Article

Quantum-Inspired Classification Based on Voronoi Tessellation and Pretty-Good Measurements

Roberto Leporini [1,*] and Davide Pastorello [2,3]

1 Department of Economics, University of Bergamo, via dei Caniana 2, I-24127 Bergamo, Italy
2 Department of Information Engineering and Computer Science, University of Trento, via Sommarive 9, I-38123 Povo, Italy
3 Trento Institute for Fundamental Physics and Applications—INFN, via Sommarive 14, I-38123 Povo, Italy
* Correspondence: roberto.leporini@unibg.it

Abstract: In quantum machine learning, feature vectors are encoded into quantum states. Measurements for the discrimination of states are useful tools for classification problems. Classification algorithms inspired by quantum state discrimination have recently been implemented on classical computers. We present a local approach combining Vonoroi-type tessellation of a training set with pretty-good measurements for quantum state discrimination.

Keywords: quantum state discrimination; optimal quantum measurement; machine learning

1. Introduction

Quantum-inspired algorithms exploit some of the advantages of quantum computing on classical hardware, providing new models of information storing and processing. In this respect, quantum formalism, based on linear operators on Hilbert spaces, is used as a rich mathematical machinery without the need for an underlying quantum system which physically realizes the computations. In particular, quantum-inspired machine learning is a field where this kind of algorithm is developed to accomplish machine learning tasks.

In the context of supervised learning, a general classification problem is defined as the assignment of labels to new data instances given a training set of already labeled data. For example, a label can be assigned to a new data instance on the basis of its distance to the training data within a metric space where data are represented. In many relevant cases, data instances are represented in a real vector space, called feature space, equipped with the Euclidean distance. Classification with quantum computers is a widely investigated topic (e.g., [1–3]), but the quantum-inspired paradigm can also be applied. Some quantum-inspired classification algorithms based on a geometric approach have recently been presented in [4] and compared with well-known classical methods.

This paper is devoted to the investigation of some quantum-inspired classification algorithms, based on the notion of pretty-good measurement, within a local approach. More precisely, we consider a Vonoroi-type tessellation of the dataset proposed in [5] to classify an unlabeled instance without considering the entire dataset but, instead, only a neighborhood of the test point. Therefore, the present proposal involves investigation that goes beyond the quantum-inspired classifiers studied in [4]. Here, we integrate the geometric approach to classification based on quantum discrimination with a *local strategy* that has been successfully applied in [5], but which has also been suggested as a promising path to improve classification in less recent proposals, such as [6]. However, the notion of locality that we address is uniquely related to the selection of a neighborhood of the test point into the feature space and has nothing to do with the notion of *quantum non-locality*, violation of Bell inequalities, or local realism via hidden variable formulations of quantum mechanics.

An ensemble of classifiers is a set that improves the performance of an individual classifier. These classifiers are trained with subsets of the original training set and integrated to achieve more accurate classification. Methods for constructing ensemble learning algorithms include boosting and bagging. The former learns sequentially: in each iteration it assigns higher weight to the observations misclassified by its predecessor. In the latter, different sample subsets are randomly drawn from the training dataset and each subset is used to train a basic learning model in parallel. The global decision is obtained by voting. Ensemble learning has been successfully used in diverse applications, such as text classification, speech recognition, sentiment analysis, protein-folding recognition, and streamflow forecasting [7].

In our approach, the ensemble is built using a clustering method that generates a Voronoi diagram, which splits the space into regions defined by its representative. The relative quantum-inspired classifiers are structured on the encoding of the feature vectors into density operators and on techniques for estimating the distinguishability of quantum states, such as pretty-good measurement. The classification accuracy of these quantum-inspired classifiers can be improved by increasing, in terms of tensor products, the number of preparations of the quantum states that encode the feature vectors, at the cost of increasing the computational space and time.

In Section 2, we recall the quantum encoding of data vectors into density operators and the quantum-inspired classification based on the well-known quantum state discrimination developed by Helstrom [8] and others [9]. Section 3 is focused on an encoding of feature vectors into Bloch vectors that scale efficiently, increasing the dimension of the feature space. In Section 4, we introduce the local algorithm as a procedure to restrict the training set to the nearest points around each representative point of the tessellation, enabling the local and parallel execution of the quantum-inspired classifiers. We present an application of the Iris dataset for evaluating the impact of locality in quantum-inspired classification, comparing the performances of the proposed algorithm to classical methods. In Section 5, concluding remarks are presented and future developments towards innovative techniques in machine learning are discussed.

2. Quantum-Inspired Classification

The first step of quantum-inspired classification is quantum encoding, that is, any procedure to encode classical information into quantum states. In this paper, we consider encodings of data vectors into density matrices on a Hilbert space H whose dimension depends on the dimension of the input space. Density matrices are the mathematical objects used to describe the physical states of quantum systems. A density matrix on H is a positive semidefinite operator ρ such that $\mathrm{tr}\rho = 1$. *Pure states* are all the density matrices of the form $\rho = |\psi\rangle\langle\psi|$, with $\|\psi\| = 1$, which are the rank-1 projectors that can be directly identified with unit vectors up to a phase factor. Let ρ be a density operator on a d-dimensional Hilbert space $\mathbb{C}^d$. ρ can be written in the following form:

$$\rho = \frac{1}{d}\left(\mathrm{I}_d + \sqrt{\frac{d(d-1)}{2}}\sum_{j=1}^{d^2-1} b_j^{(\rho)}\sigma_j\right), \tag{1}$$

where $\{\sigma_j\}_{j=1,\dots,d^2-1}$ are the standard generators of the special unitary group $SU(d)$, also called *generalized Pauli matrices*, and I_d is the identity matrix of size d. The vector $\mathbf{b}^{(\rho)} = (b_1^{(\rho)}, \dots, b_{d^2-1}^{(\rho)})$, with $b_j^{(\rho)} = \sqrt{\frac{d}{2(d-1)}}\mathrm{tr}(\rho\,\sigma_j) \in \mathbb{R}$, is the *Bloch vector* associated to ρ which lies within the hypersphere of radius 1 in $\mathbb{R}^{d^2-1}$. For $d = 2$, the qubit case, the density matrices are in bijective correspondence to the points of the unit ball in $\mathbb{R}^3$, where the pure states are in one-to-one correspondence with the points of the surface of the Bloch sphere. For $d > 2$, the points contained in the unit hypersphere of $\mathbb{R}^{d^2-1}$ are not in bijective correspondence with density matrices on $\mathbb{C}^d$, as in the case of a single qubit, so the Bloch vectors do not form a ball but a complicated convex body. However, any vector

within the closed ball of radius $\frac{2}{d}$ gives rise to a density operator [10]. One can apply the Bloch representation of density matrices as an efficient quantum encoding as discussed in Section 3.

Complex vectors of dimension n can be encoded into density matrices of a $(n + 1)$-dimensional Hilbert space H in the following way:

$$\mathbb{C}^n \ni \mathbf{x} \mapsto |\mathbf{x}\rangle = \frac{1}{\sqrt{\| \mathbf{x} \|^2 + 1}} \left(\sum_{\alpha=0}^{n-1} x_\alpha |\alpha\rangle + |n\rangle \right) \in \mathsf{H}, \tag{2}$$

where $\{|\alpha\rangle\}_{\alpha=0,...,n}$ is the computational basis of H, identified as the standard basis of $\mathbb{C}^{n+1}$. The map defined in (2), called *amplitude encoding*, encodes $\mathbf{x}$ into the pure state $\rho_\mathbf{x} = |\mathbf{x}\rangle\langle\mathbf{x}|$, where the additional component of $|\mathbf{x}\rangle$ stores the norm of $\mathbf{x}$. Nevertheless, the quantum encoding $\mathbf{x} \mapsto \rho_\mathbf{x}$ can be realized in terms of the Bloch vectors $\mathbf{x} \mapsto \mathbf{b}^{(\rho_\mathbf{x})}$, saving space resources. The improvement in memory occupation within the Bloch representation is evident when we take multiple tensor products $\rho \otimes \cdots \otimes \rho$ of a density matrix ρ, constructing a feature map to enlarge the dimension of the representation space [4].

Quantum-inspired classifiers are based on quantum encoding of data vectors into density matrices, calculations of centroids, and various criteria of quantum state distinguishability, such as the Helstrom state discrimination [11], the pretty-good measurement [12], and the geometric construction of a minimum-error measurement [13].

Let us briefly recall the notion of quantum state discrimination. Given a set of arbitrary quantum states with respective a priori probabilities $R = \{(\rho_1, p_1), \ldots, (\rho_N, p_N)\}$, in general, there is no measurement process that discriminates the states without errors. More formally, there does not exist a POVM, i.e., a collection $E = \{E_i\}_{i=1,...,N}$ of positive semidefinite operators such that $\sum_{i=1}^N E_i = \mathsf{I}$, satisfying the following property: $\mathrm{tr}(E_i\rho_j) = 0$ when $i \neq j$ for all $i, j = 1, \ldots, N$. The probability of a successful state discrimination of the states in R performing the measurement E is:

$$\mathbb{P}_E(R) = \sum_{i=1}^N p_i \mathrm{tr}(E_i\rho_i). \tag{3}$$

An interesting and useful task is finding the optimal measurement that maximizes the probability (3). Helstrom provided a complete characterization of the optimal measurement E_{opt} for $R = \{(\rho_1, p_1), (\rho_2, p_2)\}$ [8]. E_{opt} can be constructed as follows: Let $\Lambda := p_1\rho_1 - p_2\rho_2$ be the *Helstrom observable*, whose positive and negative eigenvalues are, respectively, collected in the sets D_+ and D_-. Consider the two orthogonal projectors:

$$P_\pm := \sum_{\lambda \in D_\pm} P_\lambda, \tag{4}$$

where P_λ projects onto the eigenspace of λ. The measurement $E_{opt} := \{P_+, P_-\}$ maximizes the probability (3) that attains the *Helstrom bound* $h_b(\rho_1, \rho_2) = p_1\mathrm{tr}(P_+\rho_1) + p_2\mathrm{tr}(P_-\rho_2)$.

Helstrom quantum state discrimination can be used to implement a quantum-inspired binary classifier with promising performances [11]. Let $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_M, y_M)\}$ be a training set with $y_i \in \{1, 2\} \ \forall i = 1, \ldots, M$. Once a quantum encoding $\mathbb{C}^n \ni \mathbf{x} \mapsto \rho_\mathbf{x} \in \mathfrak{S}(\mathsf{H})$ has been selected, one can construct the quantum centroids ρ_1 and ρ_2 of the two classes $C_{1,2} = \{\mathbf{x}_i : y_i = 1, 2\}$. Let $\{P_+, P_-\}$ be the Helstrom measurement defined by the set $R = \{(\rho_1, p_1), (\rho_2, p_2)\}$, where the probabilities attached to the centroids are $p_{1,2} = \frac{|C_{1,2}|}{|C_1|+|C_2|}$. The *Helstrom classifier* applies the optimal measurement for the discrimination of the two quantum centroids to assign the label y to a new data instance $\mathbf{x}$, encoded into the state $\rho_\mathbf{x}$, as follows:

$$y(\mathbf{x}) = \begin{cases} 1 & \text{if} \quad \mathrm{tr}(P_+\rho_\mathbf{x}) \geq \mathrm{tr}(P_-\rho_\mathbf{x}) \\ 2 & \text{otherwise} \end{cases} \tag{5}$$

A strategy to increase the accuracy in classification is given by the construction of the tensor product of k copies of the quantum centroids $\rho_{1,2}^{\otimes k}$, enlarging the Hilbert space where data are encoded. The corresponding Helstrom measurement is $\{P_+^{\otimes k}, P_-^{\otimes k}\}$, and the Helstrom bound satisfies [11]:

$$h_b(\rho_1^{\otimes k}, \rho_2^{\otimes k}) \leq h_b\left(\rho_1^{\otimes (k+1)}, \rho_2^{\otimes (k+1)}\right) \qquad \forall k \in \mathbb{N}. \tag{6}$$

By enlarging the Hilbert space of the quantum encoding, one increases the Helstrom bound, obtaining a more accurate classifier. Since the Helstrom classifier is similar to a support vector machine (SVM) with linear kernel [14], considering many copies of the encoding quantum states gives rise to a kernel trick. The corresponding computational cost is evident, but the space can be enlarged, saving time and space by means of the encoding into Bloch vectors because fewer significant features are used.

A popular method for quantum state discrimination for distinguishing more than two states is the square-root measurement, also known as pretty-good measurement, defined by:

$$E_i = p_i \rho^{-\frac{1}{2}} \rho_i \rho^{-\frac{1}{2}}, \tag{7}$$

where $\rho = \sum_i p_i \rho_i$. This method gives the optimal minimum-error when states satisfy certain symmetry properties [12]. Clearly, to distinguish between n centroids, we need a measurement with at most n outcomes. It is sometimes optimal to avoid measurement and simply guess that the state is the a priori most likely state.

3. Bloch Representation

In quantum-inspired machine learning, the encoding of data instances into Bloch vectors of density operators turns out to be a useful geometric tool to reduce memory consumption in defining feature maps into higher dimensional spaces. Within the quantum encoding (2), a real vector $\mathbf{x} \in \mathbb{R}^{d-1}$ is encoded in a projector operator $\rho_\mathbf{x} = |\mathbf{x}\rangle\langle\mathbf{x}|$, on a d-dimensional Hilbert space where $d \geq 2$. For simplicity, we consider an input vector $[x_1, x_2] \in \mathbb{R}^2$ and the corresponding projector operator $\rho_{[x_1, x_2]}$ on $\mathbb{C}^3$. By easy computations, one can see that the Bloch vector of $\rho_{[x_1, x_2]}$ has null components:

$$\mathbf{b}^{(x_1, x_2)} = \frac{1}{1 + x_1^2 + x_2^2}\left[2x_1 x_2, 2x_1, 2x_2, 0, 0, 0, x_1^2 - x_2^2, \frac{x_1^2 + x_2^2 - 2}{\sqrt{3}}\right]. \tag{8}$$

Instead of using a matrix with nine real elements to represent $\rho_{[x_1, x_2]}$, memory occupation can be improved by considering only the non-zero components of the Bloch vector. In general, the technique of removing the components that are zero, or are repeated several times, allows reducing of the space and the calculation time, considering only the significant values that enable carrying out the classification. More precisely, the encoding of a real data vector $\mathbf{x}$ into the amplitudes of a pure state $\rho_\mathbf{x}$ enables an encoding of data into the corresponding Bloch vector $\mathbf{x} \mapsto \mathbf{b}^{(\mathbf{x})}$. As observed in the example given by (8), the Bloch vector can be reduced to a vector of lesser dimension reducing the zero components. In higher dimension, a reduced Bloch representation can be obtained by also removing the redundant components, as discussed below. In this sense, representing real vectors into Bloch vectors, by means of the amplitude encoding, represents efficient quantum encoding.

In general, the definition of a quantum encoding is equivalent to selection of a feature map for representing data vectors into a space of higher dimension. In this sense, data representation into quantum states can be considered a way to perform kernel tricks. In the case of the considered quantum encoding $\mathbb{R}^2 \ni [x_1, x_2] \mapsto \rho_{[x_1, x_2]} \in \mathfrak{S}(\mathbb{C}^3)$, in view of (8) the non-linear explicit injective function $\varphi : \mathbb{R}^2 \to \mathbb{R}^5$, to encode data into Bloch vectors can be defined as follows:

$$\varphi([x_1, x_2]) := \frac{1}{x_1^2 + x_2^2 + 1}\left[2x_1 x_2, 2x_1, 2x_2, x_1^2 - x_2^2, \frac{x_1^2 + x_2^2 - 2}{\sqrt{3}}\right]. \tag{9}$$

From a geometric point of view, the mapped feature vectors are points on the surface of a hyper-hemisphere. Within this representation, the centroids for the classification can be calculated as:

$$\bar{\mathbf{x}}_i^{(\varphi)} := \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \varphi(\mathbf{x}) \qquad i = 1, 2. \tag{10}$$

In general, such centroids are points inside the hypersphere that do not have an inverse image in terms of a density operator; however, they can be rescaled to Bloch vectors within the closed ball of radius $\frac{2}{d+1}$.

To improve the accuracy of the classification, one can increase the dimension of the representation space providing k copies of the quantum states, in terms of a tensor product, encoding data instances and centroids into "redundant" density matrices $\rho \otimes \cdots \otimes \rho$. According to the quantum formalism, multiple copies of the states are described in a tensor product Hilbert space, whose dimension scales exponentially, with a strong impact in terms of computational space (from dimension $d - 1$ to d^{2k}) and time. However, following the geometric approach, we can take advantage of the Bloch representation of real data to deal with tensor products in an efficient way.

Let us consider the quantum encoding $\mathbb{R}^2 \ni [x_1, x_2] \mapsto \rho_{[x_1, x_2]} \in \mathfrak{S}(\mathbb{C}^3)$ introduced above. Assuming that we need to represent the data vectors into a higher dimensional space, then we define the new encoding $\mathbb{R}^2 \ni [x_1, x_2] \mapsto \rho_{[x_1, x_2]} \otimes \rho_{[x_1, x_2]} \in \mathfrak{S}(\mathbb{C}^9)$, taking two copies of the density matrix $\rho_{[x_1, x_2]}$. Following the same argument applied to reduce the Bloch vector (8), we can remove the zero entries and the multiple entries from the Bloch vector of $\rho_{[x_1, x_2]} \otimes \rho_{[x_1, x_2]}$ to store and process a density matrix as a real vector of dimension 20 instead of a matrix of 81 elements (because there are 36 zero components and 24 duplicate values). Thus, the explicit function $\varphi : \mathbb{R}^2 \to \mathbb{R}^{20}$ for two copies of the density operators on $\mathbb{C}^3$ can be defined as follows:

$$\varphi([x_1, x_2]) := \frac{1}{(x_1^2 + x_2^2 + 1)^2} \Big[2x_1^3 x_2, 2x_1^3, 2x_1^2 x_2^2, 2x_1^2 x_2, 2x_1^2, 2x_1 x_2^3, 2x_1 x_2^2, 2x_1 x_2, 2x_1, 2x_2^3,$$

$$2x_2^2, 2x_2, x_1^2(x_1 - x_2)(x_1 + x_2), \frac{x_1^2(x_1^2 + x_2^2 - 2)}{\sqrt{3}}, \frac{x_1^2(x_1^2 - 2x_2^2 + 1)}{\sqrt{6}},$$

$$\frac{x_1^4 - 4x_2^4 + x_1^2(2x_2^2 + 1)}{\sqrt{10}}, \frac{x_1^2 + x_1^4 - 5x_2^2 + 2x_1^2 x_2^2 + x_2^4}{\sqrt{15}}, \frac{x_1^4 + x_2^2 + x_2^4 + x_1^2(2x_2^2 - 5)}{\sqrt{21}},$$

$$\frac{x_1^4 - 6x_2^2 + x_2^4 + 2x_1^2(x_2^2 + 1)}{2\sqrt{7}}, \frac{1}{6}(x_1^2 + x_2^2 - 2)(x_1^2 + x_2^2 + 4) \Big].$$

Similarly, we can store only 51 values instead of 729 for three copies (because there are 351 zero components and 326 duplicate values), and so on. To clarify the application of the efficient Bloch representation in the general case, let us stress that the positions of the entries that can be removed are known a priori. However, one must also take into account high-precision numbers and track the propagation of the numerical error.

The calculation of the centroids for the classification can be related to the quantum encoding in different ways. Consider the amplitude encoding (2) of a d-dimensional real feature vector $\mathbf{x}$ into the pure state $\rho_\mathbf{x}$, the centroids of the classes $C_1, \ldots, C_M$ of training points can be defined in the following alternative terms:

- Quantum centroid $\rho_i := \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \rho_\mathbf{x}$;
- Quantum encoding $\rho_{\bar{\mathbf{x}}_i}$ of the classical centroid $\bar{\mathbf{x}}_i := \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{x}$;
- Mean of the Bloch vectors $\mathbf{b}^{(i)} := \frac{1}{|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{b}^{(\mathbf{x})}$ that is not a Bloch vector in general;
- Contracted centroid $\widehat{\mathbf{b}}^{(i)} := \frac{2}{d+1} \mathbf{b}^{(i)}$ that is a Bloch vector itself.

In general, we have that $\rho_i \neq \rho_{\bar{\mathbf{x}}_i}$ and $\widehat{\mathbf{b}}^{(i)}$ is not the Bloch vector of ρ_i or $\rho_{\bar{\mathbf{x}}_i}$. In the following, we choose $\widehat{\mathbf{b}}^{(i)}$ as the definition of centroid in order to select the encoding that is

less memory consuming and to also represent the centroids as density matrices to perform a meaningful quantum state discrimination.

4. Local Pretty-Good Classifiers

In this section, we introduce the local approach to quantum-inspired classification. More precisely, we consider the execution of the classifiers based on quantum state discrimination described in Section 2 after selection of the training points. In particular, the local process in the feature space is based on the notion of Voronoi tessellation that we sketch in the following way: Let (X, d) be a metric space and $\{E_i\}_{i \in I}$ be a collection of non-empty subsets of X called *generators*. The *Voronoi cell V_i* is the set of points in X defined by:

$$V_i := \{x \in X \,:\, d(x, E_i) \leq d(x, E_j), \text{for all} j \neq i\}, \tag{11}$$

where $d(x, E_i) := \min_{y \in E_i} d(x, y)$ and $\{V_i\}_{i \in I}$ is called *Voronoi tessellation*. The *Vonoroi vertices* are the points that are equidistant to three or more generators. In our case, X is a real vector space, d is the Euclidean distance and the generators are single points.

Following the approach proposed in [5], the training data is partitioned into voxels for every class. For any non-empty voxel of a class, the mean of the contained data points is calculated. The voxel means are the initial tessellation representatives of the classes and are passed through an expectation-maximization process to evenly distribute the generator points over the class instances. Thus, possible degenerate vertices are removed, and the generator points move closer to the centroids of the Voronoi cells. The obtained tessellations are merged into a single tessellation combining the generator points of every class. In Algorithm 1, line 1 refers to this construction of the Voronoi tessellation over the training set. For any generator point, there is the selection of the k nearest neighbors (kNNs) w.r.t. the Euclidean distance (line 3) that are encoded into Bloch vectors using the representation described in Section 3 (line 4). Then the contracted centroid of any class is constructed out to form the considered k Bloch vectors (line 5). The corresponding n density operators can be used to define the pretty-good measurement, according to (7), obtaining a quantum state discrimination procedure for any cluster of the tessellation (line 6). Given an unlabeled point $\hat{\mathbf{x}}$, there is calculation of the h nearest generator points in the input space (line 8) which correspond to as many pretty-good measurements. The unlabeled point is encoded into a pure state $\rho_{\hat{\mathbf{x}}}$, according to the amplitude encoding (2). Then, each of the h pretty-good measurements is run to attach $\rho_{\hat{\mathbf{x}}}$ to the most likely class (line 11). Finally, the algorithm returns the label given by the majority voting.

Algorithm 1: Local pretty-good classifier hNNPGM.

> **Input:** Training set X, n classes C_i of training points, unlabeled point $\hat{\mathbf{x}}$
> **Result:** Label y of $\hat{\mathbf{x}}$
> find the generator points of the tessellation;
> **foreach** x_g *generator point* **do**
> > find the k nearest neighbors $\mathbf{x}_1, \ldots, \mathbf{x}_k$ to $\mathbf{x}_g$ in X w.r.t. the Euclidean distance;
> > encode $\mathbf{x}_1, \ldots, \mathbf{x}_k$ into Bloch vectors $\mathbf{b}^{(\mathbf{x}_1)}, \ldots, \mathbf{b}^{(\mathbf{x}_k)}$;
> > construct the contracted centroids $\hat{\mathbf{b}}^{(i)} = \frac{2}{(d+1)|C_i|} \sum_{\mathbf{x} \in C_i} \mathbf{b}^{(\mathbf{x})}$ and the corresponding density operators ρ_i;
> > determine the pretty-good measurement relative to the contracted centroids ρ_i.
> **end**
> find the h nearest generator points $\mathbf{x}_1, \ldots, \mathbf{x}_h$ to $\hat{\mathbf{x}}$ in X w.r.t. the Euclidean distance;
> encode $\hat{\mathbf{x}}$ into a pure state $\rho_{\mathbf{x}}$;
> **foreach** $j = 1, \ldots, h$ **do**
> > run the quantum-inspired classifier applying the pretty-good measurement to $\rho_{\mathbf{x}}$ and assigning the class with the highest probability
> **end**
> **return** *The label assignment is attributed on the basis of the majority*

The locality is imposed in a two-fold way by running the k-nearest neighbors on the input space, finding the training vectors that are closest to any generator point of the tessellation and by the h generator points nearest to the unlabeled point $\hat{x}$. Then, there is the quantum encoding into pure states and h quantum-inspired classifiers are parallel executed over the restricted training sets. The considered method is a combination of a kNN (and a hNN) algorithm to the quantum-inspired classifier based on the pretty-good measurement. Since the parallel execution of the h pretty-good measurements selected on the basis of the closest generator points to the test instance, let us denote this local quantum-inspired classifier as *hNNPGM*.

The Iris dataset is a well-known multi-class classification dataset that we considered for the experimental implementation of the hNNPGM classifier. The considered dataset is characterized by five features and three classes where two classes are linearly separable (for this reason the SVMs with non-linear kernel do not perform better than the SVM with linear kernel, as reported in Table 1). The dataset was randomly divided into a training set and a test set. The average accuracy is a useful index for comparison of our proposal to some popular classical methods. We considered the SVM with different kernels: linear, radial basis function, polynomial, and sigmoid. Then, we ran a random forest with 10 trees and maximum depth 5, a naive Bayes classifier, and the nearest neighbor algorithm. hNNPGM requires the choice of the hyperparameters k and h, as it is known from the standard kNN algorithm that there is no general strategy to choose k a priori and hyperparameter tuning must be performed. In Table 1, the results obtained for $k = 10$ and $h = 3$ for the average accuracy of the tested classifiers are shown. The value of h is low to reduce the parallel execution of the pretty-good classifiers, and the value of k is reasonable to construct the centroids of the three classes of the Iris dataset for any generator point of the tessellation. From the experiments, it was observed that the proposed local quantum-inspired classifier performed well against the classical competitor, having the best average accuracy.

Table 1. Classification comparison, in terms of test average accuracy for 10 runs, for Iris dataset with hNNPGM hyperparameters $k = 10$ and $h = 3$.

hNNPGM	Linear	RadialBasisFunction	Polynomial	Sigmoid	RandomForest	NaiveBayes	NearestNeighbors
0.973 ± 0.009	0.903 ± 0.017	0.893 ± 0.017	0.903 ± 0.017	0.887 ± 0.018	0.913 ± 0.016	0.903 ± 0.017	0.927 ± 0.015

5. Conclusions

The present paper focuses on some methods of quantum-inspired machine learning, in particular, classification algorithms based on quantum state discrimination. We adopted a geometric formulation in defining quantum encodings of classical data in terms of Bloch vectors of density operators, as in previous work [4], which is a crucial procedure to save computational resources when defining feature maps into high dimensional spaces, considering multiple copies of the encoding quantum states. A novel contribution of the present paper is the local approach adopted to execute the classifier, not over the entire training set, but in a neighborhood of the test point after separate voxelization of the data classes, and the subsequent construction of a Vonoroi tessellation. Once partitioned, for the training set, for any generator point of the tessellation, the k nearest data points are encoded into Bloch vectors and used to define the quantum centroid of each class as a contracted Bloch vector. Then a pretty-good measurement is constructed from the obtained centroids for any generator point. Finally, a selection is made of the h nearest generator points to the test data instance to classify them by h pretty-good classifiers in parallel. The final label is chosen by majority voting. The local quantum-inspired classifier considered, hNNPGM, for reasonable values of the hyperparameters, was implemented and was found to be a method with performance comparable to well-known classical algorithms for multi-class classification. We performed some experiments using the Iris dataset and found that hNNPGM was even more accurate than an SVM with different kernels, a random forest, a naive Bayes classifier and the NN classification algorithm. We do not consider the obtained

results to be disruptive but, more cautiously, as evidence that local quantum-inspired classifiers can be considered to be an interesting kind of classification algorithm that can be investigated from the point of view of foundations and applications.

The present proposal, based on a local approach to quantum-inspired classification, offers a family of classifiers, rather than a single classification algorithm. In fact, several strategies to impose a notion of locality over a training set, and several procedures of quantum state discrimination, can be applied. Both the local approach to classification and the quantum-inspired data encoding/processing deserve careful scientific investigation to clarify the impact of these ideas on machine learning. In a future paper, we will present some numerical results obtained from the implementation of these local quantum-inspired classifiers.

References

1. Rebentrost, P.; Mohseni, M.; Lloyd, S. Quantum support vector machine for big data classification. *Phys. Rev. Lett.* **2014**, *113*, 130503.
2. Sheng, Y.-B.; Zhou, L. Distributed secure quantum machine learning. *Sci. Bull.* **2017**, *62*, 1025.
3. Schuld, M.; Fingerhuth, M.; Petruccione, F. Implementing a distance-based classifier with a quantum interference circuit. *EPL* **2017**, *119*, 60002.
4. Leporini, R.; Pastorello, D. An efficient geometric approach to quantum-inspired classifications. *Sci. Rep.* **2022**, *12*, 8781. https://doi.org/10.1038/s41598-022-12392-1.
5. Zengin, R.S.; Sezer, V. Super-k: A Piecewise Linear Classifier Based on Voronoi Tessellations. *arXiv* **2021**, arXiv:2012.15492.
6. Blanzieri, E.; Melgani, F. An Adaptive SVM Nearest Neighbor Classifier for Remotely Sensed Imagery. In Proceedings of the IEEE International Conference on Geoscience and Remote Sensing Symposium, Denver, CO, USA, 31 July–4 August 2006; pp. 3931–3934.
7. Alfaro, C.; Gomez, J.; Moguerza, J.M.; Castillo, J.; Martinez, J.I. Toward Accelerated Training of Parallel Support Vector Machines Based on Voronoi Diagrams. *Entropy* **2021** ,*23*, 1605. https://doi.org/10.3390/e23121605.
8. Helstrom, C.W. Quantum detection and estimation theory. *J. Stat. Phys.* **1969**, *1*, 231–252.
9. Bae, J.; Kwek, L.C. Quantum state discrimination and its applications. *J. Phys. A* **2015**, *48*, 083001.
10. Kimura, G.; Kossakowski, A. The Bloch-Vector Space for N-Level Systems: The Spherical-Coordinate Point of View. *Open Syst. Inf. Dyn.* **2005**, *12*, 207–229. https://doi.org/10.1007/s11080-005-0919-y.
11. Giuntini, R.; Freytes, H.; Park, D.K.; Blank, C.; Holik, F.; Chow, K.L.; Sergioli, G. Quantum state discrimination for supervised classification. *arXiv* **2021**, arXiv:2104.00971.
12. Mochon, C. Family of generalized pretty good measurements and the minimal-error pure-state discrimination problems for which they are optimal. *Phys. Rev. A* **2006**, *73*, 032328. https://doi.org/10.1103/PhysRevA.73.032328.
13. Bae, J. Structure of minimum-error quantum state discrimination. *New J. Phys.* **2013**, *15*, 073037. https://doi.org/10.1088/1367-2630/15/7/073037.
14. Leporini, R.; Pastorello, D. Support Vector Machines with Quantum State Discrimination. *Quantum Rep.* **2021**, *3*, 482–499. https://doi.org/10.3390/quantum3030032.