Learning multiple order parameters with interpretable machines

Ke Liu,[1, *] Jonas Greitemann,[1] and Lode Pollet[1]

[1]*Arnold Sommerfeld Center for Theoretical Physics,*
University of Munich, Theresienstrasse 37, 80333 Munich, Germany

Machine learning techniques are evolving into an subsidiary tool for studying phase transitions in many-body systems. However, most studies are tied to situations involving only one phase transition and one order parameter. Systems that accommodate multiple phases of coexisting and competing orders, which are common in condensed matter physics, remain largely unexplored from a machine learning perspective. In this paper, we investigate multiclassification of phases using Support Vector Machines (SVMs) and apply a recently introduced kernel method for detecting hidden spin and orbital orders to learn multiple phases and their analytical order parameters. Our focus is on multipolar orders and their tensorial order parameters whose identification is difficult with traditional methods. The importance of interpretability is emphasized for physical applications of multiclassification. Furthermore, we discuss an intrinsic parameter of SVM, the bias, which allows for a special interpretation in the classification of phases, and its utility in diagnosing the existence of phase transitions. We show that it can be exploited as an efficient way to explore the topology of unknown phase diagrams where the supervision is entirely delegated to the machine.

I. INTRODUCTION

Order parameters play a central role in condensed matter physics[1,2]. They represent the important degrees of freedom in a system, without tracking all the microscopic details. They define states of matter and identify their elementary excitations and topological defects. Recognizing the order parameter is often a basic step towards understanding the physics and constructing the effective theory of a many-body system.

For some systems, the order parameter is obvious. E.g., in ferromagnets, the order parameter, namely the magnetization, is simply given by a sum of local magnetic moments. In general, however, identifying the order parameter can be a difficult task.

This is particularly the case in frustrated spin and orbital systems[3]. For instance, the classical Heisenberg antiferromagnet on a Kagomé lattice was originally thought to be a spin liquid. However, the existence of a order-by-disorder phenomenon has been realized by Chalker and collaborators in the early 1990s, and as a consequence a quadrupolar spin-nematic order has been found[4]. More than a decade later, Zhitomirsky pointed out that in addition to the quadrupolar order, it also develops an octupolar order[5,6]. The subtlety of identifying these orders may be partially understood in the following way: They appear as collective motion and higher moments of ordinary spins. To define their order parameter, one needs to recognize the relevant degrees of freedom and the pattern in which those are combined.

The quadrupolar and octupolar orders in the above Kagomé antiferromagnet are instances of multipolar orders. Such orders also commonly occur in other frustrated systems, such as pyrochlores[7–10], garnets[11], triangular antiferromagnets[12], and frustrated orbital systems[13–15]. However, no general guiding principle is known to predict their existence and to identify them.

Recent developments in physical applications of machine-learning techniques may provide a means to detect order parameters of many-body systems. The most popular algorithms for this include a variety of neural networks[16–21] and kernel methods[22–24]. Algorithms of the former type, such as Restricted Boltzmann Machines (RBMs), Convolutional Neural Networks (CNNs), and Generative Adversarial Networks (GANs), feature a high expressiveness. Given a sufficient number of hidden units, they can in principle represent any nonlinear function. However, these methods face problems in their interpretability: Except for the simplest models, it is in general very difficult to understand the machine results. On the other hand, the kernel methods, such as Support Vector Machines (SVMs) and kernel Principal Component Analysis (kernel PCA), are strongly interpretable, which permits theoretical insight from machine results. Nevertheless, the scope of their applicability crucially relies on the choice of the kernel function, and standard SVM and PCA kernels may only apply to simple physical problems.

The trade-off between expressiveness and interpretability is a prevalent theme in machine learning. For detecting hidden multipolar orders in frustrated systems, it is possible to reconcile them. In Ref. 25, we have introduced a kernel for SVMs that can be used to probe general classical $O(3)$-breaking orientational order. We demonstrated its capabilities by learning the analytical order parameter of multipolar orders up to rank 6 and discussed its application to the identification of novel spin nematics and for ruling out spurious spin liquid candidates.

In Ref. 25, as in most studies related to the machine learning of phases, the focus is on situations where only two phases (defining one order parameter) are involved, and a binary classification is achieved. In the current paper, we apply our kernel to multiclassification to handle multiple phases. Moreover, we also introduce a new criterion which can be used to explore the initially unknown

phase diagram.

Although the classification of multiple labels is a standard extension to SVMs, the physical application is far from trivial and demands strong interpretability. This can be explained in the following way: One obtains numerical classifiers that each distinguish a pair of labels (representing two phases). If a system accommodates multiple phases, we need to interpret those classifiers to discern and define each phase in physical language. Moreover, if a phase possesses a number of orders simultaneously, the machine will combine them into a single classifier. In order to correctly understand the underlying physics, we need to be able to isolate them individually.

The remainder of the paper is organized as follows. In Sec. II, we review some basic concepts of SVM, including a brief introduction of binary and multiclassification. In Sec. III, we review the construction of our kernel, and discuss the bias parameter in SVM and its expected value as applied to phase classification. In Sec. IV, we define a model that generates multipolar-ordered spin configurations serving as training and testing data. In Sec. V, we take a simple quadrupolar order as example, and explain in detail the procedure employed to extract the analytical order parameter and the use of the aforementioned bias parameter. Moreover, in this section we also compare different choices of training data. This discussion may also be applicable to other machine-learning methods. In Sec. VI, we use our method to detect multiple (multipolar) phases and learn their order parameters. A generalized quadrupolar and the octupolar order will be considered as examples. In Sec. VII, we examine the ability of SVM to explore the phase diagram in the absence of prior knowledge on its topology. Our motivation there is to obtain a tentative phase diagram, and use it to guide the subsequent learning of the order parameters. The precise phase diagram can be obtained after the order parameters have been extracted. However, as will be discussed, the scheme is actually capable of achieving a decent precision before the order parameters are even considered. We conclude with an outlook in Sec. VIII.

II. SUPPORT VECTOR MACHINES

In this section we will provide a brief introduction of SVM. A comprehensive discussion can be found in Ref. 26. The key message here is that whether SVM is able to distinguish two classes depends on the choice of the kernel function. Readers who are already familiar with SVM or are not interested in the technical details may skip to Sec. III where we construct the kernel suited for the physical problem.

A. Definition of the optimization problem

Support Vector Machines provide a means to construct a classifier from N_s training data samples, $(\mathbf{x}^{(k)}, y^{(k)})$,

where $\mathbf{x}^{(k)} \in \mathbb{R}^d$ and $y^{(k)}$ labels the class of the training sample. We will consider the case of binary classification, $y^{(k)} \in \{-1, +1\}$, first and explain the extension to more than two classes afterwards[27–29].

The aim of the SVM is to find a hyperplane, dubbed decision boundary,

$$\mathbf{w} \cdot \mathbf{x} - \rho = 0, \tag{1}$$

defined by parameters $\mathbf{w} \in \mathbb{R}^d$ and $\rho \in \mathbb{R}$, that separates the data into the two classes. If the data are separable, typically infinitely many solutions exist. In order to select the solution that separates the data most clearly, a finite margin is imposed which must not contain any of the training data samples, and the width of the margin is sought to be maximal. When defining $\mathbf{w} \cdot \mathbf{x} - \rho = \pm 1$ as the boundaries of the margin, such that the width of the margin is given by $2/\|w\|$, the optimization problem may then be written as

$$\begin{cases} \text{minimize} & \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{w.r.t.} \quad \mathbf{w}, \rho \\ \text{subject to} & y^{(k)} \left(\mathbf{w} \cdot \mathbf{x}^{(k)} - \rho \right) \geq 1 \quad \forall\, k \end{cases} \tag{2}$$

Most of the time, the data may not be perfectly separable and the constraint Eq. (2) must be relaxed, allowing for incursions into the margin at a cost proportional to $C > 0$,

$$\begin{cases} \text{min.} & \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_k \xi_k \quad \text{w.r.t.} \quad \mathbf{w}, \rho, \{\xi_k\} \\ \text{subj.} & y^{(k)} \left(\mathbf{w} \cdot \mathbf{x}^{(k)} - \rho \right) \geq 1 - \xi_k, \quad \xi_k \geq 0 \quad \forall\, k \end{cases} \tag{3}$$

where ξ_k are slack variables. $\xi_k > 1$ corresponds to a misclassified sample. Since the training data enter the optimization problem only as inner products in the constraint, the optimal $\mathbf{w}$ will lie in the span of the training data,

$$\mathbf{w} = \sum_k \lambda_k y^{(k)} \mathbf{x}^{(k)}. \tag{4}$$

Indeed, only those samples which violate (or touch) the margin necessitate a nonzero slack variable and contribute to $\mathbf{w}$ with a nonzero λ_k. These are the support vectors.

Larger values of C 'harden' the margin, making it more narrow and reducing the number of support vectors. This comes at the risk of overfitting noise. In the limit $C \to \infty$, the solution to Eq. (2) is reproduced. Smaller values of C regularize the problem more strongly, allowing for a wider margin and consequently a higher number of support vectors, but potentially discard faithful information in the training data set.

The optimal choice of the regularization parameter C is ultimately problem-specific. In principle, values of C can span many orders of magnitude. There exists however an alternative reparametrization of Eq. (3) in terms of a regularization parameter $\nu \in [0, 1)$ which has been shown to impose an upper bound on the fraction of training samples that violate the margin and a lower bound

on the number of training samples that serve as support vectors[30]. ν-SVM thus admits a more universal interpretation and we found it to simplify the selection of an appropriate regularization. The optimization problem may not have a feasible solution for all values of ν if the training data are unbalanced. In fact, the maximum feasible value is given by $\nu_{\max} = 2\min(N_s^+, N_s^-)/N_s$ where $N_s^{\pm}$ are the number of training samples in either class[28].

B. Decision function

The quadratic programming problem Eq. (3) can be numerically solved using standard methods such as Sequential Minimal Optimization[31,32]. The complexity of this algorithm is somewhat dependent on the nature of the training data. Ref. 31 gives empirical data of better than $\mathcal{O}(N_s^{2.2})$ which is roughly in line with our observations. Note that in the present work, the computational effort was dominated by the generation of independent samples, rather than the SVM optimization, in all cases but Sec. VII.

The decision function

$$d(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} - \rho \tag{5}$$

determines the orientated distance of a test sample $\mathbf{x}$ from the hyperplane and its sign can be used to predict the class label. Plugging Eq. (4) into the above allows for the practical calculation of the decision function as a sum over inner products with the support vectors:

$$d(\mathbf{x}) = \sum_k \lambda_k y^{(k)} \mathbf{x}^{(k)} \cdot \mathbf{x} - \rho \tag{6}$$

The solution to the optimization problem is thus entirely given by the optimal variables $\{\lambda_k\}$ and the bias ρ, whereas the optimal values of the slack variables are given by the so-called hinge loss,

$$\xi_k = \max\{0, -y^{(k)} d(\mathbf{x}^{(k)})\}. \tag{7}$$

C. The kernel trick

Often, the data in the raw d-dimensional feature space is not expected to be separable by a hyperplane. The solution is to invoke a mapping $\mathbf{x} \mapsto \boldsymbol{\varphi}(\mathbf{x})$ to a higher-dimensional auxiliary space where the mapped training data become linearly separable. Further, since the decision function and the optimization problem Eq. (3) involves only inner products in the auxiliary space, the details of the mapping $\boldsymbol{\varphi}$ (and even the dimensionality of the auxiliary space) need not be known as long as one can compute a kernel function K of the original data in the raw feature space, such that

$$K(\mathbf{x}, \mathbf{y}) = \boldsymbol{\varphi}(\mathbf{x}) \cdot \boldsymbol{\varphi}(\mathbf{y}). \tag{8}$$

The decision function is consequently obtained by replacing the inner product in Eq. (6) with an evaluation of the kernel function,

$$d(\mathbf{x}) = \sum_k \lambda_k y^{(k)} K(\mathbf{x}^{(k)}, \mathbf{x}) - \rho. \tag{9}$$

This is also done in the definition of the optimization problem, Eq. 3, where $\mathbf{w}$ is first replaced by its expansion in terms of support vectors, Eq. 4. E.g., in the optimization objective, one replaces

$$\|\mathbf{w}\|^2 = \mathbf{w} \cdot \mathbf{w} = \sum_{k,k'} \lambda_k \lambda_{k'} y^{(k)} y^{(k')} \mathbf{x}^{(k)} \cdot \mathbf{x}^{(k')}$$
$$\rightarrow \sum_{k,k'} \lambda_k \lambda_{k'} y^{(k)} y^{(k')} K(\mathbf{x}^{(k)}, \mathbf{x}^{(k')}). \tag{10}$$

D. Multiclassification

The extension of SVM to the case where $M > 2$ distinct labels are assigned to the training samples is most effectively accomplished by considering all $M(M-1)/2$ pairs of labels one-by-one[33], considering only the training samples belonging to either label and solving the binary classification problems individually. This produces $M(M-1)/2$ distinct decision functions. Any training sample that contributes to any decision function by a nonzero λ_k is considered a support vector of the multiclassification problem. Note that depending on the nature of the problem, the share of support vectors contributing to multiple decision functions may be significant, allowing for a space-efficient representation.

When it comes to predicting the label for a test sample, each decision function establishes a precedence of one label over the other. Ideally, these relations are collectively compatible with transitivity and one can unambiguously assign a label. When this is not the case, the approach to reconcile the relations followed by most SVM packages is to 'poll' by giving one vote to each decision function and picking the label that accumulates the majority of votes[29]. This approach is not well suited to situations where the labeling of the training data does not necessarily correspond to the 'physical' reality and multiple labels in fact represent the same class, resulting in the vote being split among them. It also fails to recognize situations where the decision function is incapable of distinguishing between two labels but rather overfits noise.

For the purpose of this work, we therefore do not follow a blanket polling scheme but consider the decision functions individually and may discard them based on physical insight.

III. KERNEL FOR GENERAL MULTIPOLAR ORDERS

The problem of distinguishing an ordered phase from a disordered one may be viewed as a binary classification

problem. The training data consist of microscopic configurations which are labeled as being from the ordered or disordered phase.

The decision function, which quantifies the distance of a sample from the decision boundary, i.e. the phase transition, serves a similar role to the magnitude of an order parameter. Given a suitable choice of the kernel, the decision function will hence reproduce the true order parameter. This relation was first pointed out by Ponte and Melko in Ref. 24. There, the authors studied the Ising model and several of its variants using a standard quadratic kernel, which is highly interpretable but only applies to linear orders such as the Ising or XY magnetization.

In Ref. 25, we have introduced an interpretable kernel and shown it to be capable of capturing general $O(N)$-breaking orientational orders, with $N \leq 3$. Below we will first review the construction of this kernel and then discuss its further potential in the detection of phase transitions.

A. Definition of the kernel

Without assuming the specific form of a potential multipolar order, there are two basic properties we can exploit to construct our kernel: (i) local order can be defined by a finite number of local fields; (ii) a multipolar order can generally be formulated in terms of tensors (or polynomials) with a finite rank (degree) that are invariant under certain point group transformations[34,35].

Hence we can partition the system into clusters, each containing a finite number of spins,

$$\mathbf{x} = \{\mathbf{S}_i\} = \{\mathbf{S}_I^\alpha\} = \{S_{I,a}^\alpha\} \tag{11}$$

where $\mathbf{x}$ is the configuration vector and $\mathbf{S}_i$ are $O(3)$ spins at lattice site i. The index I enumerates spin clusters and $\alpha = \mathrm{l}, \mathrm{m}, \ldots$ identifies spins within a cluster, such that $i = (I, \alpha)$. $a = x, y, z$ runs over the components of each spin. We then map the spin components within each cluster to all monomials of degree n, and consequently perform a lattice average over all clusters,

$$\mathbf{x} \mapsto \boldsymbol{\phi}(\mathbf{x}) = \{\phi_\mu\} = \{\langle S_{a_1}^{\alpha_1} \ldots S_{a_n}^{\alpha_n} \rangle_{cl}\}, \tag{12}$$

where $\langle \ldots \rangle_{cl}$ denotes the lattice average over spin clusters I. We also introduce a multi-index to collectively refer to the individual monomials, $\mu = (\alpha_1, \ldots, \alpha_n; a_1, \ldots, a_n)$.

The kernel is then defined as

$$K(\mathbf{x}, \mathbf{x}') = [\boldsymbol{\phi}(\mathbf{x}) \cdot \boldsymbol{\phi}(\mathbf{x}')]^2. \tag{13}$$

Formally, it is a quadratic kernel with respect to the feature vector $\boldsymbol{\phi}(\mathbf{x})$ after the monomial mapping, Eq. (12). With this kernel, the decision function can be expressed

as

$$\begin{aligned} d(\mathbf{x}) &= \sum_k \lambda_k y_k \left[\boldsymbol{\phi}(\mathbf{x}^{(k)}) \cdot \boldsymbol{\phi}(\mathbf{x}) \right]^2 - \rho \\ &= \sum_{\mu\nu} C_{\mu\nu} \phi_\mu \phi_\nu - \rho, \end{aligned} \tag{14}$$

$$C_{\mu\nu} = \sum_k \lambda_k y_k \langle S_{a_1}^{\alpha_1} \ldots S_{a_n}^{\alpha_n} \rangle_{cl}^{(k)} \langle S_{a_1'}^{\alpha_1'} \ldots S_{a_n'}^{\alpha_n'} \rangle_{cl}^{(k)}, \tag{15}$$

where $C_{\mu\nu}$ is a coefficient matrix that is calculated from the learned support vectors.

The rank-n order parameter tensor, $\mathbb{O}$, can be written as a linear combination of basis tensors,

$$\mathbb{O} = \sum_{\boldsymbol{\alpha}} c_{\boldsymbol{\alpha}} \mathbf{S}^{\alpha_1} \otimes \mathbf{S}^{\alpha_2} \otimes \ldots \otimes \mathbf{S}^{\alpha_n}, \tag{16}$$

where the coordinates $c_{\boldsymbol{\alpha}}$ encode the sought after analytical structure of $\mathbb{O}$. The corresponding magnitude of the ordering is obtained by taking a tensor-analogue of the Frobenius norm $\|\mathbb{O}\|_F^2 = \sum_{a_1 \ldots a_n} |\mathbb{O}_{a_1 \ldots a_n}|^2$. Thus, when regarding the decision function as the magnitude of the order, we aim to write Eq. (14) as a tensor norm square $\|\mathbb{O}\|_F^2$ and extract the definition of $\mathbb{O}$ in terms of its basis tensors, Eq. (16). An example of this is demonstrated in Sec. V A.

In the above construction, we made the ansatz that the spins within the cluster are sufficient to define the underlying local order. This is done mostly to reduce the complexity of both the optimization itself and the subsequent analysis by eliminating the scaling with system size. Note that we do not assume the cluster to be the optimal choice to accommodate the given order. The choice of the cluster is guided by information of the lattice geometry or the Hamiltonian. E.g., one may use a number of lattice cells as a tentative cluster. If the cluster is chosen to be larger than necessary, one will find a reducible form of the order parameter and can likely infer the optimal cluster size.

B. The bias parameter in phase classification

We will now investigate the role of the bias parameter ρ in the decision function. Readers who are not interested in the technical details may revisit this section later, after seeing the examples in Sec. V. However, the basic idea can be intuitively summarized as follows. For a fully disordered spin configuration $\tilde{\mathbf{x}}$, the magnitude of the ordering and thereby the first term in the decision function Eq. (14) vanishes, leading to $d(\tilde{\mathbf{x}}) = -\rho$. Consequently, $\rho = -d(\tilde{\mathbf{x}}) = 1$, as (ideally) all disordered configurations $\tilde{\mathbf{x}}$ will fall onto the lower margin boundary. Therefore, we may use the behavior of the bias ρ as an indicator to signify the presence or absence of a phase transition: $\rho = 1$, if the data of either label correspond to the disordered and ordered phase, respectively; $\rho \neq 1$

(with significant violations) if the samples are in fact collected from the same phase or a significant portion of samples is mislabeled.

To support the above proposition, we begin by noting that the first term in SVM's optimization objective, Eq. (3), after being kernelized (Eq. (10)) with the multipolar kernel, Eq. (13), amounts to the Frobenius norm of the coefficient matrix, Eq. (15),

$$\|\mathbf{w}\|^2 \to \sum_{k,k'} \lambda_k \lambda_{k'} y^{(k)} y^{(k')} \left(\sum_\mu \phi_\mu^{(k)} \phi_\mu^{(k')} \right)^2$$
$$= \sum_{\mu\nu} \sum_k \lambda_k y^{(k)} \phi_\mu^{(k)} \phi_\nu^{(k)} \sum_{k'} \lambda_{k'} y^{(k')} \phi_\mu^{(k')} \phi_\nu^{(k')}$$
$$= \sum_{\mu\nu} C_{\mu\nu}^2 = \|C\|_F^2. \qquad (17)$$

As for the second part of the optimization objective, the optimal slack variables will assume a value given by the hinge loss, Eq. (7), i.e., they satisfy their constraint by equality, or they are unnecessary and won't incur any penalty to the objective.

The data ϕ_μ may obey some internal constraints (such as the normalization of spin vectors, orthogonality among spins, etc.) which allow for freedom in the choice of $C_{\mu\nu}$ and ρ whilst keeping the decision function invariant. In particular, given some matrix $D_{\mu\nu}$ which produces merely a constant when contracted with any valid feature vector ϕ,

$$\sum_{\mu\nu} D_{\mu\nu} \phi_\mu \phi_\nu = D_0, \qquad (18)$$

one can transform $C_{\mu\nu} \mapsto C_{\mu\nu} + \epsilon D_{\mu\nu}$ and absorb the additional constant by $\rho \mapsto \rho - \epsilon D_0$ without affecting the values of the decision function. Since only the decision function enters the hinge loss and the inequality constraints, SVM will choose the parameter ϵ freely in a way that minimizes the Frobenius norm $\|C + \epsilon D\|^2$. The solution will thus obey

$$\frac{\mathrm{d}}{\mathrm{d}\epsilon} \sum_{\mu\nu} (C_{\mu\nu} + \epsilon D_{\mu\nu})^2 = 2 \sum_{\mu\nu} (C_{\mu\nu} + \epsilon D_{\mu\nu}) D_{\mu\nu} = 0. \qquad (19)$$

The coefficient matrix $C_{\mu\nu}$ extracted from SVM already manifests the optimal choice with respect to $D_{\mu\nu}$, thus $\|C + \epsilon D\|$ is minimal for $\epsilon = 0$ which implies

$$\sum_{\mu\nu} C_{\mu\nu} D_{\mu\nu} = 0. \qquad (20)$$

One particular choice is given by $\tilde{D}_{\mu\nu} := \langle \tilde{\phi}_\mu \tilde{\phi}_\nu \rangle_{\mathrm{diso}}$ which denotes an ensemble average over configurations $\tilde{\phi}$ from the disordered phase. The requirement of Eq. (18) is in fact fulfilled,

$$\sum_{\mu\nu} \tilde{D}_{\mu\nu} \phi_\mu \phi_\nu = \langle (\tilde{\phi} \cdot \phi)^2 \rangle_{\mathrm{diso}} = \mathrm{const.}, \qquad (21)$$

as the disorder average amounts to an isotropic integral over the disordered spins $\tilde{\mathbf{S}}$ independently and thus the fixed, arbitrary feature vector ϕ can be eliminated from the integrand by a change of variables.

We now calculate the value of the decision function as it is measured in the disordered phase, i.e. we subject its argument to the same disorder average,

$$\langle d(\tilde{\mathbf{x}}) \rangle_{\mathrm{diso}} = \left\langle \sum_{\mu\nu} C_{\mu\nu} \tilde{\phi}_\mu \tilde{\phi}_\nu - \rho \right\rangle_{\mathrm{diso}}$$
$$= \sum_{\mu\nu} C_{\mu\nu} \langle \tilde{\phi}_\mu \tilde{\phi}_\nu \rangle_{\mathrm{diso}} - \rho$$
$$= \sum_{\mu\nu} C_{\mu\nu} \tilde{D}_{\mu\nu} - \rho = -\rho, \qquad (22)$$

by virtue of Eq. (20). This implies that the decision function assumes a constant value throughout the disordered phase. Indeed, local order parameters are typically zero throughout the disordered phase and pick up finite values as the transition to the ordered phase takes place. One should therefore shift the decision function by ρ to interpret it as an order parameter.

Since deep in the disordered phase, the individual spins are independent, the lattice average in the definition of the feature vector $\tilde{\phi}$ already averages over many disordered spins. Thus, given a sufficiently large system, the statement can be refined to $d(\tilde{\mathbf{x}}) = -\rho$ for spin configurations $\tilde{\mathbf{x}}$ in the disordered phase.

The phase classification problem is distinct from generic classification problems in the sense that all the data from one class, the disordered phase, (on average) trace out an isosurface of the decision function. The decision boundary as well as the 'upper' and 'lower' margin boundaries are isosurfaces of the decision function too, corresponding to values, of 0, +1, and −1, cf. Sec. II. Thus, the 'lower' margin boundary will fall onto the disordered samples, i.e. $d(\tilde{\mathbf{x}}) \approx -1$, which implies $\rho = 1$.

In Sec. V, we will illustrate the above interpretation in various scenarios. Further, in Sec. VII, we base our analysis of the topology of an unknown phase diagram solely on the bias.

IV. MODEL AND SAMPLES

To validate the capability of the kernel Eq. (13) to detect general multipolar orders, we employ the following classical Hamiltonian to generate samples[36],

$$H = \sum_{\langle i,j \rangle} J_{ab}^{\alpha\beta} S_{i,a}^\alpha U_{ij}^{\beta\gamma} S_{j,b}^\gamma. \qquad (23)$$

Here, $\langle i, j \rangle$ enumerates nearest-neighboring sites on a cubic lattice. $S_{i,a}^\alpha$ denotes the components $a \in \{x, y, z\}$ of three orthogonal $O(3)$ spins at lattice site i, which are distinguished by a color index $\alpha \in \{\mathrm{l, m, n}\}$. These spins form a local triad that can represent general spin orientations. In the notation of Sec. III A, these triads constitute

the spin clusters. $J_{ab}^{\alpha\beta}$ denotes a general exchange interaction. Einstein summation over color and component indices is implied.

The Hamiltonian Eq. (23) is reminiscent of the general exchange interaction of nearest-neighboring spins, $H_{\text{ex}} = \sum_{\langle i,j \rangle} J_{ab} S_{i,a} S_{j,b}$. However, in the Eq. (23), there are additional fields U_{ij} living on the bonds $\langle i,j \rangle$. These are rotation matrices realizing elements of a three-dimensional point group G, $U_{ij} \in G \subset O(3)$, and rotate the color indices of a spin, $S_a^\alpha = U^{\alpha\beta} S_a^\beta$. These fields are in fact gauge fields, and the Hamiltonian Eq.(23) possesses a local point-group symmetry,

$$S_{i,a}^\alpha \mapsto \Lambda_i^{\alpha\alpha'} S_{i,a}^{\alpha'}, \qquad (24)$$
$$U_{ij}^{\alpha\beta} \mapsto \Lambda_i^{\alpha\alpha'} U_{ij}^{\alpha'\beta'} \Lambda_j^{\beta'\beta}, \qquad \forall\, \Lambda_i, \Lambda_j \in G, \qquad (25)$$

in addition to a global $O(3)$ symmetry

$$S_{i,a}^\alpha \mapsto S_{i,a'}^\alpha \Omega_{a'a}, \qquad \forall\, \Omega \in O(3). \qquad (26)$$

The details of the gauge theory are however not relevant to the present work (cf. Ref. 36 for a comprehensive introduction to the theory). It is utilized merely for its flexibility to produce configurations of various multipolar orders. Specifically, since it is impossible to spontaneously break a gauge symmetry[37], the Hamiltonian (23) develops orders characterized by a ground state manifold $O(3)/G$. Therefore, by choosing the gauge symmetry G, we can effectively simulate different spin orders. For example, when $G = O(2)$, the gauge theory recovers the Heisenberg model with general exchange interaction,

$$H_{\text{Heis}} = \sum_{\langle i,j \rangle} J'_{ab} S_{i,a}^{\text{n}} S_{j,b}^{\text{n}}, \qquad (27)$$

while with $G = D_{\infty h}$ it reduces to the Lebwohl-Lasher model[38,39]

$$H_{\text{LL}} = \sum_{\langle i,j \rangle} J''_{ab} (S_{i,a}^{\text{n}} S_{j,b}^{\text{n}})^2. \qquad (28)$$

We refer the reader to Ref. 35 for a mathematical derivation of Eqs. (27) and (28).

For the purpose of the current manuscript, we will focus on a simple quadrupolar order, Eq. (28), a generalized quadrupolar order ($G = D_{2h}$) and a coplanar octupolar order ($G = D_{3h}$). The input data to SVM are raw spin configurations $\mathbf{x} = \{S_{i,a}^\alpha\}$ which are generated by performing Monte Carlo simulations on the effective gauge theory. We note that neither the existence of any order, nor the gauge fields are known to SVM. The task of SVM is to detect those orders and their order parameter without prior knowledge.

V. LEARNING A SINGLE QUADRUPOLAR ORDER

In this section, we will take a simple quadrupolar order as an example to illustrate the basic idea of our method.

FIG. 1. Coefficient matrix $C_{\mu\nu} = C_{ab,a'b'}$ for the quadrupolar order with the rank-2 kernel. The pattern can be decomposed according to Eq. (30) and leads to the uniaxial nematic tensor Q_{ab}. 28k samples have been used for training.

We will discuss the detection of the order and the decoding of the machine result to extract the order parameter. Further, we will compare the performance of different training schemes; this discussion may also be applicable to other machine learning methods for phase classification.

A. Extracting the order parameter

One can generate spin configurations with an emerging quadrupolar order by working with the gauge symmetry $G = D_{\infty h}$ and an anisotropic coupling where $J_{ab}^{\alpha\beta} = -J\delta_{ab}$ for $\alpha\beta = $ nn and $J_{ab}^{\alpha\beta} = 0$ for other $\alpha\beta$'s.

The resulting model is then equivalent to the Lebwohl-Lasher model in Eq. (28) and the $\mathbf{S}^{\text{l}}$ and $\mathbf{S}^{\text{m}}$ spins become irrelevant. Thus, to simplify the exposition of our analysis, we take only the $\mathbf{S}^{\text{n}}$ spins as input data to train the SVM, yielding the configuration vector $\mathbf{x} = \{\mathbf{S}_i^{\text{n}}\}$. However, as will be discussed in Sec. VI A, including $\mathbf{S}^{\text{l}}$ and $\mathbf{S}^{\text{m}}$ spins does not change the result.

We train the SVM successively with the multipolar kernel for increasing tensor rank, i.e. for increasing degrees n of the monomials in the mapping Eq. (12). As we have discussed in Ref. 25, when the rank is not sufficient to capture the order parameter, the SVM overfits the training data. Here, this is the case for rank $n = 1$ which produces an erratic decision function, whereas we capture the order at rank $n = 2$, resulting in a curve.

By calculating the coefficient matrix (cf. Sec. III A), we can interpret the resulting decision function to access its analytical structure.

$$d(\mathbf{x}) = \sum_{\mu\nu} C_{\mu\nu} \phi_\mu \phi_\nu - \rho$$
$$= \sum_{ab} \sum_{a'b'} C_{ab,a'b'} \langle S_a^{\text{n}} S_b^{\text{n}} \rangle \langle S_{a'}^{\text{n}} S_{b'}^{\text{n}} \rangle - \rho. \qquad (29)$$

The coefficient matrix $C_{ab,a'b'}$ is shown in Fig. 1 and encodes the contribution of the individual terms in the sum. It can be readily decomposed into three contractions

$$C_{ab,a'b'} = p_1 \boxed{\diagbox} + p_2 \boxed{\cdots} + p_3 \boxed{\cdots} \qquad (30)$$

$$= p_1 \delta_{aa'}\delta_{bb'} + p_2 \delta_{ab'}\delta_{ba'} + p_3 \delta_{ab}\delta_{a'b'}, \qquad (31)$$

where we read off $p_1 = p_2 \approx \frac{3}{4}$ and $p_3 \approx -\frac{1}{2}$. The first two contractions are compatible with the form $\|\mathbb{O}\|_F^2 = \sum_{ab} \mathbb{O}_{ab}^2$, as they contract indices between two tensors. We shall call them 'proper' contractions. The third contraction on the other hand is not, but it only produces a constant $\sum_{ab} \langle S_a^n S_a^n \rangle \langle S_b^n S_b^n \rangle = 1$ independent of the configuration vector. This 'self-contraction' is an example where an internal constraint of the data (here, the normalization of the spins) allows the SVM to choose the weight p_3 freely in line with its optimization objective, without affecting the ability of the decision function to distinguish the phases, as pointed out in Sec. III B.

Substituting Eq. (31) back to the decision function, and making use of the properties $\langle S_a^n S_b^n \rangle = \langle S_b^n S_a^n \rangle$ and $\|\mathbf{S}^n\| = 1$, we can write the decision function as the (squared) magnitude of a tensor order parameter, up to a linear rescaling:

$$d(\mathbf{x}) = \frac{3}{2} \sum_{ab} \left(\langle S_a^n S_b^n \rangle - \frac{1}{3}\delta_{ab} \right)^2 - \rho. \qquad (32)$$

One identifies the (uniaxial) nematic tensor[40] $Q_{ab} = \langle S_a^n S_b^n \rangle - \frac{1}{3}\delta_{ab}$. In Fig. 3b, the rescaled decision functions are compared to the true order parameter measured from Monte Carlo simulations.

B. Comparison of training schemes

We now examine several different schemes to generate training data for the SVM. These are illustrated in Fig. 2 and labeled (a) through (f).

One can either use samples generated from a uniform distribution of temperatures crossing the phase transition, (a) through (c), or at two discrete temperatures deep inside each phase, as shown in Fig. 2 (d). When adopting the first approach, one has to set a discriminatory temperature, T_{disc}, to label samples taken at temperatures below and above T_{disc}. Ideally, it should coincide with the critical temperature, $T_{\text{disc}} = T_c$, which we consider in scheme (a), whereas we deliberately chose $T_{\text{disc}} \neq T_c$ in schemes (b) and (c) to study the effect of misclassified samples. If the temperature distribution is discrete, the notion (and choice) of a discriminatory temperature is not necessary. Schemes (e) and (f) are discussed in the next section.

In all cases, we measured the decision function and examined the $C_{\mu\nu}$ matrix as before. As anticipated, the coefficient matrix exhibits the same pattern as shown in
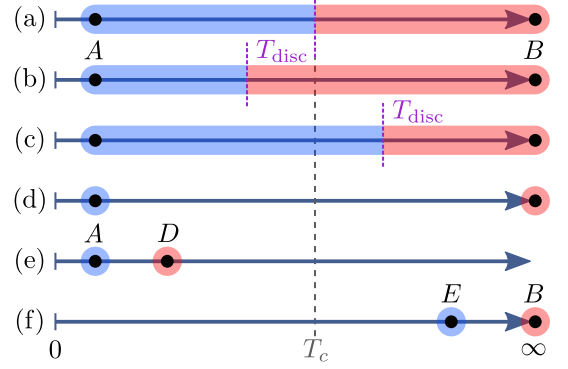
FIG. 2. Graphical representation of different training schemes. Samples are taken from temperatures corresponding to the shaded regions and labeled as 'ordered' (blue) or 'disordered' (red). The first three schemes use a continuous distribution of temperatures from $T_A \ll T_c$ to $T_B = \infty$ and differ by the discriminatory temperature T_{disc} used to assign the labels, where $T_{\text{disc}} = T_c = 0.52$ (a), $T_{\text{disc}} = 0.4 < T_c$ (b), and $T_{\text{disc}} = 0.64 > T_c$ (c) have been chosen, respectively. The last three schemes take samples at two discrete temperatures. (a)-(d) are discussed in Sec. V B; (e) and (f) in Sec. V C. The concrete temperatures read $T_A = 0.2$, $T_B = \infty$, $T_D = 0.25$, $T_E = 2$.

	T_{disc}	$\nu = 0.1$		$\nu = 0.4$	
		ρ	δ	ρ	δ
(a)	0.52	1.012	0.14	1.001	0.063
(b)	0.4	6.83	1.97	1.003	0.042
(c)	0.64	2.32	0.14	1.0012	0.087
(d)	n/a	1.0012	0.0014	1.0007	0.0013
(e)	n/a	33.9	0.0090	25.2	0.0048
(f)	n/a	0.15	1.78	6.1	0.41

TABLE I. The SVM bias parameter ρ and the deviation δ from the true uniaxial nematic order parameter are given for 28k samples generated according to the training schemes (a)-(f) and for weak ($\nu = 0.1$) and strong ($\nu = 0.4$) regularization. $\rho \approx 1$ indicates a phase transition was captured.

Fig. 1 for continuous (a) and discrete (d) temperature distributions, meaning the physical order parameter is captured regardless.

To further quantify the performance of these training schemes, we introduce a deviation metric,

$$\delta := \frac{\|\mathbf{C} - \tilde{\mathbf{C}}\|_F}{\|\tilde{\mathbf{C}}\|_F} \geq 0, \qquad (33)$$

which measures the element-wise discrepancy between the learned $C_{\mu\nu}$ and the theoretical one, $\tilde{C}_{\mu\nu}$. We note that this metric is rather sensitive and a value of $\delta = 0.1$ already constitutes a rather good result.

Table I shows results for the various training schemes and for weak ($\nu = 0.1$) and strong ($\nu = 0.4$) regularization. Besides the deviation δ, we tabulate the bias ρ which is accessible as part of the optimization result and

we expect it to attain a value of 1 if the SVM learned a physical order parameter for reasons laid out in Sec. III B.

First, we observe that the order parameters extracted using strong regularization are always better than those obtained using weaker regularization. This is consistent with our previous findings for high-rank order parameter tensors in Ref. 25.

We also observe that training set (d) in fact yields the best results, even compared to set (a) where the accurate critical temperature was used to discriminate the phases. Thus, SVM seems to work best if the training data exhibit the characteristics of the ordered phase most pronounced. It does not benefit from training data in the vicinity of the critical point.

However the continuous training sets (a) through (c) allow for a validation of the tentative phase diagram. Whereas in (a) the correct order parameter is learned for both degrees of regularization, the misclassification of the samples with temperatures between T_{disc} and T_c in sets (b) and (c) induces significant deviations from the true order parameter at weak regularization which is compensated for at stronger regularization. If continuous training is used in this way to verify the existence of a transition at T_{disc}, a relatively weak regularization may be desired.

We also note the correlation between a small deviation δ and a bias ρ close to one in all cases for (a) through (d). This is consistent with our expectation of $\rho = 1$ for a physical order parameter and enables us to use the bias to gauge the quality of the learned order parameter without examining its analytical structure. In case (b), samples from the ordered phase are wrongly labeled as disordered and push the SVM margin boundary away from the manifold of truly disordered samples which both induces a bias $\rho \gg 1$ and deforms the decision boundary, resulting in a faulty order parameter, $\delta \gg 0$. Likewise, in the opposite case (c), which falsely classifies disordered samples as ordered ones, the margin is forcibly kept very narrow which makes it prone to overfitting.

In the left panel of Fig. 3, we show the 'raw' decision functions without any rescaling. Note that the constant value attained in the disordered phase corresponds to $-\rho$. The absolute scale of the decision function is related to the inverse width of the SVM margin. For the continuous sampling schemes, the margin is more narrow as it is constrained by samples close to the transition. In the discrete scheme (d), the decision function gives values between -1 and 1, corresponding to the extreme temperatures it was trained at. Further, we point out that the zero-crossing of the raw decision function, which determines the label of a testing sample, approximately reproduces the discriminatory temperatures used in the continuous training schemes (a) and (b), whereas the discrete scheme (d) is not trained to classify samples close to the transition.

While these features of the 'raw' decision function can provide a deeper understanding of the workings of the SVM, they are physically irrelevant and we therefore shift

(a) raw decision functions   (b) rescaled
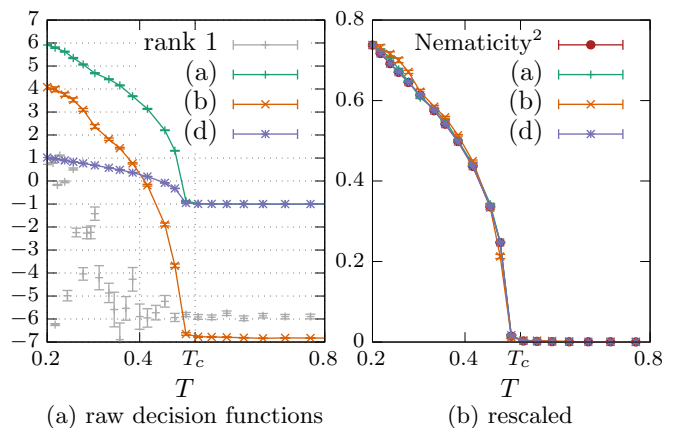
FIG. 3. The decision function, obtained from the earlier training with regularization $\nu = 0.1$, is measured as an observable in Monte Carlo simulations at temperatures spanning the phase transition. The left panel shows the numerical values of the decision functions as obtained from SVM. When training the SVM with the rank-1 kernel (shown in gray), the quadrupolar order cannot be captured, resulting in an erratic decision function. The curves (a), (b), and (d) are trained with the rank-2 kernel and correspond to the eponymous training schemes, cf. Fig. 2. The zero-crossing of the decision functions marks the decision boundary, whereas the constant value for $T > T_c$ is given by $-\rho$, cf. Tab. I. The right panel shows the same curves, but shifted up by ρ, and are seen to match the square of the uniaxial nematicity after being rescaled.

the decision functions by ρ to obtain a value of zero in the disordered phase and rescale them to match the true nematicity. As can be seen from the right panel in Fig. 3, the result is very close in all three cases, even for scheme (b).

C. Training in the same phase

We now examine two more training schemes where the samples are taken at two discrete temperatures which are both in the ordered phase (e) or disordered phase (f). We do so to assess the capability to distinguish these cases where no phase transition takes place from those where it does, (a)-(d). This can be physically relevant in situations where the phase boundary is not known, source data is limited, or when multiple order parameters are involved, while not all of them experience a phase transition.

Interestingly, when both training temperatures lie within the ordered phase (e), the SVM still manages to learn the order parameter, as is apparent from the very small deviations δ in Tab. I. This can be explained by the fact that the magnitude of the true order parameter is not constant within the ordered phase, but increases in value as one moves deeper inside the phase. The true order parameter is still the best decision function the SVM can learn in this situation to distinguish samples from the

two temperatures. Note that this enables us to measure the decision function at a range of temperatures which can even exceed the training temperatures and read off the factual transition point from the decision function.

Despite the excellent δ in scenario (e), the corresponding biases ρ are far away from unity. This is no surprise as the argument put forth in Sec. III B relied on the properties of disordered samples. Thus, the bias enables us to distinguish this situation from case (d) where a transition takes place between the training temperatures.

We get a different picture when we attempt to train the SVM with samples from two temperatures in the disordered phase (f). The magnitude of the true tensor order parameter is constant (zero) throughout the disordered phase which makes it unsuitable to distinguish samples from the two temperatures. Thus, SVM does not learn the correct order parameter, as is manifest from the large value of δ in Tab. I, but instead overfits the training data in an attempt to construct a better decision function. As a further indication of overfitting, the decision function is not reproducible. Likewise, also the bias ρ is fluctuating and in general will not be close to one.

VI. LEARNING MULTIPLE ORDERS

If a spin or orbital system of interest develops an order which needs to be characterized by some axial point group, in general, it may require more than one (multipolar) order parameter. For example, a generalized quadrupolar order is defined by two rank-2 tensors. In this section, we demonstrate the ability of our kernel Eq. (13) to detect multiple orders simultaneously. We consider situations where the orderings occur at the same rank and at different ranks and assume here that the topology of the phase diagram is approximately known. In Sec. VII, we propose a scheme that yields a phase diagram in situations where such information is absent.

A. Order parameters of the same rank

A biaxial order D_{2h} is considered. The training samples are prepared by choosing the gauge symmetry $G = D_{2h}$ and the exchange coupling $J_{ab}^{\mathrm{ll}} = J_{ab}^{\mathrm{mm}} = -J_1\delta_{ab}$, $J_{ab}^{\mathrm{nn}} = -J_3\delta_{ab}$ in Eq. (23). The gauge model is then equivalent to the Straley model of generalized quadrupolar orders[41]. J_1 and J_3 are used as tuning parameters for convenience.

The associated phase diagram is shown in Fig. 4. It requires at least two order parameters to characterize the three phases and their transitions. If the two orders are weakly coupled, they will experience (dis-)ordering separately with one of them being irrelevant in the corresponding phase transition. However, if the coupling between them is strong, they may develop ordering in a single phase transition and need to be taken into account
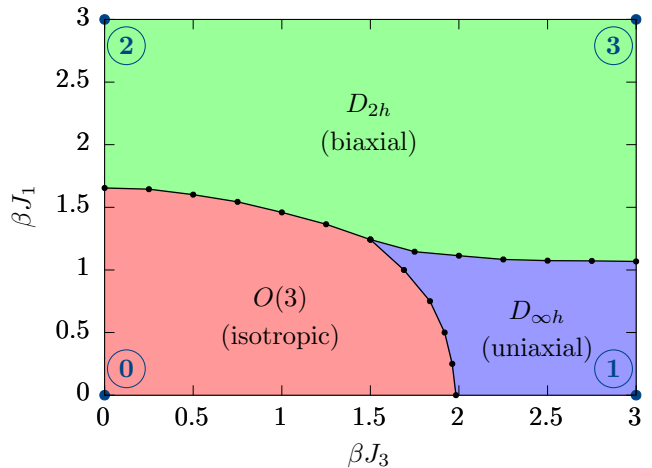
FIG. 4. The phase diagram for the generalized quadrupolar order, cf. Ref. 42. The points labeled ⓪ through ③ are used to generate training samples.

	$\nu = 0.1$	$\nu = 0.5$
	ρ	ρ
⓪ / ①	1.0017	1.0007
⓪ / ②	1.0013	1.0007
⓪ / ③	1.0010	1.0006
① / ②	0.972	0.9995
① / ③	1.27	1.34
② / ③	8.05	7.50

TABLE II. The value of the bias parameter in learning the phases of the generalized quadrupolar order. Training samples are collected from the corresponding regimes in Fig. 4) and labeled accordingly. This results in six classifiers, each distinguishing between a pair of those four labels. A total of 25k samples have been used in the training, and results from weak ($\nu = 0.1$) and strong ($\nu = 0.5$) regularization are shown for comparison. Note that for the classifier between ② and ③, we observe $\rho \gg 1$, indicating that no phase transition takes place between those points.

simultaneously. This is the case in the transition between the biaxial phase and the isotropic phase.

Given the topology of the phase diagram, without knowledge of the exact phase boundary, we can train the SVM with configurations sampled at points deep in each phase. Here, we took samples at four points, ⓪-③, corresponding to the corners of the phase diagram shown in Fig. 4.

With the multipolar kernel at rank 2, we now perform multiclassification as described in Sec. II D where we labeled the samples with $M = 4$ labels corresponding to the points ⓪-③. This yields six classifiers. Their respective biases are tabulated in Tab. II. We can note straight away that the classifier distinguishing points ② and ③ does not correspond to a phase transition as its bias is far away from one. This is consistent with our

| | $\nu = 0.1$ | $\nu = 0.5$ | |
	ρ	ρ	Fig.
$O(3)$ / $D_{\infty h}$	1.0017	1.0007	5
$D_{\infty h}$ / D_{2h}	0.985	1.0004	6a
$O(3)$ / D_{2h}	1.0011	1.0005	6b

TABLE III. Same as Tab. II, but now the data sampled at points ② and ③ are assigned the same label 'D_{2h}'; correspondingly ⓪ is labeled as '$O(3)$' and ① as '$D_{\infty h}$'. In all cases, the biases are close to one, indicating that the labeling indeed represents the phases. The last column refers to the figures showing the (block structure of the) coefficient matrix corresponding to that classifier.

(a) full $C_{\mu\nu}$ (b) block structure
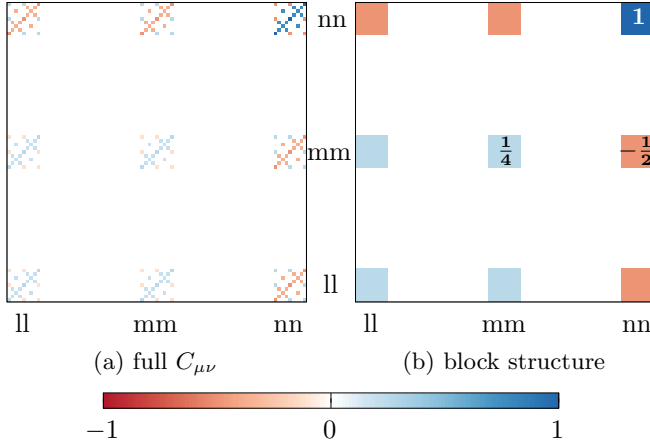
FIG. 5. The coefficient matrix distinguishing the isotropic ($O(3)$) and uniaxial ($D_{\infty h}$) phase of the generalized quadrupolar order. The left panel displays the full matrix $C_{\mu\nu}$ where the multi-indices μ and ν are sorted lexicographically and where the color indices are more significant. The resulting block structure is shown in the right panel. The block weights have been obtained by summing all matrix elements within each block over their component indices and normalizing to the top-right (nn, nn) block. The interpretation leads to an alternative expression of the nematic Q_{ab} tensor, given in Eq. (34). The pattern exhibited within each block is identical to Fig. 1.

knowledge that these points are indeed both in the biaxial phase. The bias of the classifier between ① and ③ is also somewhat elevated, so it is not clear that a phase transition takes place. However, after merging the labels ② and ③ and repeating the multiclassification with $M = 3$ distinct labels (which now correspond to the phases), we find that the bias of the corresponding classifier is closer to one (cf. Tab. III), affirming that a phase transition is present. The larger deviation from one in this case can be explained by the fact that the uniaxial order is developed in both phases whereas the magnitude of the ordering is larger in the biaxial phase.

We now proceed with the analysis of the three coefficient matrices obtained from the latter classification with respect to three labels. Let us first revisit the nematic or-

der which is now responsible for the isotropic-to-uniaxial transition in the regime of small J_1 in Fig. 4. The corresponding $C_{\mu\nu}$ matrix is shown in Fig. 5a. With $\mathbf{S}^l$ and $\mathbf{S}^m$ degrees of freedom now included, it contains 81×81 elements, defining contractions between $\phi_\mu = \langle S_{a_1}^{\alpha_1} S_{a_2}^{\alpha_2} \rangle$ and $\phi_\nu = \langle S_{b_1}^{\beta_1} S_{b_2}^{\beta_2} \rangle$. However, when arranging the multi-indices such that color indices are more significant than component indices (i.e. $\mu = (\alpha_1, \alpha_2; a_1, a_2)$), it can be divided into a 9-by-9 block structure in terms of the color indices $(\alpha_1 \alpha_2)$, $(\beta_1 \beta_2)$, corresponding to the 9 basis tensors in Eq. (16). Each block in fact features the same pattern of Fig. 1 up to a weight factor. In Sec. V A, its relation to the order parameter tensor was explained. In the following analysis, we can ignore the internal pattern of the blocks entirely and instead infer the coordinates of the order parameter tensor in the 9-dimensional tensor space from the relative weight factors of those blocks. The weight of a block may be obtained by a sum over all elements within the block as $\sum_{ab} C_{ab}^{\alpha\beta}$, or alternatively by the Frobenius norm as $\sqrt{\sum_{ab}(C_{ab}^{\alpha\beta})^2}$. We have used both definitions and verified that they yield the same weights up to a sign. The first definition preserves the sign of the block weights, so we adopted it. With this definition, normalized to the most pronounced (nn, nn) block in the top-right corner in Fig. 5a, the weights of the (ll, ll) and (mm, mm) blocks turn out to be $\frac{1}{4}$, while those of the (nn, ll) and (nn, mm) blocks are $-\frac{1}{2}$, cf. Fig. 5b. This gives an order parameter of the form

$$\mathbb{O}_{\mathrm{uni}} = \mathbf{S}^n \otimes \mathbf{S}^n - \frac{1}{2}\mathbf{S}^m \otimes \mathbf{S}^m - \frac{1}{2}\mathbf{S}^l \otimes \mathbf{S}^l. \quad (34)$$

Using the relation $\sum_{\alpha=l,m,n} \mathbf{S}^\alpha \otimes \mathbf{S}^\alpha = \mathbb{1}$, we recover the nematic tensor of Eq. (32),

$$\mathbb{O}_{\mathrm{uni}} = \frac{3}{2}(\mathbf{S}^n \otimes \mathbf{S}^n - \frac{1}{3}\mathbb{1}). \quad (35)$$

Thus, to extract the order parameter, it is sufficient to rely on the block structure of $C_{\mu\nu}$, rather than the full matrix. This simplifies the interpretation significantly, especially for learning high-rank[25] and multiple simultaneous orders.

In Fig. 6, we show the block structure of the coefficient matrix for the other two classifications. The direct sum is used to compute the block weights. Fig. 6a is the weight matrix trained with data sampled deep inside the uniaxial and biaxial phase. One immediately realizes a tensor of the form

$$\mathbb{O}_{\mathrm{bi}}^{(D_{2h})} = \mathbf{S}^l \otimes \mathbf{S}^l - \mathbf{S}^m \otimes \mathbf{S}^m, \quad (36)$$

which is indeed the biaxial order parameter describing the uniaxial-to-biaxial phase transition[40].

Fig. 6b is the weight matrix trained with samples from the isotropic and the biaxial phase. It is a superposition of that in Fig. 5b and Fig. 6a, with the weight denoted by p_1 and p_2. Its interpretation then leads to a set of two order parameters, $\{\mathbb{O}_{\mathrm{uni}}, \mathbb{O}_{\mathrm{bi}}^{(D_{2h})}\}$, which is indeed the

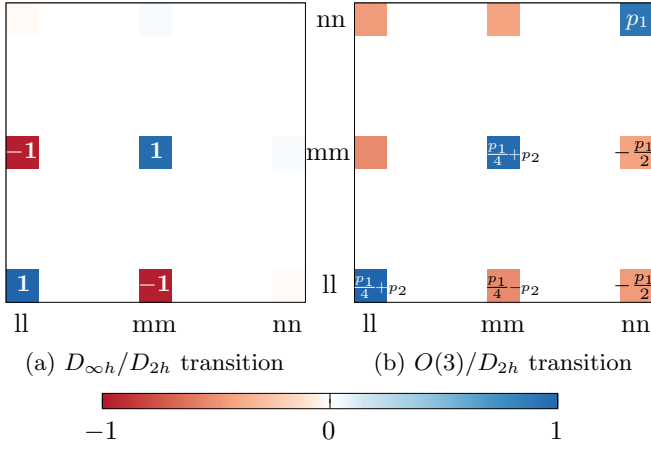(a) $D_{\infty h}/D_{2h}$ transition   (b) $O(3)/D_{2h}$ transition

FIG. 6. The block structures of the $C_{\mu\nu}$ matrix of the remaining two decision functions of the generalized quadrupolar order, (a) the uniaxial ($D_{\infty h}$) and biaxial (D_{2h}) phase, and (b) the isotropic ($O(3)$) and biaxial (D_{2h}) phase. The latter is a superposition of the patterns exhibited by the other two decision functions, shown in Fig. 5b and Fig. 6a. The block weights are not fixed, but rather the coefficients p_1 and p_2 vary with the relative strength of the two orderings.

order parameter set that uniquely defines the D_{2h} biaxial phase. (Fig. 6b can also be viewed as a superposition of three order parameters, $\mathbf{S}^l \otimes \mathbf{S}^l - \mathbf{S}^m \otimes \mathbf{S}^m$, $\mathbf{S}^m \otimes \mathbf{S}^m - \mathbf{S}^n \otimes \mathbf{S}^n$ and $\mathbf{S}^n \otimes \mathbf{S}^n - \mathbf{S}^l \otimes \mathbf{S}^l$, where two of them are independent. This is just an equivalent expression of the same ordering.)

We find that the weight of p_1 and p_2 vary as we change the points in phase diagram from which we sample. This does not come unexpected, as the cooccurrence and the expression of the two orders are protected by symmetry (of their ground state manifold), so their relative ratio depends on the microscopic coupling strengths (the value of J_1 and J_3 in this case).

This further emphasizes the importance of interpretability. In a physical problem, we typically treat different order parameters as individual quantities and measure them separately. However, a machine may combine them into a single numerical classifier. Therefore, a machine trained by samples with certain relative ratios of those order parameters may not be optimal or even misleading when being applied to samples where those ratios vary. In order to make correct predictions, one may need to be able to identify and isolate each order parameter from the machine result. This is however not an issue for our kernel method (or in general for an interpretable machine). The occurrence of multiple orders only leads to a *linear* superposition of the pattern of each single order, even though those orders may be strongly coupled.

B.   Order parameters of different ranks

The quadrupolar and octupolar orders in classical Kagomé antiferromagnets provide examples for spin systems which simultaneously develop multipolar orders of different ranks. Indeed, the quadrupolar order in Eq. (35) is compatible with all dihedral symmetries D_n, D_{nh}, D_{nd}, and axial symmetries S_{2n}, C_{nh}. That is, if the ground state manifold of a spin order cannot be ascertained theoretically, we should not immediately conclude a uniaxial phase after observing a quadrupolar order. This is also true for dipolar (rank-1) orders which are compatible with axial symmetries C_n and C_{nv}, while the Heisenberg and Néel magnetization are just limiting cases for $n \to \infty$, $O(2) = C_{\infty v}$. Therefore, to correctly characterize a multipolar phase, in principle we need to identify all relevant orders.

The kernel Eq. (13) handles these situations in a straightforward manner. We simply train the SVM with kernels at different ranks n separately. Moreover, considering the crystallographic background, it is sufficient to set an upper bound $n \leq 6$. Additional complexity may arise in distinguishing nontrivial high-rank orders from responses which are already captured by order parameters of lower rank. For example, if we measure the quadrupolar order parameter $Q_{ab} = S_a S_b - \frac{1}{3}\delta_{ab}$ in a ferromagnetic phase, it will also show a finite response. This quadrupolar order is nevertheless trivial since it is completely captured by a dipolar order, namely the magnetization. However, this is easily resolved if the machine is interpretable, as is the case for our kernel. We can extract the analytical expression of the learned order parameters at different ranks, and identify those without lower-rank origins.

To be concrete, we take a simultaneous occurrence of quadrupolar and octupolar orders as an example. This can be realized by choosing $G = D_{3h}$ in the gauge theory Eq. (23). Analogously to the previous section, the training samples are collected from deep within the ordered D_{3h} phase, the uniaxial $D_{\infty h}$ phase, and the isotropic $O(3)$ phase.

We then carry out the SVM optimization with kernels of ranks 1 through 6. The biases of the resulting classifiers are tabulated in Tab. IV. We see that rank 1 is insufficient to capture any order. At rank 2, we find an order parameter describing both the isotropic-to-uniaxial and isotropic-to-biaxial transitions, whereas no order parameter describing the uniaxial-to-biaxial transition could be found. The coefficient matrix $C_{\mu\nu}$ is identical to that of Fig. 5, indicating a quadrupolar order has been captured. At rank 3, matters are reversed and we capture the uniaxial-to-biaxial transition but not the isotropic-to-uniaxial one. The isotropic-to-biaxial transition exhibits a coexistence of both the quadrupolar and the rank-3 order parameter. At higher ranks, we do not find new nontrivial order parameters but those that we do find are functions of the lower-rank ones. At rank 4, we essentially learn the square of the quadrupolar order

rank	$O(3)/D_{\infty h}$ ρ	$D_{\infty h}/D_{3h}$ ρ	$O(3)/D_{3h}$ ρ
1	0.134	2.95	2.61
2	1.0019	5.87	1.0018
3	4.10	1.0012	1.0011
4	1.0029	0.617	1.0025
5	0.981	1.0012	1.0011
6	1.0018	1.087	1.0016

TABLE IV. The biases of the three SVM classifiers discerning the isotropic ($O(3)$), uniaxial ($D_{\infty h}$) and octupolar (D_{3h}) phases. Decision functions are monitored for ranks 1 through 6 of the kernel. As before, $\rho \approx 1$ indicates that an order parameter of that rank could be learned. This is the case for the uniaxial order at ranks 2, 4, 5, and 6, and for the biaxial order at ranks 3, 5, and 6. In both cases, the higher-rank ones are trivial functions of their lowest-rank representation. The transition between the isotropic and biaxial phase in the last column experiences simultaneous uniaxial and biaxial ordering and will thus learn an order parameter for any rank $n \geq 2$.
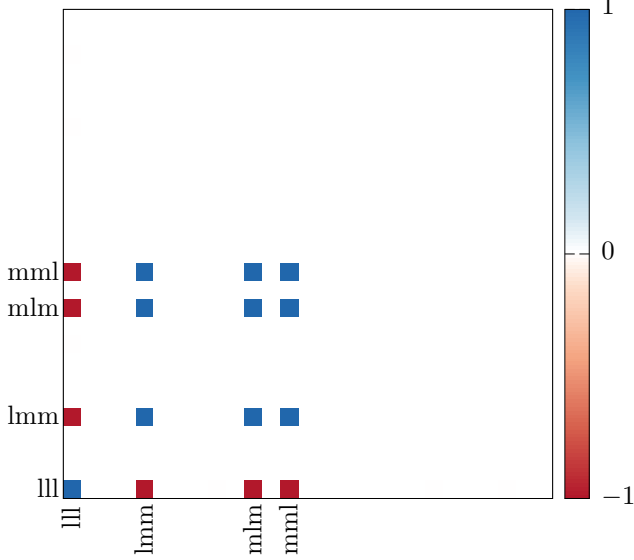
FIG. 7. The block structure of the $C_{\mu\nu}$ matrix learned by the rank-3 kernel, distinguishing the octupolar phase (D_{3h}) from the isotropic ($O(3)$) and the uniaxial phase ($D_{\infty h}$). The color indices are sorted lexicographically and those of the nonvanishing blocks are explicitly given. The interpretation of this pattern gives the rank-3 octupolar ordering tensor in Eq. (37).

parameter. At ranks 5 and 6, the tensor can be constructed from both of the lower-rank ones. Rather than relying on the bias criterion to determine if an order parameter was found, one may also measure the decision functions and discard them if they exhibit erratic behavior, similarly to how rank 1 was ruled out as an order parameter in Sec. V A. This approach leads to the same conclusions.
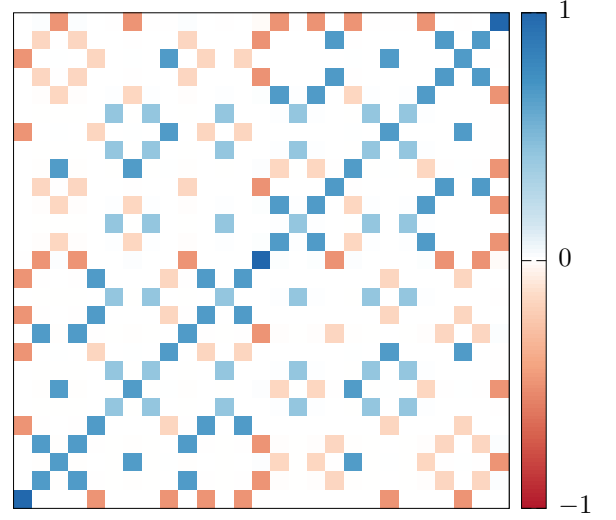
FIG. 8. Internal structure of the (lll, lll) block of the coefficient matrix in Fig. 7. The axes are labeled by lexicographically sorted triplets of component indices (xxx, xxy, ...). This pattern corresponds to the expression Eq. (38) and in turn gives the one-spin form of the octupolar ordering tensor in Eq. (40).

The coefficient matrix for the rank-3 order is shown in Fig. 7. For conciseness, we omit the full matrix and only show its block structure. We infer the contraction of two tensors of the form

$$\mathbb{O}_{\text{bi}}^{(D_{3h})} = \mathbf{S}^{\text{l}} \otimes \mathbf{S}^{\text{l}} \otimes \mathbf{S}^{\text{l}} - \mathbf{S}^{\text{l}} \otimes \mathbf{S}^{\text{m}} \otimes \mathbf{S}^{\text{m}}$$
$$- \mathbf{S}^{\text{m}} \otimes \mathbf{S}^{\text{l}} \otimes \mathbf{S}^{\text{m}} - \mathbf{S}^{\text{m}} \otimes \mathbf{S}^{\text{m}} \otimes \mathbf{S}^{\text{l}}, \quad (37)$$

which is the octupolar order parameter that was also found in classical Kagomé antiferromagnets[6]. It is a two-spin form of the D_{3h} octupolar order. However, for a coplanar order, the $\mathbf{S}^{\text{m}}$ spin is arbitrary in the sense that we can always introduce another spin orthogonal to $\mathbf{S}^{\text{l}}$. Therefore, the octupolar order may also be defined by the $\mathbf{S}^{\text{l}}$ spins alone. To obtain the one-spin representation, we simply need to examine elements inside the (lll, lll) block of the $C_{\mu\nu}$ matrix, shown in Fig. 8.

This is similar to the discussion of the quadrupolar order in Sec. V A, but involves more contractions owing to the higher rank. Still, these contractions and their weight can be readily inferred,

$$C_{abc,a'b'c'}^{\text{lll,lll}} = p_1 \big(\delta_{aa'} \delta_{bb'} \delta_{cc'} + \text{permutations of } \{a'b'c'\} \big)$$
$$+ p_0 \big(\delta_{aa'} \delta_{bc} \delta_{b'c'} + \text{other self-contractions} \big), \quad (38)$$

where p_1 and p_0 denote the weight of proper and self-contractions, respectively, and a relation $p_1 = -\frac{5}{2} p_0$ is found up to numeric precision.

As $\mathbf{S}^{\text{l} \otimes 3}$ is symmetric, the six proper contractions in Eq. (38) are equivalent if we substitute them to the decision function (as in Eq. (32)). Similarly, the nine self-contractions in this case can also be grouped into three

equivalent classes. Thus, Eq. (38) can effectively be expressed as

$$C_{abc,a'b'c'}^{\text{lll,lll}} = \delta_{aa'}\delta_{bb'}\delta_{cc'} - \frac{1}{5}\delta_{aa'}\delta_{bc}\delta_{b'c'}$$
$$- \frac{1}{5}\delta_{ac}\delta_{bb'}\delta_{a'c'} - \frac{1}{5}\delta_{ab}\delta_{a'b'}\delta_{cc'}, \qquad (39)$$

whose interpretation leads to a tensor

$$T_{abc} = S_a^{\text{l}}S_b^{\text{l}}S_c^{\text{l}} - \frac{1}{5}S_a^{\text{l}}\delta_{bc} - \frac{1}{5}S_b^{\text{l}}\delta_{ca} - \frac{1}{5}S_c^{\text{l}}\delta_{ab}. \qquad (40)$$

This is exactly the one-spin form of the octupolar order given in Ref. 6.

VII. EXPLORING THE PHASE DIAGRAM

In the preceding two sections, we have demonstrated that the bias ρ can in fact be used to ascertain whether or not the labeling of the training samples is consistent with a phase transition. Motivated by this, we embrace this criterion as the central element of a learning scheme that relegates the supervision aspect to its weakest possible sense.

In fact, we will not assume any prior knowledge of the topology of the phase diagram. We impose a uniform grid on the parameter space and sample configurations at each of those grid points, labeling the samples accordingly. This sets up a multiclassification problem where the number of classes equals the number of grid points, which we shall call M. Applying SVM to the latter yields $M(M-1)/2$ classifiers whose biases may be used to consider their respective grid points as belonging to the same phase or not. Specifically, we can build an undirected graph whose vertices correspond to the grid points and are connected by an edge if the bias ρ of the corresponding classifier exceeds a given threshold value ρ_c. Larger values of ρ_c will result in graphs with less false-positive edges at the cost of more false-negative (missing) edges. Since we are considering all pairings of grid points, the graph contains a lot of redundant information, so false-negatives are more easily compensated for. We discuss the choice of ρ_c below.

We revisit the phase diagram for the D_{2h} gauge symmetry of Sec. VI A. The graph resulting from the above procedure on a 10×10 ($M = 100$) grid is shown in Fig. 9. We used a threshold bias of $\rho_c = 2.5$.

Visual inspection of the graph immediately reveals three regions which are densely intra-connected whilst being sparsely inter-connected, corresponding to the three phases of the phase diagram (Fig. 4). Nevertheless, we point out the utility of spectral graph partitioning methods to identify the phases in an objective fashion. The adjacency matrix A (the $M \times M$ matrix having ones on the off-diagonal elements where vertices are connected by an edge) and degree matrix D (the diagonal matrix where the diagonal elements count the number of edges
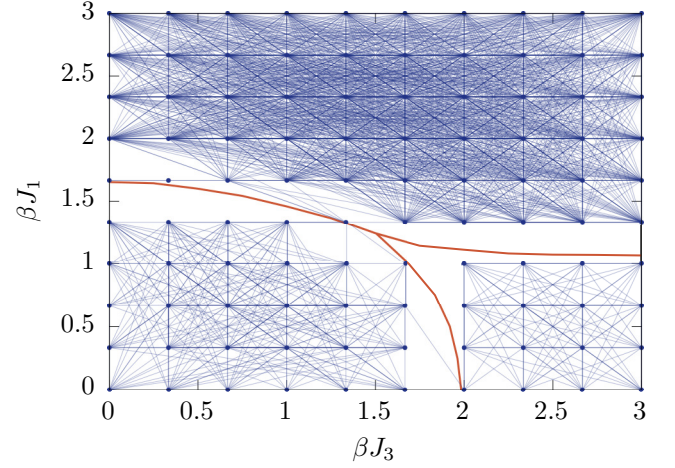

FIG. 9. Graph representation of the relation between points on a 10×10 grid as inferred from the bias ρ of the corresponding SVM classifiers ($\nu = 0.1$). Any two vertices are connected by an edge if $\rho > \rho_c = 2.5$. The phase diagram which has been found based on the peak susceptibilities (cf. Fig. 4) is shown in red for reference. The phase boundaries are intersected by only few graph edges and this happens mostly where grid points are very close to the phase transition.

incident on each vertex) together form the Laplacian matrix $L = D - A$ of the graph. Consequently, the elements of the rows and columns of L sum to zero. When calculating the eigenvalues and eigenvectors of L, the smallest eigenvalue is zero and the corresponding eigenvector is proportional to $(1)^{\otimes M}$ for a connected graph. If the graph consists of multiple disconnected components, the eigenvalue 0 will be degenerate and the eigenvectors reflect the disconnected components.

We use the degeneracy of eigenvalue 0 to guide our choice of the threshold bias ρ_c. We calculate the spectrum of the Laplacian for an assortment of different values for ρ_c and select the largest value for which the eigenvalue 0 remains nondegenerate. Thus, we tweak ρ_c such that the graph is *just* connected with the understanding that the tentative phase boundary will intersect only few edges. Generically, a higher resolution of the grid will result in more redundant information and the graph will remain connected up to larger values of ρ_c. Even larger values of ρ_c would produce disconnected graphs with increasing numbers of components. These could be interpreted as phases directly but the 'desired' number of phases may not be known a priori, so we prefer the following analysis of the (barely) connected graph.

The second smallest eigenvalue of L is referred to as the *algebraic connectivity* and its corresponding eigenvector as the *Fiedler vector*. The elements of the Fiedler vector are clustered around values, positive and negative, and can be used to partition the graph into weakly interconnected components, which in our case correspond to phases.
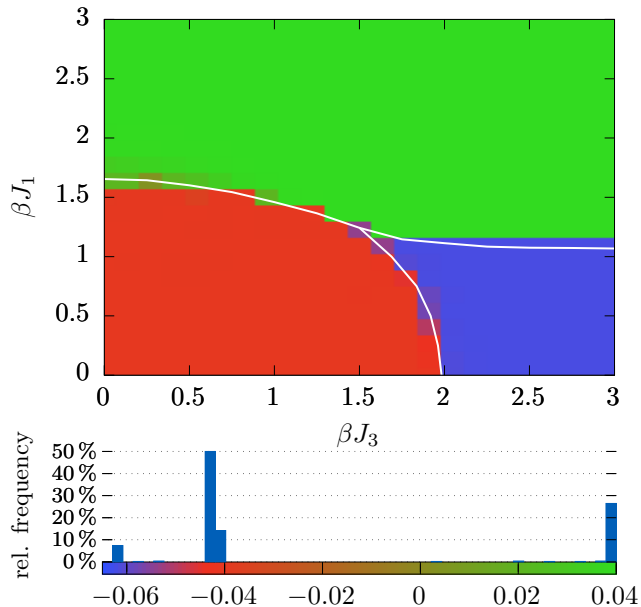
The elements of the Fiedler vector corresponding to

FIG. 10. Upper panel: elements of the Fiedler vector corresponding to the graph obtained for a 23×23 grid at a threshold value of $\rho_c = 12$. Each element is rendered as a pixel at the location of its corresponding grid point in parameter space and color-coded according to its value. The correct phase diagram (cf. Fig. 4) is shown in white for reference. Lower panel: a histogram of the elements' values is shown next to the color scale used to encode them in the upper panel.

mind. Should the computational effort of performing all $M(M-1)/2$ individual binary classification problems turn out to become infeasible, one can limit oneself to binary classification among n-th nearest neighbors, thereby exploiting locality of the grid, and reducing the number of binary classification problems from $\mathcal{O}(M^2)$ to $\mathcal{O}(C_n^2 M)$ where C_n is the number of n-th nearest neighbors, C_1 being the coordination number of the grid. (We suggest to use a grid with a high coordination in that case, e.g. a triangular grid which would result in hexagonal pixels.) Making use of this locality, one will not only gain in computational efficiency, but can also successively apply the multi-classification to parameter regimes of interest, thus effectively explore a large parameter regime of the phase diagram. Further, it is also possible to combine the graphs obtained by training SVMs with different kernels to capture orders of different ranks, such as encountered for the D_{3h} gauge group in Sec. VI B. Only those edges of the final graph persist which are edges of all constituent graphs. Finally, if the histogram of the Fiedler vector elements exhibits an ambiguous clustering, the alternative approach of *hierarchical clustering* may be taken where the graph is partitioned into two components based on the sign of the Fiedler vector elements and this procedure is iteratively repeated for the Laplacian matrices of the component graphs.

VIII. SUMMARY AND OUTLOOK

In this paper, we have explored multiclassifications using SVMs and extended a kernel method we introduced in a previous work for detecting hidden multipolar orders, to handle multiple phases and extract their order parameters. We discussed its application in spin and orbital systems and showed in detail the procedure for extracting the analytical order parameter from machine results. We demonstrated this by examining a generalized quadrupolar order and an octupolar order. The latter is also found in classical Kagomé antiferromagnets. There, it took a significant amount of time for the community to realize its existence[5,6]. However, both this octupolar order and the coexisting quadrupolar order would not elude detection by our method. While we simulate these orders here with an effective gauge theory instead of directly simulating a Kagomé antiferromagnet, this is only done in light of computational efficiency and flexibility. The source of the input data to our method is arbitrary, and the kernel Eq. (13) is suited to detect general classical multipolar orders.

Moreover, we discussed the importance of interpretability in the machine learning of multiple phases. To physically understand a phase diagram with multiple phases, the goal is not only to distinguish each phase by a symbolic label. One would also like to know their characterization and transitions. In addition, in situations where several orders coexist in a phase, which is common in many-body systems, we need to distinguish

the graph obtained by training 139656 classifiers on a 23×23 grid with threshold bias $\rho_c = 12$ is shown in Fig. 10. At that grid resolution, the graph itself becomes very hard to visualize and is therefore omitted. We histogrammized the elements of the Fiedler vector and identified three distinctive peaks. The color scale has been chosen accordingly, such that the primary colors coincide with those peak values (which are themselves arbitrary). When rendering the Fiedler vector elements in the parameter space at the position of their respective grid points and color-coding them according to that color scale, we reproduce the phase diagram perfectly within the resolution dictated by the grid.

To conclude this section, we remark that in the above cases, only about 10^3 samples per grid point were used. This is sufficient to obtain a reasonably indicative value for ρ. Thus, the total number of samples N_s is comparable to what we typically used in binary of multi-classification. The complexity of the SVM optimization, $\mathcal{O}((N_s/M)^{2.2}M^2)$, is close to that of an equivalent binary classification problem $\mathcal{O}(N_s^{2.2})$. One is also free to choose the grid to explore a more extensive parameter regime or to have higher resolution around features of interest such as the tricritical point. Once the topology of the phase diagram has been extracted, the same samples may be relabeled according to their phase to obtain clean order parameters without the need for resampling.

Lastly, several extensions to this scheme come to

and identify them individually, as they represent different physics. This can only be done if the machine is interpretable.

Furthermore, we exploited the internal bias parameter of SVM, and have shown its applications in detecting phase transitions. Although SVM is technically a supervised learning scheme and demands labeled input data, with this bias parameter, the labeling can be arbitrary and does not need to be faithful to the physical reality. SVM can differentiate between the case of a phase transition between two sets of state configurations and the case where they were sampled from the same phase. This can be used as an efficient way to explore unknown phase diagrams. As we have shown in Sec. VII, without using such physical information as the order parameter and heat capacity, simply by monitoring the behavior of this bias parameter, SVM can learn a phase diagram with decent quality. The ambiguity near the phase boundary is acceptable. An approximate phase diagram is sufficient to guide subsequent trainings used for extracting order parameters. A precise phase diagram can then be computed afterwards.

Our previous work Ref. 25 and the current manuscript provide an alternative framework for detecting unconventional orders in frustrated spin and orbital systems. It may be used as an efficient way for analyzing numerical data and extracting important features from systems of interest. The current form of the algorithm in principle can be incorporated into any (semi-)classical method where the spin or orbital degrees of freedom can be represented as $O(3)$ vectors. In future work, we plan to extend it to analyze wave functions and density matrices. However, we note that many frustrated systems already show nontrivial phenomena at the classical level, which also provide useful insight to their quantum counterparts. Examples can be found in Kagomé and triangular antiferromagnets, and in volborthite, pyrochlore and garnet materials. In particular for three-dimensional systems, owing to the lack of efficient quantum algorithms, their study often largely relies on (semi-)classical methods.

[*] ke.liu@lmu.de

[1] P. Anderson, *Basic Notions of Condensed Matter Physics*, Basic Notions of Condensed Matter Physics Series (Benjamin/Cummings, 1984).

[2] N. D. Mermin, Rev. Mod. Phys. **51**, 591 (1979).

[3] C. Lacroix, P. Mendels, and F. Mila, *Introduction to Frustrated Magnetism: Materials, Experiments, Theory*, Springer Series in Solid-State Sciences (Springer Berlin Heidelberg, 2011).

[4] J. T. Chalker, P. C. W. Holdsworth, and E. F. Shender, Phys. Rev. Lett. **68**, 855 (1992).

[5] M. E. Zhitomirsky, Phys. Rev. Lett. **88**, 057204 (2002).

[6] M. E. Zhitomirsky, Phys. Rev. B **78**, 094423 (2008).

[7] J. Yamaura, K. Ohgushi, H. Ohsumi, T. Hasegawa, I. Yamauchi, K. Sugimoto, S. Takeshita, A. Tokuda, M. Takata, M. Udagawa, M. Takigawa, H. Harima, T. Arima, and Z. Hiroi, Phys. Rev. Lett. **108**, 247205 (2012).

[8] H. Takatsu, S. Onoda, S. Kittaka, A. Kasahara, Y. Kono, T. Sakakibara, Y. Kato, B. Fåk, J. Ollivier, J. W. Lynn, T. Taniguchi, M. Wakita, and H. Kadowaki, Phys. Rev. Lett. **116**, 217201 (2016).

[9] M. Taillefumier, O. Benton, H. Yan, L. D. C. Jaubert, and N. Shannon, Phys. Rev. X **7**, 041057 (2017).

[10] H. Yan, O. Benton, L. Jaubert, and N. Shannon, Phys. Rev. B **95**, 094422 (2017).

[11] J. A. M. Paddison, H. Jacobsen, O. A. Petrenko, M. T. Fernández-Díaz, P. P. Deen, and A. L. Goodwin, Science **350**, 179 (2015).

[12] T. Momoi, P. Sindzingre, and N. Shannon, Phys. Rev. Lett. **97**, 257204 (2006).

[13] A. Orlova, E. L. Green, J. M. Law, D. I. Gorbunov, G. Chanda, S. Krämer, M. Horvatić, R. K. Kremer, J. Wosnitza, and G. L. J. A. Rikken, Phys. Rev. Lett. **118**, 247201 (2017).

[14] C. Wu, Phys. Rev. Lett. **100**, 200406 (2008).

[15] G.-W. Chern and C. Wu, Phys. Rev. Lett. **112**, 020601 (2014).

[16] J. Carrasquilla and R. G. Melko, Nature Physics **13**, 431 EP (2017).

[17] E. P. L. van Nieuwenburg, Y.-H. Liu, and S. D. Huber, Nature Physics **13**, 435 EP (2017).

[18] P. Broecker, J. Carrasquilla, R. G. Melko, and S. Trebst, Scientific Reports **7**, 8823 (2017).

[19] Y.-H. Liu and E. P. L. van Nieuwenburg, Phys. Rev. Lett. **120**, 176401 (2018).

[20] M. J. S. Beach, A. Golubeva, and R. G. Melko, Phys. Rev. B **97**, 045207 (2018).

[21] P. Suchsland and S. Wessel, Phys. Rev. B **97**, 174435 (2018).

[22] L. Wang, Phys. Rev. B **94**, 195105 (2016).

[23] C. Wang and H. Zhai, Phys. Rev. B **96**, 144432 (2017).

[24] P. Ponte and R. G. Melko, Phys. Rev. B **96**, 205146 (2017).

[25] J. Greitemann, K. Liu, and L. Pollet, arXiv preprint (2018), arXiv:1804.08557.

[26] V. Vapnik, *Statistical learning theory*, Adaptive and learning systems for signal processing, communications, and control (Wiley, 1998).

[27] C. Cortes and V. Vapnik, Machine Learning **20**, 273 (1995).

[28] C.-C. Chang and C.-J. Lin, Neural computation **13**, 2119 (2001).

[29] C.-C. Chang and C.-J. Lin, ACM Trans. Intell. Syst. Technol. **2**, 27:1 (2011).

[30] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L.

Bartlett, Neural computation **12**, 1207 (2000).

[31] J. C. Platt, in *Advances in Kernel Methods – Support Vector Learning*, edited by B. Schölkopf, C. J. C. Burges, and A. J. Smola (MIT Press, 1998).

[32] R.-E. Fan, P.-H. Chen, and C.-J. Lin, Journal of Machine Learning Research **6**, 1889 (2005).

[33] C.-W. Hsu and C.-J. Lin, IEEE transactions on Neural Networks **13**, 415 (2002).

[34] L. Michel and B. Zhilinskiı, Physics Reports **341**, 11 (2001).

[35] J. Nissinen, K. Liu, R.-J. Slager, K. Wu, and J. Zaanen, Phys. Rev. E **94**, 022701 (2016).

[36] K. Liu, J. Nissinen, R.-J. Slager, K. Wu, and J. Zaanen, Phys. Rev. X **6**, 041025 (2016).

[37] S. Elitzur, Phys. Rev. D **12**, 3978 (1975).

[38] P. A. Lebwohl and G. Lasher, Physical Review A **6**, 426 (1972).

[39] P. E. Lammert, D. S. Rokhsar, and J. Toner, Phys. Rev. Lett. **70**, 1650 (1993).

[40] P. G. de Gennes and J. Prost, *The Physics of Liquid Crystals*, International Series of Monogr (Clarendon Press, 1995).

[41] J. P. Straley, Phys. Rev. A **10**, 1881 (1974).

[42] K. Liu, J. Nissinen, J. de Boer, R.-J. Slager, and J. Zaanen, Phys. Rev. E **95**, 022704 (2017).

[43] A. Gaenko, A. Antipov, G. Carcassi, T. Chen, X. Chen, Q. Dong, L. Gamper, J. Gukelberger, R. Igarashi, S. Iskakov, M. Knz, J. LeBlanc, R. Levy, P. Ma, J. Paki, H. Shinaoka, S. Todo, M. Troyer, and E. Gull, Computer Physics Communications **213**, 235 (2017).