

Научная статья  
УДК 004.032.26, 536.911  
DOI 10.25205/2541-9447-2022-17-2-5-15

## Нейросетевой метод расчета точки Кюри двумерной модели Изинга

Алена Олеговна Король<sup>1</sup>,  
Константин Валентинович Нефедев<sup>2</sup>,  
Виталий Юрьевич Капитан<sup>3</sup>

<sup>1-3</sup>Дальневосточный федеральный университет,  
Владивосток, Россия

<sup>1</sup>korol.ao@dvfu.ru

<sup>2</sup>nefedev.kv@dvfu.ru

<sup>3</sup>kapitan.vyu@dvfu.ru

### Аннотация

В работе рассматривается метод определения критической точки фазового перехода II рода модели Изинга на квадратной решётке с применением свёрточной нейронной сети. Данные для обучения и анализа получены с помощью Монте-Карло моделирования (алгоритм Метрополиса). Нейронная сеть обучалась на данных, соответствующих низкотемпературной фазе – ферромагнитной и высокотемпературной – парамагнитной, соответственно. После обучения нейронная сеть анализировала входные данные из всего температурного диапазона: от 0,1 до 5,0 (в безразмерных величинах  $J$ ) и определяла точку Кюри  $T_c$ .

### Ключевые слова

модель Изинга, точка Кюри, метод Монте-Карло, свёрточная нейронная сеть

### Источник финансирования

Работа выполнена при финансовой поддержке гранта Президента Российской Федерации для государственной поддержки ведущих научных школ РФ № НШ-2559.2022.1.2.

### Для цитирования

Король А. О., Нефедев К. В., Капитан В. Ю. Нейросетевой метод расчета точки Кюри двумерной модели Изинга // Сибирский физический журнал. 2022. Т. 17, № 2. С. 5–15. DOI 10.25205/2541-9447-2022-17-2-5-15

## Neural Network Method for Calculation of the Curie Point of the Two-Dimensional Ising Model

Alena O. Korol<sup>1</sup>,  
Konstantin V. Nevedev<sup>2</sup>,  
Vitalii Yu. Kapitan<sup>3</sup>

<sup>1</sup>Far Eastern Federal University,  
Vladivostok, Russia

<sup>1</sup>korol.ao@dvfu.ru

<sup>2</sup>nefedev.kv@dvfu.ru

<sup>3</sup>kapitan.vyu@dvfu.ru

### Abstract

The authors describe a method for determining the critical point of a second order phase transitions using a convolutional neural network based on the Ising model on a square lattice. Data for training and analysis were obtained using Monte Carlo simulations. The neural network was trained on the data corresponding to the low-temperature phase, that is a ferromagnetic one and high-temperature phase, that is a paramagnetic one, respectively. After training, the neural network analyzed input data from the entire temperature range: from 0.1 to 5.0 (in dimensionless units  $J$ ) and determined the Curie point  $T_c$ .

*Keywords*

Ising model, Curie point, Monte Carlo method, Convolutional neural network

*Funding*

This work was supported by a grant from the President of the Russian Federation for State Support of Leading Scientific Schools of the Russian Federation No. NSh-2559.2022.1.2.

*For citation*

Korol A. O., Nefedev K. V., Kapitan V. Yu. Neural network method for calculation of the Curie point of the two-dimensional Ising model. *Siberian Journal of Physics*. 2022. Vol. 17, no. 2, pp. 5–15. (in Russ.) DOI 10.25205/2541-9447-2022-17-2-5-15

## Введение

В основополагающих научных работах [1; 2], а также в современных трудах [3–5], большое внимание уделяется решеточным структурам. Например, конденсированные среды – твердые растворы с магнитными свойствами, могут быть рассмотрены как решетки из атомных или ионных магнитных моментов, каждый из которых локализован в своем конкретном месте [1; 6]. Взаимодействия между спинами в узлах решетки могут приводить к коллективному поведению и макроскопическим эффектам, например, таким широко известным как ферромагнетизм или антиферромагнетизм. Причиной спонтанного намагничивания или антиферромагнитного упорядочения является обменное взаимодействие, теория которого была представлена Гейзенбергом и Дираком в конце 1920 г. Исследование влияния пространственного упорядочения наночастиц, их геометрии, формы и законов взаимодействия между ними на термодинамические свойства и магнитные состояния имеют как фундаментальный, так и практический интерес. Как следствие этого, необходимо исследовать подходы и возможные способы управления вероятностью реализации определенных магнитных состояний ансамблей наночастиц [7–9].

Моделирование Монте-Карло или многократное имитационное моделирование вероятностей представляет собой класс вычислительных методов, с помощью которых можно оценить возможные результаты неопределенного события с помощью инструментов случайной выборки [10]. Применение суперкомпьютеров позволяет использовать новые классы алгоритмов и оперировать большими и сверхбольшими объемами данных при проведении численных экспериментов. Численные методы и компьютерное моделирование с использованием суперкомпьютеров имеют первостепенную важность в статистической и математической физике, нанопизике, статистической термодинамике, поскольку суперкомпьютеры существенно ускоряют решение задач и/или позволяют решать задачи, которые невозможно решить аналитическими методами.

Благодаря развитию суперкомпьютерных вычислительных систем [11], алгоритмы продолжают активно развиваться и совершенствоваться [12; 13]. При этом для физики конденсированного состояния характерна постоянно возрастающая сложность анализа исследуемых конфигураций в зависимости от числа частиц и уже предложены методы машинного обучения, позволяющие классифицировать и анализировать сложные наборы данных [14], но в настоящий момент ученые только начинают раскрывать весь потенциал внедрения методов машинного обучения в свои исследования [15].

## 1. Описание модели и методов

В данном исследовании рассматривается метод определения температуры Кюри двумерной модели Изинга с применением свёрточной нейронной сети. Для этого был использован алгоритм Метрополиса для генерации входных данных нейронной сети, после чего было проведено сравнение с результатами МК-моделирования, точным решением Онзагера и выходными данными после обучения нейронной сети. Все величины в работе приведены в безразмерных единицах  $J$ .

### 1.1. Модель Изинга

Ферромагнитное упорядочение возникает, когда набор атомных спинов выстраивается таким образом, где все связанные с ним магнитные моменты коллинеарны в одном направлении, что дает чистый магнитный момент макроскопического размера. Простейшее теоретическое описание ферромагнетизма задается моделью Изинга – одной из математических моделей статистической физики, применяемой для исследования фазовых переходов и критических точек. Данная модель имеет точное решение, полученное Онзагером:  $\frac{T_c}{J} = \frac{2}{\ln(1+\sqrt{2})} = 2.269$  [16], с которым в настоящей работе сравнивались данные, получаемые другими методами.

В работе использовался гамильтониан для квадратной решетки спинов Изинга с четырьмя ближайшими соседями и периодическими граничными условиями. Математическая модель представляет собой совокупность дискретных переменных (значения магнитных моментов атомных спинов), которые могут принимать одно из двух значений:  $S_i = \pm 1$ , соответствующих одному из двух состояний. Система спинов Изинга имеет размер  $N = L \times L$  и гамильтониан задается следующим выражением:

$$H = -J \sum_{\langle i,j \rangle} S_i S_j \quad (1)$$

где  $J = 1$  – константа ферромагнитного взаимодействия [16].

### 1.2. Алгоритм Метрополиса

Алгоритм Метрополиса представляет собой Монте-Карло метод с цепью Маркова для получения последовательности случайных выборок из распределения вероятностей, для которых затруднена прямая выборка. Конфигурация системы с определенной вероятностью меняется на каждой итерации выборки за счет изменения ориентации случайно выбранного спина; алгоритм был детально рассмотрен ранее [17]. Несмотря на то, что алгоритм Метрополиса является часто применяемым инструментом при исследовании классических систем, вблизи критических точек эффективность резко снижается и возникает так называемое «критическое замедление», и движение системы в фазовом пространстве становится очень медленным. Соответственно для разрешения этой проблемы используются другие методы Монте-Карло, но в пользу большей точности, мы жертвуем скоростью вычислений и размерами исследуемых систем [12; 17]. Одной из перспективных альтернатив рассматривается применение нейронных сетей.

### 1.3. Нейросетевые алгоритмы глубокого обучения

Применение нейронных сетей в компьютерном моделировании приобретает все большую популярность в силу своей точности, производительности и наличия множества программных библиотек [14].

Нейронные сети для глубокого обучения представляют собой взвешенный направленный граф, узлы которого (нейроны) расположены слой за слоем. Каждый нейрон предыдущего слоя связан с нейронами последующего. Эти связи между ними имеют определенные веса, которые являются основной частью нейронных сетей: от весов связей зависит предсказание нейронной сети. Каждый нейрон имеет активационную функцию, которая отвечает за вес связи на его выходе. Таким образом, каждый слой архитектуры анализирует значения нейронов входного слоя, а активационные функции преобразуют значения весов между нейронами.

Веса нейронной сети изначально инициализируются случайным образом, а затем обновляются каждую эпоху обучения так, чтобы повысить общую точность сети. Для вычисления ошибки обучения – в каждую эпоху метки обучающих данных сравниваются с фактическими данными с помощью функции потерь, и затем вес обновляется соответствующим образом.

То, как обновить вес, регулирует оптимизатор, т. е. алгоритм, используемый для изменения параметров сети, таких как вес и скорость обучения, с целью минимизации значения функции потерь. Для оценки производительности модели используют метрики, представляющие собой функции, которые аналогичны функциям потерь, за исключением того, что результаты оценки метрики не используются при обучении модели.

В машинном обучении задача классификации относится к классу методов обучения с учителем. Такой метод предполагает, что для каждого из множества входных векторов существует целевой вектор, который представляет из себя желаемый выход. Размер выборки для обучения сети должен быть достаточно большим для формирования корректного отображения. Чтобы сеть корректно и быстро обучилась, необходимо правильно подобрать размер пакета данных для обучения. Размер пакета выбирают максимально возможным в зависимости от вычислительной мощности GPU. В свою очередь, количество эпох обучения зависит от размера обучающей выборки.

Обучение с учителем методом обратного распространения ошибки проходит в несколько этапов, которые повторяются на протяжении заданного количества эпох:

- 1) расчет выхода сети для каждого экземпляра входных данных;
- 2) сравнение полученного результата с соответствующим значением целевого вектора;
- 3) расчет ошибки с помощью функции потерь;
- 4) обратное распространение ошибки на вход;
- 5) корректировка параметров сети в соответствии с выбранным алгоритмом оптимизации (при условии ошибочного ответа).

Критерием останова обучения является заданный уровень ошибки на обучающей выборке [18].

### 1.3.1. Структура свёрточной нейронной сети

В данной работе была реализована архитектура свёрточной нейронной сети (convolutional neural network – CNN) [19; 20] с помощью программной библиотеки *TensorFlow* [21]. CNN – это класс глубоких нейронных сетей, наиболее часто применяемых для анализа и классификации визуальных образов. Созданная в ходе данного исследования свёрточная нейронная сеть состоит из нескольких слоев (см. рис. 1).

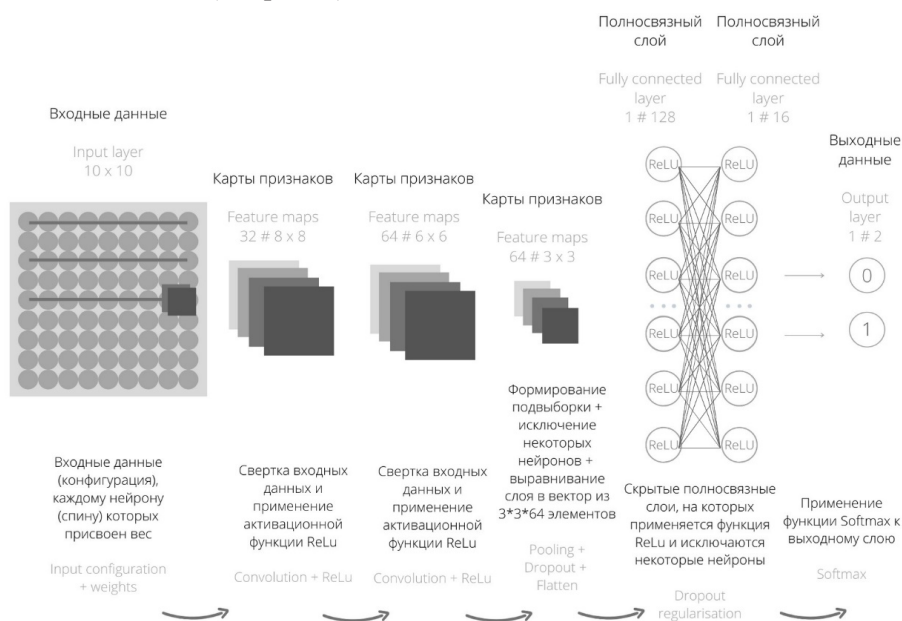


Рис. 1. Структура свёрточной нейронной сети

Входной слой содержит исходные значения спинов конфигурации модели Изинга. Над входным слоем производится операция свертки с использованием фильтра  $3 \times 3$ , а именно – выделение признаков и формирование карты признаков (feature map) (см. рис. 2).

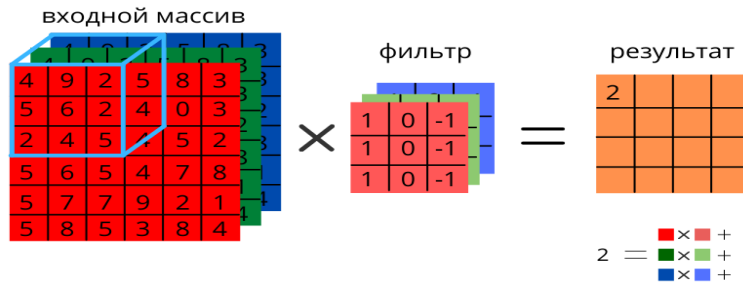


Рис. 2. Свёртка с фильтром  $3 \times 3$

Для выделения признаков в каком-либо слое используют фильтры или так называемые «ядра». При движении фильтра по матрице происходит умножение каждого элемента фильтра на значение нейрона, далее значения суммируются и записываются в карту признаков. Если некоторая искомая характеристика присутствует в матрице, операция свёртки на выходе будет выдавать число с относительно большим значением. Если же характеристика отсутствует, выходное число будет маленьким:

$$x_j^l = f(\sum_i x_i^{l-1} * k_j^l + b_j^l), \tag{2}$$

где  $x_j^l$  – карта признаков  $j$  (выходной сигнал на уровне  $l$ ),  $f()$  – функция активации, применяемая поэлементно,  $b_j^l$  – пороговое значение (сдвиг) для карты признаков  $j$ ,  $k_j^l$  – ядро свёртки с номером  $j$ ,  $x_i^{l-1}$  – карты признаков предыдущего уровня,  $*$  – операция свёртки.

Для того, чтобы обучение весов было эффективным, после операции свёртки следует ввести некоторое смещение (bias) и нелинейность (функцию активации), в качестве функции активации была использована *ReLU* (Rectified Linear Unit – усеченное линейное преобразование), эта кусочно-линейная функция, задается следующим образом:

$$f(x) = \max(0, x). \tag{3}$$

Данное преобразование заменяет все отрицательные значения нейронов в карте объекта на ноль и, таким образом, решает проблему аннулирования, а также приводит к гораздо более разреженному виду функции активации на его выходе. Разреженность в первую очередь обеспечивает нечувствительность к небольшим изменениям на входе, таким как шум [22].

Слой подвыборки (pooling layer) отвечает за уменьшение пространственного размера карт признаков. Несмотря на то, что он уменьшает размерность каждой карты, он сохраняет наиболее важную информацию (в разработанной сети  $pool_{size} = 2$ ). Существуют различные стратегии подвыборки: максимальная, средняя и вероятностная. Подвыборка помогает сети быть инвариантной к небольшим трансформациям и искажениям данных входного слоя. Это также уменьшает количество параметров и вычислений в сети, а также минимизирует вероятность переобучения:

$$x^l = f(a^l \text{subsample}(x^{l-1}) + b^l),$$

где  $x^l$  – выход уровня  $l$ ,  $f()$  – функция активации, применяемая поэлементно,  $a^l$ ,  $b^l$  – коэффициенты, *subsample* – операция выборки локальных максимальных значений (при условии, что выбрана стратегия максимальной подвыборки), т. е. максимальных значений в каждом окне *max-pooling*.

При тестировании нейронной сети нейроны уже не выбрасываются, но выход каждого нейрона умножается на величину  $(1 - p)$ .

После нескольких свёрточных слоев и блока подвыборки, трехмерное представление входных данных разворачивается в вектор – так называемый полносвязный слой (fully connected layer), который далее будет передан в многослойный перцептрон – обычную полносвязную нейронную сеть. Полносвязные слои используются для классификации. Все слои, идущие до полносвязных, используются для выделения различных признаков, которые подаются на вход классификатору:

$$x_j^k = f \left( \sum_{i=1}^m w_{ij} x_i^{k-1} \right), \quad (5)$$

где  $x_j^k$  – выходной сигнал для нейрона  $j$  на слое  $k$ ,  $x_i^{k-1}$  – входной сигнал,  $w_{ij}$  – весовой коэффициент,  $f()$  – функция активации.

Выходной слой отвечает за формирование вероятностей принадлежности входного образа тому или иному классу и определяется значениями нейронов выходных вероятностей нейронной сети. Сумма выходных вероятностей полносвязного слоя составляет 1, что обеспечивается путем использования *Softmax* в качестве функции активации. Пусть  $M$  – количество нейронов в уровне с функцией активации *Softmax*,  $s_j$  – сигнал активации на  $j$ -ом нейроне. Тогда выходом на  $j$ -ом нейроне с функцией активации *Softmax* будет  $f_j(s_1, \dots, s_M)$ :

$$f_j(s_1, \dots, s_M) = \frac{e^{s_j}}{\sum_{k=1}^M e^{s_k}}, j = 1, \dots, M. \quad (6)$$

Функция *Softmax* принимает вектор произвольных реальных значений и сжимает его до вектора значений между нулем и единицей.

### 1.3.2. Процесс обучения

После инициализации необходимо скомпилировать и обучить модель. Аргументами метода *compile* являются: функция потерь, оптимизатор и метрика. Для задач классификации используют различные функции потерь. Поскольку в нашем исследовании используется бинарная система маркировки, была использована *BinaryCrossentropy*, и, соответственно, в качестве метрики – *BinaryAccuracy*. В качестве оптимизатора использован *Adam*, который является одним из лучших среди адаптивных оптимизаторов.

Обучение модели задается методом *fit*, основными аргументами данного метода являются: массивы входных данных и целевых меток, количество экземпляров входных данных на одно обновление параметров ( $\text{batch}_{\text{size}}$ ) и количество эпох обучения (*epochs*). В качестве входных данных выступают конфигурации систем  $10 \times 10$  и  $20 \times 20$  спинов Изинга, в качестве целевых меток: 0 – если температура конфигурации ниже критической или 1 в противном случае. Гиперпараметры:  $\text{batch}_{\text{size}} = 100$  и *epochs* = 6.

Переобучение сети происходит, когда точность результатов на учебной выборке больше точности на тестовой выборке. Для того, чтобы исключить возможность переобучения сети, используется метод *Dropout*, который исключает нейроны из сети с заданной вероятностью  $p$  и они не вносят вклад в дальнейшее обучение сети. Таким образом, получается одна из возможных  $2^N$  структур связей между нейронами, где  $N$  – суммарное число нейронов [18; 21].

## 2. Результаты и обсуждение

В ходе исследования были проверены различные наборы входных данных нейронной сети, полученные при различных параметрах работы алгоритма Метрополиса для систем  $10 \times 10$  и  $20 \times 20$  спинов Изинга, для выбора оптимальных параметров моделирования, которые будут

в дальнейшем применяться при изучении более сложных спиновых систем. Проведен сравнительный анализ с результатами МК-моделирования и точным решением Онзагера.

На рис. 3, *а*, *б* представлены результаты применения свёрточной нейронной сети к вычислению критической точки  $T_c$  в сравнении с точным решением Онзагера и результатами МК-моделирования. На первом этапе сеть обучалась на спиновых конфигурациях, полученных в ходе МК-моделирования со следующими параметрами: размер системы:  $10 \times 10$   $T = 0,1 \dots 5,0$  с шагом 0,01, число МК-шагов для предварительного уравнивания системы: 10000, число МК-шагов для расчета термодинамических средних в алгоритме Метрополиса: 10000, размер выборки конфигураций для обучения сети: 50 на один шаг по температуре, результаты представлены на рис. 3, *а*. При этом в качестве функции активации в нейронной сети, был построен график сигмоиды:  $f(s) = 1/(1 + \exp(-s))$  на котором можно увидеть, как обучалась сеть на данных, полученных ниже и выше области критической температуры, соответственно. На рис. 3, *б* показан результат анализа более «грубых» входных данных: шаг по  $T = 0,1$ , а число МК-шагов было сокращено до 1000 при прочих одинаковых параметрах моделирования. Как видно из графиков, оценка  $T_c$  для алгоритма Метрополиса ухудшилась с уменьшением числа МК-шагов и увеличением шага по температуре. На результаты работы нейронной сети это не повлияло.

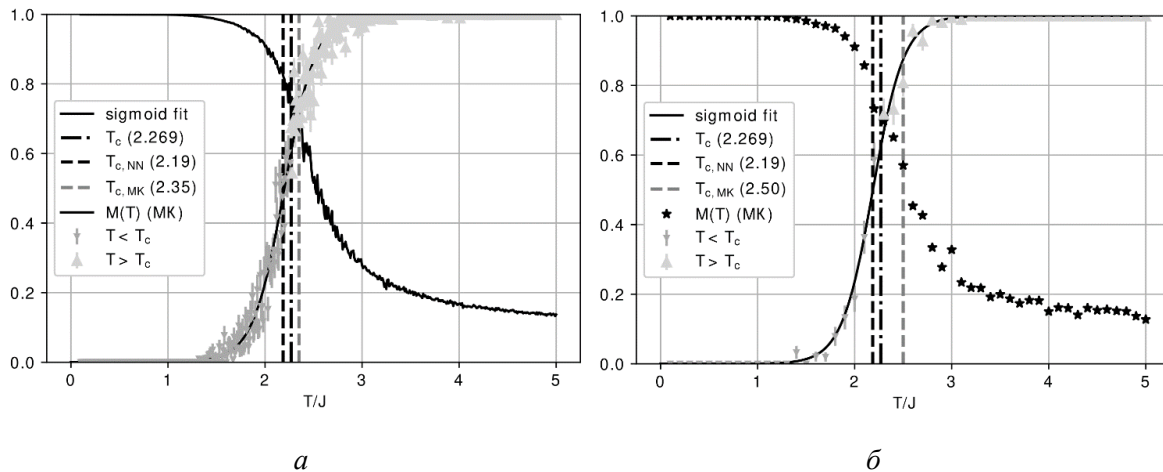


Рис. 3. Результаты расчетов  $T_c$  различными методами для системы  $10 \times 10$  спинов Изинга: *а* – шаг по  $T = 0,01$ , число МК-шагов равно 10000; *б* – шаг по  $T = 0,1$ , число МК-шагов равно 1000

Также было проверено влияние размера системы на результат работы сети, на примере системы  $20 \times 20$ . В ходе моделирования также были использованы различные параметры для МК-моделирования, аналогично описанию, приведенному выше: изменялась величина шага по  $T$  и число МК-шагов. Полученные результаты в целом аналогичны приведенным выше. Стоит отметить, что увеличение размера системы положительно повлияло на результаты МК-моделирования за счет уменьшения влияния размерного эффекта при расчетах критической температуры:  $T_c = 2,29$ , но существенно не отразилось на результатах работы нейронной сети – предсказанное значение точки Кюри составило  $T_c = 2,21$ . Однако, в некоторых численных экспериментах отклонение получаемых значений критической точки от точного решения Онзагера  $T_c^J = 2.269$  увеличились, так как обучение нейронной сети строится на вероятностном подходе. Также можно отметить, что авторы не зафиксировали существенной разницы между обучением сначала на точных данных, а потом проверки на более грубых, и наоборот. Результаты в обоих случаях были примерно одинаковы. Увеличение размера выборки и для обучения, и для тестирования также сильно не повлияли на качество работы свёрточной нейронной сети.

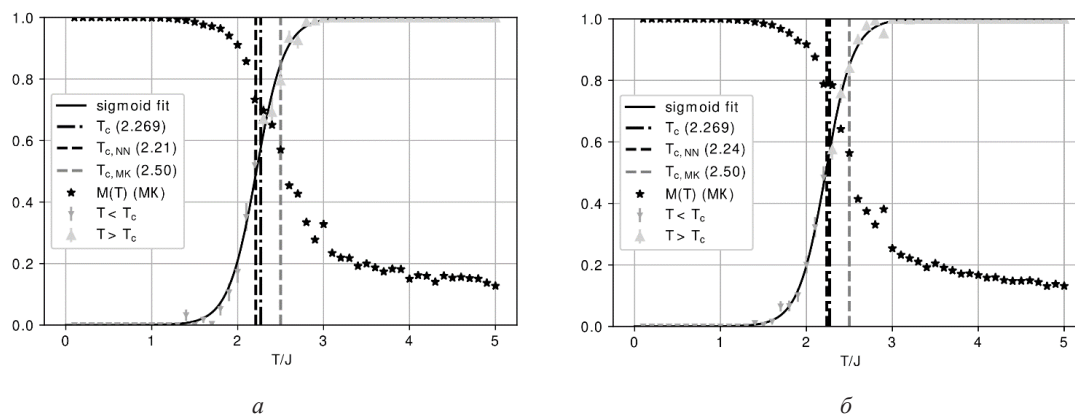


Рис. 4. Результаты расчетов  $TC$  различными методами для системы  $10 \times 10$  спинов Изинга:  $a$  – выборка равна 50;  $b$  – выборка равна 100

С точки зрения дальнейшего использования нейронных сетей при анализе спиновых систем, заслуживающий внимания результат был зафиксирован, когда сеть обучалась на «грубых» данных, аналогичных для случая описанного на рис. 3,  $b$  с числом выборки 50. А вот тестирование проводилось на увеличенной до 100 образцов выборке. При этом другие параметры МК-моделирования не менялись (см. рис. 4,  $a$ ,  $b$ ).

При анализе большей по размеру (чем обучающей) выборки были получены более точные данные (см. рис. 4,  $b$ ). Уменьшение обучающей выборки также позволило сократить время численных экспериментов. В среднем можно отметить большую точность получаемых результатов по сравнению с Монте-Карло моделированием.

### Заключение

В работе было рассмотрено применение нейронных сетей для определения критической температуры фазового перехода II рода в сравнении с известными решениями. Успешно была показана возможность применения сверточных нейронных сетей к задаче классификации спиновых состояний при разных температурах и предсказании точки Кюри в двумерной модели Изинга. Показано влияние количества Монте-Карло шагов и размера выборки на качество обучения сети в сравнении с результатами алгоритма Метрополиса. Рассмотрены системы различных размеров и влияние размерного эффекта на результаты, получаемые нейронной сетью и алгоритмом Метрополиса в сравнении с точным решением Онзагера.

В ходе дальнейших исследований планируется применить нейронные сети к изучению более сложных моделей и решеток, в частности для изучения фрустрированных систем, таких, например, как спиновые стекла.

### Список литературы

1. **Cipra B. A.** An introduction to the Ising model // *The American Mathematical Monthly*, 1987, vol. 94. № 10. P. 937–959.
2. **Fisher D. S., Huse D. A.** Ordered phase of short-range Ising spin-glasses // *Physical review letters*, 1986, vol. 56. № 15. P. 1601.
3. **Kruis J., Maris G.** Three representations of the Ising model // *Scientific reports*, 2016, vol. 6, no. 1, pp. 1–11.
4. **Падалко М. А. и др.** Ускоренный алгоритм исчерпывающего перечисления в модели Изинга // *Дальневосточный математический журнал*. 2019. Т. 19. № 2. С. 235–244.



5. **Прудников В. В. и др.** Исследование маргинального влияния дефектов структуры на неравновесное критическое поведение двумерной модели Изинга // *ЖЭТФ*. 2020. Т. 157. № 2. С. 308.
6. **Anderson P. W.** Ordering and antiferromagnetism in ferrites // *Physical Review*, 1956. Vol. 102. No. 4. P. 1008.
7. **Narita N. et al.** Design and numerical study of flux control effect dominant MAMR head: FC writer // *IEEE Transactions on Magnetics*, 2020. Vol. 57. No. 3. P. 1–5.
8. **Васильев Е. В. и др.** Численное моделирование двумерных магнитных скирмионных структур // *Компьютерные исследования и моделирование*. 2020. Т. 12. № 5. С. 1051–1061.
9. **Солдатов К. С. и др.** Конечно-размерный скейлинг в ферромагнитных спиновых системах на решетке пирохлора // *Дальневосточный математический журнал*. 2020. Т. 20. № 2. С. 255–266.
10. **Landau D., Binder K.** *A guide to Monte Carlo simulations in statistical physics*. Cambridge university press, 2003.
11. **Шаповалова К. В. и др.** Зачем суперкомпьютер Дальневосточному федеральному университету? // *Современные наукоемкие технологии*. 2017. № 1. С. 81–87.
12. **Шаповалова К. В. и др.** Методы канонического и мультиканонического семплирования пространства состояний векторных моделей // *Дальневосточный математический журнал*. 2017. Т. 17. № 1. С. 124–130.
13. **Макаров А. Г. и др.** К численному расчету фрустраций в модели Изинга // *Письма в Журнал экспериментальной и теоретической физики*. 2019. Т. 110. № 10. С. 700–705.
14. **Carrasquilla J., Melko R.G.** Machine learning phases of matter // *Nature Physics*, 2017. Vol. 13. No. 5. Pp. 431–434.
15. **Kenta S., et al.** Machine-Learning Studies on Spin Models // *Scientific reports*, 2020. Vol. 10. No. 1. Pp. 1–6.
16. **Onsager L.** A Two-Dimensional Model with an Order-Disorder Transition // *Phys. Rev.*, 1944, vol. 65. pp. 117–149.
17. **Капитан В. Ю., Шевченко Ю. А., Андрущенко П. Д., Нефедев К. В.** Суперкомпьютерное моделирование и численные решения задач статфизики, Изд-во ДВФУ, Вл-к. 2017. 195 с.
18. **Goodfellow I., Bengio Y., Courville A.** *Deep learning*. MIT press, 2016.
19. **LeCun Y. et al.** Handwritten digit recognition with a back-propagation network // *Advances in Neural Information Processing Systems (NIPS)*, 1989. Pp. 396–404.
20. **LeCun Y. et al.** Gradient-based learning applied to document recognition // *Proceedings of the IEEE*, 1998. Vol. 86. No. 11. Pp. 2278–2324.
21. **Abadi M. et al.** Tensorflow: Large-scale machine learning on heterogeneous distributed systems // *arXiv preprint arXiv.*, 2016, p. 1603.04467.
22. **Glorot X., Bordes A., Bengio Y.** Deep sparse rectifier neural networks // *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. – JMLR Workshop and Conference Proceedings, 2011. Pp. 315–323.

### References

1. **Cipra B.A.** An introduction to the Ising model // *The American Mathematical Monthly*, 1987, vol. 94, no. 10, pp. 937–959.
2. **Fisher D. S., Huse D. A.** Ordered phase of short-range Ising spin-glasses // *Physical review letters*, 1986, vol. 56, no. 15, p. 1601.
3. **Kruis J., Maris G.** Three representations of the Ising model // *Scientific reports*, 2016, vol. 6, no. 1, pp. 1–11.

4. **Padalko M. A. et al.** An accelerated exhaustive enumeration algorithm in the Ising model // *Dal'nevostochnyi Matematicheskii Zhurnal [Far Eastern Mathematical Journal]*, 2019, vol. 19, no. 2, pp. 235–244.
5. **Prudnikov V. V. et al.** Marginal Effect of Structural Defects on the Nonequilibrium Critical Behavior of the Two-Dimensional Ising Model // *JETP*, 2020, vol. 130, no. 2, p. 258–273.
6. **Anderson P. W.** Ordering and antiferromagnetism in ferrites // *Physical Review*, 1956, vol. 102, no. 4, pp. 1008.
7. **Narita N. et al.** Design and numerical study of flux control effect dominant MAMR head: FC writer // *IEEE Transactions on Magnetics*, 2020, vol. 57, no. 3, pp. 1–5.
8. **Vasil'ev E. V. et al.** Numerical simulation of two-dimensional magnetic skyrmion structures // *Computer Research and Modeling*, 2020, vol. 12, no. 5, pp. 1051–1061.
9. **Soldatov K. S. et al.** Finite-size scaling in ferromagnetic spin systems on the pyrochlore lattice // *Dal'nevostochnyi Matematicheskii Zhurnal [Far Eastern Mathematical Journal]*, 2020, vol. 20, no. 2, pp. 255–266.
10. **Landau D., Binder K.** A guide to Monte Carlo simulations in statistical physics. Cambridge university press, 2003.
11. **Shapovalova K. V. et al.** Why does Far East Federal University need supercomputer? // *Sovremennye naukoemkie tekhnologii*, 2017, no. 1, pp. 81–87. (in Russ.)
12. **Shapovalova K. V. et al.** Methods of canonical and multicanonical sampling of the space of vector models // *Dal'nevostochnyi Matematicheskii Zhurnal*, 2017, vol. 17, no. 1, p. 124–130. (in Russ.)
13. **Makarov A. G. et al.** On the Numerical Calculation of Frustrations in the Ising Model // *JETP Letters*, 2019, vol. 110, no. 10, pp. 702–706.
14. **Carrasquilla J., Melko R.G.** Machine learning phases of matter // *Nature Physics*, 2017, vol. 13, no. 5, pp. 431–434.
15. **Kenta S. et al.** Machine-Learning Studies on Spin Models // *Scientific reports*, 2020, vol. 10, no. 1, pp. 1–6.
16. **Onsager L.** A Two-Dimensional Model with an Order-Disorder Transition // *Phys. Rev.*, 1944, vol. 65, pp. 117–149.
17. **Kapitan V. Yu., Shevchenko Yu. A., Andriushchenko P. D., Nefedev K. V.** Supercomputer modeling and numerical solutions of statistical physics problems. Publishing house of FEFU, Vladivostok, 2017. 195 p.
18. **Goodfellow I., Bengio Y., Courville A.** Deep learning. MIT press, 2016.
19. **LeCun Y. et al.** Handwritten digit recognition with a back-propagation network // *Advances in Neural Information Processing Systems (NIPS)*, 1989, pp. 396–404.
20. **LeCun Y. et al.** Gradient-based learning applied to document recognition // *Proceedings of the IEEE*, 1998, vol. 86, no. 11, pp. 2278–2324.
21. **Abadi M. et al.** Tensorflow: Large-scale machine learning on heterogeneous distributed systems // *arXiv preprint arXiv: 1603.04467*, 2016.
22. **Glorot X., Bordes A., Bengio Y.** Deep sparse rectifier neural networks // *Proceedings of the fourteenth international conference on artificial intelligence and statistics. – JMLR Workshop and Conference Proceedings*, 2011, pp. 315–323.

#### Сведения об авторах

**Король Алена Олеговна**, магистрант, инженер-исследователь, Департамент теоретической физики и интеллектуальных технологий, Институт наукоёмких технологий и передовых материалов, Дальневосточный федеральный университет (Владивосток, Россия)

**Нефедев Константин Валентинович**, доктор физико-математических наук, профессор, директор, Департамент теоретической физики и интеллектуальных технологий, Институт

научноёмких технологий и передовых материалов, Дальневосточный федеральный университет (Владивосток, Россия)

**Капитан Виталий Юрьевич**, кандидат физико-математических наук, доцент, Департамент информационной безопасности, Институт математики и компьютерных технологий, Дальневосточный федеральный университет (Владивосток, Россия)

### Information about the Authors

**Alena O. Korol**, master, research-engineer, Department of Theoretical Physics and Intelligent Technologies, Institute of High Technologies and Advanced Materials, Far Eastern Federal University (Vladivostok, Russian Federation)

**Konstantin V. Nefedev**, Doctor of Science (Physics and Mathematics), professor, director, Department of Theoretical Physics and Intelligent Technologies, Institute of High Technologies and Advanced Materials, Far Eastern Federal University (Vladivostok, Russian Federation)

**Vitalii Yu. Kapitan**, Candidate of Science (Physics and Mathematics), Associate professor, Information Security Department, Institute of Mathematics and Computer Technology, Far Eastern Federal University (Vladivostok, Russian Federation)

*Статья поступила в редакцию 10.03.2021;*

*одобрена после рецензирования 08.02.2022; принята к публикации 22.06.2022*

*The article was submitted 10.03.2021;*

*approved after reviewing 08.02.2022; accepted for publication 22.06.2022*