Compositional optimization of quantum circuits for quantum kernels of support vector machines

Elham Torabian ⬤ and Roman V. Krems

Department of Chemistry, University of British Columbia, Vancouver, British Columbia, Canada V6T 1Z1
and Stewart Blusson Quantum Matter Institute, Vancouver, British Columbia, Canada V6T 1Z4

While quantum machine learning (ML) has been proposed to be one of the most promising applications of quantum computing, how to build quantum ML models that outperform classical ML remains a major open question. Here, we demonstrate a Bayesian algorithm for constructing quantum kernels for support vector machines that adapts quantum gate sequences to data. The algorithm increases the complexity of quantum circuits incrementally by appending quantum gates selected with Bayesian information criterion as circuit selection metric and Bayesian optimization of the parameters of the locally optimal quantum circuits identified. The performance of the resulting quantum models for the classification problems considered here significantly exceeds that of optimized classical models with conventional kernels.

I. INTRODUCTION

Quantum machine learning (QML) has recently emerged as a new research field aiming to take advantage of quantum computing for machine learning (ML) tasks. It has been shown that embedding data into gate-based quantum circuits can be used to produce kernels for ML models by quantum measurements [1–15]. A major question that remains open is whether quantum kernels thus produced can outperform classical kernels for practical ML applications. The advantage of quantum kernels may manifest itself in reducing the computational hardness of big data problems. For example, it has been shown that a classification problem based on the discrete logarithm problem, which is believed to be in the NP and bounded-error quantum polynomial (BQP) complexity classes, can be efficiently solved with quantum kernels [16]. It has also been shown that ML models with quantum kernels can efficiently solve a classification problem with the complexity of the k-Forrelation problem [17], which is proven to be promiseBQP-complete [18,19]. This proves that all classification problems with BQP (and promiseBQP) complexity can be solved efficiently with QML. However, it is not yet clear if classical big data problems of practical importance can benefit from QML.

The opposite limit is relevant for applications where observations are exceedingly expensive. For such applications, learning is based on small data. If quantum kernels can produce models with better inference accuracy for small data problems than classical kernels, QML may offer a practical application of quantum computing for classification, interpolation, and optimization of functions that are exceedingly expensive to evaluate. However, the generalization

performance of quantum kernels is sensitive to the quantum circuit ansatz [1,20–22]. The choice of the ansatz is an open problem in QML. For variational quantum optimization and quantum simulation applications, a common strategy is to choose a fixed ansatz and optimize the parameters of the quantum circuit [23–35]. However, for most optimal ML models, it is necessary to develop algorithms that adapt the ansatz to data. Previous work has addressed this problem by optimizing gate sequences using quantum combs [36] for cloning, storage retrieving, discrimination, estimation, and tomographic characterization of quantum circuits, graph theory [37] to develop quantum Fourier transforms and reversible benchmarks, quantum triple annealing minimization [38] for quantum annealing applications, reinforcement learning [39] to solve the max-cut problem, a variable structure approach [40] for the variational quantum eigensolver for condensed matter and quantum chemistry applications, an adaptive variational algorithm [41] for molecular simulations, quantum local search [42] for finding the maximum independent set of a graph, layerwise learning [43] for quantum neural networks, and multi-objective genetic algorithms [44] for quantum classification models.

The most general approach to identifying the best quantum circuit (QC) for a given ML task involves (i) search in the space of all possible permutations of quantum gates; and (ii) optimization of quantum gate parameters for each QC during the search. Both (i) and (ii) become quickly intractable as the number of gates increases. Both (i) and (ii) also require a circuit selection metric to guide the search and optimization. Optimization of the architecture and parameters of quantum kernels is further complicated by the complexity of quantum measurements. Estimating the kernel matrix of a ML model by measurements on a quantum computer scales with the number of training points N as $\mathcal{O}(N^3)$ [34]. Therefore, it is important to develop algorithms for optimization of quantum circuits with small N. Once found, and if robust, the quantum kernels can then be used for nonparametric models with large N, in order to make accurate predictions. Within the Bayesian

approach, ML models can be discriminated by marginal likelihood (see Sec. III below). However, computation of marginal likelihood also becomes quickly intractable as the model complexity increases. Marginal likelihood can be approximated by the Bayesian information criterion (BIC) [45]. However, this approximation is valid in the limit of large N. It is not known if BIC can be used as a QC selection metric for small data problems.

In the present paper, we demonstrate an algorithm that builds performant quantum circuits for small data classification problems using BIC as a circuit selection metric and a compositional search that involves two stages: (i) incremental construction of the quantum circuit by appending individual quantum gates to K optimal QCs of lower complexity; and (ii) Bayesian optimization of the parameters of M locally optimal quantum circuits. This algorithm reduces to greedy search in the limit of $K = 1$ and full search of the QC space in the large K and M limit. The choice of K thus balances the efficiency of greedy search with the volume of explored circuit space. By limiting optimization of QC parameters to M locally optimal circuits, this algorithm reduces the computational complexity of simultaneous optimization of QC architecture and parameters.

We demonstrate the approach by building an optimal ansatz for quantum support vector machines (SVM). SVM is a kernel method that can be used for both classification and regression problems [45]. SVM kernels allow classification of data that are not linearly separable. SVM is well suited for small data problems. The quantum analog of SVM uses quantum kernels. We show that the performance of quantum SVM can be systematically enhanced by building quantum kernels in a data-adaptable approach. In order to achieve this, we convert the output of the SVM classifier to probabilistic predictions and compute the Bayesian information criterion [46,47]. Using two unrelated classification problems, we show that quantum kernels thus obtained outperform standard classical kernels and achieve enhanced inference accuracy for small data problems. The method proposed here is inspired by the work of Duvenaud *et al.* [48], that demonstrated the possibility of enhancing the performance of classical Gaussian process models through compositional kernel search guided by BIC.

II. QUANTUM CIRCUIT DESCRIPTORS

Supervised learning for classification uses N input-output pairs $\{\boldsymbol{x}, y\}$, with inputs $\boldsymbol{x}$ represented by multi-dimensional vectors and outputs y encoding the class labels. We consider binary classification problems with $y = \{0, 1\}$. For a QML algorithm, the input vectors $\boldsymbol{x}$ must be encoded into quantum states. Here, we use the following encoding of n-dimensional vectors $\boldsymbol{x} \in \mathbb{R}^n$ into states of n qubits, each initially in state $|0\rangle$,

$$|\Phi(\boldsymbol{x})\rangle = \exp\left(i \sum_k x_k Z_k\right) H^{\otimes n} |0\rangle^{\otimes n} \qquad (1)$$

where x_k is the kth component of vector $\boldsymbol{x}$, Z_k is the Pauli Z gate acting on kth qubit, and H represents the Hadamard gate putting qubits into coherent superpositions. The state $|\Phi(\boldsymbol{x})\rangle$ is
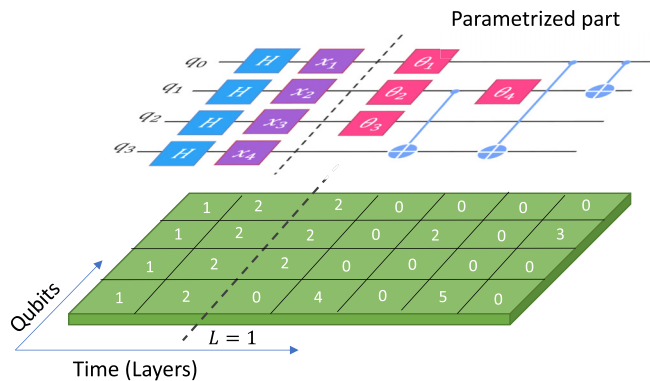

FIG. 1. Schematic diagram of a quantum circuit (upper) used for quantum kernels of SVM and a matrix descriptor (lower) of the gate sequence. Each gate is assigned a numerical descriptor, encoding the type of gate and the index of qubits entangled for CNOT gates shown by blue crosses in the upper diagram.

then operated on by a sequence of one- and two-qubit gates to produce a general quantum state $\mathcal{U}(\boldsymbol{x}) |\Phi(\boldsymbol{x})\rangle$. The measurable quantities

$$k(\boldsymbol{x}, \boldsymbol{x}') = |\langle\Phi(\boldsymbol{x}')|\mathcal{U}^\dagger(\boldsymbol{x}')\mathcal{U}(\boldsymbol{x})|\Phi(\boldsymbol{x})\rangle|^2 \qquad (2)$$

have the properties of the kernel of a reproducing kernel Hilbert space and can thus be used as kernels of SVM.

Here, we propose and demonstrate an algorithm that systematically increases the complexity of $\mathcal{U}$ to enhance the performance of quantum kernels (2). As depicted in Fig. 1, we choose $\mathcal{U}$ to consist of L layers, each consisting of one-qubit R_Z gates and/or two-qubit CNOT gates. Each of the R_Z gates for qubit k is parametrized by θ as follows:

$$R_{Z,k}(\theta) = \begin{pmatrix} e^{-i\theta x_k} & 0 \\ 0 & e^{i\theta x_k} \end{pmatrix}. \qquad (3)$$

This yields a quantum circuit with the number of free parameters θ equal to the number of R_Z gates.

Although not essential, it is convenient for the present paper to represent QCs by numerical descriptors. The numerical representation of QCs can also be used for the analysis of QCs, such as their classification into performant and nonperformant kernels. We adopt the following approach to represent QC by numerical vectors. As illustrated in Fig. 1 (lower part), each QC is described by a numerical matrix that encodes: "no gate", Hadamard, CNOT, or R_Z. Each row and column correspond to qubits and layers, respectively. We assign $\{1, 2, 3, 4, 5\}$ to $\{$Hadamard, R_Z, CNOT$(q, q-1)$, CNOT$(q, q-2)$ and CNOT$(q, q-3)\}$, respectively. The arguments of CNOT(a, b) indicate the target (a) and control (b) qubits. The numerical descriptors corresponding to n qubits and g types of gates thus generated form a total of $[(g + n - 1)!/n!(g - 1)!] \times K \times L$ quantum circuits.

III. MARGINAL LIKELIHOOD FOR QC SELECTION

Within the Bayesian approach, we can assign probability $P(\mathcal{Q}_i|\text{Data})$ to a quantum model $\mathcal{Q}_i$, which can be expressed

using Bayes' theorem as follows:

$$P(\mathcal{Q}_i|\text{Data}) = \frac{P(\text{Data}|\mathcal{Q}_i)P(\mathcal{Q}_i)}{P(\text{Data})}, \qquad (4)$$

where Data $= \{X, y\}$, X is the data matrix and y is a vector of outputs in the training data set. For two QCs $\mathcal{Q}_1$ and $\mathcal{Q}_2$, we can write

$$\frac{P(\mathcal{Q}_1|\text{Data})}{P(\mathcal{Q}_2|\text{Data})} = \frac{P(\text{Data}|\mathcal{Q}_1)}{P(\text{Data}|\mathcal{Q}_2)} \times \frac{P(\mathcal{Q}_1)}{P(\mathcal{Q}_2)}. \qquad (5)$$

Assuming no prior knowledge of circuit performance, we choose all $P(\mathcal{Q}_i)$ to be the same, which shows that QCs can be discriminated by $P(\text{Data}|\mathcal{Q}_i)$. If a QC $\mathcal{Q}_i$ is parametrized by a set of parameters denoted collectively by $\boldsymbol{\theta}$, we can define the likelihood function for the quantum model as

$$\mathcal{L}(\boldsymbol{\theta}|\text{Data}, \mathcal{Q}_i) = p(\text{Data}|\boldsymbol{\theta}, \mathcal{Q}_i), \qquad (6)$$

where $p(\text{Data}|\boldsymbol{\theta}, \mathcal{Q}_i)$ is the probability that the quantum model $\mathcal{Q}_i$ with parameters $\boldsymbol{\theta}$ yields the observations in the training set $\{X, y\}$. This shows that marginal likelihood

$$P(\text{Data}|\mathcal{Q}_i) = \int_{\boldsymbol{\theta}} p(\text{Data}|\boldsymbol{\theta}, \mathcal{Q}_i)p(\boldsymbol{\theta}|\mathcal{Q}_i)d\boldsymbol{\theta} \qquad (7)$$

can be used to discriminate between quantum models with different circuit architectures. As the number of circuit parameters grows, the integrals in Eq. (7) become intractable. However, for a large number of training points N, they can be approximated as [45]

$$\log\left(P(\text{Data}|\mathcal{Q}_i)\right) \approx \log\hat{\mathcal{L}}(\mathcal{Q}_i) - \frac{d}{2}\log(N), \qquad (8)$$

where $\hat{\mathcal{L}}(\mathcal{Q}_i) = \mathcal{L}(\hat{\boldsymbol{\theta}}|\text{Data}, \mathcal{Q}_i)$ is the maximum value of likelihood and d is the number of parameters in $\boldsymbol{\theta}$. The Bayesian information criterion (BIC) is defined as

$$\text{BIC} = -2\log\hat{\mathcal{L}} + d\log N, \qquad (9)$$

which shows that it can be used to estimate the logarithm of marginal likelihood, and hence serve as a model selection metric. In the present paper, we only parametrize the R_Z gates, so d is equal to the total number of R_Z gates in the QC. While Eq. (8) is valid for $N \to \infty$, in the present paper we provide empirical evidence that BIC is a valid selection metric for quantum models for classification problems with $N \approx 100$.

In order to compute BIC, we convert the output $\hat{f}(\boldsymbol{x}_i)$ of SVM with quantum kernels to a posterior class probability $p_i(y = 1|\boldsymbol{x}_i)$, approximated by a sigmoid function

$$p_i = \frac{1}{1 + \exp\left(a\hat{f}(\boldsymbol{x}_i) + b\right)}. \qquad (10)$$

The parameters a and b are determined by maximizing the log likelihood function $\log\hat{\mathcal{L}}$ [49]. For the binary classification problem with $y = \{0, 1\}$, we compute $\log\hat{\mathcal{L}}$ as

$$\log\hat{\mathcal{L}} = -\sum_{i=1}[y_i\log(p_i) + (1 - y_i)\log(1 - p_i)], \qquad (11)$$

where p_i is the probability of observing class with label $y_i=1$. In the current implementation, SVM is trained by maximizing Lagrange dual with a training set [45], while $\log\hat{\mathcal{L}}$ used in Eq. (9) is computed over a separate validation set. The coefficients a and b are found by fourfold cross-validation with the training set. These coefficients are then used to calculate the probabilities p_i for the validation set, which yields $\log\hat{\mathcal{L}}$ for the computation of BIC.

IV. CIRCUIT GROWTH ALGORITHM

The purpose of the present algorithm for building QCs for quantum kernels is to approximate the search of an optimal QC architecture and an optimal set of parameters in the entire space of parametrized circuits. The pseudo-code for this algorithm is presented in the Supplemental Material (SM) [50]. As seen in the pseudocode, the algorithm is controlled by two hyperparameters: K and M. The limit of $M = 0$ and $K \to \infty$ corresponds to the purely compositional search. When $M = K$, the algorithm optimizes the architecture and parameters of QCs simultaneously.

The architecture of the circuit is built incrementally, starting with all possible circuits with only one layer $L = 1$ of the parametrized part of QC in Fig. 1. Using BIC, we identify K preferred quantum kernels with $L = 1$ (i.e., K kernels yielding quantum models with the lowest BIC values). The parameters of $M \leqslant K$ of those QCs are optimized using Bayesian optimization (BO). Details of BO, including the choice of kernels for Gaussian process regression and parameters of the acquisition function, are given in the SM [50]. A new layer is then added by appending all possible combinations of gates for layer two to K selected circuits with $L = 1$. The algorithm then selects K preferred kernels with $L = 2$, and the process is iterated. This search strategy directs the algorithm to increase the complexity of quantum circuits while improving the performance of quantum kernels.

V. RESULTS

To illustrate the performance of the algorithm proposed here, we use two independent four (4D) and three (3D) dimensional classification problems. For the first example, we use the data from Ref. [51] to classify halide perovskites, with the chemical formula $A_2BB'X_6$, into metals or nonmetals. The four components of $\boldsymbol{x}$ correspond to the ionic radii of the four atoms A, B, B', and X. The second example is a 3D synthetic data set implemented in the Qiskit ML module [52]. Hereafter, we will refer to the two data sets as the perovskite dataset and the *ad hoc* dataset, respectively. We split each data set into 100 random training samples, a validation set with 100 random samples and a test set with 1442 and 4100 samples for the perovskite and *ad hoc* datasets, respectively. The training set is used to train the SVM model with a given quantum kernel, and the validation set is used to select quantum circuits, while the test set is used to illustrate the final model performance.

A. Quantum circuit search convergence

Figure 2 demonstrates the convergence of the classification accuracy with K and M. The convergence is examined using the lower value of the two class prediction accuracies over a test set for the perovskite classification problem. The results for $M \leqslant 20$ are obtained with $K = 20$. For $M \geqslant 20$, $K = M$. The results indicate that convergence is achieved for $K = 20$
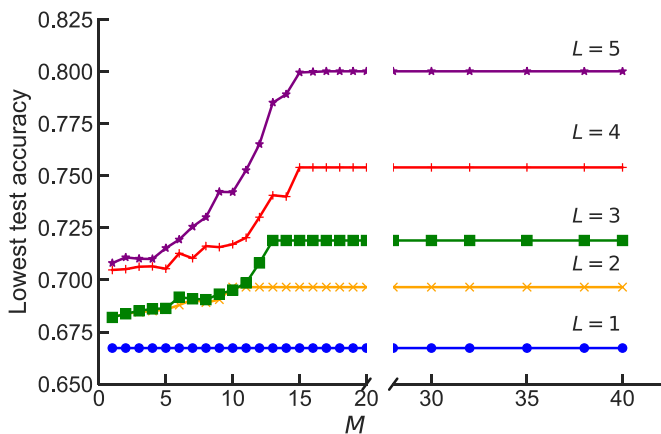
FIG. 2. Convergence of Bayesian search of the quantum circuit space quantified by the lowest test accuracy (lower of TPR and TNR as defined in the text) with respect to the algorithm hyperparameters K and M. The results for $M \leqslant 20$ are obtained with $K = 20$. For $M \geqslant 20$, $K = M$.

and $M = 15$. Figure 2 shows that circuits with more layers benefit from a larger value of M. Unless stated otherwise, we use $K = M = 20$ for benchmark calculations in this paper.

The results of Fig. 2 suggest interesting extensions of the present algorithm. For example, the computation complexity can be reduced by introducing the dependence of both M and K on the number of layers L, i.e., $M \rightarrow M(L)$ and $K \rightarrow K(L)$. For example, the results of Fig. 2 suggest that convergence can be reached with $M = 10$ for $L = 1$ and $L = 2$, $M = 13$ for $L = 3$, and $M = 15$ for $L > 3$, which requires a much smaller number of circuits to be optimized than in the present paper. To understand the relative roles of parameter vs architecture optimization, we will also compare below the performance of the fully optimized search ($K = M = 20$) with the search, where the architecture of QC is optimized first and the parameters of a single best QC are optimized at the final stage of the kernel construction. Models obtained using this algorithm are hereafter referred to as $M = 0(1)$. Finally, we note that the results of Fig. 2 indicate that the algorithm monotonically converges to an optimal circuit with significantly better performance than a randomly chosen quantum kernel. The monotonic convergence indicates that an optimal search algorithm can be determined by increasing K and $M \leqslant K$.

B. Quantum kernel selection metric

The performance of classification models such as SVM is generally quantified by test accuracy over a holdout set or equivalent metrics, such as the F score, often $F1$ [53]. It is convenient to label the two classes as positive and negative and define the true positive rate (TPR) and the true negative rate (TNR) as the ratio of correct predictions to the total number of test points for a given class. Unless otherwise specified, the test/validation accuracy of the model is defined here as the balanced average of TPR and TNR computed as (TPR + TNR)/2 over the test/validation set. The test accuracy for a specific class refers to TPR or TNR over the test set. The lowest test accuracy refers to the lower of TPR and TNR over the test set. Since the model performance is limited by the

class with the lower prediction accuracy, most of the results show the lowest test accuracy. More details about different model selection metrics used here are provided in SM [50].

To illustrate the importance of BIC for the present algorithm, it is instructive to compare the performance of BIC as a QC selection metric with other possible metrics. Figure 3 uses the perovskite dataset to compare the performance of the quantum models with quantum circuits selected using three different metrics: accuracy, $F1$ score and BIC, all computed over the validation sets. The results demonstrate that the classification accuracy or $F1$ score cannot be used as metrics for quantum circuit selection. On the other hand, BIC leads to both better models for each layer L and significant improvement of the quantum classification models as the kernel complexity increases. We observe a similar trend for the other data set (provided in SM [50]).

BIC is known to be a rigorous model selection metric asymptotically, in the limit of a large number of training points [45]. For a family of K models, the probability of model m
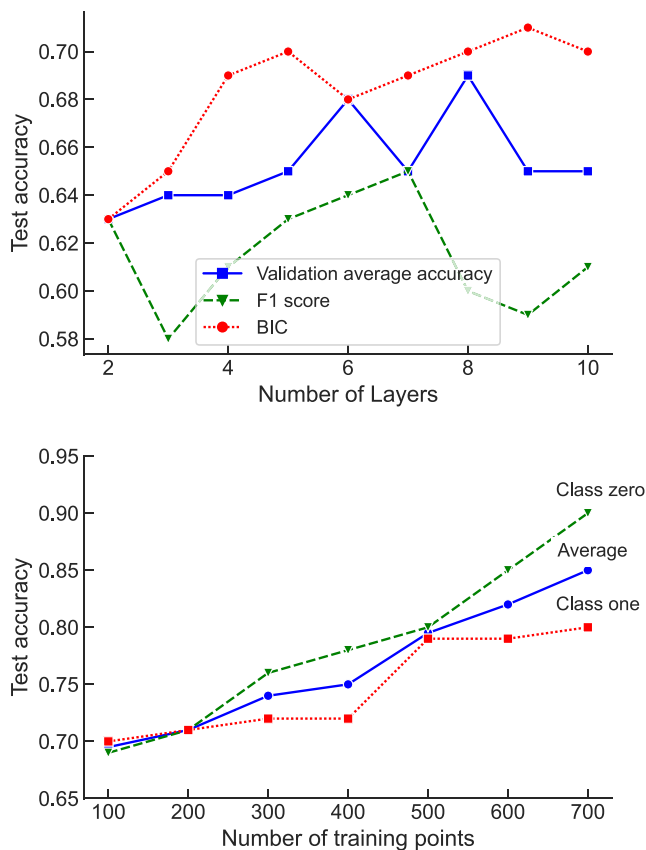

FIG. 3. Upper: Quantum classification model accuracy averaged over two classes over the holdout test set as a function of the number of layers in $\mathcal{U}$ constructed using three model selection metrics: BIC (circles), validation accuracy (squares), $F1$ score (triangles). The models are trained with 100 training points. Lower: Improvement of the quantum classification accuracy with the number of training points by applying the best obtained quantum kernel using 100 training points (fixed kernel). Class-specific test accuracy represents TPR and TNR, as defined in the text.

being selected can be estimated as

$$P(\text{Model}_m) = \frac{e^{-\text{BIC}_m/2}}{\sum_i e^{-\text{BIC}_i/2}}. \quad (12)$$

If the family of models includes the true model, the probability of selecting the correct model approaches one as $N \to \infty$ [45]. In this limit, the integral yielding marginal likelihood can be replaced with the maximized likelihood as shown in Eq. (9). However, Eq. (9) may not be valid in the small data limit. Our results in Fig. 3 show that BIC remains an effective selection metric for quantum models trained with as few as 100 training points.

To further demonstrate the expressiveness of the quantum kernels selected by BIC, we show in the lower panel of Fig. 3 the improvement of the quantum classification accuracy with the number of training points for the perovskite data set. The models illustrated in Fig. 3 are obtained with 100 training points using $K = 20$ and $M = 0$, and are subsequently used in SVM with a larger number of training points. The results illustrate two important points: (i) quantum kernels obtained by optimization of QC architecture with BIC based on a small number of training points produce models that generalize well for problems with larger training sets, as expected of any performant kernel—this is a sanity test for the resulting quantum models; (ii) performant quantum kernels can be obtained by optimization of QC architecture without optimization of QC parameters (i.e., with $M = 0$).

C. Classical vs quantum classification models

Figure 4 compares the results of the predictions of the best quantum models with the classical models based on commonly used optimized kernels. Specifically, we train four classical models with the radial basis function (RBF), linear, cubic polynomial, and sigmoid kernel functions. The models are trained with the same number and distribution of training points as for the quantum models and the parameters of each kernel function are optimized to minimize the error over identical validation sets for all models.

There are three sets of quantum models shown:

(i)   First, we construct the best quantum kernel for each number of layers L, without any optimization of the circuit parameters. These models correspond to Bayesian search hyperparameters $K = 20$ and $M = 0$ and illustrate the improvement of quantum kernels that can be achieved by optimizing only the circuit architecture. The prediction accuracy of SVM models with quantum kernels thus obtained is shown by circles. The circles in Fig. 4 thus represent the results of purely compositional optimization.

(ii) Second, we vary the parameters θ of all R_Z gates in the best quantum kernel for a given L to minimize error on the validation set. These models correspond to Bayesian search hyperparameters $K = 20$ and $M = 0$, but include parameter optimization for one best circuit at each layer. Hence, we label these results, shown by triangles in Fig. 4, as $M = 0(1)$. The purpose of this calculation is to demonstrate the improvement due to optimization of the parameters of the locally optimal circuits. The triangles in Fig. 4
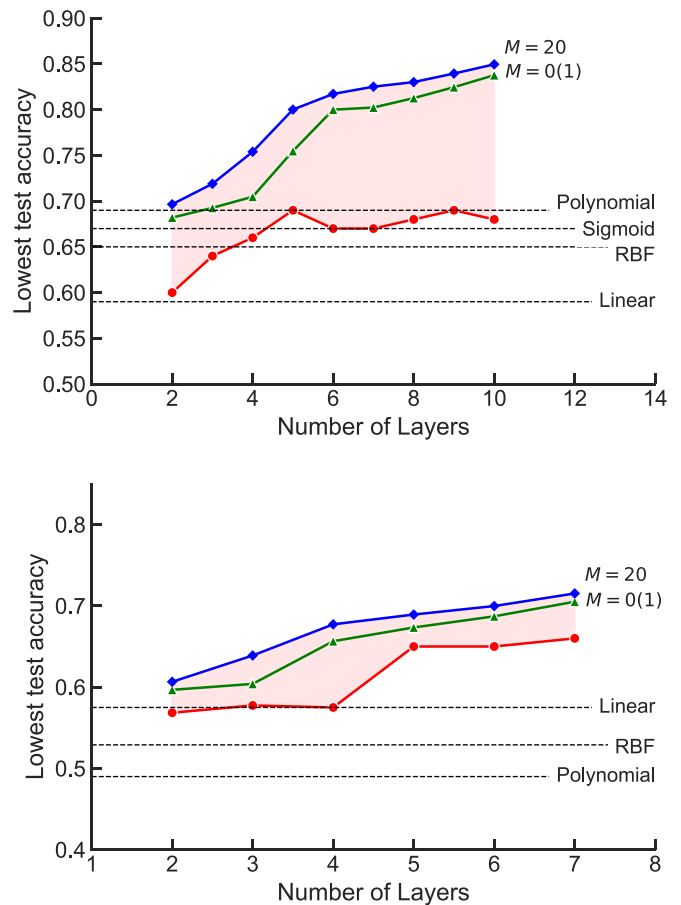

FIG. 4. Lowest prediction accuracy (lower of TPR and TNR as defined in the text) evaluated on the holdout test set for the perovskite (upper) and *ad hoc* (lower) classification problems. The horizontal broken lines show the best accuracy for SVM models with the classical kernels indicated as labels. The classical kernels are optimized using the validation set. The symbols show the results of the best quantum model for each layer L: circles, quantum kernels constructed using Bayesian search of QC architecture with fixed parameters of all quantum gates; triangles, the $K = 20, M = 0(1)$ search, where architecture is optimized without any optimization of circuit parameters, but the parameters of the best QC are optimized to enhance performance; diamonds, quantum circuits from the $K = M = 20$ Bayesian search involving simultaneous optimization of both the QC architecture and QC parameters.

thus represent the results of purely compositional optimization followed by optimization of parameters of the best QC of a given complexity.

(iii) Finally, the diamonds in Fig. 4 show the results of the fully optimized search with $K = M = 20$. The diamonds in Fig. 4 thus represent the results of simultaneous optimization of QC architecture and parameters.

For a binary classification problem, one can evaluate the model test accuracy for each class separately. The performance of classification models is generally limited by the lower of the two prediction accuracies. Figure 4 shows the lowest test accuracy of the best models trained with $N = 100$. It can be seen that the performance of quantum models

significantly exceeds the prediction accuracy achievable with classical kernels considered. SM presents the detailed comparisons of classical and quantum model predictions for each class separately.

It is interesting to observe that purely compositional optimization followed by optimization of parameters of the best QC (triangles in Fig. 4) produces quantum kernels yielding test errors comparable with the test errors of the full simultaneous optimization of QC architecture and parameters (diamonds in Fig. 4). Furthermore, the difference between the diamonds and the triangles in Fig. 4 decreases with increasing QC complexity. This trend is observed for both classification problems considered here. This suggests that the most optimal algorithm for constructing complex quantum kernels should separate the compositional optimization and the optimization of QC parameters.

VI. CONCLUSIONS

We have demonstrated an algorithm for constructing quantum kernels for quantum SVM that adapts gate sequences and gate parameters to data for classification problems. The quantum kernels for SVM are built through compositional search in the space of gate permutations and Bayesian optimization of quantum circuits with locally optimal architecture. The search algorithm is controlled by two hyperparameters (K and M) that balance the greediness, and hence the efficiency, of compositional search with the volume of quantum circuit space explored. The algorithm is designed to approach the complete search of the entire space of optimized quantum circuits in the asymptotic limit of large K and M. We have shown that quantum kernels with enhanced generalization performance can be obtained through search restricted to a small fraction ($<0.5\%$ for the present examples) of the total space.

We have illustrated the performance of the resulting quantum kernels by comparisons with classical models based on

several conventionally used kernels for two nonlinear classification problems. Our results show that optimized quantum kernels yield models with significantly better classification accuracy than the classical models considered here. It is important to note that we do not claim quantum advantage for the quantum kernels obtained in this work. Although the parameters of the classical kernels were optimized, we did not optimize the functional form of the classical models. More importantly, we consider kernels based on a small number of qubits. Such kernels are classically simulable. However, the present paper offers approach to identify QML models with enhanced performance, potentially making QML useful for practical applications, including for interpolation, classification, and optimization of objective functions that are exceedingly difficult to evaluate. We also conjecture that if quantum kernels can be constructed to be expressive for low-dimensional problems, they can also be constructed to be expressive for high-dimensional problems, for which quantum kernels are not classically simulable with existing classical hardware resources.

Critical to any algorithm of automated quantum ansatz construction is the quantum circuit selection metric. Our results show that performant quantum kernels for classification problems can be selected by Bayesian information criterion, while the validation error and $F1$ score fail as quantum model selection metrics for the problems considered here. The choice of the ansatz remains an open problem in QML. The present paper offers an algorithm to identify subdomains of quantum gate permutations that yield good kernels and bad kernels. This can potentially be used to identify the essential elements of quantum circuits that are responsible for the expressiveness of the ensuing kernels.

[1] M. Schuld, Supervised quantum machine learning models are kernel methods, arXiv:2101.11020.

[2] M. Schuld and N. Killoran, Quantum Machine Learning in Feature Hilbert Spaces, Phys. Rev. Lett. **122**, 040504 (2019).

[3] P. Rebentrost, M. Mohseni, and S. Lloyd, Quantum Support Vector Machine for Big Data Classification, Phys. Rev. Lett. **113**, 130503 (2014).

[4] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, Nature (London) **567**, 209 (2019).

[5] R. Mengoni and A. Di Pierro, Kernel methods in quantum machine learning, Quantum Mach. Intell. **1**, 65 (2019).

[6] K. Bartkiewicz, C. Gneiting, A. Černoch, K. Jiráková, K. Lemr, and F. Nori, Experimental kernel-based quantum machine learning in finite feature space, Sci. Rep. **10**, 12356 (2020).

[7] D. K. Park, C. Blank, and F. Petruccione, The theory of the quantum kernel-based binary classifier, Phys. Lett. A **384**, 126422 (2020).

[8] Y. Suzuki, H. Yano, Q. Gao, S. Uno, T. Tanaka, M. Akiyama, and N. Yamamoto, Analysis and synthesis of feature map for kernel-based quantum classifier, Quantum Mach. Intell. **2**, 9 (2020).

[9] J. E. Park, B. Quanz, S. Wood, H. Higgins, and R. Harishankar, Practical application improvement to Quantum SVM: Theory to practice, arXiv:2012.07725.

[10] R. Chatterjee and T. Yu, Generalized coherent states, reproducing kernels, and quantum support vector machines, Quantum Inf. Commun. **17**, 1292 (2017).

[11] J. R. Glick, T. P. Gujarati, A. D. Corcoles, Y. Kim, A. Kandala, J. M. Gambetta, and K. Temme, Covariant quantum kernels for data with group structure, arXiv:2105.03406.

[12] C. Blank, D. K. Park, J. K. K. Rhee, and F. Petruccione, Quantum classifier with tailored quantum kernel, npj Quantum Inf. **6**, 41 (2020).

[13] S. L. Wu, S. Sun, W. Guan, C. Zhou, J. Chan, C. L. Cheng, T. Pham, Y. Qian, A. Z. Wang, R. Zhang, and M. Livny, Application of quantum machine learning using the quantum kernel algorithm on high energy physics analysis at the LHC, Phys. Rev. Res. **3**, 033221 (2021).

[14] T. Haug, C. N. Self, and M. S. Kim, Quantum machine learning of large datasets using randomized measurements, Mach. Learn.: Sci. Technol. **4**, 015005 (2023).

[15] M. Otten, I. R. Goumiri, B. W. Priest, G. F. Chapline, and M. D. Schneider, Quantum machine learning using Gaussian processes with performant quantum kernels, arXiv:2004.11280.

[16] Y. Liu, S. Arunachalam, and K. Temme, A rigorous and robust quantum speed-up in supervised machine learning, Nat. Phys. **17**, 1013 (2021).

[17] J. Jäger and R. V. Krems, Universal expressiveness of variational quantum classifiers and quantum kernels for support vector machines, Nat Commun **14**, 576 (2023).

[18] S. Aaronson, BQP and the polynomial hierarchy, Proc. Annu. ACM Symp. Theory Comput., 141 (2010).

[19] S. Aaronson and A. Ambainis, Forrelation: A problem that optimally separates quantum from classical computing, Proc. Annu. ACM Symp. Theory Comput., 307 (2015).

[20] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms, Nat. Rev. Phys. **3**, 625 (2021).

[21] K. Nakaji, H. Tezuka, and N. Yamamoto, Quantum-enhanced neural networks in the neural tangent kernel framework, arXiv:2109.03786.

[22] E. Peters, J. Caldeira, A. Ho, S. Leichenauer, M. Mohseni, H. Neven, P. Spentzouris, D. Strain, and G. N. Perdue, Machine learning of high dimensional data on a noisy quantum processor, npj Quantum Inf. **7**, 161 (2021).

[23] M. Schuld, R. Sweke, and J. J. Meyer, Effect of data encoding on the expressive power of variational quantum-machine-learning models, Phys. Rev. A **103**, 032430 (2021).

[24] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, Quantum circuit learning, Phys. Rev. A **98**, 032309 (2018).

[25] M. Watabe, K. Shiba, M. Sogabe, K. Sakamoto, and T. Sogabe, Quantum circuit parameters learning with gradient descent using backpropagation, arXiv:1910.14266.

[26] J. Qi, C. H. H. Yang, and P. Y. Chen, QTN-VQC: An end-to-end learning framework for quantum neural networks, arXiv:2110.03861.

[27] K. Terashi, M. Kaneda, T. Kishimoto, M. Saito, R. Sawada, and J. Tanaka, Event classification with quantum machine learning in high-energy physics, Comput. Softw. Big Sci. **5**, 2 (2021).

[28] S. Y. C. Chen, C. M. Huang, C. W. Hsing, and Y. J. Kao, Hybrid quantum-classical classifier based on tensor network and variational quantum circuit, arXiv:2011.14651.

[29] A. Blance and M. Spannowsky, Quantum machine learning for particle physics using a variational quantum classifier, J. High Energy Phys. **2021**, 212 (2021).

[30] Y. Kwak, W. J. Yun, S. Jung, J. K. Kim, and J. Kim, Introduction to quantum reinforcement learning: Theory and pennylane-based implementation, Int. Conf. ICT Converg., 416 (2021).

[31] D. Sierra-Sosa, J. Arcila-Moreno, B. Garcia-Zapirain, C. Castillo-Olea, and A. Elmaghraby, Dementia prediction applying variational quantum classifier, arXiv:2007.08653.

[32] S. Y. C. Chen, C. H. H. Yang, J. Qi, P. Y. Chen, X. Ma, and H. S. Goan, Variational quantum circuits for deep reinforcement learning, IEEE Access **8**, 141007 (2020).

[33] S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, and N. Killoran, Quantum embeddings for machine learning, arXiv:2001.03622.

[34] T. Hubregtsen, D. Wierichs, E. Gil-Fuster, P. J. H. Derks, P. K. Faehrmann, and J. J. Meyer, Training quantum embedding kernels on near-term quantum computers, Phys. Rev. A **106**, 042431 (2022).

[35] L. P. Henry, S. Thabet, C. Dalyac, and L. Henriet, Quantum evolution kernel: Machine learning on graphs with programmable arrays of qubits, Phys. Rev. A **104**, 032416 (2021).

[36] G. Chiribella, G. M. D'Ariano, and P. Perinotti, Quantum Circuit Architecture, Phys. Rev. Lett. **101**, 060401 (2008).

[37] A. Shafaei, M. Saeedi, and M. Pedram, Optimization of quantum circuits for interaction distance in linear nearest neighbor architectures, 2013 50th ACM/EDAC/IEEE Design Automation Conference (DAC), 1 (2013).

[38] L. Gyongyosi and S. Imre, Quantum circuit design for objective function maximization in gate-model quantum computers, Quantum Inf. Process. **18**, 225 (2019).

[39] T. Fösel, M. Y. Niu, F. Marquardt, and L. Li, Quantum circuit optimization with deep reinforcement learning, arXiv:2103.07585.

[40] M. Bilkis, M. Cerezo, G. Verdon, P. J. Coles, and L. Cincio, A semi-agnostic ansatz with variable structure for quantum machine learning, arXiv:2103.06712.

[41] H. R. Grimsley, S. E. Economou, E. Barnes, and N. J. Mayhall, An adaptive variational algorithm for exact molecular simulations on a quantum computer, Nat. Commun. **10**, 3007 (2019).

[42] T. Tomesh, Z. H. Saleem, and M. Suchara, Quantum local search with quantum alternating operator ansatz, Quantum **6**, 781 (2022).

[43] A. Skolik, J. R. McClean, M. Mohseni, P. van der Smagt, and M. Leib, Layerwise learning for quantum neural networks, Quantum Mach. Intell. **3**, 5 (2021).

[44] S. Altares-López, A. Ribeiro, and J. J. García-Ripoll, Automatic design of quantum feature maps, Quantum Sci. Technol. **6**, 045015 (2021).

[45] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd edition (Springer-Verlag, Berlin, 2009).

[46] A. A. Neath and J. E. Cavanaugh, The Bayesian information criterion: background, derivation, and applications, Wiley Interdisc. Rev. Comput. Stat. **4**, 199 (2012).

[47] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, Taking the human out of the loop: A review of Bayesian optimization, Proc. IEEE **104**, 148 (2015).

[48] D. Duvenaud, J. Lloyd, R. Grosse, J. Tenenbaum, and G. Zoubin, Structure discovery in nonparametric regression through compositional kernel search, Proceedings of the 30th International Conference on Machine Learning Research **28**, 1166 (2013).

[49] T. F. Wu, C. J. Lin, and R. Weng, Probability estimates for multi-class classification by pairwise coupling, Adv. Neural Inf. Process. Syst. **16** (2003).

[50] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevResearch.5.013211 for the proposed

algorithm's pseudocode in this research, technical details of the Bayesian Optimization (BO) procedure, analysis of the datasets, and a detailed comparison of classical and quantum model predictions for each class individually.

[51] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson, The Materials Project: A materials genome approach to accelerating materials innovation, APL Mater. **1**, 011002 (2013).

[52] Qiskit Machine Learning Development Team, Qiskit machine learning, https://qiskit.org/documentation/machine-learning.

[53] T. Fawcett, An introduction to ROC analysis, Pattern Recognit. Lett. **27**, 861 (2006).