan, "Flow-based generative models for Markov chain Monte Carlo in lattice field theory," Phys. Rev. D **100**, no.3, 034515 (2019).

[4] K. Zhou, G. Endrodi, L.-G. Pang, and H. Stöcker, "Regressive and generative neural networks for scalar field theory", Phys. Rev. D **100**, no.1, 011501(R), (2019).

[5] A. Tanaka and A. Tomiya, "Detection of phase transition via convolutional neural network", J. Phys. Soc. Jap. **86**, no.6, 063001 (2017).

[6] B. Yoon, T. Bhattacharya and R. Gupta, "Machine Learning Estimators for Lattice QCD Observables," Phys. Rev. D **100**, no.1, 014504 (2019).

[7] G. Kanwar, M. S. Albergo, D. Boyda, K. Cranmer, D. C. Hackett, S. Racanière, D. J. Rezende, P. E. Shanahan, "Equivariant flow-based sampling for lattice gauge theory", Phys. Rev. Lett. **125**, 121601 (2020).

[8] M. Favoni, A. Ipp, D. I. Müller, D. Schuh, "Lattice gauge equivariant convolutional neural networks", [arXiv:2012.12901 [hep-lat]]

[9] D. L. Boyda, M. N. Chernodub, N. V. Gerasimeniuk, V. A. Goy, S. D. Liubimov, A. V. Molochkov, "Machine-learning physics from unphysics: Finding deconfinement temperature in lattice Yang-Mills theories from outside the scaling window", Phys. Rev. D **103**, 014509 (2021)