EPiC Series in Computing

Volume 104, 2024, Pages 151–162

Proceedings of 3rd International Workshop on Mathematical Modeling and Scientific Computing

Convolutional Neural Networks Application for an Average Thermodynamic Characteristics Calculation of Spin Glasses

Dmitrii Kapitan[1]*, Petr Andriushchenko[2], Konstantin Nefedev[1], and Vitalii Kapitan[3]

[1] Department of Theoretical Physics and Intelligent Technologies, Far Eastern Federal University, Vladivostok, Russia
kapitan.diu@dvfu.ru, nefedev.kv@dvfu.ru

[2] Heidelberg Institute of Global Health (HIGH), Heidelberg University Hospital (UKHD), Heidelberg University, Heidelberg, Germany
petr.andriushchenko@uni-heidelberg.de

[3] Department of Statistics and Data Science, National University of Singapore, Singapore
kapitanv@nus.edu.sg

Abstract

In this work, we introduce a novel methodology for studying the low-temperature phase of frustrated spin glass models using convolutional neural networks (CNNs). Our approach addresses the regression of thermodynamic properties, specifically the average energy $\langle E \rangle$, as a function of temperature T for spin glasses on a square lattice. By modelling the spin glass as a weighted graph, where exchange interaction values J_k are represented by the edges and mapped to lattice coordinates, we explore the functional relationship between $\langle E \rangle$ and the spatial distribution of J. We evaluate CNNs for their performance across various spin glass sizes and distributions of exchange integrals, demonstrating the potential of CNNs in capturing complex spin interactions and advancing the understanding of frustrated systems.

1 Introduction

Spin glasses (SG) are paradigmatic complex systems, for which disorder plays a central role. The study of spin glasses is intrinsically interdisciplinary because, despite the term's original use to characterize specific magnetic materials that display unusual phase behaviour, related models and concepts have since found use in a wide range of domains [1].

Two important characteristics that distinguish spin glasses from other lattice models are disorder brought on by the freezing of spins at low temperatures and frustration, where competing magnetic interactions prohibit all interactions in the system from being satisfied simultaneously [12]. Thus, energy minimization and ground state identification in spin glass models are classified as NP-complete problems [20] because of their substantial theoretical and computational difficulties. Therefore, the development of efficient algorithms for calculating low-energy states and calculating their thermodynamic properties is a major task in the theory of frustrated magnetism.

V.L. Turova, A.E. Kovtanyuk and J. Zimmer (eds.), MMSC 2024 (EPiC Series in Computing, vol. 104), pp. 151–162

Analytical solutions for Ising spin lattice models are limited [5], and no exact polynomial-time method exists for finding the ground state. Spin glass thermodynamics are typically studied with Monte Carlo methods [19, 29, 2, 4, 26], though their efficiency is hampered by long relaxation times, complex energy landscapes, ground state degeneracy, and critical slowdown. While the system requires numerous states to reach equilibrium, advances in computing power and Monte Carlo techniques have eased the growing complexity of spin glass calculations [18, 24, 21, 30, 22].

Machine learning offers an innovative solution to these issues by analyzing through massive volumes of data, finding hidden patterns, and predicting novel phenomena. Neural networks can effectively capture relevant features about conventional ordered and non-conventional phases, including Anderson's localized phases or lattice Ising gauge theory [9]. Researchers can apply Machine Learning to explore previously undiscovered areas and alter our understanding of the natural world by using the power of data-driven algorithms. Machine learning algorithms have the remarkable capacity to find intricate patterns in large amounts of data, which is of significant practical value to statistical physics [28, 7, 8, 6, 14, 17, 13, 25].

In this paper we propose to solve a relatively general problem: using machine learning methods to solve the problem of regression of the basic thermodynamic characteristic $\langle E \rangle$ (average energy), but it might be any other system characteristics, as a function of temperature T for spin glasses on a square lattice. To achieve this, we view the spin glass as a weighted graph, where the exchange interaction values are represented by the edges' values and the graph's architecture is matched to the lattice. Therefore, we are searching for a functional dependence between the primary average thermodynamic property of the system $\langle E \rangle$ and the spatial distribution of the exchange integrals on the square lattice of the spin glass $J_k = f_J(x_k, y_k)$ using a convolutional neural network (CNN). Here, x_k and y_k represent the bond coordinates for bond k, J_k is the bond value, and f_J is the function of the spatial distribution of spin glass bond values. To solve this problem, we used several CNN architectures described below to determine the optimal architecture for solving the above problem.

2 Spin glass model

This paper examines SG in the frame of two-dimensional Edwards - Anderson (EA) model [16] with periodic boundary conditions (PBC) on an Ising square lattice $N = L \times L$ (refer to Fig. 1 from [3]), where N is a total number of spins and L is a linear size of the system. Each spin in these models is an Ising spin that implies that it can be in two states $S_i = \pm 1$. On a square 2D lattice with periodic boundary conditions, every spin has four nearest neighbours, whose interactions are determined by the exchange integral $J_k = \pm 1$. For two spins the interaction energy is defined by:

$$E_{ij} = -J_{ij} S_i S_j, \tag{1}$$

where S_i, S_j are the interacting neighbouring spins in the lattice, J_{ij} is the exchange integral between S_i and S_j. Hence, the Hamiltonian of the whole system will be determined by:

$$H = -\sum_{<i,j>} J_{ij} S_i S_j, \tag{2}$$

where summation is only on nearest neighbours.

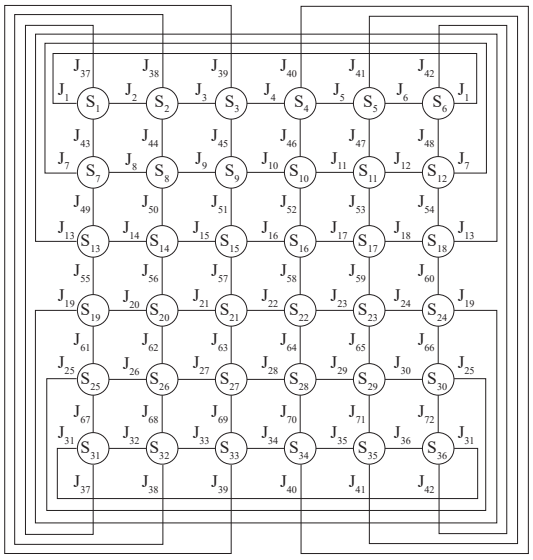


Figure 1: Example of the spin glass model on a square lattice 6×6 of Ising spins with periodical boundaries condition. $S_i = \pm 1$ – spins of the lattice, $J_k = \pm 1$ – exchange integral.

3 Data

For a lattice of size $N = L \times L$, there are 2^{2N} possible configurations of the exchange integrals. These configurations have a range from a fully antiferromagnetic case to a fully ferromagnetic case. For the former, all exchange interactions are $J_k = -1, \forall k$ leading to $\sum_{k=1}^{2N} J_k = -2N$. For the latter, all interactions are $J_k = 1, \forall k$ and $\sum_{k=1}^{2N} J_k = 2N$. To capture this relation, a neural network can be trained to recognize patterns in the arrangement of exchange interaction values on the lattice and their impact on macroscopic parameters. In particular, the spin glass configuration defined by the set $\{J_1, J_2, J_3, \ldots, J_{2N}\}$ and temperature T should be provided to the network, that then can be trained to predict the average energy $\langle E \rangle$. The same approach can be extended. For the extension, other thermodynamic quantities can be predicted. That includes the probability density of states, residual entropy, heat capacity, susceptibility and others.

To train the CNN, it was necessary to prepare datasets for training, validation, and testing of the models. First, two sizes of 6×6 and 10×10 spin glass models with different random distributions of the exchange integral J were calculated. For each considered spin glass configuration, 60 temperatures were calculated, ranging from 0.1 to 6 with a step size of 0.1. For the CNN architecture, we partitioned J into two channels: J_{hor} - representing horizontal bonds of the system and J_{ver} - representing vertical bonds. This approach uses the spatial information of the exchange integral distribution,

Two datasets were prepared using two different methods:

1. **Exact method**. We used an exact method to obtain the SG partition function, which uses high-performance parallel computing based on the transfer matrix method [15]. Although the algorithm has exponential time, it is still able to perform calculations on a relatively big lattice. From the partition function, it is possible to calculate the main average thermodynamic characteristics of the system. The method also allows us to find both

the ground state energy and the spin configurations of the state [23]. After specifying the size of the system, the boundary conditions and the values of all exchange integrals, we can calculate the number of all configurations (degeneration degree) $g(E_q, M_r)$ with all possible values of energy E_q and magnetisation M_r, where q runs through all possible values of the system energy and r are all possible values of the system magnetisation. For that, the partition function of the system can be written in the form:

$$Z = \sum_i \exp\left(-E_i/k_B T\right), \tag{3}$$

where i runs through all possible configurations of the system and E_i is the energy of the corresponding configuration. We can also rewrite the partition function in a different form:

$$Z = \sum_j g(e_j) \exp\left(-e_j/k_B T\right), \tag{4}$$

where j runs through all possible energy values (levels) of the system, e_j is the corresponding energy level, and $g(e_j)$ is the degeneracy of the corresponding energy level e_j - the number of configurations with the corresponding energy.

Any average thermodynamic quantity $\langle K \rangle$ can be calculated through a partition function [27]:

$$\langle K \rangle = \frac{1}{Z} \sum_i K_i \exp\left(-E_i/k_B T\right) = \frac{\sum_i K_i \exp\left(-E_i/k_B T\right)}{\sum_i \exp\left(-E_i/k_B T\right)}. \tag{5}$$

Then the average energy can be calculated as:

$$\langle E \rangle = \frac{\sum_i E_i \exp\left(-E_i/k_B T\right)}{\sum_i \exp\left(-E_i/k_B T\right)} = \frac{\sum_j e_j g(e_j) \exp\left(-e_j/k_B T\right)}{\sum_j g(e_j) \exp\left(-e_j/k_B T\right)}. \tag{6}$$

In this method, we calculated $6,860$ configurations for $N = 10 \times 10$ (the dataset dimensionality is $411,600$ since each spin glass model was calculated at 60 different temperature values) and $2,010$ configurations for $N = 6 \times 6$ (dataset dimensionality is $120,600$).

2. **Parallel tempering Monte Carlo method**. Also, we used Monte Carlo modelling. There were 10^4 equilibration MC steps, and then the energy and magnetization of the system were calculated and averaged over the next 10^5 MC steps. To overcome the effects of critical slowing down and avoid getting trapped in local minima, the system was simulated in parallel at 60 different temperatures. System configurations were exchanged every 10^3 MC steps, with the exchange probability dependent on the energy difference between configurations, shown in Equation (7). This exchange process is designed to allow high-temperature configurations, which explore the energy landscape more freely, to replace low-temperature configurations that may have become trapped in local minima. The algorithm proceeds by selecting pairs of neighbouring temperatures, starting from the highest, and attempting to swap configurations based on the Metropolis-Hastings acceptance criterion. This approach enhances convergence by enabling configurations at high temperatures to 'rescue' replicas stuck in local minima, facilitating a more thorough exploration of the state space across all temperatures.

$$p(X \to X') = \begin{cases} 1 & \text{if } \Delta \leq 0 \\ \exp(\Delta) & \text{if } \Delta > 0 \end{cases}, \tag{7}$$

where $\Delta = (1/T' - 1/T)(E' - E)$, E and E' are the energies corresponding to X and X' configurations respectively.

For the spin glass model with spin number $N = 6 \times 6$, the dataset consisted of $3,000$ configurations (with dimension 180,000 since each spin glass configuration was calculated at 60 different temperature values). For the model with $N = 10 \times 10$ number of spins, the dataset consisted of $8,000$ configurations (with dimension 480,000).

4 CNN architecture

Convolutional Neural Networks (CNNs) have emerged as a powerful and versatile tool in the field of machine learning, particularly for tasks involving spatial data. Their architecture is designed to learn spatial hierarchies of features automatically during the training process, adapting to the data through the use of convolutional layers, which enables them to effectively capture complex patterns and structures. One of the primary advantages of CNNs is their ability to recognize and exploit local dependencies in data through the use of convolutional filters (or kernels), which scan the input to identify patterns. This reduces the need for manual feature extraction and allows for more efficient training.

In the context of predicting the average energy levels in spin glass models, CNNs are particularly well-suited due to their capability to learn from the intricate patterns present in the bond configurations. By employing separate CNN channels for vertical and horizontal interactions, our approach leverages the inherent structure of spin glass systems, where bond interactions exhibit distinct spatial orientations. This dual-channel architecture not only enhances the model's ability to capture orientation-specific features but also improves the overall accuracy of predictions by allowing the network to process and integrate information from both dimensions effectively. In selecting the CNN architecture, we experimented with different configurations, including models with different numbers of fully connected (FC) layers. Since our goal in this paper is to present CNN architecture that has empirically demonstrated superior predictive performance, we focus below on describing the final architectures that performed best among the tested architectures.

The architecture of our first proposed CNN model, as illustrated in Figure 2, is designed to process the bond configurations through a series of convolutional and pooling layers. The model starts with two convolutional layers that extract basic features from the input bond configurations, followed by max pooling layers to reduce the spatial dimensions while retaining essential features. A dropout layer is introduced to mitigate overfitting, enhancing the model's generalizability. Subsequent convolutional layers further refine the feature extraction process, with additional pooling layers reducing the dimensionality. The global average pooling layer aggregates the extracted features into a single vector, which is then concatenated with an additional input representing the temperature. This concatenated vector is passed through fully connected layers to perform regression tasks, predicting the average energy levels. The integration of bond configuration and temperature inputs enables the model to effectively learn and predict the energy dynamics of the spin glass system.

CNN1 detailed architecture (Fig. 2): $Conv2D : (32, 3 \times 3) \to Conv2D : (64, 3 \times 3) \to MaxPool : (2 \times 2) \to Dropout : (0.2) \to Conv2D : (128, 3 \times 3) \to Conv2D : (256, 3 \times 3) \to$

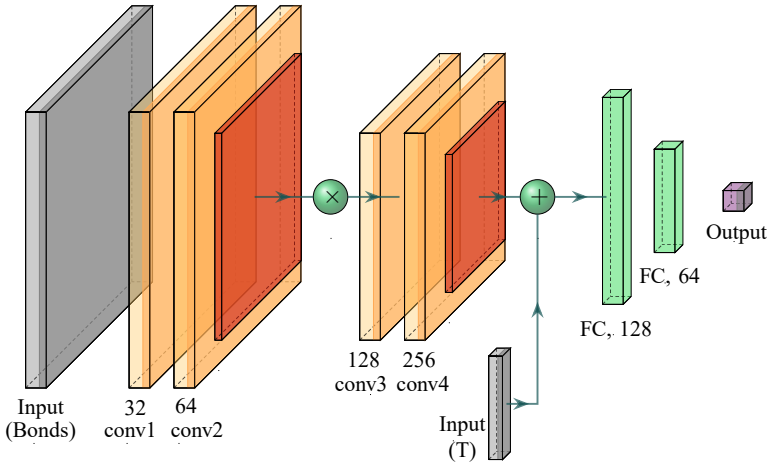


Figure 2: CNN1 architecture.

$MaxPool : (2 \times 2) \rightarrow GAP \rightarrow Concatenate$: (Temp input) $\rightarrow Dense : (128) \rightarrow Dense : (64) \rightarrow Dense : (1)$.

The second CNN model, detailed in Figure 3, incorporates a combination of convolutional, upsampling, and fully connected layers to capture and process the complex patterns within spin configurations. A key feature of this architecture is the inclusion of upsampling layers, which utilize Conv2DTranspose operations to reconstruct the spatial dimensions of the feature maps. This approach allows the model to produce high-resolution output by reversing the effects of max pooling and capturing finer details from the input data. The model then flattens the output from the convolutional layers using global average pooling, which aggregates the features into a compact representation. This representation is concatenated with an additional scalar input representing the temperature. The concatenated vector is passed through several fully connected layers.

CNN2 detailed architecture (Fig. 3): $Conv2D : (32, 3 \times 3) \rightarrow Conv2D : (64, 3 \times 3) \rightarrow MaxPool : (2 \times 2) \rightarrow Dropout : (0.2) \rightarrow Conv2D : (64, 3 \times 3) \rightarrow Conv2D : (128, 3 \times 3) \rightarrow Conv2DTranspose : (128, 3 \times 3) \rightarrow Conv2DTranspose : (64, 3 \times 3) \rightarrow UpSampling : (2 \times 2) \rightarrow Conv2DTranspose : (64, 3 \times 3) \rightarrow Conv2DTranspose : (32, 3 \times 3) \rightarrow GAP \rightarrow Dense : (128) \rightarrow Dense : (64) \rightarrow Dense : (32)$.

4.1 Gradient scalling

In addition, to optimize neural networks, we used the gradient scaling approach - Landscape Modification (LM) method [11, 10] for the proposed NN architectures and compared them with the results of the default version.

The LM method improves optimization by transforming the objective function $g(x)$ into a modified form $\hat{g}(x)$ controlled by parameters a, and a threshold c. This transformation helps gradient-based optimizers like Adam better avoid local minima and saddle points, leading to faster convergence to global or high-quality local optima. For differentiable functions g, the gradient of $\hat{g}$ computes as [10]:

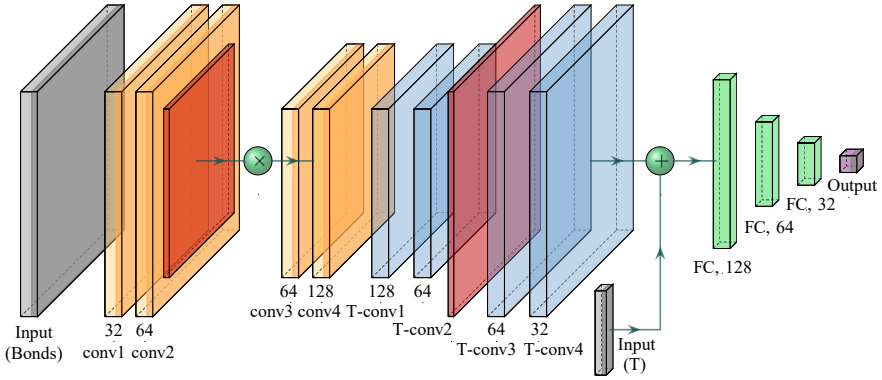


Figure 3: CNN2 architecture.

$$\nabla \hat{g}(x) = \frac{\nabla g(x)}{af((g(x) - c)_+) + 1}. \tag{8}$$

In the special case of choosing $f = 0$ or $a = 0$, we see that $\nabla \hat{g} = \nabla g$ and hence the original landscape is used.

In this context, f is a transformation function that can smooth the optimization landscape based on the values of $g(x)$. Empirical studies show that using a linear function f was the optimal choice in our case, which also supports the hypothesis about the choice of function f put forward in [10]. In turn, the scaling parameter $a \in \mathbb{R}_+$ controls the magnitude of the landscape modification. A range of values such as $1, 0.1, 0.01, ...$ was evaluated to find the optimal balance. The threshold parameter $c \geq \inf_z g(z)$ adjusts the landscape to improve the exploration properties and convergence of the algorithms, and ideally, it should be set as close to the global minimum $g(x)$ as possible. The three parameters f, a and c are to be chosen suitably. The function $f : \mathbb{R}_+ \to \mathbb{R}_+$ is assumed to be chosen such that it satisfies $f(0) = 0$. A commonly used choice of f is given by $f(x) = x$.

In the context of the problem under consideration, we perform the next scaling procedure: the gradient ∇g_t is scaled using a function f of the running loss rl_t and the parameter c_t on a step t:

$$\nabla \hat{g}_t = \frac{\nabla g_t}{af((rl_t - c_t)_+) + 1}, \tag{9}$$

where c_t was set close to the minimum value of the loss function of the default NN model.

In this way, the optimized architectures CNN1LM and CNN2LM are transformed from CNN1 and CNN2 as follows:

- CNN1LM: $Conv2D : (32, 3 \times 3) \to Conv2D : (64, 3 \times 3) \to MaxPool : (2 \times 2) \to Dropout : (0.2) \to Conv2D : (128, 3 \times 3) \to Conv2D : (256, 3 \times 3) \to MaxPool : (2 \times 2) \to GAP \to Concatenate$: (Temp input) $\to Dense : (128) \to Dense : (64) \to Dense : (1)$ $\to$ Landscape Modification for $\nabla \hat{g}_t$;
- CNN2LM: $Conv2D : (32, 3 \times 3) \to Conv2D : (64, 3 \times 3) \to MaxPool : (2 \times 2) \to Dropout : (0.2) \to Conv2D : (64, 3 \times 3) \to Conv2D : (128, 3 \times 3) \to Conv2DTranspose : (128, 3 \times 3) \to Conv2DTranspose : (64, 3 \times 3) \to UpSampling : (2 \times 2) \to Conv2DTranspose : (64, 3 \times 3) \to Conv2DTranspose : (32, 3 \times 3) \to GAP \to Dense : (128) \to Dense : (64) \to Dense : (32) \to$ Landscape Modification for $\nabla \hat{g}_t$.

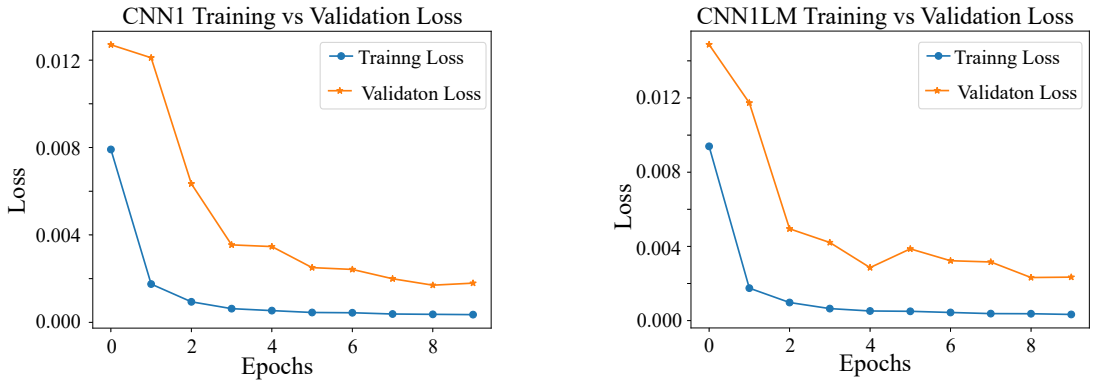


Figure 4: Evolution of the training loss function (MSE) against the epoch number: a) CNN1, b) CNN1LM. These graphs are plotted for the learning process on datasets of the system with size $N = 10 \times 10$.

5 Results and discussion

In this study, we utilized data proportions of 0.8 : 0.15 : 0.05 for training, validation, and testing, respectively. This choice reflects our focus on maximizing the amount of data available for training while still reserving a meaningful subset for validation and testing. We used one data set each for 6×6 and 10×10 systems. These datasets were obtained by combining Monte Carlo and Complete enumeration results for systems of the same size in a 1:1 ratio. The models were trained first on a dataset for 6×6 dimensions and then further trained on a dataset for 10×10. The number of epochs for each training step was 10 and the batch size was 256.

We tested two neural network architectures on described datasets: CNN1 represents the first architecture without the LM approach, as illustrated in Fig. 2. CNN1LM uses the same architecture but incorporates LM. CNN2 represents a more complex architecture, as shown in Fig. 3 as well as CNN2LM, but with LM. Results show that CNN1 performed well for 10×10, while CNN1LM with the addition of LM improved performance for both SG sizes. CNN2 and CNN2LM outperformed the first models, demonstrating the importance of the enhanced architecture, see Table 1. In this study, we tested two hyperparameter selection strategies for LM: with respect to 6×6 and 10×10 systems, which may explain the differences in the obtained results. This outcome could be attributed to suboptimal choices of the hyperparameters f, a, or c, as identifying the ideal values can be challenging, particularly since the optimal settings may vary depending on the system size. In this case, the adaptive tuning strategy for c may be a better option, we plan to test this in further research.

	RMSE			
	CNN1	CNN1LM	CNN2	CNN2LM
6×6	0.01182	0.00338	0.00249	0.00437
10×10	0.00178	0.00309	0.00187	0.00172

Table 1: The root mean squared error (RMSE) metric for the predicted $\langle E \rangle(T)$ using the proposed CNN architectures.

The root mean square errors (RMSE) of the mean energy were calculated with the different CNNs from the initial values obtained with the replica exchange Monte Carlo and complete enumeration. The RMSE was calculated for 6×6 and 10×10 over the entire test dataset and averaged over both temperature and different test configurations of the exchange integral J distribution. The resulting RMSE values depending on CNN architecture and system size are presented in Table 1. The table shows that the CNN models with Landscape Modification showed slightly more accurate results than the regular version. The graphs of the loss function value $\mathscr{L}$ as a function of the number of training epochs are shown in Fig. 4 for CNN1 and CNN1LM. Loss metric reflects model performance improvements over epochs.

Figure 5 shows test examples of mean energy calculation using a complete enumeration and

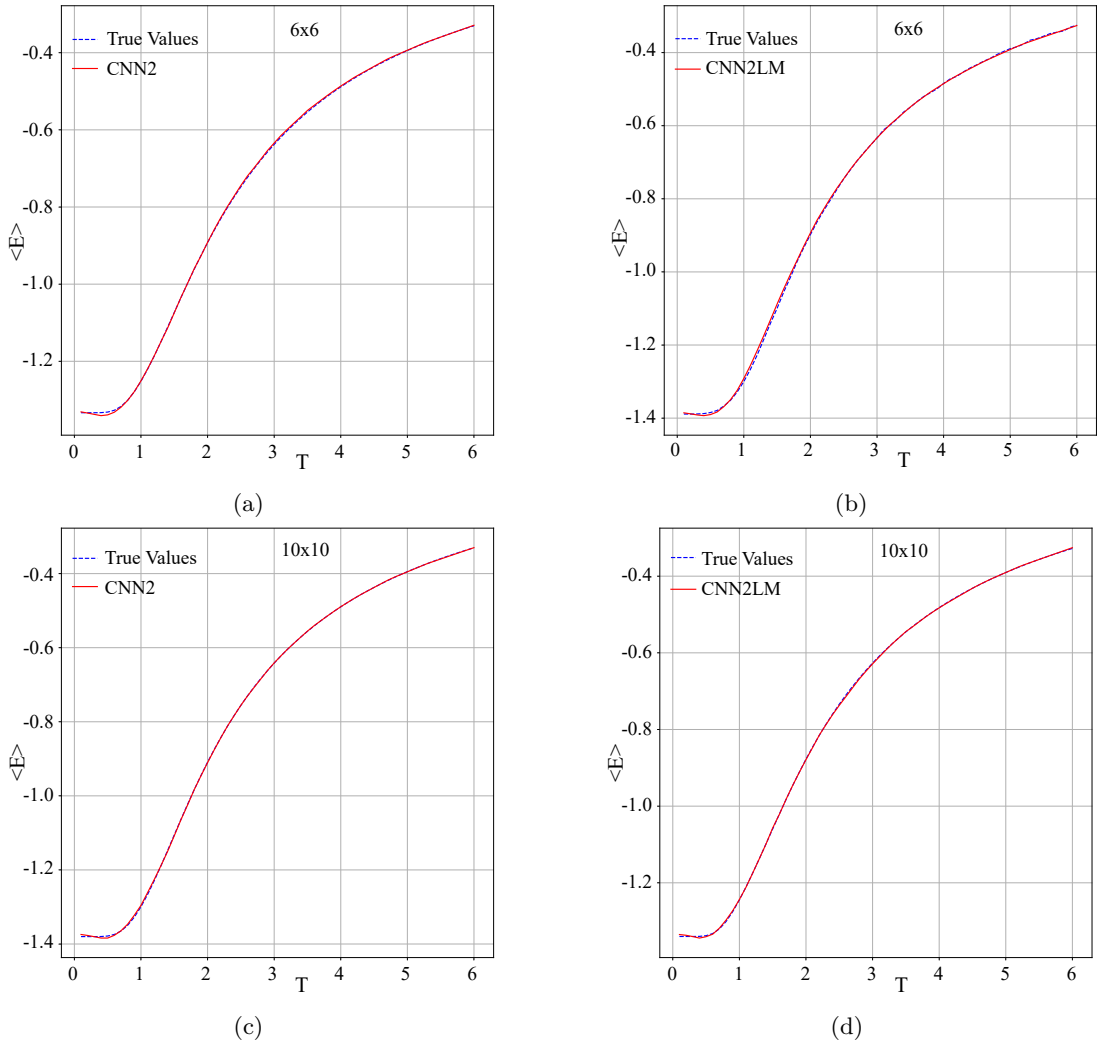
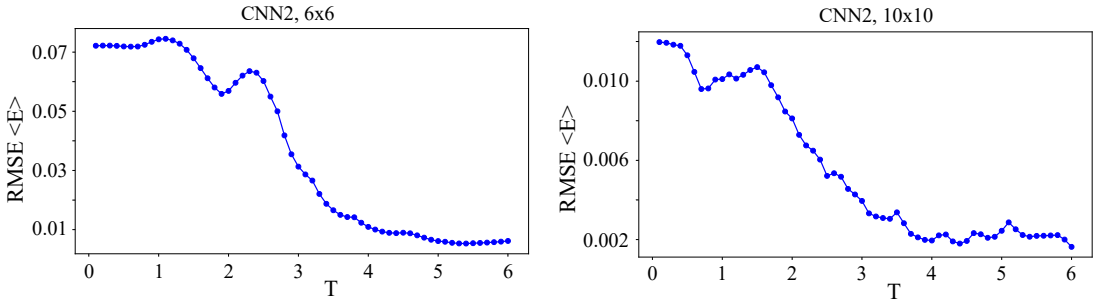


Figure 5: Comparison of mean energy-temperature dependencies for random configurations obtained using the exact method and neural networks (CNN2 and CNN2LM) for different system sizes: a) 6×6, CNN2; b) 6×6, CNN2LM; c) 10×10, CNN2; d) 10×10, CNN2LM.



(a) (b)

Figure 6: RMSE dependencies on temperature for CNN2 a) 6×6, b) 10×10.

CNNs of different architectures (CNN2 and CNN2LM). These figures show well the very high accuracy of the proposed approach at $T > 1$ for all considered CNN architectures. In the region $T < 1$, the difference between the architectures already starts to be seen. One can observe that CNN2 and CNN2LM show results closer to the original values, which is due to differences in architecture and training optimisations. The RMSE dependences on temperature for CNN2 are shown in Fig. 6. These figures clearly show that the CNN accuracy decreases with decreasing temperature. This behaviour is explained by the dramatically increasing influence of the ground states of the spin glass on the thermodynamic characteristics as the temperature approaches 0.

6 Conclusion

In this paper, we considered the regression problem of predicting the average energy $\langle E \rangle$ as a function of temperature T for spin glasses on a square lattice using machine learning methods. Representing the spin glass as a weighted graph with exchange interactions as edges, we explored the relationship between the spatial distribution of connections and $\langle E \rangle$ using CNNs. We tested several CNN architectures and found that increasing the complexity and depth of the NN architecture and implementing the LM approach improves performance in some cases. We assume that increasing the training dataset as well as optimising the architecture of CNN can further increase the accuracy of this method.

7 Acknowledgement

V. Kapitan performed the optimization using the LM method and contributed to the manuscript with financial support provided by the project “MAPLE” with grant number 22-5715-P0001 under the National University of Singapore Faculty of Science Ministry of Education Tier 1 grant Data for Science and Science for Data collaborative scheme.

References

[1] A. Altieri and M. Baity-Jesi. An introduction to the theory of spin glasses. In Tapash Chakraborty, editor, *Encyclopedia of Condensed Matter Physics (Second Edition)*, pages 361–370. Academic Press, Oxford, second edition edition, 2024.

[2] P. Andriushchenko. Influence of cutoff dipole interaction radius and dilution on phase transition in kagome artificial spin ice. *Journal of Magnetism and Magnetic Materials*, 476:284–288, 2019.

[3] P. Andriushchenko, D. Kapitan, and V. Kapitan. A new look at the spin glass problem from a deep learning perspective. *Entropy*, 24(5):697, 2022.

[4] P. Andriushchenko, K. Soldatov, A. Peretyatko, Y. Shevchenko, K. Nefedev, H. Otsuka, and Y. Okabe. Large peaks in the entropy of the diluted nearest-neighbor spin-ice model on the pyrochlore lattice in a [111] magnetic field. *Physical Review E*, 99(2):022138, 2019.

[5] A. Barzegar, C. Pattison, W. Wang, and H.G. Katzgraber. Optimization of population annealing monte carlo for large-scale spin-glass simulations. *Physical Review E*, 98(5):053308, 2018.

[6] M. Bukov, M. Schmitt, and M. Dupont. Learning the ground state of a non-stoquastic quantum hamiltonian in a rugged neural network landscape. *SciPost Physics*, 10(6):147, 2021.

[7] K.T. Butler, D.W. Davies, H. Cartwright, O. Isayev, and A. Walsh. Machine learning for molecular and materials science. *Nature*, 559(7715):547–555, 2018.

[8] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová. Machine learning and the physical sciences. *Reviews of Modern Physics*, 91(4):045002, 2019.

[9] J. Carrasquilla and R.G. Melko. Machine learning phases of matter. *Nature Physics*, 13(5):431–434, 2017.

[10] M. Choi. Improved metropolis–hastings algorithms via landscape modification with applications to simulated annealing and the curie–weiss model. *Advances in Applied Probability*, 56(2):587–620, 2024.

[11] M. Choi and Y. Wang. Improved langevin monte carlo for stochastic optimization via landscape modification. *arXiv preprint arXiv:2302.03973*, 2023. https://arxiv.org/abs/2302.03973.

[12] M. J. Harris, S.T. Bramwell, D.F. McMorrow, T.H. Zeiske, and K.W. Godfrey. Geometrical frustration in the ferromagnetic pyrochlore Ho2Ti2O7. *Physical Review Letters*, 79(13):2554, 1997.

[13] D. Kapitan, A. Korol, E. Vasiliev, P. Ovchinnikov, A. Rybin, E. Lobanova, K. Soldatov, Y. Shevchenko, and V. Kapitan. Chapter one - application of machine learning in solid state physics. In Rair Macedo and Robert L. Stamps, editors, *Solid State Physics*, volume 74 of *Solid State Physics*, pages 1–65. Academic Press, 2023.

[14] V. Kapitan, E. Vasiliev, A. Perzhu, D. Kapitan, A. Rybin, A. Korol, K. Soldatov, and Y. Shevchenko. Numerical simulation of magnetic skyrmions on flat lattices. *AIP Advances*, 11(1):015041, 2021.

[15] B. Kaufman. Crystal statistics. ii. partition function evaluated by spinor analysis. *Physical Review*, 76(8):1232, 1949.

[16] S. Kirkpatrick and D. Sherrington. Infinite-ranged models of spin-glasses. *Physical Review B*, 17(11):4384, 1978.

[17] A.O. Korol, V.Yu. Kapitan, A.V. Perzhu, M.A. Padalko, D.Yu. Kapitan, R.A. Volotovskii, E.V. Vasiliev, A.E. Rybin, P.A. Ovchinnikov, P.D. Andriushchenko, Yu.A. Makarov, A.G. Shevchenko, I.G. Il'yushin, and K.S. Soldatov. Calculation of the ground states of spin glasses using the restricted boltzmann machine. *JETP Letters*, 115(8):500, 2022.

[18] A. Kovtanyuk, K. Nefedev, and I. Prokhorov. Advanced computing method for solving of the polarized-radiation transfer equation. In *Russia-Taiwan Symposium on Methods and Tools of Parallel Processing*, pages 268–276. Springer, 2010.

[19] D. Landau and K. Binder. *A guide to Monte Carlo simulations in statistical physics*. Cambridge university press, 2021.

[20] A. Lucas. Ising formulations of many np problems. *Frontiers in physics*, 2:5, 2014.

[21] A. Makarov, K. Makarova, Y. Shevchenko, P. Andriushchenko, V. Kapitan, K. Soldatov, A. Perzhu, A. Rybin, D. Kapitan, E. Vasil'ev, et al. On the numerical calculation of frustrations in the ising model. *JETP Letters*, 110(10):702–706, 2019.

[22] K. Makarova, A. Makarov, V. Strongin, Iu. Titovets, Y. Shevchenko, V. Kapitan, A. Rybin, D. Kapitan, A. Korol, E. Vasiliev, et al. Canonical monte carlo multispin cluster method. *Journal of Computational and Applied Mathematics*, 427:115153, 2023.

[23] M. Padalko, Y. Shevchenko, V. Kapitan, and K. Nefedev. Parallel computing of edwards-anderson model. *Algorithms*, 15(1):13, 2022.

[24] Y. Shevchenko, A. Makarov, P. Andriushchenko, and K. Nefedev. Multicanonical sampling of the space of states of h(2, n)-vector models. *Journal of Experimental and Theoretical Physics*, 124(6):982–993, 2017.

[25] K. Shiina, H. Mori, Y. Okabe, and H.K. Lee. Super-resolution of spin configurations based on flow-based generative models. *Journal of Physics A: Mathematical and Theoretical*, 2024.

[26] K. Soldatov, A. Peretyatko, P. Andriushchenko, K. Nefedev, and Y. Okabe. Comparison of diluted antiferromagnetic ising models on frustrated lattices in a magnetic field. *Physics Letters A*, 383(12):1229–1234, 2019.

[27] K.S. Soldatov, K.V. Nefedev, V.Y. Kapitan, and P.D. Andriushchenko. Approaches to numerical solution of 2d ising model. In *Journal of Physics: Conference Series*, volume 741, 012199, 2016.

[28] P. Suchsland and S. Wessel. Parameter diagnostics of phases and phase transition learning by neural networks. *Physical Review B*, 97(17):174435, 2018.

[29] R.H. Swendsen and J. Wang. Replica monte carlo simulation of spin-glasses. *Physical review letters*, 57(21):2607, 1986.

[30] E. Vasil'ev, A. Perzhu, A. Korol, D. Kapitan, A. Rybin, K. Soldatov, and V. Kapitan. Numerical simulation of two-dimensional magnetic skyrmion structures. *Computer Research and Modeling*, 12(5):1051–1061, 2020.