COS424 Final Project: Machine Learning in Physics

Nicholas Haubrich
Princeton University
njh@princeton.edu

Alan Morningstar
Princeton University
aorm@princeton.edu

Abstract

We present applications of machine learning methods to physics problems. By analytically solving for the parameters of a restricted Boltzmann machine (RBM) we find a solution that allows us to represent an Ising model with an RBM exactly. By doing this, spin configurations of the Ising model are generated via block Gibbs sampling of the RBM. We also investigate the prospect of using machine learning techniques to rediscover the phases of the two-dimensional Ising model from spin configurations alone. In another application we used supervised machine learning to distinguish particle decay processes by their final products. By constructing additional discriminating features motivated by physical principles, a neural net was able to achieve a 8.2σ search significance for the particular process h→aa→4b, a large improvement over the previous significance of 4.2σ.

1 Introduction

Machine learning techniques have yet to be fully integrated into all of the quantitative sciences. In the case of physics, problems of pattern recognition, classification, and unsupervised discovery in large datasets arise often and are a natural setting in which to apply modern machine learning methods. In this project, inspired by recent advances in physics aided by machine learning technology [1, 2, 3], we apply generative modelling and supervised learning to problems in statistical and particle physics.

2 Machine Learning in Statistical Physics

A physical system (such as a macroscopic collection of water molecules) is in a certain *phase of matter* (such as liquid or vapour) if its control parameters (such as temperature and pressure) can be varied by a finite amount while leaving the large-scale qualitative character of the system invariant. Certain values of the control parameters called *critical points* separate these distinct phases. One of the broad goals of statistical physics is to characterize the set of possible phases of matter and the critical points between them.

In this project we consider two scenarios common to statistical physics research which can benefit from some machine learning input. This work contains variations of recent ideas from the machine-learning-in-physics literature. References to the relevant papers are included throughout. The first is the task of generating configurations of microscopic degrees of freedom (such as the momenta and positions of water molecules) given a known theory governing the system: the so-called "sampling" task [4]. The idea is to make a generative model that is easy to sample whose samples are distributed according to some physical scenario [5, 6]. The second is the task of identifying phases (large-scale, emergent, qualitative behavior) given configurations of the microscopic degrees of freedom in a system: the so-called "phase discovery" task [3, 7]. This is relevant to the analysis of experimental data and the discovery of new phases of matter.

We will carry out the sampling and phase discovery tasks using a canonical model of phases and phase transitions: the Ising model at finite temperatures. This model is simpler than considering the problem of, for example, water at different pressures and temperatures, but it still contains two phases and a critical point. The Ising model [8] consist of binary degrees of freedom called *spins*, $s_i \in \{+1, -1\}$, that exist on the sites, i, of a two-dimensional square lattice. The model is specified by its *Hamiltonian*,

$$H(\vec{s}) = -\sum_{\langle i_1 i_2 \rangle} s_{i_1} s_{i_2}, \tag{1}$$

which can be thought of as a function that maps configurations of spins, $\vec{s}$, to the total energy of the configuration. The angular brackets $\langle i_1 i_2 \rangle$ indicate the sites i_1 and i_2 must be physical neighbors to be included in the sum. Statistical physics assigns a Boltzmann probability distribution to configurations $\vec{s}$ based on their energy and the temperature, T, of the system:

$$p(\vec{s}) \propto \mathrm{e}^{-\frac{H(\vec{s})}{T}}. \tag{2}$$

In this Ising system, the temperature is the control parameter that tunes the system from the *paramagnetic* phase ($T > T_\mathrm{c}$) to the *ferromagnetic* phase ($T < T_\mathrm{c}$), where T_c marks the critical point. It is known that for such an Ising model on an infinite lattice, the critical temperature is $T \approx 2.3$ (in the dimensionless energies we are working with) [9].

To summarize the plan, we will define a generative model that can be efficiently sampled and whose samples follow the distribution of Equation 2. By doing this we are using machine learning technology to design a statistical physics sampling algorithm. Once this has been done and we can generate configurations of spins at any temperature, we will generate a dataset of such configurations and use unsupervised learning techniques to rediscover the known phases of the Ising model.

2.1 The Sampling Task

We choose to use a generative graphical model that was originally based on the Ising model: the restricted Boltzmann machine (RBM) [10]. The RBM consists of visible nodes which we identify with physical spins, s_i, and hidden nodes which we call hidden spins, h_j. Traditionally these nodes take values $\{1, 0\}$ but we will consider $s_i, h_j \in \{+1, -1\}$ to align with our physical Ising model variables. The RBM joint distribution takes the form

$$q_\theta(\vec{s}, \vec{h}) \propto \mathrm{e}^{\vec{h} \cdot W \vec{s} + \vec{a} \cdot \vec{s} + \vec{b} \cdot \vec{h}}, \tag{3}$$

where $\theta \equiv \{W, \vec{a}, \vec{b}\}$ are the parameters of the model.

At this point it is important to mention that $q_\theta(\vec{s}|\vec{h})$ and $q_\theta(\vec{h}|\vec{s})$ both factorize over visible and hidden spins respectively, so efficient block Gibbs sampling is possible. This is crucial and makes the RBM efficient to sample with Markov chain Monte Carlo (MCMC).

It is possible to explicitly sum over all hidden-spin configurations, $\vec{h}$, to get the marginal distribution $q_\theta(\vec{s})$:

$$\begin{aligned} q_\theta(\vec{s}) &= \sum_{\vec{h}} q_\theta(\vec{s}, \vec{h}) && (4) \\ &\propto \sum_{\vec{h}} \mathrm{e}^{\sum_i a_i s_i + \sum_j b_j h_j + \sum_{ij} h_j W_{ji} s_i} && (5) \\ &= \prod_i \mathrm{e}^{a_i s_i} \prod_\alpha 2\cosh(\sum_i W_{ji} s_i + b_j). && (6) \end{aligned}$$

Putting Equation 2 into a similar form yields $p(\vec{s}) \propto \prod_{\langle i_1 i_2 \rangle} \mathrm{e}^{\frac{s_{i_1} s_{i_2}}{T}}$. It follows that if we set $\vec{a} = \vec{0}$ and allow one hidden spin per bond between visible spins then solving

$$2\cosh(W_{ji_1} s_{i_1} + W_{ji_2} s_{i_2} + b_j) \propto \mathrm{e}^{\frac{s_{i_1} s_{i_2}}{T}} \tag{7}$$

for the possible values of s_{i_1} and s_{i_2} will result in $q_\theta(\vec{s}) \propto p(\vec{s})$. Indeed this can be solved by

$$b_j = 0 \tag{8}$$

$$W_{ji_1} = -W_{ji_2} = \frac{1}{2}\mathrm{arccosh}(\mathrm{e}^{2/T}). \tag{9}$$

To summarize this derivation, if we want to generate configurations from the Ising model at temperature T, all we need to do is set the parameters of a binary ±1 RBM with one hidden node per Ising bond to $\vec{a} = \vec{b} = \vec{0}$ and W according to Equation 9 and then run block Gibbs sampling from a random initial state until the samples approach an equilibrium distribution. This analytic solution for the RBM parameters was inspired by a similar problem solved in [5].

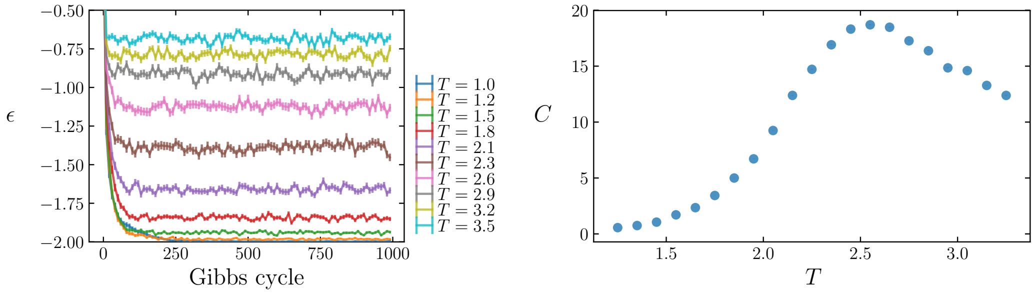

Figure 1: (left) A demonstration of "burn in" for block-Gibbs sampling of the RBM. We see that for any temperature T the energy density ϵ equilibrates before 500 cycles. (right) The heat capacity C as a function of temperature T. The peak indicates the finite-size remnant of the phase transition in the Ising model, which demonstrates that our RBM does reproduce correct Ising-model configurations.

Due to the factorization of the RBM conditional probability distribution, and the sparsity of the weight matrix defined above (only two non-zero weights per hidden node) this algorithm effectively performs parallel local updates similar to the simple Metropolis-Hastings algorithm. We therefore expect this algorithm not to outperform the best physics algorithms for sampling the Ising model (the autocorrelation time of the MCMC is still long compared to the best algorithms). Nonetheless it is still an interesting perspective gained from machine learning on an old problem in physics. For a better solution which involves a custom RBM with interaction terms $W_{i_1 i_2 j} s_{i_1} s_{i_2} h_j$ see [5].

We wrote code to construct an `ising_rbm` as described above and to perform block-Gibbs sampling to generate Ising model configurations at specified temperatures. To verify that our RBM does indeed produce samples from Equation 2 we construct an `ising_rbm` for a 6 × 6 system. After checking how long it takes to "burn in" the Markov chain (see the left of Figure 1) we generated 10^4 uncorrelated configurations and computed a physically relevant quantity called the *heat capacity* ($C = \mathrm{var}(E)/T^2$ where E is the energy of a configuration) which should show a peak at the critical temperature. On the right of Figure 1 we show this peak, which is not exactly at $T_c = 2.3$ because our system is finite. This verifies that our `ising_RBM` does indeed generate the desired thermal configurations.

2.2 The Phase Discovery Task

After verifying that our RBM solution to the sampling task works, we generated 10^3 configurations from each of one hundred temperatures in the range $T \in [1.5, 3.5]$ for an $N = 36$ (6×6) Ising model. This data is a list of ±1 variables on each of the N sites, with an accompanying temperature value T. To re-discover the two phases of the Ising model via machine learning we used what is called the "learning by confusion" scheme developed in [7]. This scheme is based on first choosing a partition temperature T_p, then using the values of T to generate assumed class labels $c = 1$ $(T < T_p)$ and $c = 0$ $(T > T_p)$ (we must assume a number of phases). We then train a classifier to perform the classification task $\vec{s} \mapsto c$ and record the test-set accuracy of this task. By varying the partition temperature we obtain the classification accuracy as a function of T_p. The idea behind the confusion scheme is that when T_p is equal to the true critical temperature then this classification task should be at its easiest (highest accuracy) because there actually are two qualitatively distinct regimes (phases) separated by that temperature. The critical point therefore is signalled by a local maximum in the classification accuracy as a function of T_p.

We applied the confusion scheme to configurations of both the 2d ($N = 6 \times 6$) and 1d ($N = 16$) Ising model, which we generated using our RBM method described in the previous section. This was done because unlike the 2d model, the 1d Ising model does not have a phase transition. As the classifier we used a single-hidden-layer perceptron with the same number of hidden neurons as inputs. In Figure 2 we show the resulting test-set classification accuracy as a function of the partition temperature. In the 1d case (right) the V shape indicates no existing critical point. The V shape occurs because naturally the task of binary classification is easy when there are many more samples of one class than another. Therefore these edge effects are not "signal", but a mere artifact

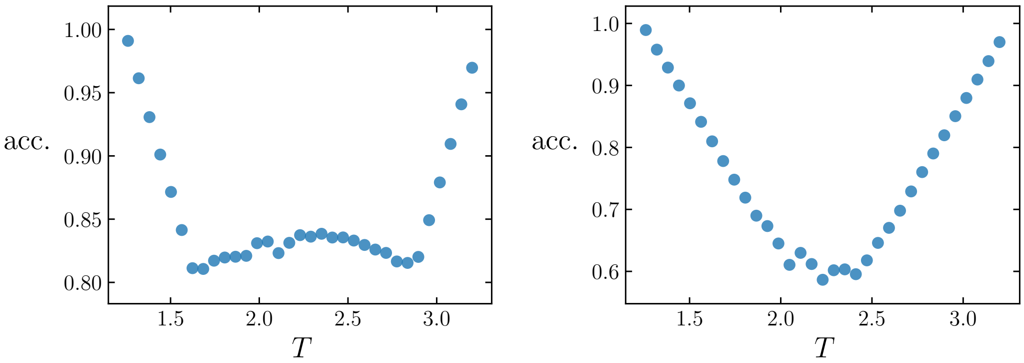

Figure 2: The learning by confusion scheme applied to the 2d (left) and 1d (right) Ising models.

of the method. The absence of a local maximum in the 1d case indicates there is no phase transition. In the 2d case (left) there is a broad local maximum centered around what we know to be the critical temperature. We therefore have re-discovered the existence of the 2d Ising critical point via machine learning. The local maximum is broad because this proof-of-principle method was applied to a small 6×6 system where the critical regime is broad due to finite size effects. In the limit of large system sizes (like in an experimental application) this peak would sharpen.

3 Machine Learning in Particle Physics

3.1 Overview of Particle Searches

The Compact Muon Solenoid Experiment (CMS)[11] at the Large Hadron Collider, is an all-purpose particle detector with the goal of discovering new particles beyond the Standard Model of particle physics. It detects the final-state particles of proton-proton collisions at the Large Hadron Collider with center-of-mass energies of 14 TeV that occur every 25 ns during operation. By analyzing the resulting particles, statistical limits on the presence (or lack thereof) of specific proposed particles can be obtained. The process of interest is two colliding protons forming a standard-model Higgs boson [12] that decays into two hypothetical "a" particles, each of which decays into two standard-model bottom quarks. This process (referred to as h→aa→4b) is phenomenologically well-motivated [13]. Put simply, this is a search for a theoretical "a" particle with a mass of 60 GeV via the Higgs boson and bottom quarks. The challenge in any search is to discriminate rare signal events (i.e. occurrences of the process of interest) from overwhelmingly common background events that produce a similar signature.

The CMS detector is comprised of a 3.8 T magnet and a number of complex subdetectors that surround the proton-proton collision point. The magnetic field causes the path of produced charged particles to curve, and the particles interact with electronics in the layers of the detector. The physical properties (namely, momentum, position, charge, and mass) of particles can then be reconstructed based on the traces left in the detector and knowledge of the strength of the magnetic field. Bottom quarks, due to hadronization, are seen as a "shower" of particles, instead of just a single particle. The anti-KT jet[14] clustering algorithm is used to convert the "showers" into reconstructed bottom quarks called "jets" with well-defined position, momentum, and mass. These jets also contain a "b-tag", which is an indicator of the likelihood that a jet came from a bottom quark compared to another quark. All physical objects resulting from a particular proton-proton collision are grouped together and called an "event".

A search for a process goes as follows. First, loose criteria on the final-state particles that narrow down the space of the search are defined. For h→aa→4b, this was requiring in every event at least four jets, at least two of which pass a loose b-tag requirement. Then, simulated Monte Carlo data for all of the background processes and the signal process are produced. A strategy to discriminate signal events from the background is created by examining the Monte Carlo data for regions of high signal-vs-background, and then imposing selections that discard events in low-signal-vs-background regions (for example, requiring all particles pass a threshold momentum and have a certain charge). This is a task well-suited for machine-learning, and many other searched have made use of it[1].

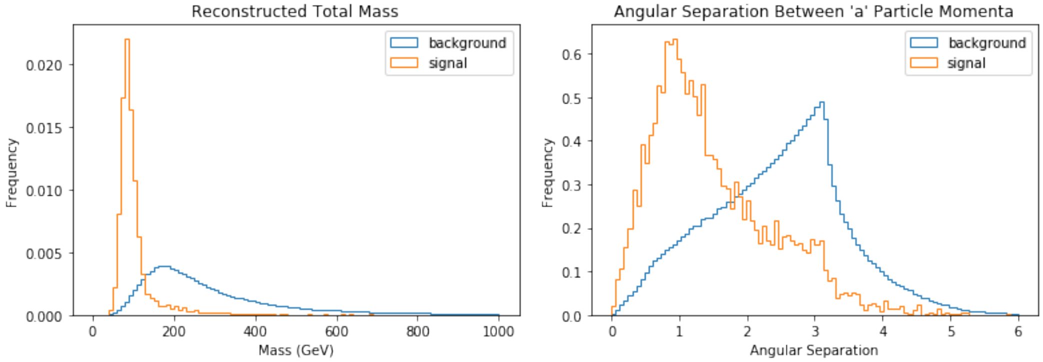

Figure 3: Discriminating features based on physical principles. The reconstructed total mass of the event is peaked around 125 GeV for signal because the four jets all decayed from a 125 GeV Higgs boson, while background events have a much larger spread due to the variety of background processes. The reconstructed angular separation of the reconstructed intermediate "a" particles also is discriminatory. For signal events that actually contained "a" particles, they were produced from a single Higgs boson with nonzero momentum, so they are approximately collimated. For background events, the reconstructed "a" properties are less correlated and the angular separation is greater. Introducing these features with others greatly improved the accuracy of the classifiers.

Here, we will evaluate the use of machine learning techniques for the h→aa→4b search and compare the sensitivity to the initial sensitivity estimate performed the human way.

3.2 Data and Methods

The training data consisted of 770,000 Monte Carlo events with 20 features: the transverse momentum, two coordinates that describe position, the reconstructed mass, and the b-tag of each of the four jets. An equal-sized validation dataset was used to evaluate performance. The equal-sized partitions were chosen in order to ensure the statistical evaluation of the classifiers was accurate, even if this lead to a sub-optimal performance of the classifiers. The training dataset contained 5000 h→aa→4b signal events and 765000 background events that arise from common processes that can produce four jets, with at least two passing the loose b-tag requirement. The largest contributing background process is $t\bar{t}$, where the protons produce a top and anti-top quark pair, each of which decay into a bottom quark plus a jet or lepton, thus sometimes yielding four jets, with two having high b-tags.

Before any training, the features were scaled to have a mean of zero and a standard deviation of approximately 1 via Scikit Learn's StandardScaler [15] fitted with the training data. Due to the large number of events and complex relations between the features, neural networks were chosen as a suitable classifier model for our purposes. Four networks were implemented with Scikit Learn, trained, and evaluated. A baseline classifier consisted of 20 nodes and was trained on the 20 features. Three other classifiers, with 20 nodes, 38 and 2 nodes, and 38 and 38 nodes, were also trained, but using additional constructed features as described below. The neural networks all used the "L-bfgs" solver and were trained over a single epoch.

In an attempt to improve accuracy, a number of additional features were created from the original jet features. On the whole, these were physical properties that can be discriminatory for different processes. Two of these variables are shown in Fig 3 for signal and background events of the training data. For instance, the momenta, positions, and mass of the four jets can be combined to find the mass of the particle they decayed from. Physically, this is done by treating the decays as relativistic and elastic, and summing the momentum vectors of the four particles. For the signal process, this value is peaked around 125 GeV (the mass of the Higgs), while the background processes have a broad distribution of masses due to the variety of different interactions they contain. Similarly, properties of the intermediate "a" particles can be reconstructed. There is an ambiguity regarding which pair of jets to use for each "a". The jets were paired so that the two reconstructed "a" masses were as close as possible. This is admittedly a heuristic, and could be improved upon. Nevertheless, the angular separation of the momenta of the reconstructed "a" particles is visually-evident a discriminatory variable. The momentum, angular position, and mass of the singular original particle,

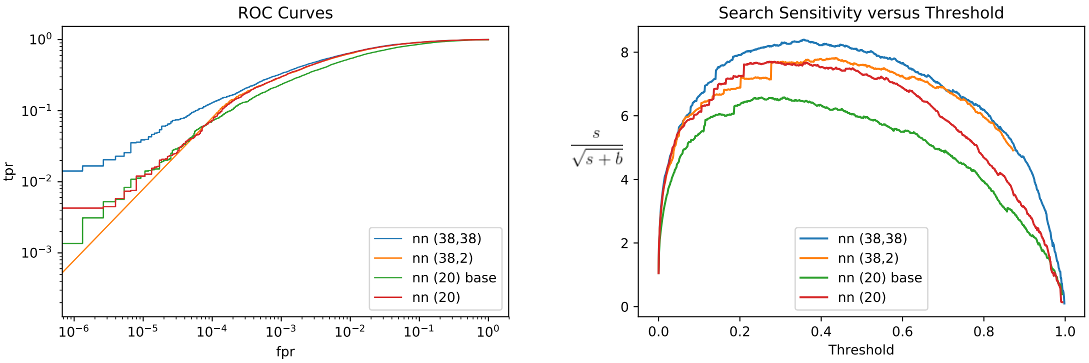

Figure 4: ROC curves and search significance for the classifiers. The ROC curve plot (left) shows clearly that the (38,38) node classifier on the expanded set of features has the best performance. The significances of the classifiers at various thresholds (right) shows that peak significance occurs for the (38,38) node classifier at a significance of 8.2σ at a threshold of .38.

and of each of the “a” particles (ranked by momentum, so the “a1” features correspond to the “a” with the higher momentum), as well as the angular separation between the two “a” particles and also between the pairs of jets used to reconstruct the “a” particles were added to the features. The scalar sum of the momenta of the four jets was also added, as well as a reconstruction of the top quark mass, which should be peaked for $t\bar{t}$ background events but not signal.

3.3 Results and Evaluation

Fig 4 (left) shows the receiver operating characteristic curve for the four classifiers. The baseline classifier trained on just the original twenty features is outperformed by the other classifiers, and the neural net with two 38-node layers clearly is the most effective. However, while the ROC curve demonstrates the relative performance of the classifiers, it does not give the overall statistical significance, σ of the search. In the context of particle physics, the significance gives the expected confidence level of observing the signal process[16]. A typical way to estimate the significance is, given s signal events and b background events in the signal region:

$$\sigma = \frac{s}{\sqrt{s+b}} \tag{10}$$

In the final evaluation of a search, there is ultimately just a number of real events that are classified as signal, and a predicted number of background events. The sensitivity gives the standard deviation that corresponds to the probability the excess number of events is just due to fluctuations in the background. For example, a 2σ result corresponds to a 97.72% confidence that the excess was not due to statistical fluctuations. In the particular case of h→aa→4b, the cross-section of the process (which gives the expected amount of true signal events) is a free parameter. The data used assumed a 1 femtobarn cross-section for the process, but different cross-sections can be assumed by simply scaling the number of signal events in proportion with the cross section.

An important subtlety with the data is that the background processes are not simulated perfectly one-to-one. Instead, a given number of events for a process are generated, then these events are assigned a weight that gives the real number of events corresponding to the simulated event. For instance, a single $t\bar{t}$ event may be generated, but have a weight of 5, meaning it counts for essentially 5 actual events. Notably, the neural net does not see the weight of the training events, which may prevent maximum effectiveness when training the classifier. Ideally, the weight of each event would factor into the loss function of the classifier so that wrongly classifying an event with a high weight is treated as worse than misclassifying an event with low weight. However, this was technically infeasible, and the classifiers still performed well without this.

The sensitivities of the neural nets were computed using equation (10) with the weighted sum of signal events and background events classified as positive with each classifier. The sensitivity depends on the threshold of the classifiers, as the amount of true positives and false positives depends on this quantity. Fig 4 (right) shows the significances of the different classifiers over the full range of the thresholds. Unlike the ROC curve, where the classifiers are judged by the curves total area, all that matters in this case is the peak significance. As expected from the ROC curves, the (38,38) node classifier performs best, with the (20) and (38,2) node classifiers somewhat less sensitive, and the base classifier that lacked the additional features even lower. At its optimal threshold, the (38,38) node classifier has a significance of 8.2σ, meaning that if the h→aa→4b process existed at the assumed 1 femtobarn cross-section, the detector would observe it at above 99% confidence. The sensitivity of the search when performed without machine learning by manually placing selections on discriminating variables such as the reconstructed total mass was 4.7σ. Thus, using the techniques of machine learning greatly improved upon the statistical power of the search.

4 Conclusion

In this project we explored applications of generative modelling and supervised learning to physics problems. First, we demonstrated that the probability distribution of a RBM could be set exactly equal to the thermal distribution of an Ising model with a certain choice of parameters. This allowed us to generate sample spin configurations of the Ising model using efficient block Gibbs sampling of the RBM. This was verified by sampling configurations from the Ising RBM and computing the physical heat capacity as a function of temperature. Following this we showed that the "learning by confusion" scheme of van Nieuwenburg et al. [7] could be used to leverage supervised classification to rediscover the phases of the Ising model. In a final application we developed a discriminatory neural net that had a 8.2σ significance for detecting the h→aa→4b process with the Compact Muon Solenoid experiment. Essential to this result was creating additional features such as the reconstructed total mass of the event that were highly discriminating between the signal and background due to the physical properties of those processes. The neural net significantly out-performed the 4.7σ manually-constructed discriminator previously used. These three successful applications of machine learning in physics demonstrate the rich potential of the union of these two fields.

References

[1] Peter J Sadowski, Daniel Whiteson, and Pierre Baldi. Searching for higgs boson decay modes with deep learning. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2393–2401. Curran Associates, Inc., 2014.

[2] Lei Wang. Discovering phase transitions with unsupervised learning. *Phys. Rev. B*, 94:195105, Nov 2016.

[3] Juan Carrasquilla and Roger G. Melko. Machine learning phases of matter. *Nature Physics*, 13:431 EP –, 02 2017.

[4] M Newman and G Barkema. *Monte carlo methods in statistical physics chapter 1-4*. Oxford University Press: New York, USA, 1999.

[5] Lei Wang. Exploring cluster monte carlo updates with boltzmann machines. *Phys. Rev. E*, 96:051301, Nov 2017.

[6] Li Huang and Lei Wang. Accelerated monte carlo simulations with restricted boltzmann machines. *Phys. Rev. B*, 95:035105, Jan 2017.

[7] Evert P. L. van Nieuwenburg, Ye-Hua Liu, and Sebastian D. Huber. Learning phase transitions by confusion. *Nature Physics*, 13:435 EP –, 02 2017.

[8] STEPHEN G. BRUSH. History of the lenz-ising model. *Rev. Mod. Phys.*, 39:883–893, Oct 1967.

[9] Lars Onsager. Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Phys. Rev.*, 65:117–149, Feb 1944.

[10] Geoffrey E. Hinton. *A Practical Guide to Training Restricted Boltzmann Machines*, pages 599–619. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[11] S. Chatrchyan et al. The CMS Experiment at the CERN LHC. *JINST*, 3:S08004, 2008.

[12] CMS Collaboration. Observation of a new boson at a mass of 125 gev with the cms experiment at the lhc. *Physics Letters B*, 716(1):30 – 61, 2012.

[13] David Curtin et al. Exotic decays of the 125 GeV Higgs boson. *Phys. Rev.*, D90(7):075004, 2014.

[14] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. The anti-k_t jet clustering algorithm. *JHEP*, 04:063, 2008.

[15] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[16] Alexander L. Read. Modified frequentist analysis of search results (The CL(s) method). In *Workshop on confidence limits, CERN, Geneva, Switzerland, 17-18 Jan 2000: Proceedings*, pages 81–101, 2000.