

Estimating quantum coherence using limited quantum resources

Bin Zou(邹斌), Kai Wu(吴凯), and Zhihua Chen(陈芝花)
Citation: Chin. Phys. B, 2026, 35 (**3**): 030301. DOI: 10.1088/1674-1056/adfb56
http://cpb.iphy.ac.cn; http://iopscience.iop.org/cpb

What follows is a list of articles you may be interested in

Control of the Liouvillian gap in the finite open quantum system

Kai-Li Li(李凯丽), Yan-Sheng Liu(刘延盛), and Xi-Zheng Zhang(张禧征)
Chin. Phys. B, 2026, 35 (**1**): 010306. DOI: 10.1088/1674-1056/adeb60

Witnessing the distribution of sources in quantum networks via hierarchical nonlocality

Shu-Yuan Yang(杨舒媛), Jin-Chuan Hou(侯晋川), and Kan He(贺衍)
Chin. Phys. B, 2025, 34 (**6**): 060303. DOI: 10.1088/1674-1056/add507

Uncertainties of the standard quantum teleportation channel

Zhihua Zhang(章志华), Zehao Guo(郭泽豪), and Zhipeng Qiu(邱志鹏)
Chin. Phys. B, 2025, 34 (**4**): 040301. DOI: 10.1088/1674-1056/adb263

Sudden change of interferometric power for X-shape states

Dian Zhu(朱典), Fu-Lin Zhang(张福林), and Jing-Ling Chen(陈景灵)
Chin. Phys. B, 2025, 34 (**2**): 020305. DOI: 10.1088/1674-1056/ad9732

Phase-matching enhanced quantum phase and amplitude estimation of a two-level system in a squeezed reservoir

Yan-Ling Li(李艳玲), Cai-Hong Liao(廖彩红), and Xing Xiao(肖兴)
Chin. Phys. B, 2025, 34 (**1**): 010307. DOI: 10.1088/1674-1056/ad8f9e

---

Estimating quantum coherence using limited quantum resources

Bin Zou(邹斌)[1], Kai Wu(吴凯)[2,3], and Zhihua Chen(陈芝花)[1,†]

[1] *School of Science, Jimei University, Xiamen* 361021, *China*
[2] *Key Laboratory of Low-Dimensional Quantum Structures, and Quantum Control of Ministry of Education, Hunan Normal University, Changsha* 410081, *China*
[3] *Department of Physics and Synergetic Innovation Center for Quantum Effects and Applications, Hunan Normal University, Changsha* 410081, *China*

(Received 28 May 2025; revised manuscript received 23 July 2025; accepted manuscript online 14 August 2025)

Quantum coherence, as one of the most fundamental non-classical features in quantum mechanics, plays a pivotal role in various quantum information processing tasks, including quantum computing and quantum metrology. The robustness of quantum coherence (RoC) offers an operational interpretation by quantifying the advantage provided by a quantum state in phase discrimination tasks. To achieve verification RoC with high precision via semidefinite programming (SDP), complete knowledge of quantum states is typically required. Relying solely on expectation values of observables may introduce significant errors in SDP-based estimations. To estimate RoC with high precision using limited data extracted from quantum states, firstly, we propose a semi-supervised K-nearest neighbor (KNN) algorithm (semi-KNN) and a semi-supervised method that combines the KNN and random forest (RF) models with a dynamical threshold (semi-KNN-RF). Then we implement the semi-KNN and semi-KNN-RF models to efficiently estimate quantum coherence by analyzing statistical data obtained from randomly generated local projective measurements performed on unknown quantum states. The semi-KNN-RF model performs better than the semi-KNN model. This innovative methodology allows for accurate coherence estimation, even for high-dimensional quantum systems.

Keywords: robustness of quantum coherence, semi-K-nearest neighbors model, semi-K-nearest neighbors and random forest model

PACS: 03.65.Aa, 03.67.–a, 03.65.Ta, 03.65.–w

DOI: 10.1088/1674-1056/adfb56 **CSTR:** 32038.14.CPB.adfb56

1. Introduction

Emerging from the fundamental principle of quantum superposition, quantum coherence represents an essential nonclassical phenomenon that distinguishes quantum systems from classical ones.[1] This indispensable resource underpins critical quantum technologies spanning from quantum computation and quantum metrology to quantum communication.[2–6] Various measures, including coherence distillation and cost, Fisher information-based measures, and relative entropy of coherence, have been proposed to characterize quantum coherence.[8–14] Although theoretically comprehensive, these methods often require complex quantum state tomography in experiments, making them impractical for large-scale systems. While its theoretical foundations are firmly established, significant challenges persist in developing reliable methods for both precise quantification and practical manipulation of coherence, which are limitations that currently hinder its full exploitation in quantum information processing.

Among these measures, the robustness of quantum coherence (RoC) serves as an operational measure of quantum coherence, its lower bound is not only experimentally accessible but also numerically computed through semidefinite programming (SDP).[2] SDP can also be used to estimate other coherence measures, such as the geometric measure of quantum coherence.[15] However, due to its reliance on convex optimization techniques, the computational complexity increases significantly with the system dimension, posing a major challenge. In addition, lower and upper bounds of quantum coherence can be estimated by the spectrum estimation method with a small number of measurements or the traces and the entries of the density matrix.[16,17] A novel methodology has been proposed for estimating quantum coherence using limited experimental datasets.[18] Furthermore, a protocol has been developed for the direct measurement of quantum coherence in unknown quantum states through a two-copy collective measurement approach.[19] Recent advances have demonstrated the simultaneous estimation of quantum entanglement and coherence via purity detection techniques.[20] Recently, machine learning methods have been introduced to the field of quantum physics and quantum information to tackle various problems such as quantum state tomography and estimation, quantum error correction, and wave function reconstruction. Notably, machine learning has proven effective in entanglement detection and quantification.[21–29]

Machine learning has also been proposed to estimate the RoC, but all the entries of the density matrices were used as features for the machine learning models, which requires

[†]Corresponding author. E-mail: chenzhihua77@sina.com

© 2026 Chinese Physical Society and IOP Publishing Ltd. All rights, including for text and data mining, AI training, and similar technologies, are reserved.

quantum tomography and is quite resource-consuming.[30] Motivated by recent research progress, we integrate machine learning techniques with the quantification of quantum coherence based on the RoC. Firstly, we derive the SDP method to compute RoC values directly, and then we construct a labeled dataset using SDP. Lastly, we apply two semi-supervised regression models to estimate coherence values across different quantum states. We use the probabilities obtained by performing a small number of measurements as the features of two models. This method allows us to effectively determine the range of robustness while requiring a relatively small sample size. Our results demonstrate that a high coefficient of determination (R^2) can be achieved, indicating the potential of machine learning for efficient and scalable quantum coherence quantification. Furthermore, the algorithm we proposed can be used to estimate other resources, such as quantum entanglement measures, provided that the measures can be calculated or approximated using numerical algorithms, such as semi-definite programming or the descent gradient method. This framework is highly scalable and can be extended to higher-dimensional quantum systems, offering new directions for estimating quantum correlations in quantum information science.

2. SDP for calculating the RoC and the semi-supervised regression model

Let $\mathfrak{D}(\mathcal{C}^d)$ be the convex set of density operators acting on a d-dimensional Hilbert space, and let $\mathfrak{I} \subset \mathfrak{D}(\mathcal{C}^d)$ be the subset of incoherent states, then the RoC, introduced in Ref. [2], is defined as the minimum weight of another state τ such that its convex mixture with ρ yields an incoherent state δ, namely

$$\mathfrak{C}_R(\rho) = \min_{\tau \in \mathfrak{D}(\mathcal{C}^d)} \left\{ s \geq 0 \,|\, \frac{\rho + s\tau}{1+s} =: \delta \in \mathfrak{I} \right\}. \tag{1}$$

Then the minimum s satisfies the following equality:

$$\frac{\tau s}{1+s} = -\frac{\rho}{1+s} + \delta. \tag{2}$$

The left-hand side of Eq. (2) is positive semidefinite, then the right-hand side is also positive semidefinite. Let $p = \frac{1}{1+s}$, then calculating the RoC is transformed to the following SDP problem:

$$\begin{aligned} \max \quad & p \\ \text{s.t} \quad & (-p\rho + \delta) \geq 0, \\ & \mathrm{Tr}(\delta) = 1, \\ & \delta \geq 0, \quad \delta \in \mathfrak{I}. \end{aligned} \tag{3}$$

The solution of the above equation coincides with that obtained via SDP in Ref. [2]. While the SDP approach in Ref. [2] enables high-precision calculation of the RoC when exact density matrices from quantum state tomography are available, such tomographic reconstruction is notoriously resource-intensive in terms of both time and experimental requirements. Alternatively, one may estimate the RoC using only the expectation values of certain observables, but this approximation can introduce significant errors when using SDP in Ref. [2]. For instance, in the case of two-qutrit quantum states, the relative error in RoC estimation based on just six observable measurements may reach approximately 50%. To tackle this problem, we propose a semi-supervised machine learning method to estimate the quantum coherence with high precision using the limited data extracted from quantum states.

While artificial neural networks (ANNs) could theoretically serve as a method to approximate the coherence measures, they are more computationally expensive than the K-nearest neighbor (KNN) algorithm due to complex training, hyperparameter tuning, and hardware needs. The KNN algorithm is a simple, instance-based supervised machine learning algorithm used for classification and regression, and it remains a lightweight alternative for low-resource settings. Our initial experimental results demonstrate that the KNN algorithm outperforms ANNs in the task of estimating the RoC. The KNN algorithm operates on the principle that similar data points (neighbors) tend to have similar outcomes. It predicts a sample's value by averaging (for regression) or taking a majority vote (for classification) from its k closest neighbors, where proximity is determined using distance metrics such as Euclidean distance.

The optimal k is typically selected through cross-validation to balance bias and variance. Since the KNN method relies on distance calculations, feature scaling is essential to ensure that all features contribute equally to the nearest-neighbor determination.

In this article, the Euclidean distance is chosen. For two n-dimensional vectors $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$, where x and y are two sample points and n is the number of features, x_i and y_i represent the i-th feature of samples x and y, respectively. The Euclidean distance is

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}. \tag{4}$$

Based on the distances, the k nearest neighbors to the test sample are selected. Using the output values of these k neighbors, the output of the test sample is calculated by using the average values as

$$\widetilde{z} = \frac{1}{k} \sum_{i=1}^{k} z_i, \tag{5}$$

where z_i is the output value of the i-th neighbor. If weighted averaging is used, the predicted value is the weighted average of the k neighbors' outputs, where the weights are usually the

reciprocals of the distances between the neighbors and the test sample:

$$\widetilde{z} = \frac{\sum_{i=1}^{k} \omega_i z_i}{\sum_{i=1}^{k} \omega_i}, \tag{6}$$

where $\omega_i = \frac{1}{d(x,x_i)}$ is the reverse of the distance between the test sample and the i-th neighbor. Select an appropriate value of k and predict the value of the next point using weighted averaging.

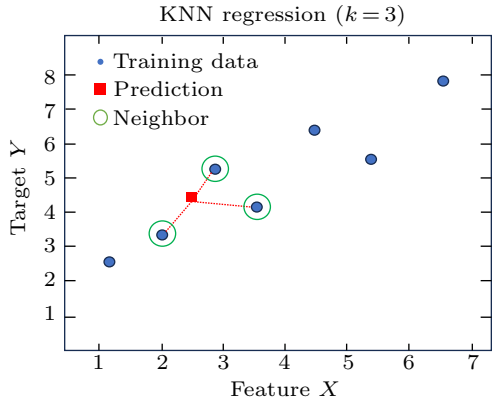


Fig. 1. The illustration depicts the scenario when $k = 3$, indicating that three nearest "neighbors" are selected to calculate the weighted average value based on Euclidean distance. The blue dots represent the sample points, the green circles denote the chosen "neighbors," and the red lines represent the Euclidean distances.

Semi-supervised learning methods leverage a small amount of labeled data and a large amount of unlabeled data to enhance the model's learning capabilities. Utilizing information from unlabeled data, semi-supervised learning methods can make full use of the information from a large number of unlabeled samples. Compared to supervised learning, which only uses labeled samples, it provides more data for model training, thereby improving the model's performance and generalization ability. Compared to supervised learning, it can reduce labeling costs, and achieve good performance by labeling only a small number of samples, thus reducing the cost and workload of labeling samples.

Semi-supervised K-nearest neighbors (semi-KNN) is an extension of the traditional KNN algorithm that incorporates both labeled and unlabeled data to improve classification or regression performance. This method leverages the strengths of supervised learning with labeled data and the unsupervised learning potential of unlabeled data, creating a more comprehensive model. In the semi-KNN approach, the algorithm initially uses the available labeled data to find the k-nearest neighbors for a given query instance. However, unlike the supervised KNN which relies solely on labeled data, semi-KNN also takes into account the unlabeled data points to refine the classification. This is achieved by considering the structure and patterns within the entire dataset, including both labeled and unlabeled examples, to make more informed predictions. One of the key advantages of semi-KNN is its ability to handle non-linear relationships and high-dimensional data without the need for explicit feature mapping, as seen in methods such as support vector machines.

Besides the semi-KNN model, we utilize another semi-supervised model, namely, a random forest (RF) model. This algorithm demonstrates distinct advantages in handling high-dimensional data, owing to its inherent feature selection capabilities and robustness against overfitting. The RF model performs random sampling on the original dataset to construct distinct n sample datasets. Based on these datasets, n different decision tree models are built, and the final regression result is obtained by averaging the predictions from these decision tree models. Decision trees are tree-structured algorithms widely used in both classification and regression tasks. They recursively partition the dataset into smaller subsets, ultimately forming a tree-like model to predict the output of new data points. Figure 2 illustrates the decision-making process of an RF.

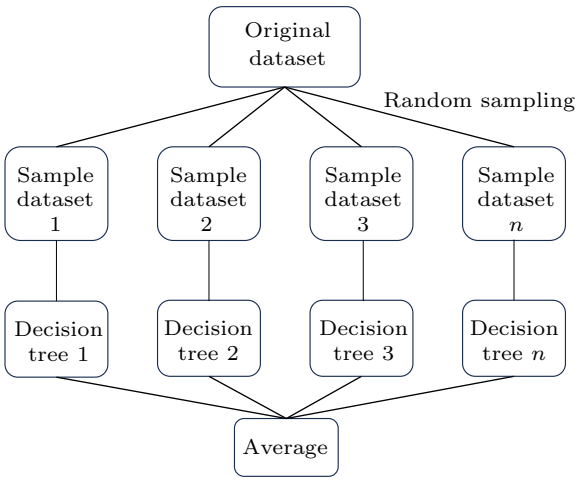


Fig. 2. The RF model splits the data, constructs n decision tree models, and outputs the average prediction of these trees as the final regression result.

We propose a semi-supervised regression model which combines the KNN and RF models with a dynamical threshold (semi-KNN-RF). We continuously use both RF and KNN models to predict unlabeled data. When the difference of their predictions is lower than some threshold, we use the average of their predictions as a pseudo-label to enrich the training data. The dynamic threshold function is used to adjust the threshold for pseudo-label generation during each iteration based on the current training epoch

$$\mathrm{thr} = t_0 + (T - t_0)\left(1 - \mathrm{e}^{\left(-\alpha \cdot \frac{\mathrm{epoch}}{\mathrm{epoch_m}}\right)}\right), \tag{7}$$

Specifically, t_0 is the initial threshold, T is the maximum threshold, epoch is the current training epoch, $\mathrm{epoch_m}$ is the total number of epochs, and α is a parameter that controls the growth rate; the larger the value of α, the faster the threshold increases. The purpose of this function is to gradually

raise the threshold as training progresses. At early stages, a lower threshold allows the model to generate pseudo-labels more flexibly. As training continues, the threshold increases, making the consistency requirement between model predictions stricter and thereby improving the quality of the pseudo-labels.

3. Training semi-supervised regression models

3.1. Preparation of the training dataset

Initially, we just generate quantum states randomly. However, numerical experiments reveal that the coherence measures RoC derived from these states exhibit a concentrated distribution, failing to cover the full range of possible values. This indicates that the generated states are not sufficiently diverse to meet our experimental requirements. To prepare the training dataset and the testing dataset, firstly we generate the quantum states as the convex mixture of randomly generated quantum states, incoherent states, and the maximal coherent states in d- dimensional Hilbert space, namely

$$\rho = p\rho' + q\delta' + (1-p-q)\omega, \tag{8}$$

where ρ' is a random state, δ' is the random incoherent state and $\omega = \frac{1}{\sqrt{d}}\sum_{i=1}^{d}|ii\rangle$ with $p \geq 0$, $q \leq 0$ and $p+q \leq 1$. Secondly, for a d-dimensional quantum state ρ_d, random d-dimensional local unitary matrices U_j $(j = 1,\ldots,M)$ are generated to act on the d-dimensional measurement basis and obtain $\{U_j|i\rangle\}_{i=0}^{d-1}$; here, $i = 0,1,\ldots,d-1$ and $|i\rangle$ is the measurement basis with the i-th entry being 1 and other entries 0, U_j is the tensor product of two, three (or four) lower dimensional unitary matrices for 4, 6, 9, 18 (or 36) dimensional quantum states. Tensor products of low-dimensional unitaries are significantly more accessible to realize experimentally compared with arbitrary nonlocal unitaries, as they can be decomposed into sequences of fundamental gates in most quantum hardware architectures. This approach substantially reduces the demand for physical resources (e.g., fewer entangling gates are required). Performing the measurements $\{U_j|i\rangle\}_{i=0}^{d-1}$ on the quantum state ρ_d for $j = 1,\ldots,M$, we can get the features of semi-KNN and semi-KNN-RF as $\{\mathrm{tr}[\rho_d U_j|i\rangle\langle i|U_j^+]\}_{i,j}$. To improve generality of the model, we adopt the strategy of increasing the number of feature vectors. In the numerical experiment, we let $M = 3$ for three-dimensional (3D) and six-dimensional (6D) quantum states, and $M = 4$ for four-dimensional (4D), nine-dimensional (9D), eighteen-dimensional (18D) and thirty-six-dimensional (36D) quantum states. The labels are obtained by performing SDP process in Eq. (3).

3.2. The training process of the semi-supervised regression model

Following the preparation of the dataset, the training process begins with splitting the training set into labeled data and unlabeled data. The labeled subset is used to train a KNN regression model or an RF, while the unlabeled portion is reserved for subsequent testing. During the process of KNN, cross-validation is employed to determine the optimal k-value (number of neighbors) to ensure model robustness. Once trained, the model predicts labels for the unlabeled data. High-confidence samples, which are those exhibiting the smallest distances to their nearest neighbors, are identified and incorporated back into the labeled training set. This iterative process continues, retraining the model with the augmented dataset in each cycle, until no further high-confidence samples can be reliably selected or a predefined maximum iteration count is reached. Upon completion of training, the remaining unlabeled data points form the test set, which is used to evaluate the model's final performance. The left figure in Fig. 3 provides a flowchart illustrating this semi-supervised training process. The right figure in Fig. 3 presents a flowchart of the semi-KNN-RF model. After splitting the data, both KNN and RF regression models are trained on the labeled data. The absolute difference between the predictions of the two models is used to determine their consistency. If the difference is smaller than a dynamic threshold, the prediction is considered consistent, and the average of the two predictions is used as a pseudo-label. If the difference exceeds the threshold, the sample is excluded. This process continues until the maximum number of iterations is reached, completing the training of the entire model.

The mean squared error (MSE), the mean absolute percentage error (MAPE) and the coefficient of determination (R^2) are employed to evaluate the model's performances. The MSE quantifies the average squared discrepancy between the predicted values and their corresponding true values. Mathematically, it is defined as

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \tag{9}$$

where n is the total number of the samples, y_i is the i-th true value, and $\hat{y}_i$ is the i-th predicted value. MAPE represents the average of the absolute percentage errors of the predictions from the true values, which is defined as

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|/y_i. \tag{10}$$

The coefficient of determination (R^2) serves as a key metric for evaluating regression model performance. It measures the proportion of variance in the dependent variable that is predictable from the independent variables. Its values range from 0 to 1, where $R^2 = 1$ indicates a perfect fit where the model explains all the variability of the response data, and $R^2 = 0$ suggests the model fails to capture any linear relationship. R^2 is computed as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{11}$$

where $\bar{y}$ represents the mean of the observed data.

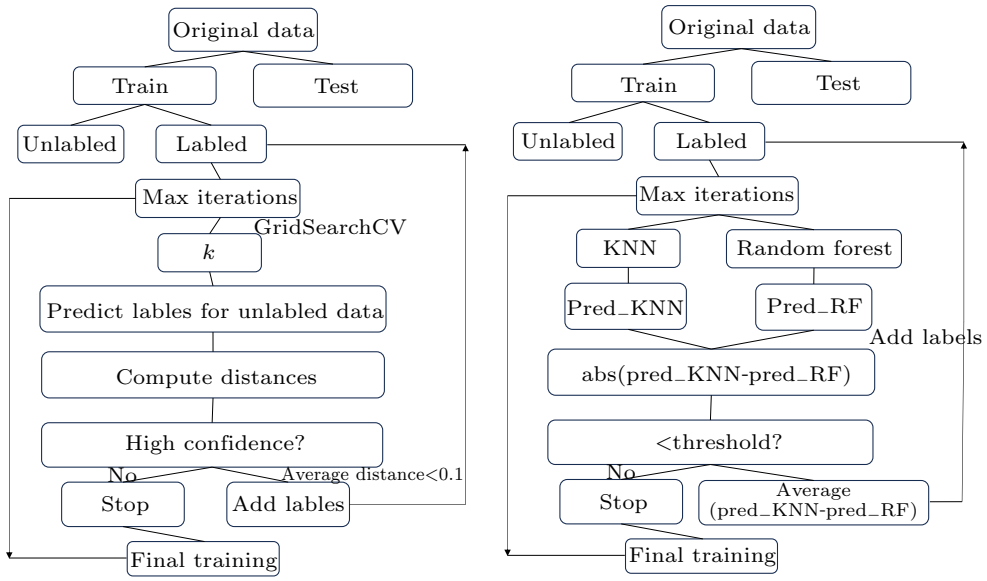


Fig. 3. The left diagram illustrates the workflow of the semi-KNN model. A maximum number of iterations is set (6 in the semi-KNN model). The KNN model is then trained using the training set, and cross-validation is used to determine the optimal value of k. Data points with high confidence (distance less than 0.1) are selected, labeled accordingly, and added to the labeled dataset for further training. The right diagram introduces a semi-KNN-RF model after the labeled and unlabeled data have been split. The RF and KNN are trained, and the absolute difference between the predictions of the KNN and RFs is computed. A dynamic threshold function is defined to generate a new threshold in each iteration using Eq. (7). Data points with a prediction difference smaller than the current threshold are selected, added to the labeled dataset, and used for further training.

4. Results

In the experiments, for different dimensional quantum states, different numbers of data are generated. For 3D random quantum states, a total of 30000 data points are generated. 80% of these data are randomly selected as the training data, which includes 21600 data points as original labeled dataset, and the remaining 2400 data points as unlabeled data. The experimental results consistently demonstrate $R^2 > 0.95$. For 4D,6D, 9D, 18D and 36D random quantum states, 35000 data points were prepared, with 22400 data points assigned as the original labeled dataset and 5600 data points as unlabeled data, yielding R^2 exceeding 0.95. These experimental results are summarized in Table 1, we present the numerical experimental results for quantum states of varying dimensions: 3D, 4D, 6D, 9D, 18D and 36D. The first row for each corresponding dimension shows the average results of four runs using the semi-KNN model, while the second row shows the average results of four runs using the semi-KNN-RF model based on a dynamic threshold. The parameters for the dynamic threshold can be set as $t_0 = 0.015$, $T = 0.085$, $\alpha = 3$ and $\text{epch}_\text{m} = 6$ for all states. To ensure proper alignment with semi-supervised learning principles, we implemented the semi-KNN-RF model using a small labeled dataset (5,600 or fewer quantum states) combined with a large unlabeled dataset (22,400 or more quantum states) for RoC estimation. The dynamic threshold mechanism was carefully configured with dimension-specific parameters: For 3D, 4D, 6D and 9D quantum states, we set the initial threshold $t_0 = 0.15$, maximum threshold $T = 0.16$, growth rate $\alpha = 3$ and the number of the training epochs $\text{epch}_\text{m} = 6$. For higher-dimensional 18D and 36D systems, we use $t_0 = 0.10$, $T = 0.15$, $\alpha = 2$ while maintaining $\text{epch}_\text{m} = 6$. These parameter choices enable the model to begin with more flexible pseudo-labeling during early training stages before gradually enforcing stricter prediction consistency. Through proper hyperparameter tuning, this approach maintains high precision while adhering to the fundamental semi-supervised learning paradigm of leveraging minimal labeled data alongside abundant unlabeled data.

Numerical experimental results show that our method performs well when dealing with different dimensional quantum states, achieving a high R^2 with relatively small sample sizes. Beyond random unitary matrices, we investigate the relationship between the given unitary matrices and accuracy of estimating the RoC. When we replace one of the unitary matrices generated randomly with the identity matrix (i.e., using the original basis), the R^2 decreases significantly. For example, we conducted experiments with one unitary matrix replaced by the identity matrix for the four-dimensional and six-dimensional quantum states, R^2 becomes 0.556 and 0.882, respectively. However, replacing the unitary matrix with a fixed nonsparce matrix as Eq. (12) significantly improves the R^2 back to its original performance. These examples demonstrate that the performance tends to be better when the unitary matrices are less sparse.

$$U_1 = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{j}{\sqrt{2}} \\ \frac{j}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix},$$
$$F_3 = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 1 & 1 \\ 1 & -\frac{1}{2} + \frac{\sqrt{3}}{2} & -\frac{1}{2} - \frac{\sqrt{3}}{2} \\ 1 & -\frac{1}{2} - \frac{\sqrt{3}}{2} & -\frac{1}{2} + \frac{\sqrt{3}}{2} \end{pmatrix}. \quad (12)$$

Table 1. The R^2, MSE, MAPE and the number of labeled, unlabeled and test data for predicting the RoC of different quantum states.

Dimension	R^2	MSE	MAPE	Labeled data	Unlabeled data	Test data	Range of RoC
3	0.9525	0.0196	0.0408	21600	2400	6000	0.0422–1.9900
	0.9544	0.0053	0.0416	21600	2400	6000	
	0.9503	0.0086	0.0503	5600	22400	6000	
4	0.9643	0.0648	0.0448	22400	5600	7000	0.0305–2.9808
	0.9646	0.0092	0.0336	22400	5600	7000	
	0.9566	0.0163	0.0500	5600	22400	7000	
6	0.9662	0.0248	0.0643	22400	5600	7000	0.0253–4.9439
	0.9542	0.0463	0.0541	22400	5600	7000	
	0.9421	0.0584	0.0667	5600	22400	7000	
9	0.9430	0.0413	0.0572	22400	5600	7000	0.0167–7.9410
	0.9609	0.0413	0.0425	22400	5600	7000	
	0.9423	0.1427	0.0555	5600	22400	7000	
18	0.9575	0.3427	0.0480	22400	5600	7000	0.1337–16.9645
	0.9635	0.1940	0.0322	22400	5600	7000	
	0.9731	0.3083	0.0427	5600	22400	7000	
	0.9507	0.5653	0.0615	840	27160	7000	
36	0.9379	2.9452	0.0541	22400	5600	7000	0.3046–34.9594
	0.9876	0.5270	0.0235	22400	5600	7000	
	0.9810	0.9030	0.0304	5600	22400	7000	
	0.9588	1.9527	0.0441	1120	26880	7000	

For 18D quantum states, we perform experiments by replacing the first two random local unitary matrices with $U_1 = U_3^1(\frac{\pi}{4}) \otimes U_3^1(\frac{\pi}{4}) \otimes X$ and $U_2 = U_3^1(\frac{\pi}{3}) \otimes U_3^1(\frac{\pi}{3}) \otimes X$, where $U_3^1(t_1)$ is defined in Eq. (13). Under this configuration, the semi-KNN model achieved $R^2 = 0.8656$, while the semi-KNN-RF model yielded $R^2 = 0.9175$. Next, we further replace the third random local unitary matrix with $U_3 = U_3^2(\frac{\pi}{4}) \otimes U_3^2(\frac{\pi}{4}) \otimes X$, where $U_3^2(t_2)$ is given in Eq. (13), then we have $R^2 = 0.8123$, MSE $= 2.1512$ and MAPE $= 0.1124$ for semi-KNN and $R^2 = 0.8456$, MSE $= 1.7703$ and MAPE $= 0.1035$ for the semi-KNN-RF model. Besides U_j $(j = 1,2,3)$, we finally substitute the fourth random matrix with $U_3^2(\frac{\pi}{3}) \otimes U_3^2(\frac{\pi}{3}) \otimes X$, then $R^2 = 0.7972$, MSE is 2.3245 and MAPE is 0.1129 for semi-KNN and $R^2 = 0.8255$, MSE is 2.0006 and MAPE is 0.1078 for the semi-KNN-RF model.

$$U_3^1(t_1) = \begin{pmatrix} \cos(t_1) & 0 & \mathrm{i}\sin(t_1) \\ 0 & 1 & 0 \\ \mathrm{i}\sin(t_1) & 0 & \cos(t_1) \end{pmatrix},$$
$$U_3^2(t_2) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(t_2) & \sin(t_2) \\ 0 & -\sin(t_2) & \cos(t_2) \end{pmatrix}. \quad (13)$$

5. Conclusion

By leveraging the probabilities derived from three or four sets of projective measurements on unknown quantum states as input features for a semi-supervised KNN model and a semi-supervised hybrid KNN and RF model, we demonstrate that the robustness of quantum coherence can be estimated with high precision, even when using limited training data. Our results indicate that the semi-KNN-RF model generally outperforms the semi-KNN approach, achieving higher accuracy and reliability in coherence estimation. This suggests that a hybrid model combining the KNN and RF algorithms can enhance the predictive power of quantum machine learning models, particularly in scenarios where data availability is constrained. This work highlights the potential of integrating classical machine learning models for efficient quantum property estimation, especially in data-scarce scenarios.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (Grant No. 12071179).

References

[1] Streltsov A, Adesso G and Plenio M B 2017 *Rev. Mod. Phys.* **89** 041003
[2] Napoli C, Bromley T R, Cianciaruso M, Piani M, Johnston N and Adesso G 2016 *Phys. Rev. Lett.* **116** 150502
[3] Piani M, Cianciaruso M, Bromley T R, Napoli C, Johnston N and Adesso G 2016 *Phys. Rev. A* **93** 042107
[4] Shi H L, Liu S Y, Wang X H, Yang W L, Yang Z Y and Fan H 2017 *Phys. Rev. A* **95** 032307
[5] Giorda P and Allegra M 2017 *J. Phys. A: Math. Theor.* **51** 025302
[6] Palacios A, Martínez-Peña R, Soriano M C, Giorgi G L and Zambrini R 2024 *Commun. Phys.* **7** 369
[7] Baumgratz T, Cramer M and Plenio M B 2014 *Phys. Rev. Lett.* **113** 140401
[8] Hu M L, Hu X Y, Wang J C, Peng Y, Zhang Y R and Fan H 2018 *Phys. Rep.* **762** 1
[9] Winter A and Yang D 2016 *Phys. Rev. Lett.* **116** 120404
[10] Yuan X, Zhou H, Cao Z and Ma X 2015 *Phys. Rev. A* **92** 022124
[11] Yu C S 2017 *Phys. Rev. A* **95** 042337
[12] Li L, Wang Q W, Shen S Q and Li M 2021 *Phys. Rev. A* **103** 012401
[13] Shi X 2024 *Phys. Rev. A* **109** 052443
[14] Qi X, Gao T and Yan F 2017 *J. Phys. A: Math. Theor.* **50** 285301
[15] Zhang Z, Dai Y, Dong Y L and Zhang C J 2020 *Sci. Rep.* **10** 12122
[16] Ding Q M, Fang X X, Yuan X, Zhang T and Lu H 2021 *Phys. Rev. Res.* **3** 023228

[17] Zhang H J, Chen B, Li M, Fei S M and Long G L 2017 *Commun. Theor. Phys.* **67** 166

[18] Zhang D J, Liu C L, Yu X D and Tong D M 2018 *Phys. Rev. Lett.* **120** 170501

[19] Yuan Y, Hou Z B, Tang J F, Streltsov A, Xiang G Y, Li C F and Guo G C 2020 *npj Quantum Inf.* **6** 28

[20] Zhang T, Smith G, Smolin J A, Liu L, Peng X J, Zhao Q, Girolami D, Ma X, Yuan X and Lu H 2024 *npj Quantum Inf.* **10** 60

[21] Chen Z, Lin X and Wei Z 2023 *Quantum Sci. Technol.* **8** 035029

[22] Lin X, Chen Z and Wei Z 2023 *Phys. Rev. A* **107** 062409

[23] Tian Y, Che L, Long X, Ren C and Lu D 2022 *Adv. Quantum Technol.* **5** 2200025

[24] Canabarro A, Brito S and Chaves R 2019 *Phys. Rev. Lett.* **122** 200401

[25] Luo Y J, Liu J M and Zhang C 2023 *Phys. Rev. A* **108** 052424

[26] Zhang L, Chen Z and Fei S M 2023 *Phys. Rev. A* **108** 022427

[27] Zhang J, He K, Zhang Y, Hao Y Y, Hou J C, Lan F P and Niu B N 2022 *Phys. Rev. A* **105** 032408

[28] Gao X, Isoard M, Sun F, Lopetegui C E, Xiang Y, Parigi V, He Q and Walschaers M 2024 *Phys. Rev. Lett.* **132** 220202

[29] Wu Y D, Zhu Y, Wang Y and Chiribella G 2024 *Nat. Commun.* **15** 9796

[30] Zhang L, Chen L, He Q and Zhang Y 2024 *Appl. Sci.* **14** 7312