Machine learning representation of the F_2 structure function over all charted Q^2 and x range

S. Brown
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061, USA

G. Niculescu and I. Niculescu
James Madison University, Harrisonburg, Virginia 22807, USA

(Received 29 June 2021; accepted 6 December 2021; published 23 December 2021)

Structure function data provide insight into the nucleon quark distribution. They are relatively straightforward to extract from the world's vast, and growing, amount of inclusive leptoproduction data. In turn, structure functions can be used to model the physical processes needed for planning and optimizing future experiments. In this paper a machine learning algorithm capable of predicting, using a unique set of parameters, the F_2 structure function, for four-momentum transfer $0.055 \leqslant Q^2 \leqslant 800.0$ GeV^2 and for Bjorken x from 2.8×10^{-5} to the pion threshold, is presented. The model was trained and reproduces the hydrogen and the deuterium data at a level comparable with the average uncertainty of the experimental data. Extending the model to heavier nuclei or expanding the kinematic range is straightforward. The model is at least ten times faster than existing grid-based structure functions parametrizations that rely on interpolation and a hundred times faster than models requiring convolutions, making it an ideal candidate for event generators and systematic studies.

DOI: 10.1103/PhysRevC.104.064321

I. INTRODUCTION

Inclusive electron scattering experiments have been used for more than fifty years to gain insight into the structure of subatomic particles. As far back as the 1960s this type of experiments, carried out at the Stanford Linear Accelerator (SLAC), provided the experimental evidence for the existence of quarks [1,2].

Starting with the experimental cross section for inclusive scattering one can define and extract the so-called structure function(s), which parametrize the spatial extent of the target. Using the wealth of data accumulated, several structure function models have been developed. Some of these models are predominantly phenomenological [3–5] while others build the structure function starting from the underlying parton distribution functions (PDFs). "Hybrid" approaches that combine the different approaches traditionally used in the deep inelastic and resonance regimes are also available [6]. For a recent review of these see [7] and references therein.

The present work uses machine learning (ML) to develop a structure function model, named "inclAI" (which is a shortened form for "inclusive artificial intelligence"). The model aims to provide accurate F_2 predictions over as large a Bjorken x and four-momentum transfer, Q^2, kinematic region as possible. This includes both deep inelastic scattering (DIS) data as well as resonance region data. The model is built *ab initio* to handle both hydrogen and deuterium data and it can be easily extended to heavier nuclei. The model does not make assumptions, implicit or explicit, about the data, and a unique set of parameters is used to predict the structure function regardless of the target or the kinematic regime. While this model does not directly provide PDFs, it is fast and reasonably accurate, making it attractive for use in applications where very large number of predictions are needed (event generators, acceptance simulation, radiative correction estimation).

This paper is structured as follows. In Sec. II we briefly review the basics of inclusive electron scattering and main approaches in structure function modeling. Section III describes the input data set while Sec. IV introduces the machine learning approach used in this study. Section V presents the structure function results including the associated uncertainty studies that were undertaken. The last section presents our conclusions.

II. INCLUSIVE ELECTRON SCATTERING AND STRUCTURE FUNCTION MODELS

The data used in this study come primarily from fixed target charged lepton-nucleon scattering experiments: a lepton of energy E scatters from a stationary nucleon and is detected at an angle ϑ with an energy E', while the final hadronic state is not detected. In the one-photon-exchange approximation the lepton-nucleon scattering process is mediated by the exchange of a virtual photon and can be represented by the Feynman diagram shown in Fig. 1, where l, l' are the incident and scattered leptons respectively, N is the target nucleon (of mass M),

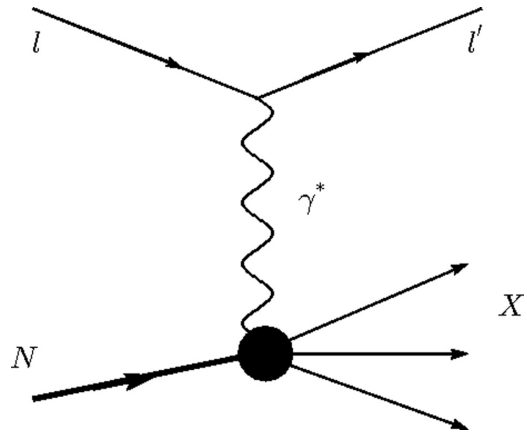

FIG. 1. Feynman diagram for inclusive lepton-nucleon scattering in the one-photon-exchange approximation.

and X represents the recoiling system. Some of the kinematic variables used to describe the inclusive lepton-nucleon scattering process are the four-momentum transferred from the lepton to the target nucleon, Q^2; the fraction of the nucleon's momentum carried by the struck quark, x; the energy lost by the lepton, $\nu = E - E'$; and the invariant mass squared of the hadronic final state, W^2:

$$Q^2 = 4EE' \sin^2(\vartheta/2), \tag{1}$$

$$x = \frac{Q^2}{2M\nu}, \tag{2}$$

$$W^2 = M^2 + 2M\nu - Q^2. \tag{3}$$

Using this approximation the differential cross section can be expressed in terms of structure functions F_1 and F_2, which parametrize the spatial extent of the charge distribution of partons inside the nucleon:

$$\frac{d^2\sigma}{d\Omega\, dE'} = \sigma_{\text{Mott}}\left(\frac{2}{M}F_1(x, Q^2)\tan^2\frac{\vartheta}{2} + \frac{1}{\nu}F_2(x, Q^2)\right), \tag{4}$$

with σ_{Mott} the cross section for scattering off of a pointlike particle. F_1 and F_2 can be related to the cross section for absorbing either a transversely (σ_T) or a longitudinally polarized virtual photon (σ_L) [8].

In the quark-parton model one can write the structure functions as combinations of the underlying quark and antiquark distribution functions. Various groups have used this formalism to extract parton distribution functions (PDFs). A list and brief discussion of the most recent PDF parametrizations available, including machine learning approaches, can be found in [7]. These parametrizations focus on the deep inelastic scattering process (large Q^2 and W^2) and are not suitable in the resonance region.

Phenomenological models of the inclusive cross section in the resonance region have been developed, with the most recent ones being by Christy and Bosted [3,4]. These models describe the cross section as a resonant contribution overlayed on top of a nonresonant background. The structure functions can then be obtained from the differential cross section using the ratio of the longitudinal and transverse cross sections, $R = \sigma_L/\sigma_T$.

Both approaches can be computationally intensive when calculating experimental observables such as cross section or structure functions. This is due to the interpolations or convolutions required for each and every prediction. For PDF-based models the convolution is carried out over the parton distributions themselves. For the phenomenological parametrizations convolutions over the Fermi distribution are needed when calculating structure functions for deuterons or heavier nuclei. This can significantly impact several important data analysis steps where a very large number of predictions are needed, such as radiative correction estimation and detector response function modeling (i.e., acceptance calculations).

III. INPUT DATA SELECTION

To develop the machine learning model described in this work the input data were selected and curated as follows:

(i) The data must be published or available from public sources/databases.
(ii) The data must provide either the F_2 structure function or the differential cross section. For the datasets providing only the latter the R1998 parametrization [9] was used to extract F_2, with a small increase in the uncertainty budget.
(iii) For each data point the statistical and systematic uncertainties were added in quadrature to obtain the point-to-point uncertainty. This was subsequently used, in conjunction with the overall normalization uncertainty of each experiment, when known, to estimate the influence of the data uncertainty on the model predictions as described in Sec. V.
(iv) Only data above the pion threshold were used.
(v) No additional Q^2 or x cuts were imposed, resulting in the most extensive data set available.
(vi) Even though this study is limited to hydrogen and deuterium data, an extension to heavier targets is easily achievable.

The data set selected includes both electron-nucleon and as muon-nucleon experiments, originating from several international laboratories: SLAC [10–12], DESY [13], CERN [14–16], and JLab [17–20]. With such a large dataset, spanning several decades and laboratories, some "tensions" between datasets covering the same or adjacent phase space regions can be expected and have been documented [10,21–23]. As it is difficult, if not impossible to carry out a full, *ab initio*, reanalysis of decades-old experiments, the input data were left "as is," not modified or renormalized *post hoc*. In other words, the input data was taken directly from the original publications stemming from the various experiments or from near-contemporaneous analyses thereof [10]. Figs. 2 and 3 show the coverage of the selected input data set for hydrogen and deuterium, respectively. The curve shown represents $W^2 = 4$ GeV^2, the nominal boundary between the resonance and the deep inelastic scattering regions.

A total of 12036 data points, 7526 on hydrogen and 4510 on deuterium targets, were used in this study. For both

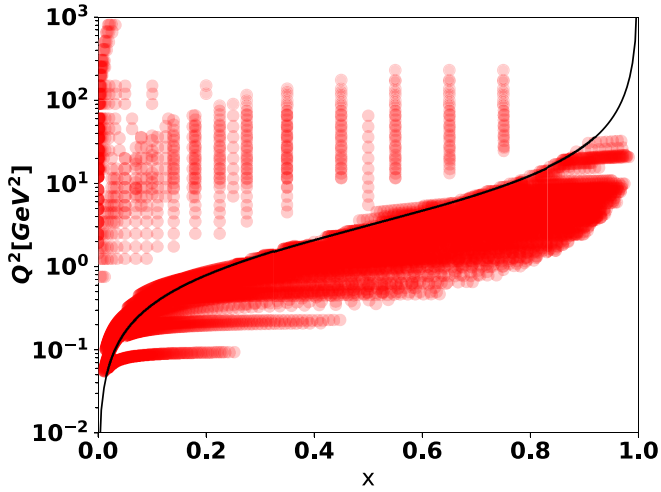

FIG. 2. x and Q^2 coverage of the input hydrogen data set chosen for this study. The curve shown represents $W^2 = 4$ GeV^2, the nominal boundary between the resonance and the deep inelastic scattering regions.

targets the bulk of the data is in the so-called "resonance region" ($W^2 < 4.0$ GeV^2), ≈80% for hydrogen and ≈90% for deuterium. The absolute value of the deuteron F_2 structure function was used and not its relative value (i.e., the "per nucleon" F_2). The availability of a large number of data points in the region where the F_2 structure function exhibits substantial nonlinearity should help guide the training of the machine learning model.

IV. MACHINE LEARNING MODEL

A. Rationale

As noted above, several approaches have been used, with varying degrees of success, to obtain fits of the structure function. All of these approaches assume a specific functional form (be it theory inspired or phenomenological) for the fitted quantity. Subsequently, a minimization procedure is used to find the best values for the free parameters of this function.

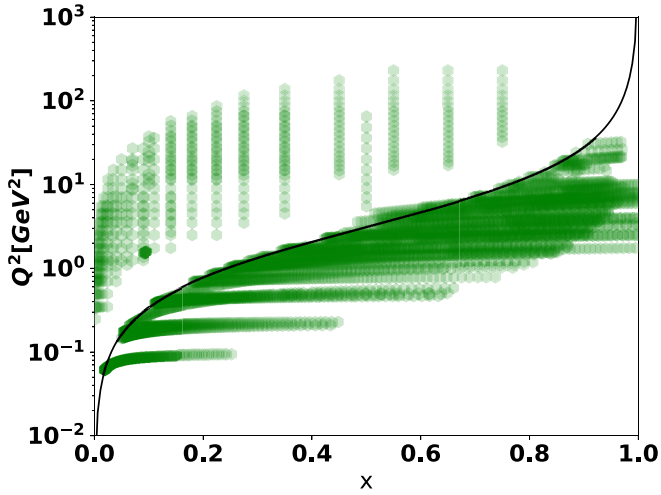

FIG. 3. x and Q^2 coverage of the input deuterium data set chosen for this study. The curve shown represents $W^2 = 4$ GeV^2, the nominal boundary between the resonance and the deep inelastic scattering regions.

While in principle one can obtain the Hessian matrix associated with the fit and use error propagation to infer the uncertainties associated with quantities of interest that depend on the structure function (cross section, moments), this procedure becomes impractical as the parameter space increases. Though a number of alternative methods circumventing the calculation of the full error matrix are available (truncated Newton, quasi-Newton methods [24]) one still runs the risk of potentially under- or overestimating the uncertainty of the fit.

Furthermore, once the choice of the functional form is made, one effectively has biased the algorithm against all other possible functions that share the same domain and codomain as the initial choice. There are three substantial drawbacks associated with this choice, as described below.

First, lacking a clear theoretical guidance the functional form choice is somewhat arbitrary, and it requires constant refitting and/or redefinition (as in changing the functional form) as new data emerge. While for any data reduction problem the addition of a large body of new information does warrant the revision of one's model, the constant refitting of the model's parameters or the addition of new parameters casts doubts about a model's predictive power.

Second, the functional form choice might limit the model's ability of making predictions over the whole domain of the observables studied. For example, PDF-based models are theoretically limited to the DIS region where the observables are varying smoothly with respect to x and Q^2 and thus they are not able to reproduce the low W^2 resonance region data. Similarly, models that explicitly include resonance behavior (Breit-Wigner or similar shapes) are generally optimized for data below $W^2 = 4$ GeV^2. Some hybrid approaches attempt to address this issue by combining two or more models, each known to perform reasonably in its own portion of the structure function domain. The models are then combined using a designated "merging phase space region" where the prediction is simply a linear combination of the individual models. While this type of solution does work in practical implementations, the continuity of the prediction in the merging region suffers. The approach described here does not make any explicit or implicit cuts on W^2 and/or Q^2 and aims to reproduce, with a unique set of parameters, both resonance and DIS data.

Lastly, most of the theoretical insights are cast at the parton distribution level. For all cases where experimental observables, such as structure functions or cross sections, need to be evaluated, CPU-intensive convolutions or interpolations are required. This greatly increases the computational resources needed, especially for applications in which very large number of events are generated (Monte Carlo simulations, radiative corrections, etc.).

B. Artificial neural network architecture

In recent years artificial intelligence and machine learning approaches have seen increased use in many fields,

including substantial strides in nuclear and particle physics (pattern recognition, event reconstruction and topology, accelerator control and maintenance [25]). Significant strides have been made even in the specific area of structure function modeling [22].

The ML approach used here attempts to address or circumvent the issues listed in Sec. IV A and provide fast, accurate predictions for the structure function F_2. The model is completely data driven, with no assumptions (or biases) of any underlying physics. The implementation is based on artificial neural networks (ANNs) and a back-propagation algorithm for the optimization of the network parameters.

The most important design constraints and features are presented below, grouped separately into physics choices and machine learning or implementation choices. As using ML is a relatively new addition to the computational capabilities of nuclear and particle physicists, the latter set of choices will be more extensively detailed, highlighting the differences (and introducing some data-analytics-specific vocabulary) between ML and traditional fitting procedures that readers might be familiar with.

Physics choices are

(i) The ML code shall provide F_2 predictions over as large Q^2 (from $Q^2 < 1$ to $Q^2 \approx 1000$ GeV^2) and x [from very small ($\approx 10^{-5}$) to the pion threshold] range as possible.
(ii) The ML input set shall incorporate charged lepton-nucleon data (DIS, resonance region, using electron as well as muon beams) that provide either the structure function F_2 itself or the differential cross section (from which the structure function can be obtained). In the latter cases the generally accepted R1998 [9] function shall be used for the ratio of the longitudinal and transverse cross sections.
(iii) The ML model shall consider both hydrogen and deuterium data, with no bias or explicit provisions for either. Furthermore, the model shall provide a transparent way of generalizing the approach to other nuclei (see ML implementation below).
(iv) The total uncertainty associated with the input data shall be used in assessing the error associated with the ML predictions. No attempt to second guess or rescale the original, published, data shall be implemented.

ML implementation choices are

(i) The ML model shall consist of an assembly of ANNs with varying topologies. Simple majority voting (average) shall be used as the as a robust ML prediction. As a faster alternative one can pick a single topology as "the" model. In either approach the spread in the F_2 predictions will be used as a measure of the model uncertainty due to the topology choice.
(ii) The data uncertainty influence on the model prediction shall be estimated using the Monte Carlo method (see Sec. V).
(iii) The ML model shall be implemented using "industry standard" tools and libraries and should be able to complete its training using modest computation means.
(iv) The ML model shall run in a consistent manner and shall have a way of assessing the quality of its predictions.
(v) The residual between the data and the ML model shall be commensurable with or better than the uncertainty associated with the input data points themselves.
(vi) The ML model shall try to minimize the mean square error (MSE):

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(F_2^{\mathrm{data}} - F_2^{\mathrm{model}}\right)^2 \tag{5}$$

as it is an unbiased statistic.

From the ML or data-analytics standpoint, the F_2 prediction is a supervised learning exercise where one seeks to infer the best possible values for the parameters of the learning function given a set of "labels" [i.e., the observable(s) to be fitted, in this case F_2] and their corresponding "features" (i.e., the variables on which the observable depends). As the labels are known in advance for each set of inputs, the ML method is deemed "supervised." Furthermore, given that the structure function F_2 takes real values, the ML algorithm is a regression rather than a classification.

Given that the behavior of the structure function, even considering the resonance region, is reasonably smooth and continuous, and given the time and hardware constraints of the project, a relatively simple ML was chosen, namely a fully connected ANN with one input layer, one output layer, and a number of hidden layers. While the numbers of neurons in the input and output layers are determined by the number of features and, respectively, by the number of labels (one), the number of hidden layers was varied: networks with up to ten hidden layers were tested. The number of neurons per layer was varied as well. Shallow networks (one or two hidden layers) required a relatively large number of neurons to achieve any significant performance. For one hidden layer network topologies with up to 1000 neurons/layer were tested while for two hidden layer networks widths of up to 70×70 neurons were used. For networks deeper than two hidden layers topologies with 4 to 15 neurons/layer for layers two and above were used. As recommended in the machine learning literature [26] the number of neurons in the first layer was kept larger, 20 to 50 neurons. The number of neurons in layers two and above was kept the same for all layers. This substantially reduced the hyperparameter tuning task (less parameters that required study and optimization) without loss of model precision.

Thus each network's topology can be summarized as

$$(n_i, n_1, n_2, \ldots, n_m, n_o), \tag{6}$$

where n_i is the size of the input layer (i.e., the number of features used in the model), n_o is the size of the output layer (the number of labels, which for this project was one, F_2), and n_j $(j = 1, 2, \ldots m)$ are the sizes of the m hidden layers. A graphical representation of one of the ANNs used in this work is shown in Fig. 4.

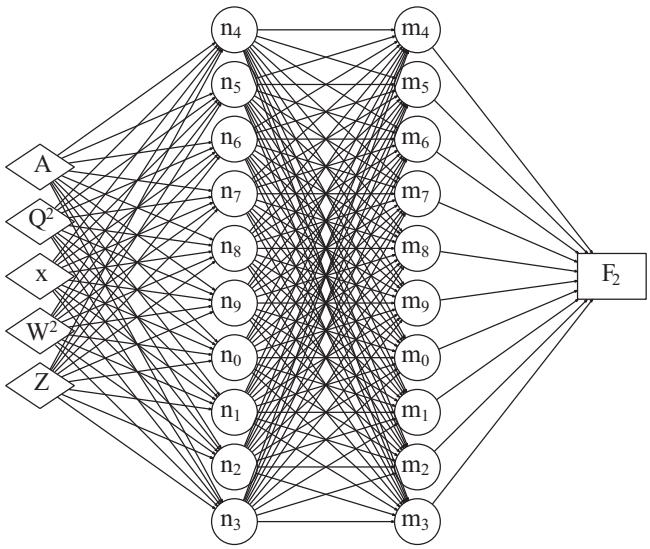

FIG. 4. Sample topology for the artificial neural networks used in this work.

Theoretically, for a given target, F_2 is only a function of x and Q^2. However, as the data set includes many resonance region entries, where several peaks are prominent (especially at low Q^2), W^2 was also added as a feature of the model. W^2 is, of course, not independent of x and Q^2 and in traditional fitting approaches one seeks, as much as possible, independent input variables. ML models, however, often benefit from such "feature engineering" (i.e., creating new features based on existing ones) as one can model nonlinear behavior using less neurons in the hidden layers than in the case when the network will only have the absolute minimum set of features.

Lastly, as the model is to handle both hydrogen and deuterium data, the atomic (Z) and mass (A) numbers of the target were introduced as features. This brings the total number of features to five and also provides a straightforward path of extending the model to heavier nuclei if desired.

Two types of activation functions[1] were used in these networks: a sigmoid [27] for the output layer and a rectified linear unit (ReLu) activation function, $f(x) = \max(0, x)$ [28], for the hidden layers. The input layer simply copies its input to its output. The model outlined above was implemented in Python, using the Keras package [29] with a Tensorflow back end [30].

V. NEURAL NETWORK TRAINING AND RESULTS

The class of ANNs described in Sec. IV B was trained using the experimental data presented earlier in Sec. III. The data were randomly split into training (80%) and testing (20%) sets. These sets were kept the same throughout this study. The training was done exclusively on the training sample. The performance of each ANN was evaluated on the test sample.

[1]In ANNs the *activation function* is the response function of a neuron given a set of input(s).

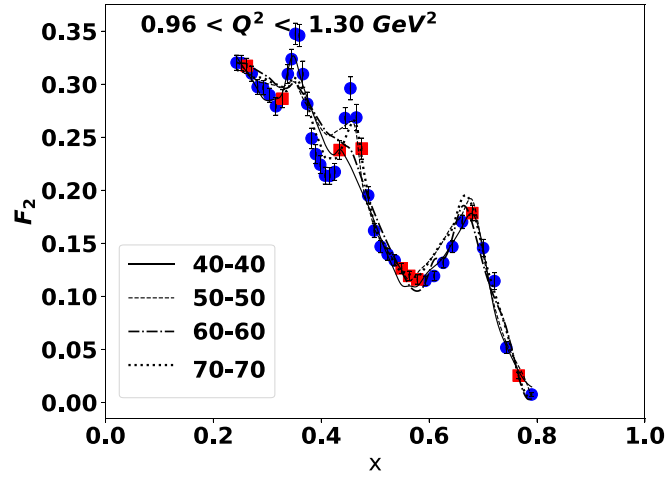

FIG. 5. ANN predictions for several two-hidden-layer networks trained as part of this study for a sample of F_2 structure function data off of hydrogen in the resonance region.

All the features (input variables) of the model were standardized (i.e., subtracting the mean and dividing by the standard deviation) prior to training. This transformation was based solely on the training sample. This procedure ensures that the transformed features will have a mean of zero and a unit standard deviation. The exact transformation (i.e., using the mean and standard deviation obtained from the train sample) was then used on the test sample. Given the statistical nature of the train/test split, the transformed test feature's mean and standard deviation might differ (slightly) from zero and unity, respectively. Though one would be tempted to use the test sample mean and standard deviation, that will bias the result, effectively defeating the purpose of having separate train/test samples.

Given the choice of activation function for the output layer, the labels (the structure function output variable) were subjected to a min-max transform (subtracting the lowest value and dividing by the max-min range), resulting in labels in the [0,1] range. As before, the procedure was based solely on the training sample labels. Once the ANN training is completed the inverse of this function needs to be applied in order to get back to the physical quantity of interest, F_2. The parameters associated with the scaling of both the features and the labels are saved as part of the ML model.

The training proceeded for up to 10 000 epochs[2] for each ANN topology. An early stopping procedure based on a minimum improvement (10^{-6}) in the cost function[3] every 500 epochs was also implemented. For convenience the algorithm could start "cold" (i.e., random starting values for the parameters) or "warm" (continuing from the best set of parameters previously found for the given ANN topology).

[2]In machine learning an *epoch* is a complete pass through the training data. It is similar to an "iteration" in a conventional fitting procedure.

[3]The statistic to be minimized during training.

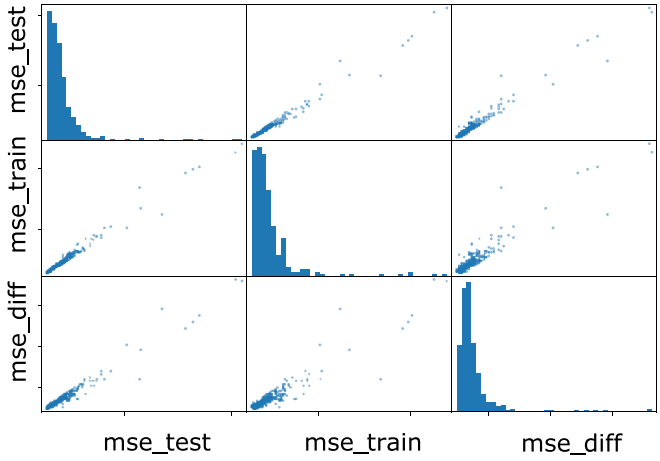

FIG. 6. Scatter matrix showing the mean squared error for the training and testing data sets. The difference between the training and testing MSEs is also shown. The diagonal plots show the histogram of each of these three values while the off-diagonal panels show the pairwise correlation between these quantities.

Several ANN networks were trained as part of this study. The number of hidden layers as well as the number of neurons per layer were varied. The networks with less neurons per layer are more biased but also have lower variance. An increase in the number of neurons per layer results in less bias but also in a substantially larger variance. An in-depth discussion about the intricacies related to the bias-variance tradeoff can be found in [31].

Figure 5 shows the ANN output for some of the two-hidden-layer networks studied here for a representative set of F_2 (on hydrogen) data points in the resonance region. Line type (solid, dashed, etc.) and line thickness help differentiate between the various ANN topologies. It can be seen that less complex networks have difficulties reproducing the sharper data features specific to the resonance region. For a fixed number of hidden layers, the network complexity is directly related to the number of parameters: in this study a 40-40 network has 1921 parameters while the 70-70 network has 5461 parameters. However, this study found that deeper networks [32] were able to achieve similar or better performance with substantially less parameters. A 40-10-10-10 network has only 881 parameters and a mean square error similar to the 70-70 network.

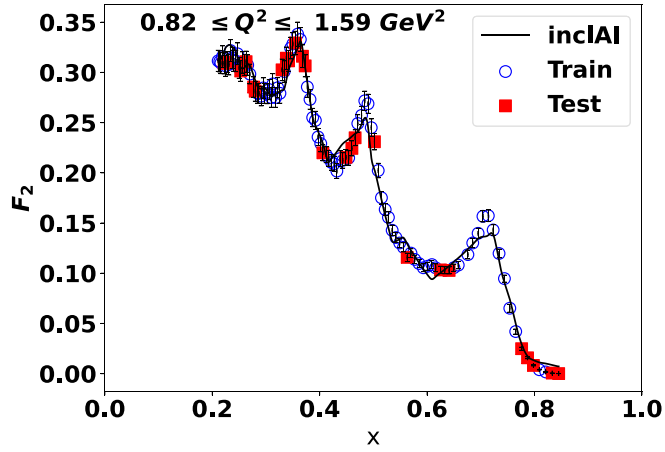

FIG. 8. Sample inclAI F_2 structure function results for hydrogen (II).

In this study more than 400 different ANN topologies were trained and tested. Beside the MSE used in the minimization procedure, additional performance criteria were recorded and can be used for "best model" selection the number of parameters, the mean absolute deviation (MAD) between the data and the ANN (this is a quantity less sensitive to outliers), and the fraction of events with a large (larger than 200%) difference between the experimental data and model prediction. Separate MSE values were saved for the DIS and resonance regions, for the test and train subsets, and for the two targets considered.

To select models with a good performance only networks with an average absolute difference between the experimental data and the prediction below 7% were retained. Additionally, models were required to have less than 4% of the data in the tail of this distribution. To limit overlearning, a cut on

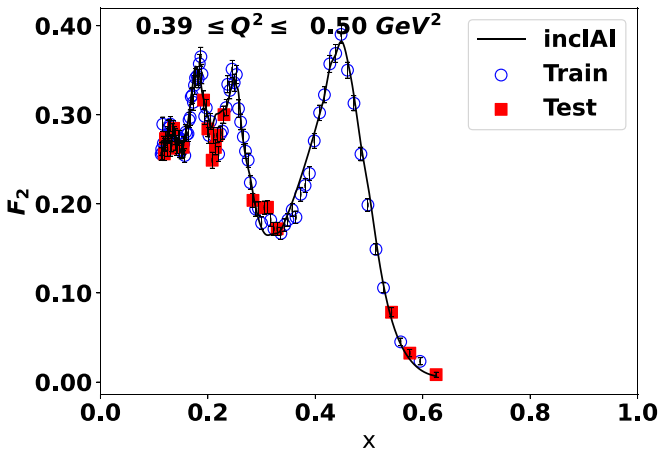

FIG. 7. Sample inclAI F_2 structure function results for hydrogen (I).

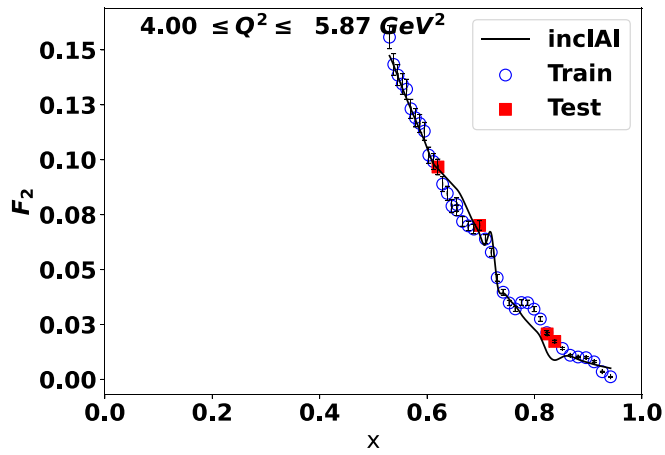

FIG. 9. Sample inclAI F_2 structure function results for hydrogen (III).

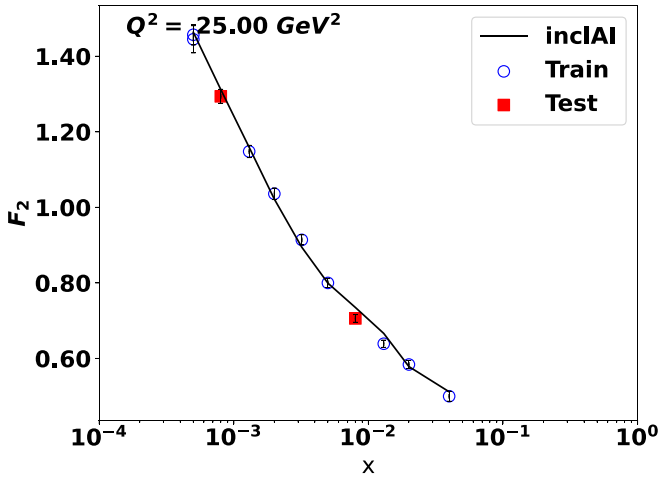

FIG. 10. Sample inclAI F_2 structure function results for hydrogen (IV).

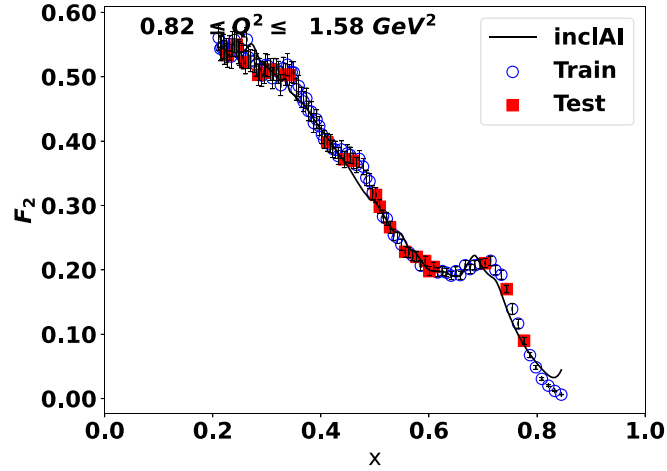

FIG. 12. Sample inclAI F_2 structure function results for deuterium (II).

the difference between the training and testing MSE values was applied ($\text{MSE}_{\text{diff}} \leqslant 0.0005$; see also Fig. 6). Finally, only models with less than 1500 parameters were retained. This also limits the potential for overlearning while promoting faster models. Combined, these criteria selected 52 models, all of which can be used to model inclusive electron or muon scattering off of hydrogen or deuterium. The spread of these predictions, 3.5%, is interpreted as a measure of the uncertainty due to the network topology.

To test the stability of the ML optimization procedure with respect to the initial choice of weights, several neural networks were repeatedly trained and their performance at the end of the training procedure was recorded. The "cold" training option (i.e., starting always from a random set of initial weights) was used in each case. The variation observed was of the order of 1.5%.

Figures 7 to 10 show representative hydrogen F_2 distributions as a function of Bjorken x for various Q^2 ranges. For all panels the training data points are shown using open circle symbols while the points used for testing are shown with solid squares. The solid line represents the AI model described in this work. Similar results for deuterium are shown in Figs. 11–14. For all these plots a 40-10-10-10-10-10 network was used. While the agreement between the data (both used in training and in testing) is good, discrepancies occasionally appear at the edge of the F_2 domain (high x in these figures), as seen, for example in Fig. 12.

One of the key strengths of the machine learning algorithm described here is its ability to model the existing experimental data over an extended kinematic range. Figure 15 illustrates this versatility by comparing experimental hydrogen F_2 structure function with the inclAI model over a large x range. All experimental data with $7 \leqslant Q^2 \leqslant 13$ GeV^2 were selected. This subset of data contains results from several laboratories (SLAC, CERN, DESY, and JLab). The experimental data are shown as closed symbols, with their respective uncertainties, while the inclAI calculation is represented by the solid line. It is worth reiterating that the machine learning approach uses a

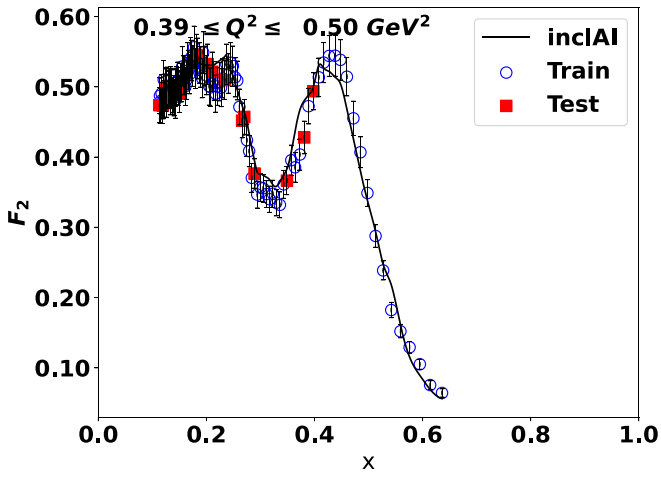

FIG. 11. Sample inclAI F_2 structure function results for deuterium (I).

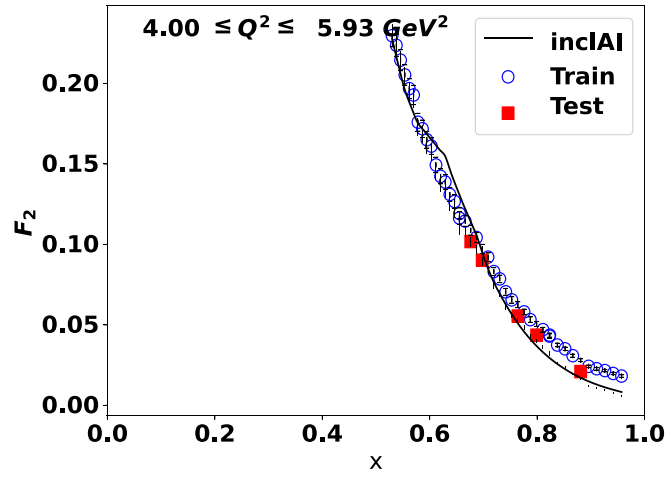

FIG. 13. Sample inclAI F_2 structure function results for deuterium (III).

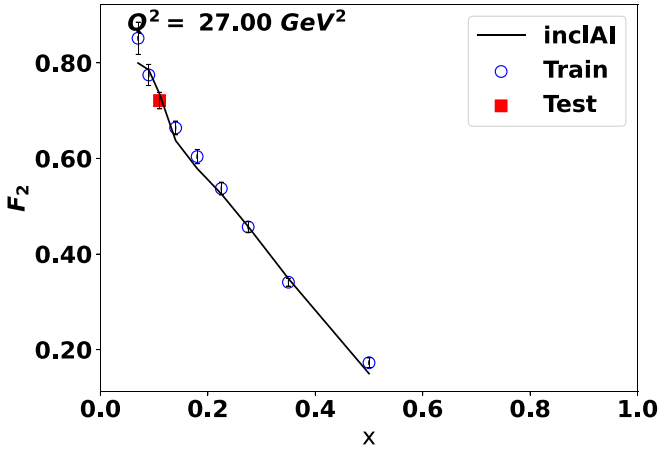

FIG. 14. Sample inclAI F_2 structure function results for deuterium (IV).

single set of parameters to describe all these data (as well as deuterium structure functions), whereas most theory-inspired models are more focused (and thus restrictive) on particular x and Q^2 ranges (resonance region, DIS).

To estimate the influence of the data uncertainties on the model, the Monte Carlo method [7,22] was used. A large (500) number of pseudodata sets were randomly generated using the total uncorrelated experimental uncertainty as the standard deviation of a normal distribution centered at the nominal (published) value for each data point. For each experiment the normalization uncertainty, when known, was interpreted as a box-shaped distribution [33]. A unique random number was uniformly generated in this box for each experiment and added, in quadrature, to the uncorrelated uncertainties. The same ANN was trained on each of these pseudodata sets, resulting in five hundred networks. These networks were used to obtain predictions for all kinematic settings used in this study. The standard deviation for each

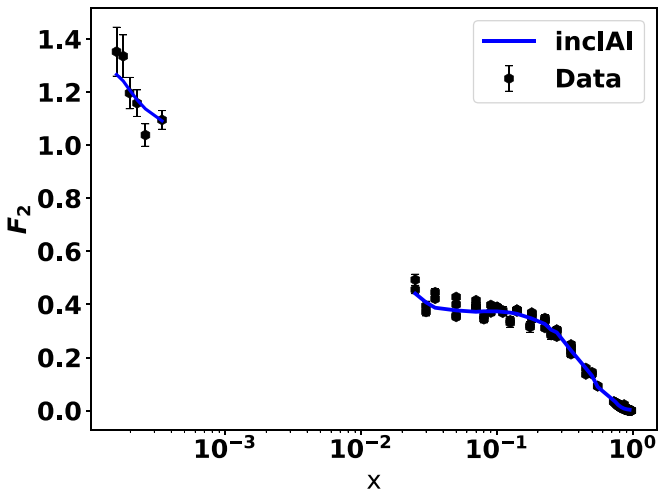

FIG. 15. Experimental hydrogen F_2 structure function (solid hexagon) compared to the inclAI results (solid curve) as a function of x for $7 \leqslant Q^2 \leqslant 13$ GeV^2.

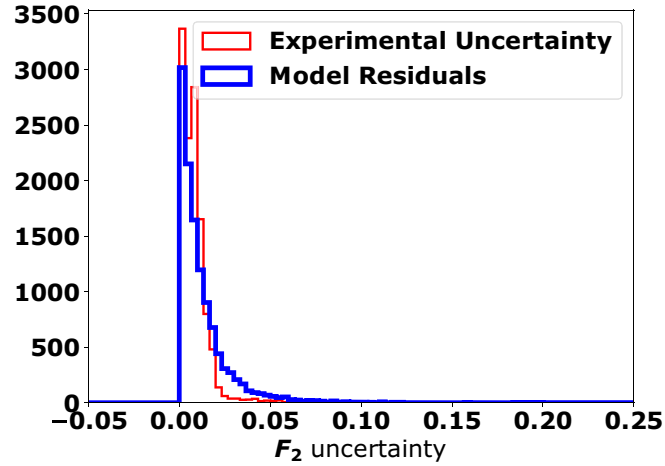

FIG. 16. Comparison between the total experimental uncertainty (thin red line) and the residual between the data and the inclAI model (thick blue line). Note that these are absolute, not relative values.

data point was then obtained and is interpreted as the model uncertainty for that point. Based on this study the overall model uncertainty due to the data errors is 6.3%. Adding in quadrature this value to the uncertainties due to network topology and the variation due to the choice of initial network weights, the overall inclAI uncertainty is ≈7.5%, comparable with the total average experimental uncertainty.

As is the case with any machine learning project where the model is left as unbiased as possible, there is the danger of overlearning. Essentially a very complex model (very deep and/or wide network) ends up "learning by heart" the training examples but performs substantially worse on the testing set. This problem can potentially exacerbate if too many training epochs are used. To mitigate this type of problems the inclAI model described here uses early stopping, a limited number of hidden layers, and possibly averaging over the

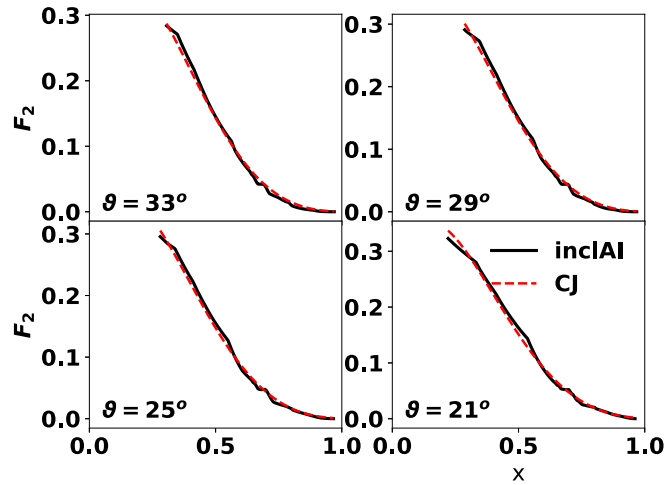

FIG. 17. Hydrogen F_2 structure function inclAI predictions for some of the kinematic settings acquired by JLab experiment E12-10-002 using a 10.6 GeV electron beam. The corresponding CJ15 predictions are shown with a dashed line.

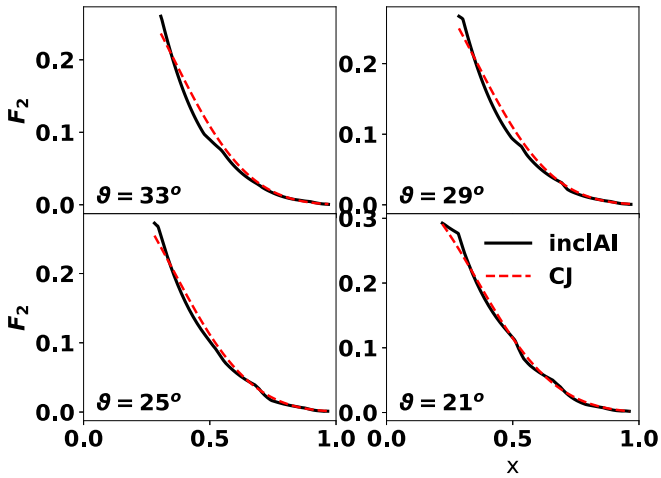

FIG. 18. Deuterium F_2 structure function inclAI predictions for some of the kinematic settings acquired by JLab Experiment E12-10-002 using a 10.6 GeV electron beam. The corresponding CJ15 predictions are shown with a dashed line.

predictions of several ANN topologies. Figure 16 shows a frequency distribution of the total experimental uncertainty for all data used in this study (thin line). The absolute value of the residual difference between the model prediction and the experimental data is also shown (thick line). The widths of the two distributions are similar, indicating that (a) the algorithm has converged and (b) the ANNs have not overlearned.

The speed of the inclAI model was compared with the speed of existing models providing F_2 structure function predictions. As noted above, theory-inspired models make extensive use of interpolations and/or convolutions. The machine learning algorithm described here is a factor of 10 faster than most grid-based models and at least 0 times faster than models that require convolutions (for example for integrating over the Fermi motion for deuterium).

Finally, to test the predictive power of inclAI, the hydrogen and deuterium structure function F_2 was calculated for the kinematic regime probed by the JLab experiment E12-10-002 [34], which took data in 2018 and is currently under final review before publication. Figures 17 and 18 show the inclAI predictions (solid line) for four spectrometer angle settings: 33°, 29°, 25°, and 21° for hydrogen and deuterium using a 10.6 GeV electron beam. The dashed line depicts the CJ15 [35] calculations. The agreement between the inclAI and the CJ15 predictions is remarkable, especially taking into account that the machine learning algorithm has no physics insight built in and is completely data driven.

VI. CONCLUSIONS

A machine learning model (inclAI) of the F_2 structure function was developed. The model implements fully connected artificial neural networks with up to ten hidden layers. Its input features are Q^2, W^2, x, Z, and A and its output (label) is the F_2 structure function. The model was trained on the inclusive leptoproduction world data in the x range from 2×10^{-5} to the pion threshold, and in Q^2 from 0.055 to 800 GeV^2.

With a single set of parameters the model reproduces equally well deep inelastic scattering and resonance data, for both hydrogen and deuterium. Based on the extensive studies carried out as part of this work the average model uncertainty is 7.5%. Furthermore, the distribution of the mean absolute deviation between this model and the data is similar with the uncertainty distribution of the global data set. As the atomic and mass numbers of the target are input features, the model can be easily extended to heavier nuclei. Compared with other available structure function parametrizations, inclAI is very fast, a factor of 10 faster compared with grid-based models and typically 100 times faster compared with models that rely on convolutions, making it an ideal candidate for event generators and Monte Carlo simulations. The Python code for defining and using the ML models described here (including parameters for pre-trained networks) is available from the authors upon request.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation, Grant No. 1913257. The authors would also like to thank Dr. Stephen Wood, and Mr. Thomas O'Neill for his help in reviewing the manuscript.

[1] E. D. Bloom *et al.*, Phys. Rev. Lett. **23**, 930 (1969).
[2] M. Breidenbach *et al.*, Phys. Rev. Lett. **23**, 935 (1969).
[3] P. E. Bosted and M. E. Christy, Phys. Rev. C **77**, 065206 (2008).
[4] M. E. Christy and P. E. Bosted, Phys. Rev. C **81**, 055213 (2010).
[5] M. Arneodo *et al.*, Phys. Lett. B **364**, 107 (1995).
[6] S. A. Kulagin and V. V. Barinov, arXiv:2103.00158.
[7] P. A. Zyla *et al.* (Particle Data Group), Prog. Theor. Exp. Phys. **2020**, 083C01 (2020).
[8] F. E. Close, *An Introduction to Quarks and Partons* (Academic, New York, 1979).
[9] K. Abe *et al.*, Phys. Lett. B **452**, 194 (1999).
[10] L. W. Whitlow *et al.*, Phys. Lett. B **282**, 475 (1992).
[11] S. Rock, R. G. Arnold, P. E. Bosted, B. T. Chertok, B. A. Mecking, I. Schmidt, Z. M. Szalata, R. C. York, and R. Zdarko, Phys. Rev. D **46**, 24 (1992).
[12] C. Keppel, in *The 5th Conference on the Intersections of Particle and Nuclear Physics*, 31 May – 6 Jun 1994, St. Petersburg, FL, edited by S. J. Seestrom, AIP Conf. Proc. No. 338 (AIP, New York, 1995), p. 675; Ph.D. thesis, American University, Washington, DC, SLAC Report No, SLAC-R-694, 1994 (unpublished).
[13] V. Andreev *et al.*, Eur. Phys. J. C **74**, 2814 (2014).
[14] A. C. Benvenuti *et al.*, Phys. Lett. B **223**, 485 (1989).
[15] A. C. Benvenuti *et al.*, Phys. Lett. B **237**, 592 (1990).
[16] M. Arneodo *et al.*, Nucl. Phys. B **483**, 3 (1997).
[17] Y. Liang *et al.*, arXiv:nucl-ex/0410027.
[18] I. Niculescu, C. S. Armstrong, J. Arrington, K. A. Assamagan, O. K. Baker, D. H. Beck, C. W. Bochna, R. D. Carlini, J. Cha, C. Cothran, D. B. Day, J. A. Dunne, D. Dutta, R. Ent, B. W. Filippone, V. V. Frolov, H. Gao, D. F. Geesaman, P. L. J.

Gueye, W. Hinton, R. J. Holt *et al.*, Phys. Rev. Lett. **85**, 1186 (2000).
[19] S. P. Malace *et al.*, Phys. Rev. C **80**, 035207 (2009).
[20] V. Tvaskis *et al.*, Phys. Rev. C **97**, 045204 (2018).
[21] W. T. Giele and S. Keller, Phys. Rev. D **58**, 094023 (1998).
[22] S. Forte *et al.*, J. High Energy Phys. 05 (2002) 062.
[23] S. Alekhin, K. Melnikov, and F. Petriello, Phys. Rev. D **74**, 054033 (2006).
[24] J. Nocedal and S. Wright, *Numerical Optimization* (Springer-Verlag, Berlin, 2000).
[25] C. Tennant, A. Carpenter, T. Powers, A. Shabalina Solopova, L. Vidyaratne, and K. Iftekharuddin, Phys. Rev. Accel. Beams **23**, 114601 (2020).
[26] A. Geron, *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*, 2nd ed. (O'Reilly, Newton, MA, 2019).
[27] T. M. Mitchell, *Machine Learning* (WCB McGraw-Hill, New York, 1997).
[28] K. Fukushima, IEEE Trans. Syst. Sci. Cybern., **5**, 322 (1969).
[29] Keras: The Python deep learning API, https://keras.io/.
[30] TensorFlow: Free and open-source software library for machine learning, https://www.tensorflow.org/.
[31] P. Mehta *et al.*, Phys. Rep. **810**, 1 (2019).
[32] H. W. Lin, M. Tegmark, and D. Rolnick, J. Stat. Phys. **168**, 1223 (2017).
[33] R. Devenish and A. Cooper-Sarkar, *Deep Inelastic Scattering* (Oxford University Press, Oxford, 2004).
[34] S. Malace, E. Christy, C. Keppel, and I. Niculescu, Precision measurements of the F_2 structure function at large x in the resonance region and beyond, https://www.jlab.org/exp_prog/proposals/10/PR12-10-002.pdf.
[35] A. Accardi, L. T. Brady, W. Melnitchouk, J. F. Owens, and N. Sato, Phys. Rev. D **93**, 114017 (2016)