

Connecting Dualities and Machine Learning

Philip Betzler and Sven Krippendorf*

Dualities are widely used in quantum field theories and string theory to obtain correlation functions at high accuracy. Here we present examples where dual data representations are useful in supervised classification, linking machine learning and typical tasks in theoretical physics. We then discuss how such beneficial representations can be enforced in the latent dimension of neural networks. We find that additional contributions to the loss based on feature separation, feature matching with respect to desired representations, and a good performance on a ‘simple’ correlation function can lead to known and unknown dual representations. This is the first proof of concept that computers can find dualities. We discuss how our examples, based on discrete Fourier transformation and Ising models, connect to other dualities in theoretical physics, for instance Seiberg duality.


1. Introduction

In many cases, when we want to describe a dynamical system in physics we identify the effective field theory governing its dynamics. However, in some cases there are multiple effective field theories describing the same system. This phenomenon is referred to as duality. Dualities are a very powerful tool in fundamental physics, ubiquitously used in dynamical systems involving gauge theories, and are extremely explored and utilised in the context of string theory (cf. [1, 2] for an overview). Such dualities provide two descriptions – often two Lagrangians with distinct sets of fields and associated couplings – of the same dynamical system. The difference between these effective field theories is that they describe certain properties of the system, i.e. correlation functions, in a more efficient way.

The efficient calculation of correlation functions or estimates of them based on sample data is also relevant in typical data

P. Betzler, Dr. S. Krippendorf
Arnold Sommerfeld Center for Theoretical Physics
Ludwig-Maximilians-Universität
Theresienstraße 37, München 80333, Germany
E-mail: sven.krippendorf@physik.uni-muenchen.de

P. Betzler
Max-Planck-Institut für Physik
Föhringer Ring 6, München 80805, Germany

 The ORCID identification number(s) for the author(s) of this article can be found under <https://doi.org/10.1002/prop.202000022>

© 2020 The Authors. *Fortschritte der Physik* published by WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim. This is an open access article under the terms of the Creative Commons

Attribution-NonCommercial-NoDerivs License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

DOI: 10.1002/prop.202000022

science applications such as classification. Here, we present examples of data questions where dualities prove to be useful (cf. Section 2). For simplicity we restrict ourselves at this stage to data questions in physical systems where we know a useful dual description. This has the added benefit that the results can be compared to interpretable solutions. We show that the classification with ‘simple’ standard network architectures works much better for data in the dual representation. Better accuracy is achieved in the dual frame with less training effort.

We then show that finding a similar level of classification is not easily

possible, i.e. by examining several standard changes to the architectures such as wider and deeper networks. In particular, this includes architectures which in principle have the capability to perform the duality transformation. We find that the network generically does not find this beneficial configuration. As a next step, we then explore opportunities how to enforce such dual representations, beyond a ‘trivial’ enforcing of dual variables when the duality transformation is known (cf. Section 3). In particular, we find positive results when we demand feature separation in the latent space. We also identify good representations with a modified autoencoder structure where we put an additional constraint (good performance on simple classification tasks) on the latent dimension. Finally we provide and exemplify a method how to enforce certain distributional properties of the dual representation. These representations found by the networks are the first examples where dual representations are obtained without the network “knowing” them a priori.

Before concluding, we comment on the connection to other dualities in physics (cf. Section 4).

2. Benefits of Dual Representations

Here we present several examples where dualities prove useful to address supervised classification tasks.

2.1. Discrete Fourier Transformation

The Fourier transformation captures the essence of many dualities relating strongly-coupled and weakly-coupled field theories (cf. also Section 4). Strongly coupled theories feature non-vanishing correlations over large distances whereas weakly coupled theories only feature seizable correlations at short distances. This is resembled in Fourier transformation, where a delta-peak in momentum space is spread out over all of position

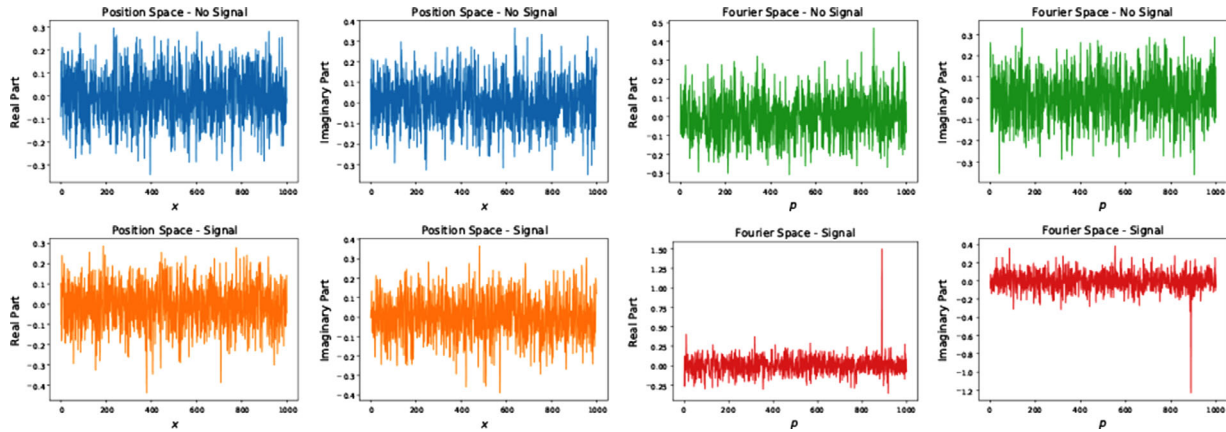


Figure 1. Comparison of noisy signals and pure noise in position and Fourier space.

space. *When is it useful to use position or momentum space representations?* A simple example is given by identifying whether there is a signal hiding under Gaussian noise. For concreteness we consider a signal which is a single peak in momentum space. An example of the data for each class in this binary classification problem is shown in **Figure 1** and the details of the construction and our neural networks and numerical experiments can be found in Appendix A.

When performing classification with a simple neural network¹, we find that a classification is possible for the data in the momentum representation (test accuracy 0.9835) but not for the position representation (test accuracy at pure guessing ~ 0.5).

When adding a single or several hidden dense layers to the position space network, we find only a marginal improvement (again details can be found in Appendix A). As the reached performance does not even come close to the perfect score in the momentum space representation, it is clear that our deeper neural networks are not adapting the position space representation.

2.2. 2D Ising Model

A very well-known example of duality in physics is that of the high-low temperature duality in the 2D Ising model^[3–5] (cf. also [6] for a review).

This Ising model lives on a $N \times N$ square lattice with periodic boundary conditions. On each lattice site there is a spin degree of freedom s_i , which can take values ± 1 . The Hamiltonian of a given state s in the original description is given by

$$H(s) = -J \sum_{\langle i,j \rangle} s_i s_j, \quad (1)$$

where we take the interaction to be ferromagnetic $J > 0$ and from now set $J = 1$, $k_B = 1$. The partition function of this system at finite temperature T is given by

$$Z(\beta) = \sum_s e^{-\beta H(s)}, \quad (2)$$

where $\beta = 1/T$. The duality in this Ising model is as follows. The partition function $Z(\beta)$ of the above system is related to that of another system at a dual temperature $\tilde{\beta} = -\frac{1}{2} \ln \tanh \beta$ by the dependency

$$Z(\beta) = \frac{1}{2} (\sinh(2\tilde{\beta}))^{-N} \sum_{\sigma} e^{-\beta H(\sigma)}, \quad (3)$$

where the dual spins σ_i also take values ± 1 on a lattice with the same geometry, and the dual system shows the same coupling strength J . This is known as the Kramers-Wannier duality^[3,4] which relates a description at low temperature with long-range correlations (strong coupling) and high temperature with short range correlations (weak coupling).²

Classification of Temperature

When is it useful to use the high temperature and when is it useful to use the low temperature phase? Similar in spirit to the Fourier case, we start with a classification task. In particular we are interested in predicting which temperature a sample is drawn from. Our experimental setup is as follows: We considered a square-lattice Ising model on a 40×40 lattice at temperatures $T = 0.25, 0.5, \dots, 2.25$ and their corresponding dual temperatures. The dataset for each temperature was split into 16000 training samples and 4000 test samples. Networks were then trained to classify states drawn from two datasets according to the respective temperature of the set they were drawn from (binary classification). We chose as architecture a simple convolutional neural network consisting of one 2×2 -convolutional layer with 8 filters and ReLU activation followed by a linear layer with sigmoid activation. The overall performance did not change significantly when increasing the number of layers to up to five and varying the number of filters between 8, 12, 16, and 32. Weights were initialised randomly; training was performed using standard Nesterov Adam optimiser with initial learning rate 0.002 and learning rate decay. No significant

¹ Here we perform a classification with a single Conv1D layer with 4 filters and ReLU activation followed by a Dense layer with a single neuron and sigmoid activation. Details on the experiment can be found in Appendix A.

² The fact that both partition functions describe the same type of Ising model implies the existence of a critical temperature $\beta_{\text{crit}} \approx 0.4407$ at which a transition between ordered and disordered phases occurs.

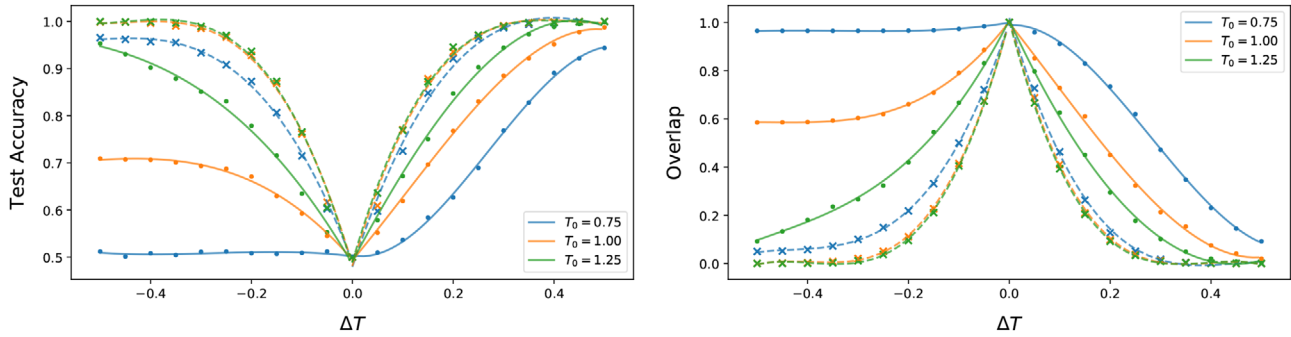


Figure 2. Classification of states according to their temperature in the square-lattice Ising model. Solid lines and dots indicate data for the original temperatures, dashed lines and crosses for the dual temperatures. Pairs of temperatures $T_0, T = T_0 + \Delta T$ were chosen by fixing a reference point T_0 and gradually increasing ΔT by increments of 0.05.

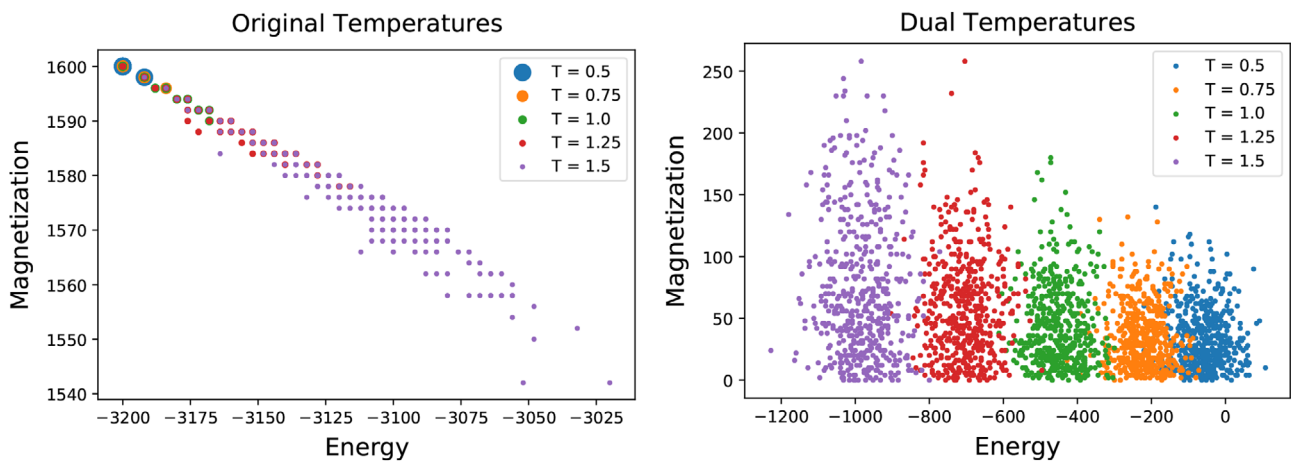


Figure 3. Distribution of energies and magnetizations of a square-lattice Ising model for various temperatures and their duals.

changes in performance were observed after a maximum of 200 training epochs.

Dataset generation and training was performed for ten different seeds to prevent outliers in performance from distorting the results. The best test set accuracies reached after 200 epochs were then averaged over the ten test-runs. The average best test set accuracies for various pairs of temperatures are shown in **Figure 2**.

As can be seen, the classification performance improves substantially when performed for the dual temperatures. This can be seen when visualising the energy and magnetisation for both representations, cf. **Figure 3**. An example of the overlap in the energy distributions for temperatures $T_1 = 1.0$ and $T_2 = 1.25$ is shown in **Figure 4**. The correlation with the classification performance and the overlap of the energy distributions is shown in **Figure 2**.

Further uses could be looked for in determining other correlation functions. In particular, we investigated several disorder correlation functions, e.g. correlators of the type $\langle \sigma_i \sigma_j \rangle$. However, as the performance difference between the two representations are not as dramatic as in the temperature classification we leave a detailed discussion of these correlators to the future.

2.3. 1D Ising Models

Other lattice systems offer different types of dualities, and here we present an example where the dual representation features a different Hamiltonian, i.e. there is no self-duality of the same system. Simple examples of this type of duality are given in the context of one-dimensional Ising models on a finite spin-chain with N spins, n -spin interactions and free boundary conditions. A discussion of such systems can be found for instance in [7], and we summarise here the important system properties for our sub-sequent analysis.

For n -spin interaction models, the Hamiltonian $H(s_1, \dots, s_N)$ takes the form

$$H(s) = -J \sum_{k=1}^{N-n+1} \prod_{l=0}^{n-1} s_{k+l} - B \sum_{k=1}^N s_k. \quad (4)$$

The free boundary conditions are to be understood in the sense that one considers only interactions of n -spin chains which can be fully embedded into the system (s_1, s_2, \dots, s_N) , and there are no identifications or interactions connecting both ends of the chain. Furthermore, there do not exist any relations which fix the values of boundary (or other) spins to specific values.

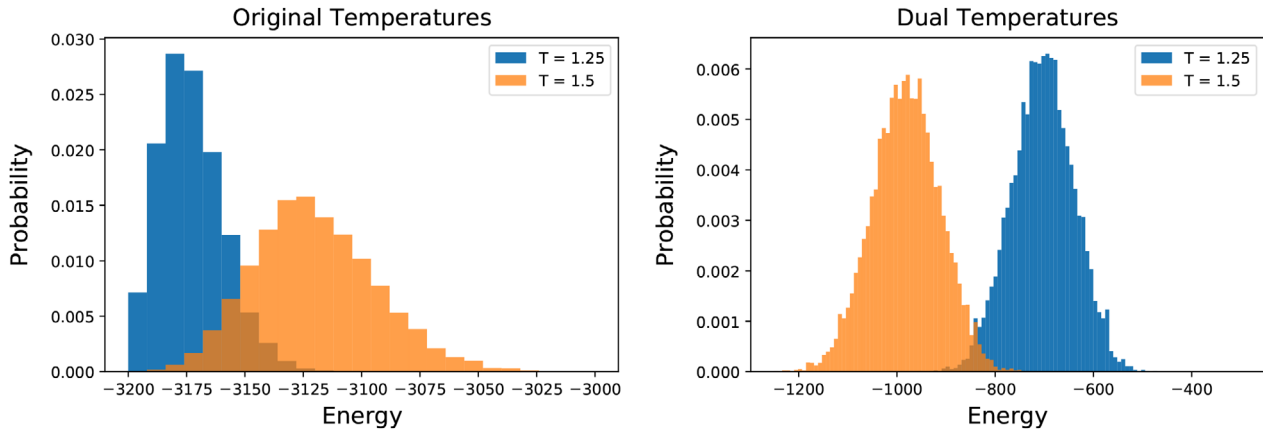


Figure 4. Energy distributions of the square-lattice Ising model for $T = 1.25, 1.5$ and their respective dual temperatures. The energies of the original representation concentrate on a very small region and show a significant overlap. Both diagrams use bins of the same width respectively.

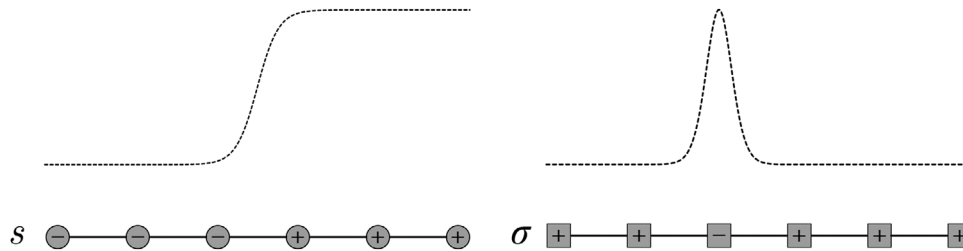


Figure 5. Comparison of spin configurations in a two-spin interaction model and a scalar field kink. Dual spins located on the interaction links represent the energy distribution of a “kink” in the spin model.

Let us now consider the special case of a purely interacting theory with $B = 0$. The Hamiltonian then reduces to

$$H(s) = -J \sum_{k=1}^{N-n+1} \prod_{l=0}^{n-1} s_{k+l}, \quad (5)$$

This can be bijectively mapped to a non-interacting theory with external field J and Hamiltonian

$$H(\sigma) = -J \sum_{k=1}^{N-n+1} \sigma_k. \quad (6)$$

The corresponding duality transformation exchanges the roles of the spins and their interaction terms,

$$\sigma_k = \prod_{l=0}^{n-1} s_{k+l}, \quad k = 1, \dots, N, \quad (7)$$

where spins s_l with $l > N$ are to be understood as ghost spins taking the fixed value 1. The inverse transformation is given by

$$s_k = \prod_{r=0}^q \sigma_{k+r} \sigma_{k+r+1}, \quad (8)$$

where q is to be chosen as the maximum value such that $k + qn \leq N$ and one again introduces a ghost spin $\sigma_{N+1} = 1$ (further ghost

spins can be introduced to generate representations of the same dimension, but they do not play any role in the inverse transformation). For n -spin interactions, the product runs over pairs of adjacent spins, starting from the position k and skipping $n - 2$ spins between the individual pairs. The involvement of spins in the duality transformation (7) and its inverse (8) is exemplified in **Figure 6** for the case $N = 10$ and $n = 3$. Notice that this can be considered a direct generalisation of the special case $n = 2$, for which the the duality transformation corresponds to an exchange of roles between the original spins and their kink variables (cf. **Figure 5**).

Identifying (Meta-)Stable States

Which task is more easily addressed in the dual representation? A simple example for this would be to compute the total energy of a given spin configuration (s_1, \dots, s_N) , which can involve high-order products in the original frame and simplifies to summing over the first $N - n + 1$ spins in the dual frame. Of course, this is more of an ad-hoc example since the duality transformation by construction computes the local energy contributions.

Generally speaking, there exist more sophisticated tasks where no such hand-crafted frame can be constructed. These tasks also can be drastically simplified by applying duality transformations known from or learned in a different context.

One such instance is the detection of states s which are (meta-)stable with respect to single-flip spin dynamics. Such

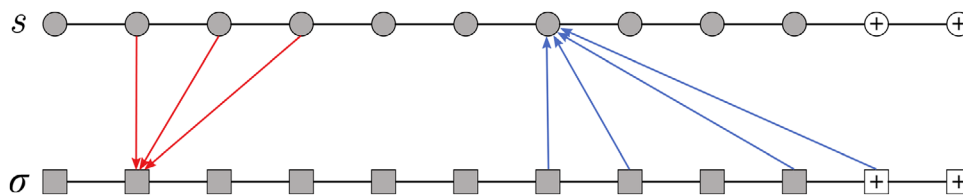


Figure 6. Structure of the duality mappings (7) (red) and the inverse duality mappings (8) (blue) for $N = 10$ and $n = 3$. White spins with “+”-sign inside indicate ghost spins with fixed value 1

Table 1. Detection of (meta-)stable states in the 1D Ising chain for different interactions and amounts of training data. The listed numbers describe the average best test accuracy over 10 training runs of 500 epochs each. Missing values indicate that the number of required samples exceeds the total number of metastable states for the considered setting. On the left are the results for the normal variables, and the right side shows the results for the dual variables.

normal	$n = 4$	$n = 5$	$n = 8$	$n = 9$	$n = 12$	dual	$n = 4$	$n = 5$	$n = 8$	$n = 9$	$n = 12$
$6 \cdot 10^2$	0.9113	0.8688	0.8788	0.8813	0.8803	$6 \cdot 10^2$	0.9911	0.9783	0.9819	0.9855	0.9909
$3 \cdot 10^3$	–	0.9243	0.9215	0.9223	0.9295	$3 \cdot 10^3$	–	0.9958	0.9977	0.9994	1.0000
$9.5 \cdot 10^3$	–	–	0.9424	0.9475	0.9739	$9.5 \cdot 10^3$	–	–	1.0000	1.0000	1.0000

single-flip stable states are defined as configurations for which flipping any of the spins causes the energy of the system to increase.³

Effect on Simple Networks

In order to get an idea whether the duality (7) is a viable tool to improve the classification of metastable states, we choose as a first benchmark how “simple” architectures of neural networks can handle this classification problem and whether transforming our variables to the dual frame can improve their performance. While, in practice, any improvement from utilising the dual frame might also be achieved by using more sophisticated architectures, this setting nevertheless serves as an important first step. A positive result justifies a further scrutinising whether the same principles also hold for tasks which state-of-the-art models fail to solve.

Since the duality transformations (7) are themselves highly nontrivial from the perspective of computational complexity, some caution is needed here to prevent distorting our results by limitations arising from a mere lack of capacity. Taking, for instance, our toy-example of energy regression, it is clear that the task cannot be solved by a linear network in the normal frame, while even a simple perceptron with sufficiently high number of neurons can do so at ease. In this case, the only benefit coming from using the dual frame thus lies in a lower network complexity, which is, however, in parts nullified by the computational complexity of the duality transformation itself.

Taking this into account, we chose a suitable benchmark for our tests a single-layer perceptron with 128 hidden neurons, ReLu activation for the hidden layer and sigmoid activation for the output layer. This architecture shows a sufficiently-high capacity to easily learn the transformation (7) directly, while at the same time keeping a relatively simple structure.

We generated all 2^{18} states for the 1D Ising chain with $N = 18$ spins and tested different networks for varying n . We split the data into states labeled as “not (meta-)stable” (0) or “(meta-)stable” (1) and normalised the training and test sets to

contain an equal number of samples for each class. We furthermore checked the performance for varying amounts of training data in order to properly analyse effects on generalisation errors and data efficiency.

The average best test accuracies and losses achieved in 10 training runs of 500 epochs are listed in Table 1. Average training curves for the case $n = 8$ and varying amounts of training data can be found in Figure 7. Further details on the training and testing modalities are discussed in Appendix B.

Results

The results show that there is indeed a major improvement of performance in the dual representation. While all networks are able to detect at least some patterns in either frame, we find several advantages from using the dual representations:

- The best performance achieved for low numbers of training samples is notably higher in the dual representation, implying that the duality transformation (7) can be useful to prevent overfitting and improve data efficiency.
- While increasing the amount of training data gradually tightens the performance gap between the original and dual representations, the learning curves in the latter remain much steeper in all cases, leading to shorter and more stable training.
- Even in cases for which the best test accuracies are high in both representations, there remains a significant difference in the actual binary cross-entropy,

$$\mathcal{L} = -[\gamma_{\text{true}} \log(\gamma_{\text{pred}}) + (1 - \gamma_{\text{true}}) \log(1 - \gamma_{\text{pred}})], \quad (9)$$

³ Such metastable states can cause standard MCMC-algorithms to be trapped in a local minimum as the temperature approaches zero and is a major reason why the performance of common simulation algorithms tends to deteriorate at low temperatures.

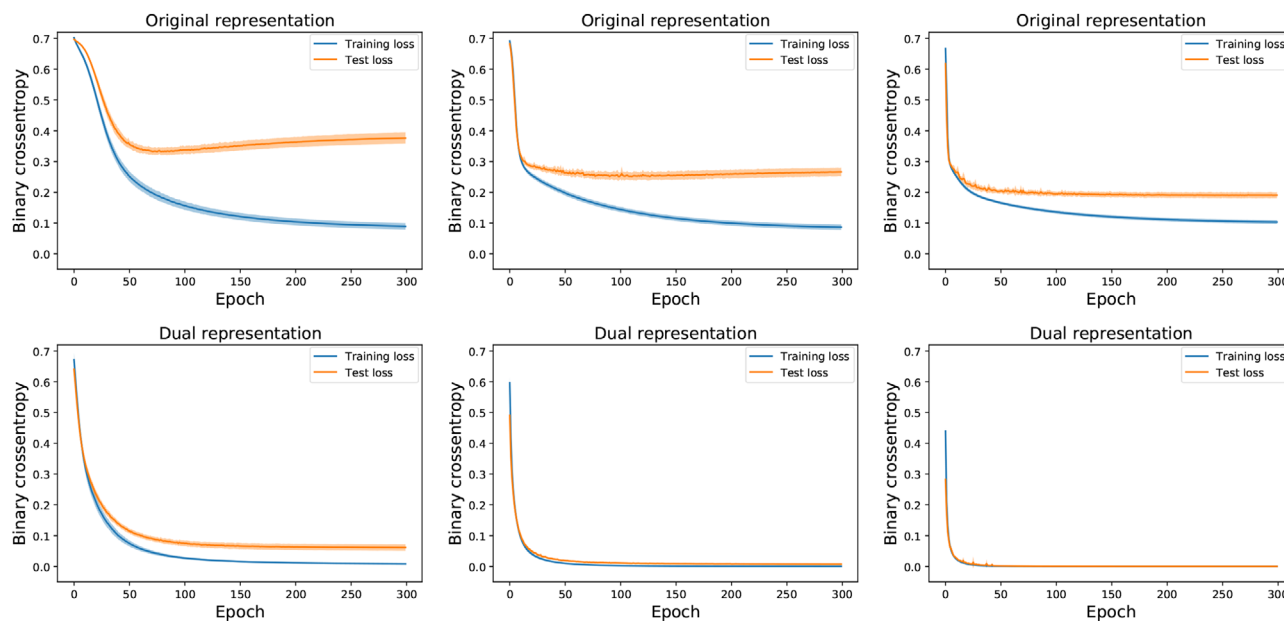


Figure 7. Example histories of training loss (blue) and test loss (orange) over the course of 300 epochs for $n = 8$ and various numbers of training samples. The plots show averaged curves computed over ten test-runs; standard deviations are indicated with shaded colours.

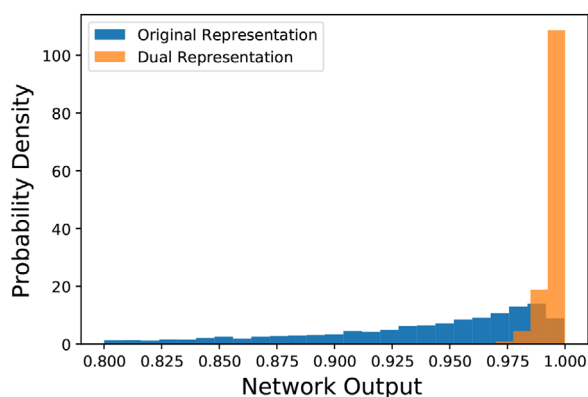


Figure 8. Output distribution of simple neural networks for states classified as (meta-)stable for $N = 18$ and $n = 8$. Both networks were trained on 3000 samples. Only values for the dual representation accumulate very close to one, implying a higher degree of certainty in this frame.

implying that networks trained on the dual representation perform classifications with a considerably higher degree of certainty. This is also reflected in the model outputs, which are commonly closer to 0 or 1 in the dual representation than in the original variables, even in settings with high test accuracies in both representations (cf. **Figure 8**).

- While overfitting is prevalent in the original representation, the loss curves additionally show signs of underfitting. This can be remedied by increasing the capacity of the network, which, however, leads to even stronger overfitting. We found that regularization techniques can slightly improve performance in this case, however, there remained a significant difference between both representations for all tested methods. Details on this are discussed in Appendix B.

Interpretation

Some sense can be made out of this result when addressing the problem from a naive analytical viewpoint. In the original representation, checking whether flipping a particular spin s_i increases the total energy of the system requires taking into account n interaction terms containing s_i , some of whose contributions might cancel each other. On the other hand, these n interaction terms are represented by a cluster of n spins σ_j , $j = i - n + 1, \dots, i$ in the dual frame, and flipping s_i causes all of those n dual spins to change sign. Since the total energy of the system can be computed by simply adding up the first $N - n + 1$ dual spins of the complete system, an overall increase in energy then occurs precisely iff more than half of the flipped dual spins take the value 1 (not counting those spins σ_j with $j \geq N - n$). In other words, the transformation (7) maps the single-flip dynamics of the original system to n -spin-cluster dynamics in the system governed by the Hamiltonian (6), thus creating a “dual task” which is considerably easier to learn for neural networks. An illustrative example for the case $N = 10$ and $n = 3$ is given in **Figure 9**.

Discussion and Limitations

There are several important aspects as well as limitations of the considered experimental settings, which shall be briefly commented on in this subsection.

• Modifications to Setup

A first important point to remark is that the discussed setting describes a very low number of spins and is therefore to be understood as a toy model. While a large-scale simulation of realistic systems is beyond the scope of this work, it is worth mentioning that we found a more drastic difference in

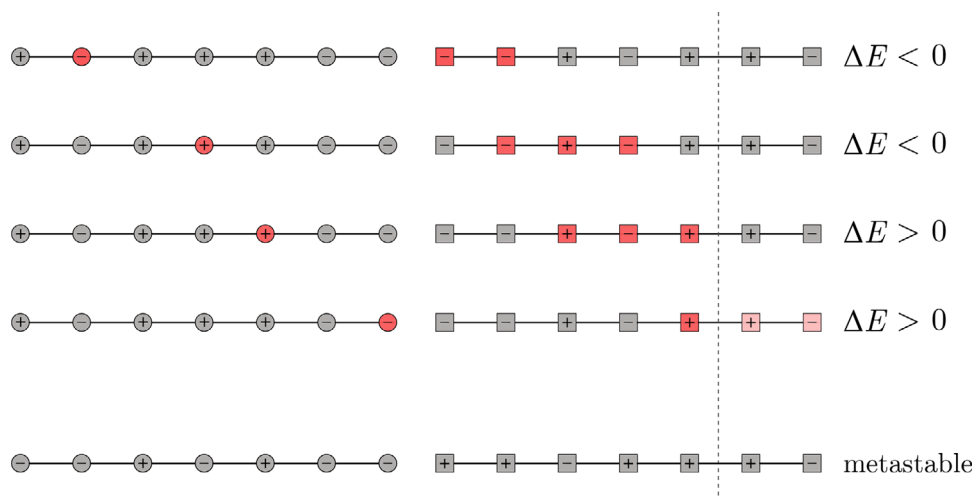


Figure 9. Single-spin flip dynamics and metastability in the normal and dual representation. **Top:** Flipping a single spin s_i in the normal representation (left) causes n dual spins σ_j with $j = i - n + 1, \dots, i$ to change sign (right), as indicated by red color for the case $n = 3$. The overall energy increases iff more than half of the involved dual spins have positive sign (counting only values j with $1 \leq j \leq N - n + 1$). **Bottom:** Example of a metastable state for $n = 3$ in the normal (left) and dual (right) representation. Flipping any of the spins in the original representation causes the overall energy of the state to increase.

performance as more complex settings such as $N = 100$ and $n = 50$ were considered. This commonly led to pure guessing on the original data, whereas accuracies higher than 0.95 could be reached with as few as 1500 training samples in the dual representation. The benefits of dualities might thus extend beyond simple toy-settings, however, further testing is required to confirm this.

• Sensitivity to Architecture

While the above tests were performed for a rather large number of different systems and training set sizes, defining a clear benchmark naturally required the utilization of a fixed model to test performance. In light of this, a natural question is to which degree the improvement is owed to the choice of architecture, and whether the results remain valid if a wider class of architectures is considered. We therefore checked the effect of various modifications on our results, as described in more detail Appendix B.

We found that, except for strong results of convolutional neural networks on very simple systems with $n \leq 4$, none of the above modifications led to a significant change in the overall results. It cannot be excluded that a similar improvement in performance can alternatively be obtained by more sophisticated network architectures. However, our tests clearly demonstrate that the benefits of the dual representation is not isolated to our experimental setting, but does extended to a wider class of architectures.

• Avoiding Shortcutting Predictors

Since the dual representation by definition describes the spin system in terms of its local energy contributions, there is one particular pitfall here which has to be treated with caution: (Meta-)stable states commonly accumulate at low energies, and relatively high accuracies in our classification task can be obtained by simply choosing a fixed energy cutoff to label states as “(meta-)stable” (see Table 2 and Figure 10). In such situations, a neural network can be prone to adopting shallow

Table 2. Classification accuracy for (meta-)stable states using only a fixed energy cutoff (cf. Figure 10).

	$n = 4$	$n = 5$	$n = 8$	$n = 9$	$n = 12$
Energy cutoff	0.9925	0.9605	0.9535	0.9269	0.8985

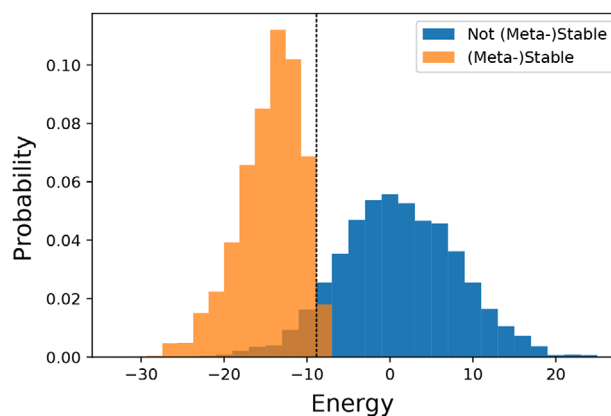


Figure 10. Energy distributions of normal and (meta-)stable states for $N = 100$ and $n = 50$ (choice for illustrative reasons). Relatively high accuracies can be obtained by choosing a fixed energy cutoff for classification (dashed line).

heuristics which perform well in many cases (in this case the total energy) instead of learning the actual task it is supposed to solve.

We found, however, that the networks trained for the settings listed in Table 1 do not rely purely on the lower energy of metastable states, and the difference in performance remains the same if tested in low-energy regions where the ratio of each class is roughly the same. Going one step further, the element

of energy can be eliminated completely by additionally training only on states with fixed energies. While this drastically tightens the performance gap in simple settings, a similar difference as before remains at more complex settings like $N = 100$ and $n = 50$.

3. Enforcing Good Representations

Having established that dual representations can be ‘beneficial’ for classification tasks, we now turn to the question how such representations can be adapted by the network dynamically. When a duality map is known explicitly, it could easily be learned by a neural network with appropriate regression. Although this can be of interest in principle, we here focus on unsupervised learning techniques for adapting dual representations.

To do this, we discuss three different training strategies which we find to lead to ‘dual-like’ representations:

1. Feature separation in the latent space.
2. An autoencoder setup with an additional latent loss. In this case, the output of the encoder is the dual-like representation.
3. Demanding properties of the dual representation, for instance that it resembles the correct energy distribution.

3.1. Feature Separation

For the discrete Fourier transform described in Section 2.1 and Appendix A, the momentum space defines a valuable data representation in which the previously infeasible task of detecting signals in noisy data becomes easy to solve. Based on our finding that deeper networks do not adapt this representation (cf. Appendix A), we now pursue the question how one can assist the neural network to find such a beneficial representation without knowledge about its explicit form.

Basic Idea and Motivation

Heuristically, the benefit is likely to come from the information of a non-localised signal in the space domain being collected in one single (complex) bin of the momentum space domain. This causes the signal in the momentum space domain being clearly separated from the background noise, which takes the same non-local form in both frames (cf. again Figure 1).

Can this “feature separation” be exploited to automatically learn such favourable representations without analytic knowledge about the structure of the signal? Assuming for the moment that there exists only one non-vanishing frequency, we would like to train a neural network to find a representation in which the outputs for pure signals and pure noise satisfy

$$|Y_{\text{signal}}|^2 - |Y_{\text{noise}}|^2 \geq \alpha. \quad (10)$$

Here, $\alpha > 0$ denotes a margin where we want to push the latent representation. Formulated as a loss function, at values larger than α , this function shall take the value 0, which avoids a run-away of the signal (vanishing gradients). Notice that this task re-

sembles the minimisation of a triplet loss,^[8,9] with the location of the noise fixed at zero. To apply this strategy to a setup with $N = 1000$ different frequencies, two aspects have to be taken into account:

1. The relation (10) should be satisfied for any frequency.
2. The information of different frequencies should be collected at different locations. Otherwise, the mapping might not be able to distinguish between clear signals and “noisy” inputs with small components in many different frequencies (as is the case for the background noise in our setting).

A viable ansatz to achieve this is by defining a loss function

$$\mathcal{L} = \max(0, \alpha - (\xi_1^2 + \xi_2^2)), \quad (11)$$

where ξ_1^2 and ξ_2^2 are defined as the two largest squared values of the $2N$ outputs for a given input sample. When using pure single-frequency signals as training data, this loss effectively urges the sum of only the two output components with largest absolute value to be pushed away from zero until the margin α is reached. The aim of this is to enforce a data representation similar to the actual Fourier transform, in which all information of the single-frequency signals is concentrated in the real and imaginary parts of the p_k .

At the same time, we keep the complexity of the network as low as possible (in this case linear). This is necessary because the loss (11) alone does not prevent the occurrence of representations in which an arbitrarily large number of bins is maximised for any frequency. As a consequence, enforcement of sparse and local representations of signals would not take place. In practice, such cases of “overfitting” are possible for any network architecture, however, we observe that they commonly occur at higher degrees of complexity, whereas the constrained parameter space of low-capacity networks seems to act as an efficient preventive measure. Somewhat remarkably, this heuristic approach clearly outperformed more elaborate methods such as forcing sparse outputs via L1 penalty or penalising for correlation of latent variables.

Note that the network has no further knowledge on the structure of Fourier transformation or the structure of background noise.

Performance and Structure of Representation

Training a linear network with $2N$ output nodes with Nesterov Adam optimiser, learning rate $1 \cdot 10^{-3}$ and $\alpha = 5$ commonly led to close-to-zero losses after less than five epochs. As can be seen in **Figure 11**, the learned representation shows characteristic properties of the actual Fourier transform when we trained just with noisy signals as input. Using this representation for our previous task of signal detection in noisy data, the mean best test accuracy of the same simple one-layer convolutional neural network as described in Section 2.1 (cf. also Appendix A for more details) indeed improved to around 0.7717.

Interestingly, the learned data representations often take the form of transformations such as rescalings, reflections or rotations of the actual Fourier transform in the $2N$ -dimensional

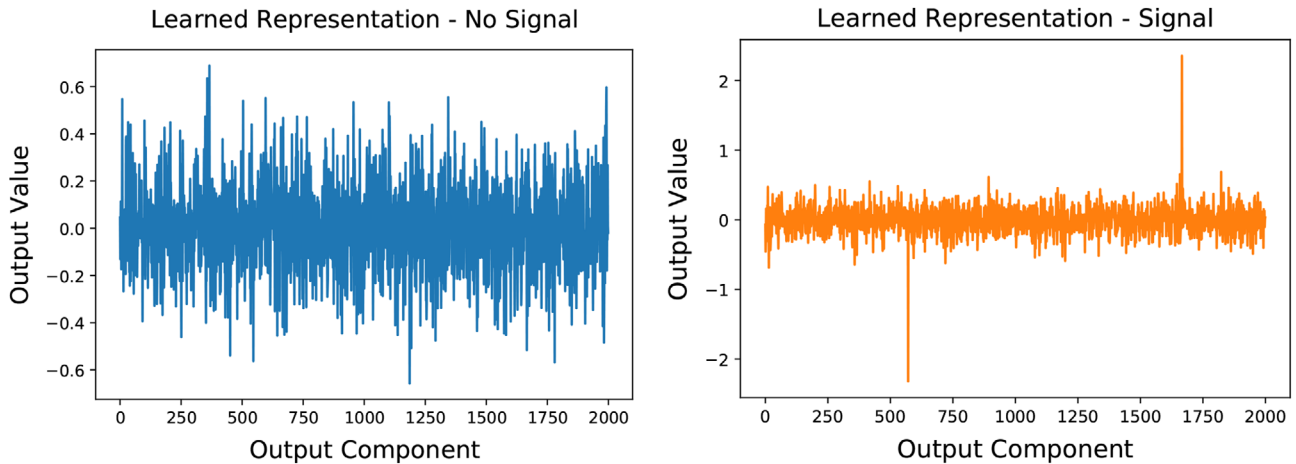


Figure 11. Output of the feature-separation network for pure noise and noisy signal.

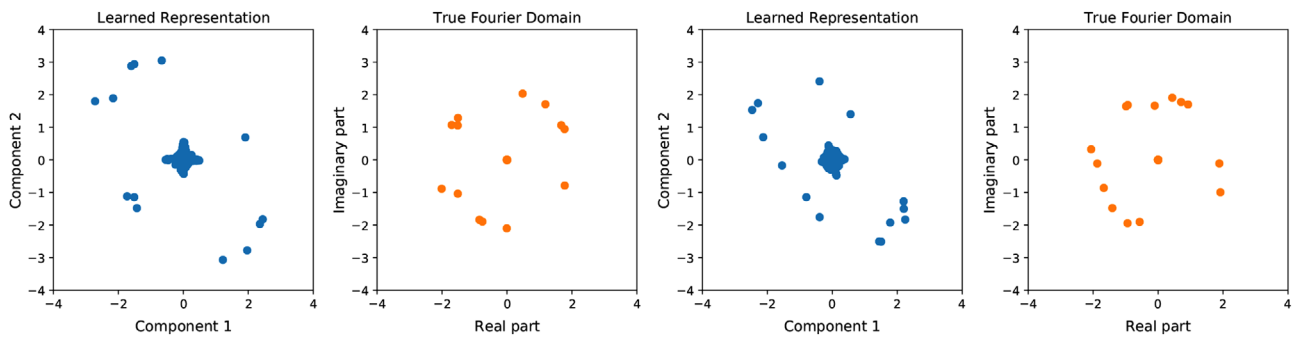


Figure 12. Comparison of representations learned via feature separation and embedding into true momentum space domain. The above plots show examples of learned representations and Fourier transforms of single-frequency signals at different frequencies without noise. Signals with non-vanishing component in the respective frequency arrange in similar shapes, while the rest accumulates close to or at the origin.

space. Projecting the output of the network for a large number of samples onto particular pairs of components, the distribution of values then corresponds to that of the real and imaginary parts of a certain value p_k in the Fourier domain. This is exemplified for two instances in Figure 12.

Response to Single-Frequency Signals

Some more insights into the structure of the feature-separation network can be gained by analysing its outputs $f_j(x)$. Here, we do this by analysing the $2N$ response values $f_j(x)|_{p_i \neq 0}$ when given pure signals with single non-vanishing frequency p_i . These can be stored a $N \times 2N$ response matrix

$$M_{ij} = \langle |f_j(x)|^2 \rangle \Big|_{p_i \neq 0}, \quad (12)$$

where the mean is taken over all samples satisfying the condition $p_i \neq 0$. The matrix generally shows a high degree of sparsity, and we find that a fraction of higher than 0.8 of all rows contain at least one large value, implying that the network makes efficient use of the $2N$ dimensions to embed the signals into the latent space. An example plot of the matrix M for the case $N = 100$

can be found in Figure 13. It can be observed that each row of the matrix commonly contains between 2 and 4 large activations, with the remaining entries being close to zero. Visualising the corresponding latent dimensions, one finds that this behaviour reflects precisely the way in which the Fourier-transform is embedded into the latent space. This is exemplified for various cases in Figure 14.

3.2. Autoencoder with Latent Loss

We now turn to the second example of adapting an appropriate latent dimension dynamically which is based on the 1D Ising setup already described in Section 2.3.

Motivation and Architecture

We have seen that by exchanging the roles of individual spins and their interaction terms, the task of detecting (meta-)stable states becomes more accessible due to the relevant information being easier to extract from a lower number of spins in the dual frame. To find such a suitable representation, we here employ the following strategy: We use the fact that a simple task can be

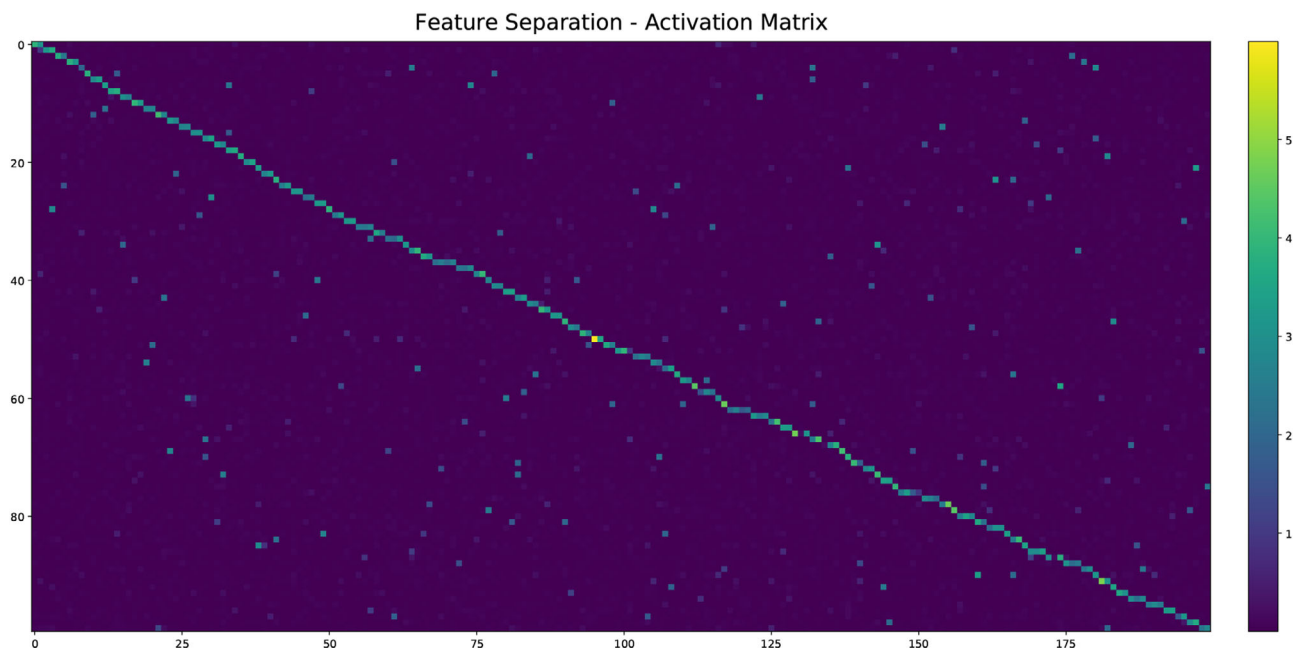


Figure 13. Example plot of an activation matrix (12) for the case $N = 100$. The columns have been reordered according to the indices of their respective largest entries. The number of non-vanishing values in a given row matches with the dimension of the subspace of the $2N$ -dimensional latent space into which the representation of signals with a corresponding non-vanishing frequency p_i is nontrivially embedded (cf. Figure 14).

performed very efficiently in the dual representation. In this case this is the (trivial) task of energy classification.

By itself, this is not sufficient and we need to ensure that no information is lost in the latent representation. A viable method to achieve this goal is to use an autoencoder-like architecture whose ‘bottleneck’ has (at least) the same dimension as the original input and is required to represent the data in a way that the total energy can be extracted by a simple linear model. This way, the model is guaranteed to learn a representation which encodes the energetic properties of a state in a manner similar to the dual frame (cf. Equation (7)), while at the same time the presence of an additional reconstruction loss forces the mapping to be information conserving.

In practice, this can be implemented by training a neural network to map an input state s_1, \dots, s_N to an intermediate output of (at least) the same dimension, which in turn serves as input for a linear model extracting the total energy of the input state and another network reconstructing the initial input configuration. Figure 15 illustrates this architecture schematically.

3.2.1. Results and Discussion

We tested the performance in classifying (meta-)stable states using the same setting as before, with the duality transformation (7) replaced by the intermediate output of a constrained autoencoder with latent dimension 18 and 50. Details on the experimental conditions are provided in Appendix B; results are shown in Table 3.

One again observes a significant improvement compared to the original representation (cf. Table 1, left), albeit not as drastic as in the actual dual representation. Autoencoders with latent dimension 18 often suffered from underfitting problems,

and further benefits were possible when increasing the latent dimension to 50. Networks trained on the learned representation mostly outperformed accuracies reachable by pure energy cutoffs in particular at latent dimension 50, but showed a slight tendency to misclassify samples which are located in energy regions dominated by the respective other class. While part of the improvement might therefore be attributed to the correlation between overall energy and (meta-)stability, the learned representation still allows to solve the classification task significantly better than by training on the original representation directly, and the networks do not resort completely to superficial energetic arguments.

Further Applications

Let us conclude this discussion by stressing that the main purpose of the above architecture is to realise transfer learning between different physically related problems. This can be beneficial when training data is limited or expensive to generate for one task but can be efficiently acquired for a simpler task. In such cases, it might not be a reduction of required overall training data, but rather a change in the type of data that eventually leads to an improvement in overall performance.

In our considered setting, we indeed found that benefits in performance are only possible when the constrained autoencoder is trained on relatively large datasets. While this obviously nullifies the improvement in overall data efficiency of analytical dualities, it can simplify the process of training due to the possibility to replace large datasets of metastable states (which might not even exist for some settings) by corresponding pairs of random states and their energy.

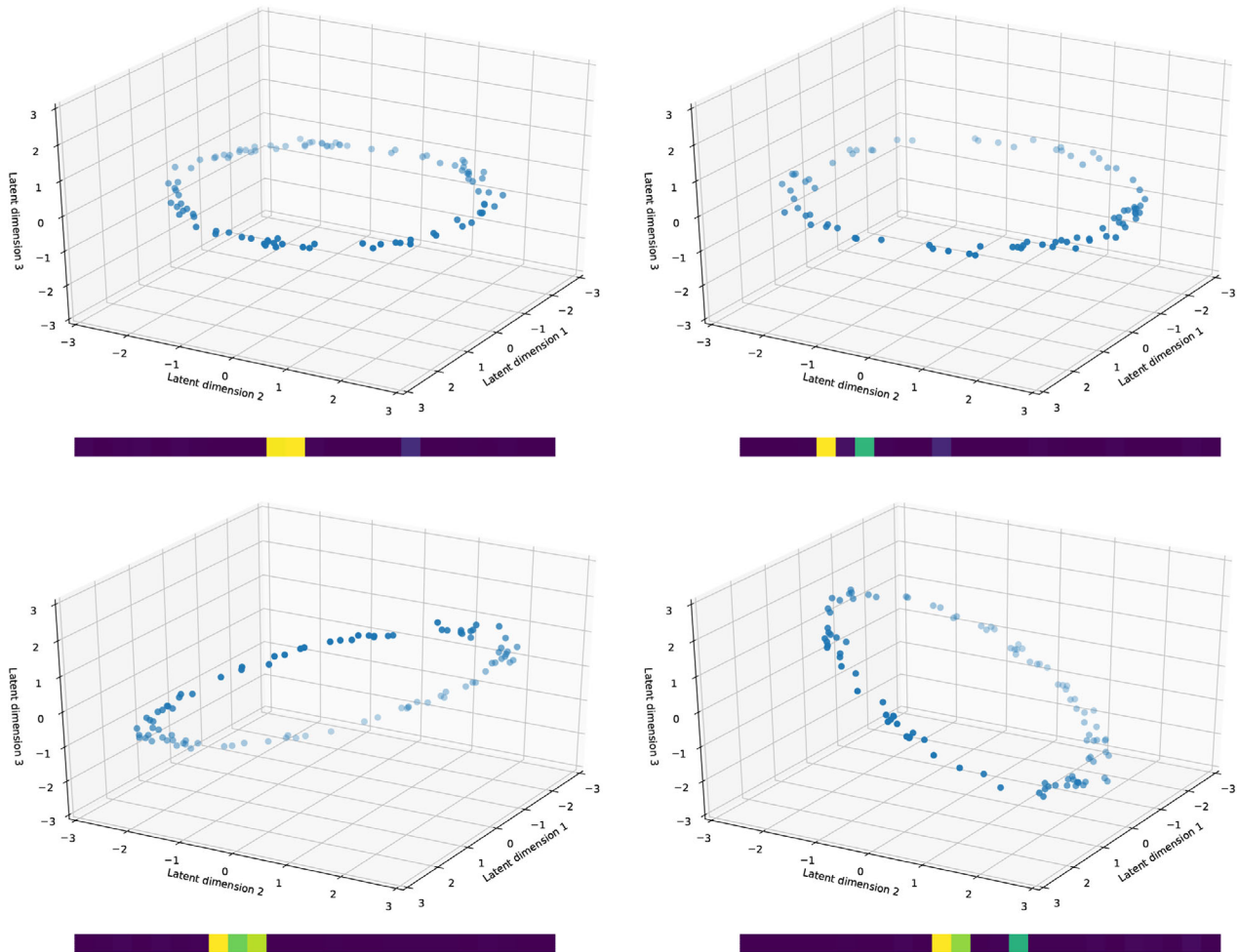


Figure 14. Interpretation of the activation matrix illustrated in Figure 13. The plotted latent dimensions correspond to the three largest entries of a given row. **(Top)** Two non-vanishing entries in one row. The Fourier transform is completely embedded into two latent dimensions. **(Bottom)** Three non-vanishing entries in one row. The Fourier transform is nontrivially embedded into three latent dimensions.

Table 3. Detection of (meta-)stable states in the 1D Ising chain for different interactions and amounts of training data. The listed numbers describe the average best test accuracy over 10 training runs of 500 epochs each when trained on the intermediate output of a constrained autoencoder with latent dimension 18 **(Left)** and 50 **(Right)**. Missing values indicate that the number of required samples exceeds the total number of metastable states for the considered setting.

lat (18)	$n = 4$	$n = 5$	$n = 8$	$n = 9$	$n = 12$	lat (50)	$n = 4$	$n = 5$	$n = 8$	$n = 9$	$n = 12$
$6 \cdot 10^2$	0.9880	0.9540	0.9180	0.9072	0.9228	$6 \cdot 10^2$	0.9887	0.9526	0.9300	0.9304	0.9500
$3 \cdot 10^3$	–	0.9677	0.9527	0.9353	0.9476	$3 \cdot 10^3$	–	0.9718	0.9787	0.9637	0.9829
$9.5 \cdot 10^3$	–	–	0.9607	0.9500	0.9597	$9.5 \cdot 10^3$	–	–	0.9910	0.9885	0.9968

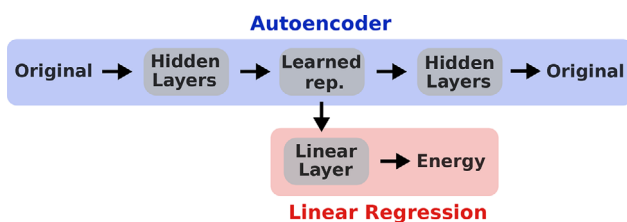


Figure 15. Schematic illustration of a task-constrained autoencoder used to learn suitable representations for difficult tasks. The intermediate output takes the role of the “dual” representation.

Generally, finding such physically related tasks commonly requires domain knowledge or heuristic arguments, but it nevertheless opens up a wide range of new possibilities going beyond known analytical dualities.

Interpretation of Intermediate Output

Before we delve into the interpretation of the intermediate output, it is important to remark that we did not impose any further constraints regarding the structure of the intermediate output as

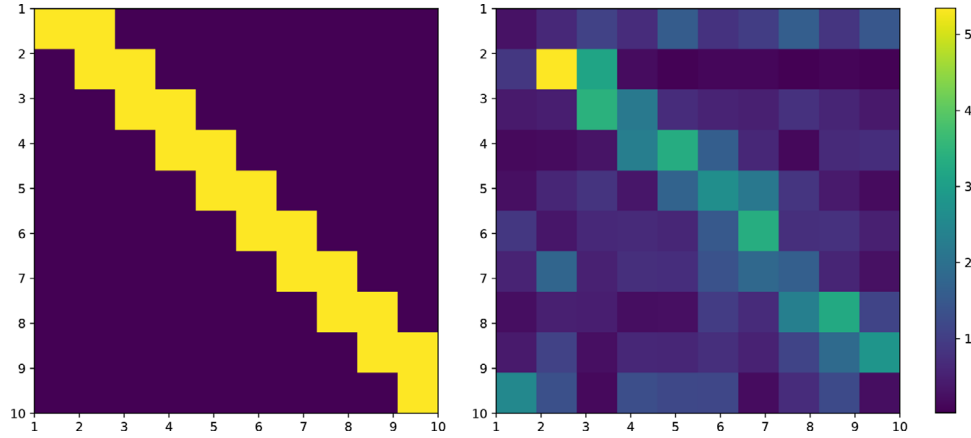


Figure 16. Plots of the sensitivity matrix (13) for the actual duality transformation (left) and a learned representation of a constrained autoencoder (right) for $N = 10$ and $n = 2$. Both matrices show characteristic nearest neighbour interactions; the latter contains additional nonlocal components.

performance commonly suffered from reduced network capacity in such cases. As a consequence, the intermediate output has no obvious physical interpretation and relations to the true dual representation are a priori not obvious.

An interesting question in this context is whether there is some way to make sense of how the relevant information is encoded in our learned representation. A viable way to study dependencies between the input and latent variables is to analyse the sensitivity of the latent variables with respect to flips of a particular spin s_j while keeping all other spins fixed. This information can be stored in the matrix

$$M_{ij} = \frac{\langle (f_i(s_1, \dots, s_j, \dots, s_N) - f_i(s_1, \dots, -s_j, \dots, s_N))^2 \rangle}{\frac{1}{N} \sum_{k=1}^N \langle (f_i(s_1, \dots, s_k, \dots, s_N) - f_i(s_1, \dots, -s_k, \dots, s_N))^2 \rangle}, \quad (13)$$

where the expectation values are to be computed for the complete (test) dataset. Heuristically, this matrix encodes the average sensitivity of the components f_i of the transformed representation with respect to flips of a particular spin s_j , normalised by the average sensitivity of f_i to flips of any spin. For the actual duality transformation (7), the numerator takes precisely the values 0 or 4, leading to a staircase-like structure as depicted on the left hand side in **Figure 16**.

We trained 25 constrained autoencoders for the simple setting $N = 10$ and $n = 2$ and compared the transformation behaviour of the learned variables to that of the true duality transformation (7). Interestingly, there exist many instances of networks with structurally similar dependencies as the proper duality transformation. These commonly include components f_i depending strongly on neighbouring pairs of spins and a distinguished value f_N which is highly sensitive to one particular spin - the matrix M_{ij} for one such example is presented on the right hand side in **Figure 16**.

Notice that this basically represents the way the duality transformations (7) encode the information of the original system in that there exist $N - 1$ terms $\sigma_i, i = 1, \dots, N - 1$ describing the nearest-neighbour interactions and one value σ_N which does not interact with the external field and stores the overall sign of the system.

3.3. Distributional Properties

The next question we analysed is to which degree neural networks are capable of learning the relation between dual Ising models on the square lattice. A minimal requirement for this is that the duality map between the two systems can be learned if samples from both data representations are provided explicitly.

Here we start with no one-to-one mapping between states of a system at temperature T and those of a system at dual temperature \tilde{T} . Instead, we match features of the dual representation on the level of the probability distributions, i.e. that the learned representation shares features with the target dual distribution. For this purpose, we consider the following architecture: States s sampled from the temperature T are used as input for a deep convolutional network and mapped onto a lattice of the same shape whose entries are interpreted as probabilities of the the respective spins to take the value 1.

Binary states are then sampled by utilising the Gumbel trick to preserve differentiability of the network. In the discussed setting, this can be realised by sampling for each site p_i of the lattice some value $\varepsilon_i \sim U(0, 1)$ uniformly and map the input state s to an output state $f(s)$ with

$$f_i(s) = 2 \cdot \text{sig}[\gamma(\log(\varepsilon_i) - \log(1 - \varepsilon_i) + \log(p_i) - \log(1 - p_i))] - 1, \quad (14)$$

where sig denotes the sigmoid function $\text{sig}(x) = \frac{1}{1 + e^{-x}}$ and γ is a scale parameter which can be used to force the output values closer to the extremal values 0 and 1⁴.

The output states $f(s)$ are then fed into a hard-coded layer to compute their total energy, and the loss function is defined as the Kullback-Leibler divergence

$$D_{\text{KL}}(P_f \| P_\sigma) = - \sum_E P_f(E) \log \left(\frac{P_\sigma(E)}{P_f(E)} \right) \quad (15)$$

⁴ Some caution is needed when choosing γ as high values can lead to vanishing or exploding gradients, resulting in poor training.

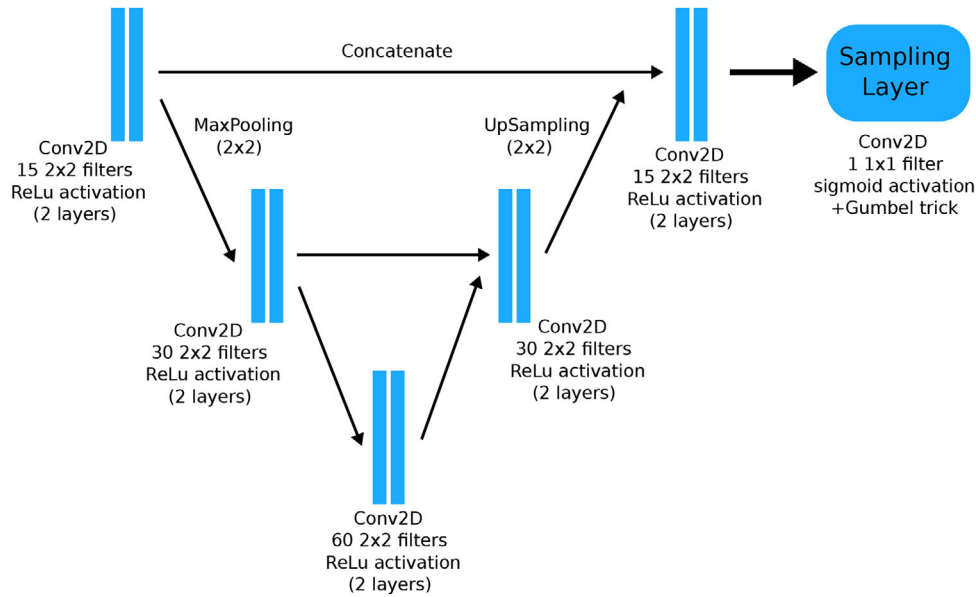


Figure 17. Schematic illustration of a U-Net architecture.

between the energy distributions $P_f(E)$ and $P_\sigma(E)$ of states sampled from the network and the true dual temperature, respectively.

The network produces binary outputs as desired, with the energy distributions closely resembling those of the actual dual system. This is depicted for two examples in Figure 18.

Results

We used a U-Net architecture as depicted in Figure 17 with three levels consisting of two layers of 15, 30 respectively 60 2×2 filters with ReLu activations. The scale parameter in (14) was set to 50. Tests were conducted for a 40×40 lattice at temperatures $T = 0.25, 0.5, \dots, 2.25$ using standard Nesterov Adam optimiser with initial learning rate 0.002 and learning rate decay. The dataset for each temperature was again split into 16000 training samples and 4000 test samples. Training equilibrium was commonly reached within 50 epochs; no significant changes were noticed after 500 epochs. Tests were again performed for 10 random seeds per temperature and showed consistent overall performance, however, there were rare instances in which poor local minima required reinitialization of the network in particular when mapping to lower temperatures.

The network produces binary outputs as desired, with the energy distributions closely resembling those of the actual dual system. This is depicted for two examples in Figure 18.

We next checked the output of U-nets trained on a single temperature for input states sampled from other temperatures. For networks trained on larger original temperatures, the output energy distribution shows some resemblance of the true dual temperatures, albeit with wrong numerical values. This behaviour is shown in Figure 19 for temperature $T = 1.80$. For lower training set temperatures, the networks gradually lose their ability to distinguish between input states.

When we trained the network with data from multiple temperatures, we have not (yet) found a significant improvement compared to Figure 19.

Generally speaking, one can think of extending this method and incorporating more and more properties, i.e. matching more and more correlators. This would lead to a more and more precise map which satisfies more and more properties of the respective dynamical system.

4. Connection to Other Dualities in Physics

We have seen in previous sections that dualities are a change in the basis which describes the system. Although we have already used this in the case of physical systems, such as the 2D Ising model (cf. Section 2.2), we would like to highlight how such a change in the basis appears analytically in physical systems and how it is connected to Fourier transformation. To do this we repeat the key steps from arguments presented for instance in [2].

To do this, one can consider electromagnetism in four dimensions without sources. The path integral is described by

$$\int \mathcal{D}A e^{iS(A)/\hbar},$$

$$S(A) = -\frac{1}{4g^2} \int d^4x (\partial^\mu A^\nu - \partial^\nu A^\mu)(\partial_\mu A_\nu - \partial_\nu A_\mu) \quad (16)$$

This can be re-formulated as a path integral over the antisymmetric tensor field $F_{\mu\nu}$ subject to the constraint that the Bianchi identity $\partial_\mu \tilde{F}^{\mu\nu} = 0$ is satisfied at each point x

$$\int \mathcal{D}F \prod_x \delta(\partial_\mu \tilde{F}^{\mu\nu}(x)) e^{-\frac{i}{4\hbar g^2} \int d^4x F_{\mu\nu} F^{\mu\nu}}, \quad (17)$$

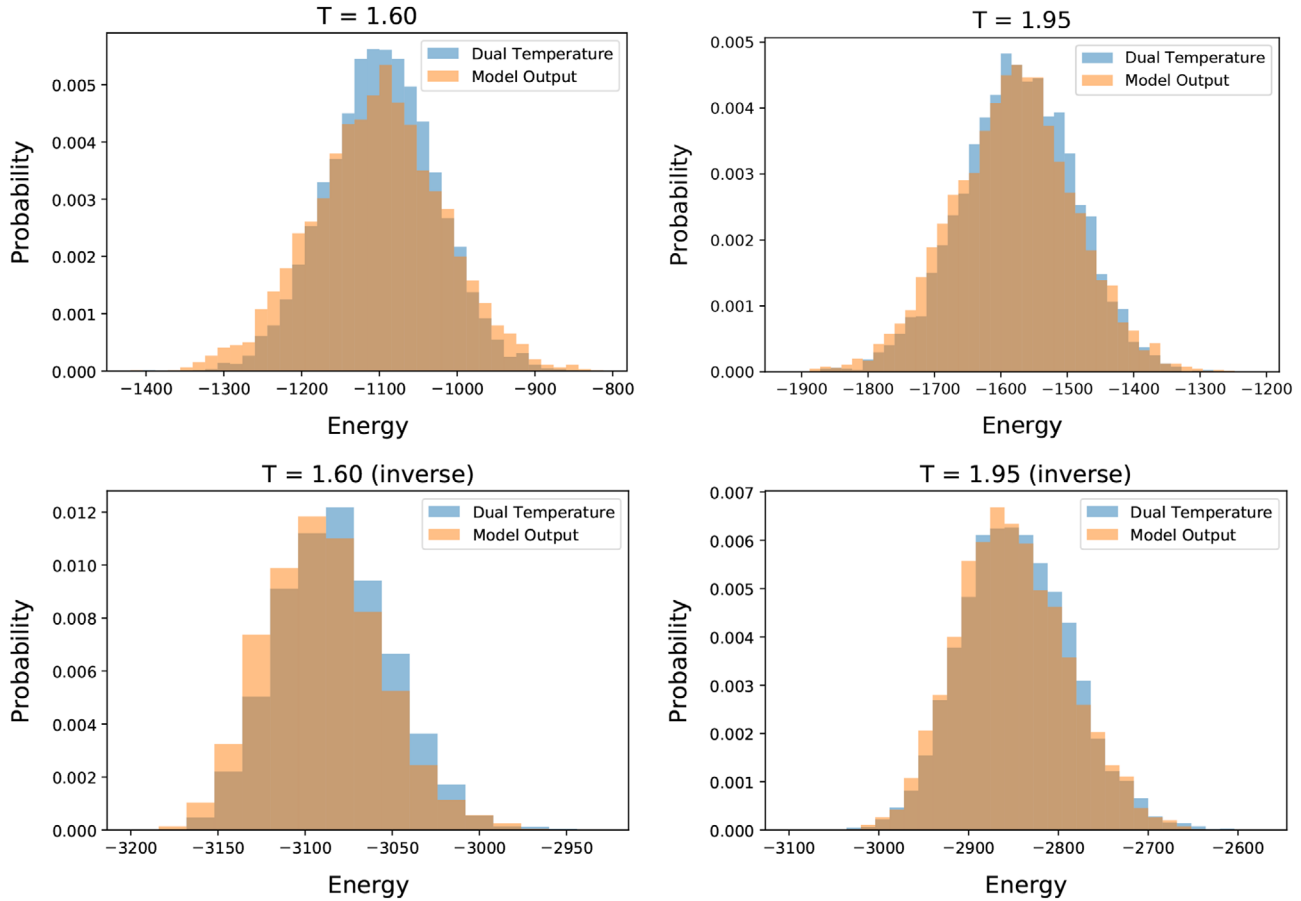


Figure 18. Energy distributions of U-Net outputs and true dual temperatures. **Top:** Mapping from low- to high-temperature regions. **Bottom:** Mapping from high- to low-temperature regions.

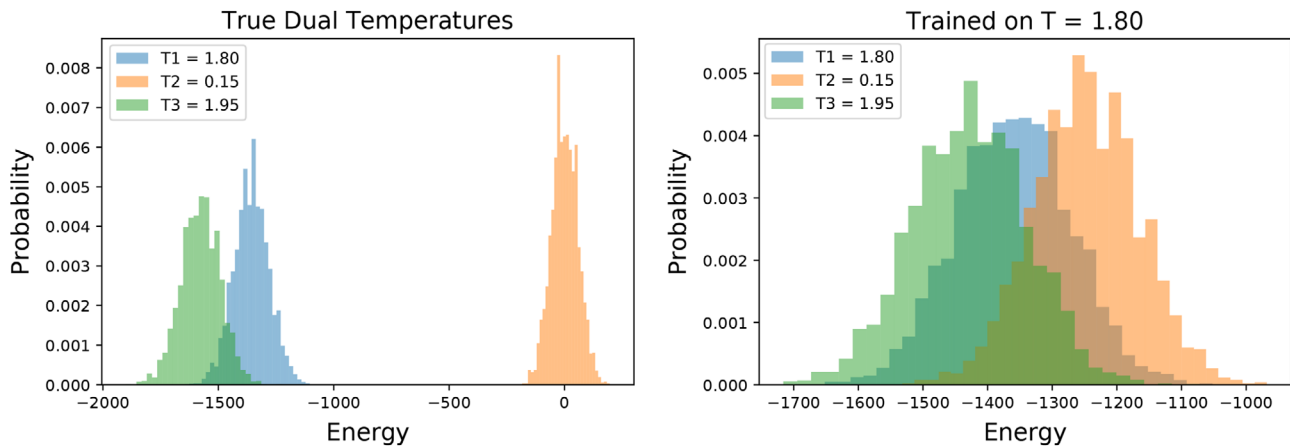


Figure 19. Output of U-networks trained on a single-temperature dataset for various temperatures. The ability to distinguish between inputs depends strongly on the original temperature. Here we show results for $T = 1.80$ where the network is able to distinguish between different inputs.

where a potential Jacobian is ignored. By using an integral representation for the δ function and some integration by parts, this action can be rewritten as

$$\int DFDV e^{-\frac{i}{\hbar} \int d^4x \frac{1}{4g^2} F_{\mu\nu} F^{\mu\nu} - \frac{1}{4\pi} (\partial_\mu V_\nu - \partial_\nu V_\mu) \bar{F}^{\mu\nu}} \quad (18)$$

In this formulation one can now also integrate out $F_{\mu\nu}$ as the integral is essentially Gaussian. This leads to

$$\int DVe^{-\frac{ig^2}{16\pi^2} \int d^4x (\partial_\mu V_\nu - \partial_\nu V_\mu) (\partial^\mu V^\nu - \partial^\nu V^\mu)} \quad (19)$$

This path integral is now over a different field V which was introduced merely as an auxiliary field. The relation between both representations can be seen from the equations of motion from the action involving both fields A and V :

$$\tilde{F}_{\mu\nu} = -\frac{g^2}{2\pi}(\partial_\mu V_\nu - \partial_\nu V_\mu) \equiv -G_{\mu\nu} \quad (20)$$

Electric and magnetic fields components are exchanged between these two descriptions and in addition the appearance of the coupling constant is inverted $g \rightarrow 1/g$.⁵ Despite the local relation (20), the map relating both representations is non-local as it involves the integration over space-time.

Note that the integration of a Gaussian from (18) to (19) corresponds precisely to the transformation of a Gaussian from position space to momentum space in the Fourier transformation. This highlights the connection between Fourier transformation and mapping fields under duality.

This analysis for electromagnetism in four dimensions can be extended to the discussion of massive p -form fields in D dimensions (cf. [1] for a review). Again a relation between the variables in terms of Fourier transformation can be established.

Applications in Physics

In the previous sections we have focused on the determination of classification tasks with the help of dual variables. In the context of physics, the use of dualities is generally speaking in the context of determining correlation functions more accurately. In turn this can be seen as properties of the data and hence can be connected with our classification tasks. To highlight the strength of these techniques we mention two major applications where the methods based on dualities outperform other techniques:

- 1. Hydrodynamic transport coefficients for quark gluon plasma:** In the context of holography, strongly coupled conformal field theories are related with weakly coupled gravitational systems⁶ in one higher dimension. Field theory correlators can be calculated by performing the appropriate perturbation analysis in the gravitational system.^[11–13] One of the prime examples includes the calculation of the shear viscosity η/s of $\mathcal{N} = 4$ super Yang-Mills theory which effectively is a two-point correlation function of the stress energy tensor.^[14,15] It has been argued that these calculations can be used to understand properties of the quark-gluon plasma and provide - at reasonably low calculational effort - quantitatively more accurate results than lattice predictions (cf. [16] for a review and further interesting applications).
- 2. Yukawa couplings in the standard embedding for the heterotic string:** Here the duality in use is referred to as mirror symmetry, a generalisation of T-duality. In the heterotic standard embedding it facilitates the calculation of Yukawa couplings in the standard embedding. Concretely, in the dual frame the

⁵ Note that this becomes a real strong-weak duality once charged fields are introduced.

⁶ See [10] for the connection between holography and deep Boltzmann machines.

Table 4. Field content of the electric phase.

Field	$SU(N_c)$	$SU(N_f)_L$	$SU(N_f)_R$	$U(1)_A$	$U(1)_B$	$U(1)_R$
Q	N_c	N_f	1	1	1	$1 - \frac{N_c}{N_f}$
\bar{Q}	\bar{N}_c	1	\bar{N}_f	1	-1	$1 - \frac{N_c}{N_f}$

Table 5. Field content of the magnetic phase.

Field	$SU(\bar{N}_c = N_f - N_c)$	$SU(N_f)_L$	$SU(N_f)_R$	$U(1)_A$	$U(1)_B$	$U(1)_R$
q	\bar{N}_c	\bar{N}_f	1	1	$\frac{N_c}{N_c}$	$1 - \frac{N_c}{N_f}$
\bar{q}	\bar{N}_c	1	N_f	1	$-\frac{N_c}{N_c}$	$1 - \frac{N_c}{N_f}$
\bar{M}	1	N_f	\bar{N}_f	-2	0	$2 \frac{N_c}{N_f}$

27^3 couplings are purely topological whereas in the original frame the couplings ($\overline{27}^3$) depend on the Kähler moduli. The topological couplings can be computed with standard methods in finding solutions to the Picard-Fuchs equations. Both couplings have to be identical due to mirror symmetry and utilising the mirror map between the dual moduli spaces allows a calculation of the Kähler moduli dependence in the $\overline{27}^3$ coupling. The direct calculation of these corrections requires counting of appropriate rational curves on the background Calabi-Yau manifold which is known as a hard problem in Mathematics. Using mirror symmetry this hard calculation can be avoided. For a physicist the Yukawa couplings in the original frame capture a tree-level part and non-perturbative corrections. It is these non-perturbative corrections which can be calculated using mirror symmetry. For explicit constructions of these dualities and more details see for instance.^[17–20] Note that the reduced calculational complexity required to calculate the Yukawa couplings in the dual frame was mentioned in [21].

Both examples highlight the capability of calculating far beyond the realm of standard perturbation theory. As a final comparison to showcase the connection of the dualities in the 1D Ising case, we discuss the connection with Seiberg duality. Here we identify a starting point for correlators which can serve as candidate replacements of metastability in the 1D Ising case.

4.1. Seiberg Duality

Let us comment on the connection to the classical example of Seiberg duality in the context of SQCD.^[22–24] Here two gauge theories share the same infrared physics but differ in the UV. These are referred to as the electric and magnetic phase. The electric phase ($3/2N_c < N_f < 3N_c$) is described by the field content presented in **Table 4** and the magnetic one in **Table 5**. The electric theory has no superpotential whereas the magnetic theory has a superpotential of the form $W = \bar{M}q\bar{q}$ where \bar{M} is related to the meson M built out of quarks in the electric phase.

Electric Phase

As a supersymmetric theory with zero tree-level superpotential, the classical Lagrangian of the electric phase involves a D-term potential whose flat directions at vanishing value parameterise the moduli space of the theory. More precisely, the corresponding quark expectation values can be determined by imposing the D-flatness condition $D^A = 0$ with

$$D^A = \sum_i Q_i^\dagger T_i^A Q_i + \tilde{Q}_i^\dagger T_i^A \tilde{Q}_i, \quad (21)$$

where the T^A denote the generators of the respective gauge group $SU(N_c)$. The classical moduli space is then defined as the space of quark vacuum expectation values modulo gauge equivalence. As argued in [24, 25], this allows for an equivalent description in terms of expectation values of gauge-invariant polynomials in the fields subject to any classical relations. For the theories considered here, such combinations are given by the $2\binom{N_f}{N_c}$ baryon and N_f^2 meson operators

$$\begin{aligned} B^{i_1 \dots i_{N_c}} &= Q_{a_1}^{i_1} \dots Q_{a_{N_c}}^{i_{N_c}} \epsilon^{a_1 \dots a_{N_c}}, \\ \tilde{B}_{i_1 \dots i_{N_c}} &= \tilde{Q}_{a_1}^{i_1} \dots \tilde{Q}_{a_{N_c}}^{i_{N_c}} \epsilon_{a_1 \dots a_{N_c}}, \\ M_j^i &= Q_a^i \tilde{Q}_j^a. \end{aligned} \quad (22)$$

Due to the identity

$$\epsilon_{a_1 \dots a_{N_c}} \epsilon^{b_1 \dots b_{N_c}} = \delta_{a_1}^{[b_1} \delta_{a_{N_c}}^{b_{N_c}]}, \quad (23)$$

these are subject to additional constraints

$$B^{i_1 \dots i_{N_c}} \tilde{B}_{j_1 \dots j_{N_c}} = M_{j_1}^{[i_1} M_{j_{N_c}}^{i_{N_c}]}, \quad (24)$$

leaving a total of $2N_f N_c - (N_c^2 - 1)$ light D-flat directions (cf. [26]). The physical interpretation of this is that the gauge group $SU(N_c)$ is completely broken, which is reflected in the number $N_c^2 - 1$ of broken generators.^[24]

Magnetic Phase

In the infrared, the above theory is dual to a magnetic description based on the gauge group $SU(\tilde{N}_c = N_f - N_c)$. The corresponding field content is listed in Table 5. Unlike the electric phase, the magnetic phase involves an additional superpotential

$$W = \tilde{M}_j^i q_i \tilde{q}^j, \quad (25)$$

where the magnetic meson \tilde{M} defines a fundamental degree of freedom and is related to its electric counterpart defined in (22) by a characteristic scale μ ,

$$\tilde{M} = \frac{1}{\mu} M. \quad (26)$$

Often both mesons are identified and one uses the notation M in either phase, which is indeed valid at the infrared fixed point. The

presence of the dimensionful parameter μ in (26) is only required to relate both meson operators in the ultraviolet limit: Here, the electric meson is a composite state with canonical dimension 2, picking up an anomalous dimension $3\frac{\tilde{N}_c}{N_f}$ during the renormalisation group flow to the infrared fixed point, while the latter defines a fundamental field of dimension one flowing to the same fixed point. It is therefore common to define a separate operator as in (26) to correctly describe the magnetic meson in the ultraviolet limit. The characteristic scale μ also appears in the matching condition

$$\Lambda^{3N_c - N_f} \tilde{\Lambda}^{3\tilde{N}_c - N_f} = (-1)^{\tilde{N}_c} \mu^{N_f} \quad (27)$$

for the scales Λ and $\tilde{\Lambda}$ of the electric and magnetic theory, respectively. From this, it can be seen that the duality relates different theories at strong and weak coupling, thus resembling the characteristic structure of a strong-weak duality.

Analogously to the electric phase, one can define $2\binom{N_f}{\tilde{N}_c}$ magnetic baryon operators as

$$\begin{aligned} b_{i_1 \dots i_{\tilde{N}_c}} &= q_{a_1}^{i_1} \dots q_{a_{\tilde{N}_c}}^{i_{\tilde{N}_c}} \epsilon_{a_1 \dots a_{\tilde{N}_c}}, \\ \tilde{b}^{i_1 \dots i_{\tilde{N}_c}} &= \tilde{q}_{a_1}^{i_1} \dots \tilde{q}_{a_{\tilde{N}_c}}^{i_{\tilde{N}_c}} \epsilon^{a_1 \dots a_{\tilde{N}_c}}, \end{aligned} \quad (28)$$

which, due to the identity $\binom{N_f}{\tilde{N}_c} = \binom{N_f}{N_f - \tilde{N}_c}$, carry the same number of degrees of freedom as their electrical counterparts. Formally, further mesons could be defined by $\tilde{m} = q\tilde{q}$, however, these do not lead to new degrees of freedom in the moduli space due to additional equations of motion $\langle q\tilde{q} \rangle = 0$ arising from the presence of the superpotential (25), thus avoiding inconsistency of the duality.^[26] A more in-depth analysis of the moduli spaces as well as further consistency checks of the duality were performed (e.g. in [24]) and we would like to refer the interested reader to the original works for more details.

Application to Neural Networks

At the infrared fixed point, there exists a direct relation between both types of baryon operators,

$$\begin{aligned} B^{i_1 \dots i_{N_c}} &= \sqrt{-(-\mu)^{N_c - N_f} \Lambda^{3N_c - N_f}} \epsilon^{i_1 \dots i_{N_c} j_1 \dots j_{\tilde{N}_c}} b_{j_1 \dots j_{\tilde{N}_c}}, \\ \tilde{B}_{i_1 \dots i_{\tilde{N}_c}} &= \sqrt{-(-\mu)^{N_c - N_f} \Lambda^{3N_c - N_f}} \epsilon_{i_1 \dots i_{\tilde{N}_c} j_1 \dots j_{N_c}} \tilde{b}^{j_1 \dots j_{N_c}}. \end{aligned} \quad (29)$$

As can be seen, the baryons in the electric and magnetic phase involve products of N_c and $\tilde{N}_c = N_f - N_c$ quarks, respectively. This is similar to our discussion of the 1D Ising chain, in which determining the total energy required the computation of n -spin products in the original representation, while the dependency was linear in the dual frame and therefore significantly easier to learn for neural networks. As the degree n of interactions was raised, the value of the total energy became increasingly sensitive to flips of single spins due to their involvement in an increasing number of n local interaction terms (cf. Figure 9), which eventually led to a complete deterioration of performance at very high n .

In the above setting, the baryon operators in (22) and (28) take the form of sums over products of N_c or \tilde{N}_c quarks, with each particular component appearing in $(N_c - 1)!$ or $(\tilde{N}_c - 1)!$ non-vanishing products (taking the role of the “local interaction terms”). Similar to the 1D Ising chain, such dependencies are likely to be learned more easily in the phase for which the number of factors is lower. In the setting discussed here, there exists a range $3/2N_c < N_f < 2N_c$ for which $\tilde{N}_c < N_c$, implying that baryon relations might be easier to be accessed in the magnetic theory. Conversely, the electric phase might be preferable in the region $2N_c < N_f < 3N_c$, where generically $\tilde{N}_c > N_c$.

It is a natural question to explore whether this fact can be used to re-discover Seiberg-like dualities following the strategy successfully applied for the 1D Ising case in Section 2.3. As this analysis promises to be too lengthy for this proof of concept paper, we leave this issue for the future.

5. Conclusion

Dualities offer a more efficient way of calculating correlation functions in physics. In particular, in the context of strongly coupled regions they provide in several examples the best technique to calculate properties of these dynamical systems. We have presented several examples where this improved way of calculating correlation functions via dual representations can be related to improved classification tasks.

Such different and more efficient data descriptions are clearly desirable, but how can one get them without knowing about the explicit map between such representations. We have shown in this work how such beneficial representations can be obtained in an unsupervised fashion, i.e. without telling the network about its existence. By reproducing several human-made dualities automatically we provide a proof of concept that machines can be programmed to find dualities. Clearly, further and more involved types of dualities need to be addressed with these kind of techniques, which then will enable the search for new dualities.

Undoubtedly our tasks are relatively simple and can be achieved for instance in the case of the 1D Ising and Fourier analysis by more sophisticated architectures. However, we want to stress that these settings serve as an important first step to address tasks which are not accessible with state-of-the-art techniques with the same strategies used here.

The dual representations obtained by our networks can be analysed and we have found a representation which is interpretable, e.g. we could recognise a Fourier-like transformation or transformations similar to the duality transformation in the 1D Ising example. This is encouraging as the neural network provides us with the explicit map to this interpretable representation.

Where will further steps in this new field of exploring dualities between different descriptions of dynamical systems with the help of machine learning take us?

Appendix A: Details on Discrete Fourier Transformation

This appendix contains further details on the experimental setup used for our discussion of the Fourier transform in section

A.1. Data

The dataset is split into two categories “pure noise” (0) and “noise with signal” (1). We consider a discretised space of size 1000 and generate 10^5 signals p_k in the Fourier domain taking the form

$$p_k = |p_k|e^{i\varphi_k}, \quad (\text{A.1})$$

where $|p_k| \sim \mathcal{N}(2, 0.1)$, $\varphi_k \sim \mathcal{U}(0, 2\pi)$ and k uniformly sampled between 0 and 1000. A signal in position space is generated by computing the inverse Fourier transform, which relates the position and Fourier domains via

$$p_k = \frac{1}{\sqrt{N}} \sum_{j=1}^N x_j e^{-2\pi i j k / N},$$

$$x_k = \frac{1}{\sqrt{N}} \sum_{j=1}^N p_j e^{2\pi i j k / N}, \quad (\text{A.2})$$

with $N = 1000$. We then generate noisy signals (“class 1”) by adding Gaussian noise following the distribution $x_{k, \text{noisy}} \sim \mathcal{N}(0, 0.1)$ and pure noise $x_{k, \text{pure noise}} \sim \mathcal{N}(0, \sigma)$ with σ chosen such that the samples of both classes show the same mean quadratic deviation from 0. The number of samples for both classes is set to 10^5 , and we employ a 4:1 train-test split. The data is formatted in such a way that each sample contains one channel representing its real part and one its imaginary part.

In this task a signal in the position space takes the form of a sine-cosine wave spread all over the domain, whereas its information is concentrated in one (complex) bin in the Fourier space (cf. Figure 1).

A.2. Experiments

All experiments were performed using Keras with TensorFlow backend. Training equilibrium in all settings was commonly reached after less than 50 epochs; the training process was run for 200 epochs to ensure that no further improvements occur after stopping. Training was performed with Nesterov Adam optimiser with learning rate $2 \cdot 10^{-3}$, batch size 128 and binary crossentropy as loss function. Data generation, preprocessing and training of networks was performed for ten different random seeds to prevent results from getting skewed due to outliers. A summary of all results related to this setting can be found in **Table A1** at the end of this appendix.

Simple Networks: We first checked whether simple networks are able to distinguish the classes in position space. We used one-dimensional convolutional neural networks with one convolutional layer consisting of 4 filters of size one with ReLu activation followed by a linear layer with sigmoid activation. The accuracy on the position space commonly stagnated at values around 0.53, which is only slightly superior to pure guessing. The poor performance could be traced back to both underfitting and overfitting, with the training set accuracy commonly remaining below 0.5800 for the entire training process. Slight improvements to test set accuracies around 0.54 could be made by including one or two additional convolutional layers, however, no notable difference in performance was observed for more complex architectures.

Table A1. Mean best test set accuracies for signal detection in noisy data reached after 200 epochs.

Model	val acc
Simple Network x-space	0.5317
Simple Network p-space	0.9879
Simple Network learned representation (feature separation)	0.7717
Simple Network (2 Conv-Layers) x-space	0.5449
Simple Network (3 Conv-Layers) x-space	0.5438
Simple Network + Dense x-space (fixed pretrained weights)	0.5005
Simple Network + Dense x-space (free pretrained weights)	0.5013
Simple Network + Dense x-space (free random weights)	0.5016
Simple Network + 2 Dense x-space (free random weights)	0.5018
Simple Network + 3 Dense x-space (free random weights)	0.5025
Simple Network + Dense x-space (free random weights) + L1-Reg (1e-5)	0.5013
Simple Network + Dense x-space (free random weights) + L1-Reg (1e-4)	0.5014
Simple Network + Dense x-space (free random weights) + L1-Reg (1e-3)	0.5011
Simple Network + Dense x-space (free random weights) + L1-Reg (1e-2)	0.5015
Simple Network + Dense x-space (free random weights) + L1-Reg (1e-1)	0.5008
Simple Network + Dense x-space (free random weights) + L2-Reg (1e-5)	0.5010
Simple Network + Dense x-space (free random weights) + L2-Reg (1e-4)	0.5019
Simple Network + Dense x-space (free random weights) + L2-Reg (1e-3)	0.5010
Simple Network + Dense x-space (free random weights) + L2-Reg (1e-2)	0.5018
Simple Network + Dense x-space (free random weights) + L2-Reg (1e-1)	0.5015
Simple Network + Dense x-space (free random weights) + Dropout (0.1)	0.5017
Simple Network + Dense x-space (free random weights) + Dropout (0.2)	0.5015
Simple Network + Dense x-space (free random weights) + Dropout (0.5)	0.5004
Simple Network + Dense x-space (free random weights) + BatchNorm	0.5013

Adding Dense Layers: We tested whether adding a dense layer of size $2N$ at the beginning of the above architecture can improve the results. The model was tested for the three different settings

- all weights randomly initialised and trainable,
- weights of the convolutional-layer pretrained on Fourier domain and trainable,
- weights of the convolutional-layer pretrained on Fourier domain and fixed.

Due to the high number of parameters, we tested the performance for L1 and L2 regularisation with parameters $10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}$, dropout with rates 0.1, 0.2, 0.5 and batch normalisation. To cover a wider range of architectures, we furthermore varied the number of dense layers between 1 and 3.

Interestingly, none of the architectures showed any improvements beyond pure guessing, implying that it is difficult for neural networks to find the Fourier transform (or similar mappings) by themselves, even if given “hints” by initializing parts of the weights to perform well in the momentum space domain.

Except for architectures with very strong regularization, the poor performance in most settings could be attributed exclusively to severe overfitting. This problem might in principle be avoidable by increasing the amount of the training data to extremely large values. This would, however, defeat the point of finding a “useful” network with reasonable resources.

Feature Separation: Tests for representations learned by the architecture described in Section 3.1 were performed using the same simple network architecture as employed for comparing the position and actual momentum space. Preprocessing and training modalities were the same as before; the feature separation network was trained separately for each of the ten test-runs. The feature separation network was trained on separately-generated noisy signals with varying noise levels of up to $\sigma = 0.075$ and showed similarly good performance in all instances. All performance values and plots presented in this paper are with respect to networks trained on noisy data with $\sigma = 0.075$. We found an improved performance when we reduced noise levels in the data.

Appendix B: Details on 1D Ising Model

This appendix contains further details on the experimental setup used for our discussion of the 1D Ising model in sections 2.3 and 3.2.

B.1. True Dual Representation

In Section 2.3 we focused on the application of simple neural networks to detect (meta-)stable states in the 1D Ising model with multi-spin interactions.

Classifying (Meta-)Stable States: For our large-scale tests, we used a single-layer perceptron with 128 hidden neurons, ReLU activation for the hidden layer and sigmoid activation for the output layer. To ensure comparability of results, we did not include any regularisation techniques or more advanced components. Weights were initialised randomly following common practice; training was performed with standard Nesterov Adam optimiser and learning rate decay.

The full dataset contained all 2^{18} states for the 1D Ising chain with $N = 18$, and tests were performed for varying orders of interaction n . We split the data into states labeled as “(meta-)stable” (0) or “(meta-)stable” (1) and normalised the training and test sets to contain an equal number of samples for each class. Experiments were performed for training set sizes of 600, 3000 and 9500 which were chosen for better comparability of results due different total numbers of metastable states for different n .

The training showed a slight dependence on the initial conditions and was therefore performed ten times for each setting and data representation. Python, NumPy and TensorFlow random seeds were fixed by hand and stored for each result. We found that 500 epochs was a viable cutoff after which no relevant changes in the overall performance occurred. The average best test accuracies and losses achieved in 10 training runs are listed in Table 1.

Modifications of Architecture: In order to check whether the results obtained for the above setting generalise to a wider class of neural networks, we performed sample-wise tests for one or more of the following modifications in architecture:

- Varying the number of hidden layers within the range 1,2,3,4,5.
- Varying the number of neurons per layer within the range 16, 32, ..., 1024.
- Employing linear, sigmoid or ReLU activation functions.
- Using L2 weight regularisation with penalty 0.01 and 0.1, dropout with rates 0.2 and 0.5 or batch normalisation.
- Including up to five Convolutional layers into the network.

Almost none of the modifications lead to any significant change in the overall results. One exception was the introduction of convolutional networks, which were able to reach close-to-perfect performance at very low $n \leq 4$, but resorted to pure guessing at higher-order interactions.

B.2. Autoencoder with Latent Loss

In Section 3.2 the discussion of (meta-)stability classification was extended to the output of a constrained autoencoder as depicted in Figure 15.

Training of Autoencoders: The constrained autoencoders consisted of one hidden layer of 128 neurons in their encoder and decoder components and bottleneck of dimension 18 and 50. The latent output of the bottleneck part was additionally fed into a linear layer. Training was then performed using standard Nesterov Adam optimiser to simultaneously minimise component-wise binary crossentropy as reconstruction loss and mean squared error as regression loss.

Weights were initialised randomly for both autoencoders. We found that this generally led to better performance in both losses compared against hard-coded layers or pretraining on the actual dual representation (the latter only possible for latent dimension 18).

Achieving good performance in both tasks required relatively large amounts of training data. The autoencoders were therefore trained on 80% of the full dataset of 2^{18} states. In case of poor performance, underfitting was prevalent, and there were no cases of overfitting. Using L1 penalties to force sparse activations or representations generally led to poor performance and therefore were not used for our tests.

Classifying (Meta-)Stable States: We tested the performance in classifying (meta-)stable states using the same setting as before, with the duality transformation (7) replaced by the intermediate output of the previously described constrained autoencoders. In order to prevent information of the metastability test set from leaking into the training set of the autoencoder, the same train-test split was employed for the metastability classification. Otherwise, training and testing modalities were identical with those of the original and true dual representation.

Data preprocessing and training of all involved networks were repeated ten times for different Python, NumPy and TensorFlow random seeds to prevent outliers from skewing the results. The average best test set accuracies reached after at most 500 epochs are stored in Table 3. More specifically, we observed the following behaviour:

- For very simple settings such as $n = 4$, the classifiers performed almost equally well on the intermediate output as on the actual dual representation. As shown in Table 2, similar performances can also be reached by using pure energy considerations, and these results should therefore be treated with caution.
- For all higher-degree interactions with $n \geq 4$, both most classifiers clearly beat the benchmark performance on the original representation. However, networks trained on outputs with latent dimension 18 often fall short of outperforming the benchmarks set by purely energetic arguments (cf. again Table 2) in particular at low training set sizes. This is not the case for latent dimension 50, and such networks are able to distinguish well between both classes in regions of high energetic overlap. Their performance is, however, not equally well to the true dual representation in these critical cases.
- The method transfers well to other values of N and n as long as the task of energy regression is sufficiently easy to solve for the considered class of networks, but breaks down in very complex cases such as $N = 100$ and $n = 50$.

In addition to the above tests, a sanity check similar to the Fourier setting was performed by placing a non-pretrained encoding architecture with latent dimension 18 or 50 in front the simple network and training on the original representation (see Table B1). The performance in this case was slightly worse than that of the simple networks alone (cf. Table 1), showing that the layer pre-trained for energy regression indeed leads to a benefit beyond a mere improvement of network capacity.

Table B1. Detection of (meta-)stable states in the 1D Ising chain for different interactions and amounts of training data. The listed numbers describe the average best test accuracy over 10 training runs of 500 epochs each when trained in the original representation with a non-pretrained encoding architecture with latent dimension 18 (**Left**) and 50 (**Right**) placed in front of the simple network. Missing values indicate that the number of required samples exceeds the total number of metastable states for the considered setting.

lat (18)	$n = 4$	$n = 5$	$n = 8$	$n = 9$	$n = 12$	lat (50)	$n = 4$	$n = 5$	$n = 8$	$n = 9$	$n = 12$
$6 \cdot 10^2$	0.9047	0.8404	0.8629	0.8507	0.8747	$6 \cdot 10^2$	0.9026	0.8570	0.8616	0.8715	0.8632
$3 \cdot 10^3$	–	0.8983	0.9039	0.9011	0.9165	$3 \cdot 10^3$	–	0.9058	0.9002	0.8991	0.9176
$9.5 \cdot 10^3$	–	–	0.9405	0.9400	0.9751	$9.5 \cdot 10^3$	–	–	0.9410	0.9360	0.9745

Acknowledgements

We would like to thank Jim Halverson, Fernando Quevedo, and Fabian Ruehle for discussions. SK thanks the Aspen Center for Physics, which is supported by National Science Foundation grant PHY-1607611, and the Simons Center for Geometry and Physics during the Neural Networks and the Data Science Revolution program for providing a very stimulating work environment to develop some of this work. Parts of these results have been presented already at the following conferences and workshops: String Phenomenology 2019, QTS 2019 (Montreal), Corfu Summer Institute, DLAP in Kyoto, 1st French-German Meeting in Physics, Mathematics and Artificial Intelligence Theory, and XAIENCE in Seoul.

Conflict of Interest

The authors have declared no conflict of interest.

Received: March 1, 2020
Published online: March 25, 2020

- [1] F. Quevedo, *Nucl. Phys. Proc. Suppl.* **1998**, 61A, 23, hep-th/9706210. [23(1997)].
- [2] J. Polchinski, *Stud. Hist. Phil. Sci.* **2017**, B59, 6, 1412–5704.
- [3] H. A. Kramers, G. H. Wannier, *Phys. Rev.* **1941**, 60, 252.
- [4] H. A. Kramers, G. H. Wannier, *Phys. Rev.* **1941**, 60, 263.
- [5] L. Onsager, *Phys. Rev.* **1944**, 65, 117.
- [6] R. Savit, *Rev. Mod. Phys.* **1980**, 52, 453.
- [7] L. Turban, *J. Phys. A: Math. Gen.* **2016**, 49, 355002, 1605.05199.
- [8] G. Chechik, V. Sharma, U. Shalit, S. Bengio, *J. Mach. Learn. Res.* **2009**, 11, 1109.
- [9] F. Schroff, D. Kalenichenko, J. Philbin, FaceNet: A Unified Embedding for Face Recognition and Clustering, *arXiv e-prints* (2015) arXiv:1503.03832, 1503.03832.
- [10] K. Hashimoto, *Phys. Rev.* **2019**, D99, 106017, 1903.04951.
- [11] J. M. Maldacena, *Int. J. Theor. Phys.* **1999**, 38, 1113, hep-th/9711200. [Adv. Theor. Math. Phys.2,231(1998)].
- [12] E. Witten, *Adv. Theor. Math. Phys.* **1998**, 2, 253, hep-th/9802150.
- [13] O. Aharony, S. S. Gubser, J. M. Maldacena, H. Ooguri, Y. Oz, *Phys. Rept.* **2000**, 323, 183, hep-th/9905111.
- [14] G. Policastro, D. T. Son, A. O. Starinets, *Phys. Rev. Lett.* **2001**, 87, 081601, hep-th/0104066.
- [15] P. Kovtun, D. T. Son, A. O. Starinets, *Phys. Rev. Lett.* **2005**, 94, 111601, hep-th/0405231.
- [16] J. Casalderrey-Solana, H. Liu, D. Mateos, K. Rajagopal, U. A. Wiedemann, Gauge/String Duality, Hot QCD and Heavy Ion Collisions, 1101.0618.
- [17] P. Candelas, X. C. De La Ossa, P. S. Green, L. Parkes, *Nucl. Phys.* **1991**, B359, 21.
- [18] P. Candelas, X. De La Ossa, A. Font, S. H. Katz, D. R. Morrison, *Nucl. Phys.* **1994**, B416, 481, hep-th/9308083. [483(1993); AMS/IP Stud. Adv. Math.1,483(1996)].
- [19] S. Hosono, A. Klemm, S. Theisen, S.-T. Yau, *Commun. Math. Phys.* **1995**, 167, 301, hep-th/9308122.
- [20] S. Hosono, A. Klemm, S. Theisen, *Lect. Notes Phys.* **1994**, 436, 235, hep-th/9403096.
- [21] J. Halverson, F. Ruehle, *Phys. Rev.* **2019**, D99, 046015, 1809.08279.
- [22] N. Seiberg, *Phys. Rev.* **1994**, D49, 6857, hep-th/9402044.
- [23] N. Seiberg, *Nucl. Phys.* **1995**, B435, 129, hep-th/9411149.
- [24] K. A. Intriligator, N. Seiberg, *Nucl. Phys. Proc. Suppl.* **1996**, 45BC, 1, hep-th/9509066. [157(1995)].
- [25] M. A. Luty, W. Taylor, *Phys. Rev.* **1996**, D53, 3399, hep-th/9506098.
- [26] B. Wecht, Introduction to supersymmetric gauge theories. Lecture Notes from lectures presented at the Isaac Newton Institute, Sept., 2007.