Int J Theor Phys
DOI 10.1007/s10773-017-3371-1

A Quantum-inspired Version of the Classification Problem

Giuseppe Sergioli[1] · Gustavo Martin Bosyk[2] ·
Enrica Santucci[1] · Roberto Giuntini[1]

Received: 31 December 2016 / Accepted: 1 April 2017

Abstract We address the problem of binary classification by using a quantum version of the Nearest Mean Classifier (NMC). Our proposal is indeed an advanced version of previous one (see Sergioli et al. 2017 that *i)* is able to be naturally generalized to arbitrary number of features and *ii)* exhibits better performances with respect to the classical NMC for several datasets. Further, we show that the quantum version of NMC is not invariant under rescaling. This allows us to introduce a free parameter, *i.e.* the rescaling factor, that could be useful to get a further improvement of the classification performance.

Keywords Nearest mean classifier · Density operators · Rescaling invariance

1 Introduction

Quantum mechanics is a probabilistic theory that turns out to be successfully predictive with respect to a wide range of microscopic phenomena. Given an arbitrary state and a certain perturbation, quantum mechanics makes a probability distribution to obtain a possible

✉ Giuseppe Sergioli
giuseppe.sergioli@gmail.com

Gustavo Martin Bosyk
gbosyk@fisica.unlp.edu.ar

Enrica Santucci
enrica.santucci@gmail.com

Roberto Giuntini
giuntini@unica.it

[1] University of Cagliari, Via Is Mirrionis 1, I-09123 Cagliari, Italy

[2] Instituto de Física La Plata, UNLP, CONICET, Facultad de Ciencias Exactas, 1900,
La Plata, Argentina

Published online: 08 April 2017

value of a given observable. The quantum mechanical formalism is particularly suitable to describe different kinds of stochastic processes, that - in principle - can also include non-microscopic domain. In recent years a number of non standard applications of the quantum formalism has arisen. For example, the quantum formalism has been widely used in game theory [7, 16], in the economic processes [17] and in cognitive sciences. Concerning this domain, Aerts et al. [1, 2, 4, 5] have shown some non-trivial analogies between the mechanisms of human behavior and the ones of the microscopic systems.

By this perspective, another non-standard application of quantum theory is devoted to apply it for solving classification problems. The basic idea is to represent classical patterns in terms of quantum objects, with the aim to boost the computational efficiency of the classification algorithms. In the last few years many efforts have been made to apply the quantum formalism to signal processing [8] and to pattern recognition [18, 19]. Exhaustive surveys concerning the applications of quantum computing in computational intelligence are provided in [14, 22]. Even if these approaches suggest possible computational advantages of this sort [12, 13], what we propose here is a different approach that consists in using quantum formalism in order to reach some benefit in classical computation. Indeed, this have already addressed in a recent works [11, 20], where a "quantum counterpart" of a the Nearest Mean Classifier (NMC) has been proposed. This proposal has been constructed on the following basis: *i)* first, an encoding of an arbitrary two-feature pattern into a density operator on the Bloch sphere has been presented; *ii)* then, a concept of "quantum centroid" that plays a similar role as the classical centroid in the NMC has been introduced; *iii)* finally, a distance measure between density operators that plays a similar role as the Euclidean distance in the NMC has been proposed.

It has been shown (in [20]) that this quantum counterpart of the NMC - named *quantum NMC* (QNMC for short) - leads to significant improvements for several 2-feature datasets.

In this paper we propose an alternative version of the QNMC, where another encoding of real vectors (n-feature patterns) into density operators is involved. The problem to encode real vectors by quantum objects is not trivial and it could turn out to be promising for the whole theory of machine learning. Quoting Petruccione: "*In order to use the strengths of quantum mechanics without being confined by classical ideas of data encoding, finding 'genuinely quantum' ways of representing and extracting information could become vital for the future of quantum machine learning*" ([21], pg.6).

We will show how the new proposed encoding leads to two relevant advantages: *i)*, unlike the old encoding, it allows us to extend the classification model to n-feature patterns in a very natural way; *ii)* it improves the efficiency of the QNMC. In the final part of the paper we also focus on a very considerable difference between the NMC and the QNMC: unlike the NMC, the QNMC is not invariant under rescaling. More precisely, the accuracy of the QNMC changes by rescaling (of an arbitrary real number) the coordinates of the dataset. This seeming shortcoming turns out to be partially beneficial to the classification process.

The paper is organized as follows: in the first Section we briefly describe the formal structure of the Nearest Mean Classifier. In the second Section we introduce the quantum version of the NMC and in the third Section we compare the NMC and the QNMC on different datasets, showing that in most of the cases the QNMC exhibits a better performance (in terms of accuracy) with respect to the NMC. In the fourth Section we show that, differently from the NMC, the QNMC is not invariant under rescaling. Starting with what, apparently, is a problem we end up that this non-invariance could even be slightly beneficial. Some conclusions and possible further developments are proposed at the end of the paper.

2 The Classification Problem and the Nearest Mean Classifier

Here, we address the problem of classification, which is an instance of supervised learning, *i.e.* learning from a training dataset of correctly labeled objects. More precisely, each object can be characterized by its features; hence, a d-feature object can be naturally represented by a d-dimensional real vector, *i.e.* $\mathbf{x} = \left[x^1, \dots, x^d\right] \in \mathbb{R}^d$.[1] Formally, a pattern can be represented as a pair $(\mathbf{x}_i, \lambda_i)$ where $\mathbf{x}$ is the d-dimension vector associated to the object and λ_i is the label that refers to the class which the object belogs to. Let $\mathcal{L} = \{\lambda_1, \dots, \lambda_N\}$ be the set of labels that are in correspondence with the respective classes. The goal of the classification process is to design a *classifier* that attributes (in the most accurate way) to any unlabled object a label (class). In supervised learning, such a classifier is obtained by learning from a training dataset $\mathcal{S}_{\text{tr}}$, *i.e.* a set $\mathcal{S}_{\text{tr}}$ of patterns (correctly labelled objects). Formally:

$$\mathcal{S}_{\text{tr}} = \{(\mathbf{x}_1, \lambda_1), \dots, (\mathbf{x}_M, \lambda_M)\},$$

where $\mathbf{x}_i \in \mathbb{R}^d$ and λ_i is the label associated to its class.

Hereafter, we will deal with binary classification (*i.e.* $N = 2$) and we will assume $\lambda_m \in \{+, -\}, \forall m \in \{1, \dots, M\}$. Given a training dataset $\mathcal{S}_{\text{tr}} = \{(\mathbf{x}_1, \lambda_1), \dots, (\mathbf{x}_M, \lambda_M)\}$ we can define the *positive class* $\mathcal{S}_{\text{tr}}^+$ and the *negative class* $\mathcal{S}_{\text{tr}}^-$ of $\mathcal{S}_{\text{tr}}$ in the following way:

$$\mathcal{S}_{\text{tr}}^+ = \{(\mathbf{x}_i, \lambda_i) \in \mathcal{S}_{\text{tr}} : \lambda_i = +\} \text{ and } \mathcal{S}_{\text{tr}}^- = \{(\mathbf{x}_i, \lambda_i) \in \mathcal{S}_{\text{tr}} : \lambda_i = -\}.$$

$\mathcal{S}_{\text{tr}}^+$ ($\mathcal{S}_{\text{tr}}^-$, respectively) is the set of all the patterns of the training dataset which belong to the class labeled by $+$ ($-$, respectively).

By M_+ (M_-, respectively) we will denote the number of elements of $\mathcal{S}_{\text{tr}}^+$ ($\mathcal{S}_{\text{tr}}^-$, respectively). Clearly, $M_+ + M_- = M$.

One of the simplest classification method in pattern recognition is the so called *Nearest Mean Classifier*. The NMC algorithm consists in the following steps:

1. Training: one has to compute the *centroid* for each class, that is,

$$\mathbf{x}^+ = \frac{1}{M_+} \sum_{i \in \{m : \lambda_m = +\}} \mathbf{x}_i \text{ and } \mathbf{x}^- = \frac{1}{M_+} \sum_{i \in \{m : \lambda_m = -\}} \mathbf{x}_i \tag{1}$$

2. Classification $Cl : \mathbb{R}^d \to \{+, -\}$ such that $\forall \mathbf{x} \in \mathbb{R}^d$:

$$Cl(\mathbf{x}) = \begin{cases} + & \text{if } d(\mathbf{x}, \mathbf{x}^+) \leq d(\mathbf{x}, \mathbf{x}^-) \\ - & \text{otherwise,} \end{cases}$$

   where $d(\mathbf{x}, \mathbf{y}) = |\mathbf{x} - \mathbf{y}|$ is the Euclidean distance.

Intiutively, the classifier associates to a d-feature object $\mathbf{x}$ the label of the closest centroid.

As a remark, it is worth noting that, depending on the particular distribution of the objects of the dataset, it is possible that a pattern belonging to a given class is uncorrectly classified. For an arbitrary pattern $(\mathbf{x}_i, \lambda_i)$, four cases are possible:

- $\mathbf{x}_i$ is a *true positive* (TP) object: the pattern $(\mathbf{x}_i, \lambda_i)$ belongs to the positive class $\mathcal{S}_{\text{tr}}^+$, and it is correctly classified, *i.e.*, $Cl(\mathbf{x}_i) = +$;

[1] For the sake of the clarity regarding the indexes, we accord to use superscript index to indicate the different components of the vector and subscript to indicate different vectors.

- $\mathbf{x}_i$ is a *true negative* (TN) object: the pattern $(\mathbf{x}_i, \lambda_i)$ belongs to the negative class $\mathcal{S}_{\text{tr}}^{-}$, and it is correctly classified, *i.e.*, $Cl(\mathbf{x}_i) = -$;
- $\mathbf{x}_i$ is a *false positive* (FP) object: the pattern $(\mathbf{x}_i, \lambda_i)$ belongs to negative class $\mathcal{S}_{\text{tr}}^{-}$, but it is uncorrectly classified, *i.e.*, $Cl(\mathbf{x}_i) = +$;
- $\mathbf{x}_i$ is a *false negative* (FN) object: the pattern$(\mathbf{x}_i, \lambda_i)$ belongs to positive class $\mathcal{S}_{\text{tr}}^{+}$, but it is uncorrectly classified, *i.e.*, $Cl(\mathbf{x}_i) = -$;

In order to evaluate the performance of NMC, one introduces another set of patterns (called *test dataset*) that does not belong to the training dataset [6]. Formally, the test dataset is a set $\mathcal{S}_{\text{ts}} = \{\{\mathbf{y}_1, \beta_1\}, \dots, \{\mathbf{y}_{M'}, \beta_{M'}\}\}$, such that $\mathcal{S}_{\text{tr}} \cap \mathcal{S}_{\text{ts}} = \emptyset$, where M'_+ is the number of test patterns belonging to the positive class, M'_- is the number of test patterns belonging to the negative class and $M'_+ + M'_- = M'$.

Then, by applying the NMC to the test set, it is possible to evaluate the performance of the classifier by considering the following statistical measures:

- True positive rate (TPR): $TPR = \frac{\#TP}{M'_+}$;
- True negative rate (TNR): $TNR = \frac{\#TN}{M'_-}$;
- False positive rate (FPR): $FPR = \frac{\#FP}{M'_-} = 1 - TPN$;
- False negative rate (FNR): $FNR = \frac{\#FN}{M'_+} = 1 - TPR$;
- Accuracy (ACC): $ACC = \frac{\#TP + \#TN}{M'}$.

Let us notice that the values of these quantities are obviously related to the training/test datasets; as a natural consequence, also the classifier performance is strictly dataset-dependent.

3 Quantum Version of the Nearest Mean Classifier

This Section is devoted to provide a quantum-inspired version of the NMC described above, that we will call *Quantum Nearest Mean Classifier* (QNMC). This representation not only could lead to exploit the well known advantages related to the quantum computation with respect to the classical one (mostly related to the speed up of the computation), but also allows us to make a full comparison between the performance of both classifiers (the NMC and the QNMC) by using a classical computer only.

In order to obtain this representation, we need to fulfill the following steps:

1. for each pattern, one has to provide a suitable encoding into a quantum object (*i.e.* a density operator) that we call *density pattern*;
2. for each class of density patterns, one has to define the quantum conterpart of the classical centroid, that we call *quantum centroid*;
3. one has to provide a suitable definition of *quantum distance* between density patterns, that plays a similar role as the Euclidean distance for the NMC.

On this basis, it is now possible to define a QNMC. Indeed, this framework has been already followed by Sergioli et al. [20] in order to classify two-feature patterns ($d = 2$). The classifier we propose here offers two relevant advantages: *i)* it allows us to work with the general case of d-feature patterns; *ii)* it provides further improvement of the performance, as we will show by applying this for several datasets.

Let us notice that there are infinite many ways to encode a real vector into a density pattern and the convenience of using one instead of others could be strictly dataset-dependent. Even though this point would deserve further investigation, here we propose the following encoding which is easy to compute and which gives us a quite natural way to deal with general d-feature objects.

In order to introduce our new encoding it will be expedient to recall the notion of *stereographic projection* as follows. Let $\tilde{\mathbf{x}} = \left[\tilde{x}^1, \ldots, \tilde{x}^{d+1}\right]$ be an arbitrary $(d+1)$-feature object of $\mathbb{R}^{d+1}$. The stereographic projection SP is a map $SP : \mathbb{R}^{d+1} \to \mathbb{R}^d$ such that:

$$\mathbf{x} = SP(\tilde{x}^1, \tilde{x}^2, ..., \tilde{x}^{d+1}) = \left(\frac{\tilde{x}^1}{1-\tilde{x}^{d+1}}, ..., \frac{\tilde{x}^d}{1-\tilde{x}^{d+1}}\right).$$

The inverse of the stereographic projection can be defined as follows. Let $\mathbf{x} = \left[x^1, \ldots, x^d\right]$ be an arbitrary d-feature object of $\mathbb{R}^d$; the inverse of the stereographic projection SP^{-1}, is a map $SP^{-1} : \mathbb{R}^d \to R^{d+1}$ such that:

$$\tilde{\mathbf{x}} = SP^{-1}(\mathbf{x}) = \frac{1}{\sum_{i=1}^d (x^i)^2 + 1}\left[2x^1, \ldots, 2x^d, \sum_{i=1}^d (x^i)^2 - 1\right], \tag{2}$$

where $\frac{1}{\sum_{i=1}^d (x^i)^2+1}$ is a normalization factor.

Definition 1 (Density pattern) The density pattern $\rho_{\mathbf{x}}$ associated to the d-feature object $\mathbf{x} \in \mathbb{R}^d$ is defined as:

$$\rho_{\mathbf{x}} \doteq \tilde{\mathbf{x}}^t \tilde{\mathbf{x}}. \tag{3}$$

Clearly, every density pattern is a quantum pure state, *i.e.* $\rho_{\mathbf{x}}^2 = \rho_{\mathbf{x}}$.

By using the above encoding, we define the quantum training dataset

$$\mathcal{S}_{\mathrm{tr}}^{\mathrm{q}} = \{\{\rho_{\mathbf{x}_1}, \lambda_1\}, \ldots, \{\rho_{\mathbf{x}_M}, \lambda_M\}\}$$

as the set of all the density patterns obtained by encoding all the elements of $\mathcal{S}_{\mathrm{tr}}$.

This fact allows us to introduce the quantum versions of the standard centroid given in (1), as following.

Definition 2 (Quantum centroids) Let $\mathcal{S}_{\mathrm{tr}}^{\mathrm{q}} = \left\{\{\rho_{\mathbf{x}_1}, \alpha_1\}, \ldots, \{\rho_{\mathbf{x}_M}, \alpha_M\}\right\}$ be a quantum training dataset of density patterns. The quantum centroids for the positive and negative class are respectively given by:

$$\rho_+ = \frac{1}{M_+} \sum_{i\in\{m:\lambda=+\}} \rho_{\mathbf{x}_i} \text{ and } \rho_- = \frac{1}{M_-} \sum_{i\in\{m:\lambda=-\}} \rho_{\mathbf{x}_i}. \tag{4}$$

Notice that the quantum centroids are now mixed states and they are not generally obtained by the encoding of the respective classical centroids $\mathbf{x}_\pm$. In other terms $\rho_\pm \neq \rho_{\mathbf{x}_\pm}$. Accordingly, the definition of the quantum centroid leads to a new object that does not have any classical counterpart. As we will see, this is the main reason of the deep difference between NMC and QNMC.

In order to introduce a suitable definition of distance between density patterns, we can recall the well known distance between quantum states that is commonly used in quantum computation (see, *e.g.* [15]).

Definition 3 (Trace distance) Let ρ and σ be two quantum density operators belonging to the same Hilbert space. The trace distance (d_{tr}) between ρ and σ is given by:

$$d_{\mathrm{tr}}(\rho, \sigma) = \frac{1}{2} \operatorname{Tr} |\rho - \sigma|, \tag{5}$$

where $|A| = \sqrt{A^{\dagger} A}$.

Notice that the trace distance is a metric. In other words it satisfies: *i)* $d_{\mathrm{tr}}(\rho, \sigma) \geq 0$ with equality iff $\rho = \sigma$ (positivity), *ii)* $d_{\mathrm{tr}}(\rho, \sigma) = d_{\mathrm{tr}}(\sigma, \rho)$ (symmetry) and *iii)* $d_{\mathrm{tr}}(\rho, \omega) + d_{\mathrm{tr}}(\omega, \sigma) \geq d_{\mathrm{tr}}(\rho, \sigma)$ (triangle inequality).

The choice of using the definition of quantum centroids and the trace distance introduced above are not completely arbitrary. They naturally arise in a related classification problem, which consists in classifying an unknown density pattern randomly extracted from the quantum training dataset. Indeed, it has been shown that this problem is equivalent to the distinguishability problem between the quantum centroids ρ_+ and ρ_- [3, 9]. It is a well-known result in quantum estimation and detection theory that the probability of distinguishing between two density operators by optimal quantum process is upper bounded by $\frac{1+d_{\mathrm{tr}}(\omega_+\rho_+, \omega_-\rho_-)}{2}$, where $\omega_{\pm} = \frac{M'_{\pm}}{M'}$ are the a-priori probabilities of each class [10].

We have introduced all the necessary ingredients to describe in details the QNMC process, which, similarly to the classical case, consists in the following steps:

- obtaining the quantum training dataset $\mathcal{S}^{\mathrm{q}}_{\mathrm{tr}}$ by applying the encoding given in Definition 1 to each pattern of the classical training dataset $\mathcal{S}_{\mathrm{tr}}$;
- calculating the quantum centroids $\rho_{\pm}$ according to Definition 2 ;
- classifying an arbitrary pattern $\mathbf{x}$ accordingly with the minimization problem: the quantum classifier is a function $QCl : \mathbb{R}^d \to \{+, -\}$ such that $\forall \mathbf{x} \in \mathbb{R}^d$:

$$QCl(\rho_{\mathbf{x}}) = \begin{cases} + & \text{if } d_{\mathrm{tr}}(\rho_{\mathbf{x}}, \rho_{\mathbf{x}^+}) \leq d_{\mathrm{tr}}(\rho_{\mathbf{x}}, \rho_{\mathbf{x}^-}) \\ - & \text{otherwise.} \end{cases}$$

4 Experimental Results

In this Section we provide a comparison between NMC and QNMC by applying (obviously, in a classical computer) both classifiers to different datasets. In particular, we consider three artificial (two-feature) datasets (Moon, Banana, and Gaussian) and four real (many-feature) datasets (Diabetes, Cancer, Liver and Ionosphere) extracted from the UC Irvine Machine Learning Repository.

In our experiment, we follow the standard methodology of randomly splitting each dataset in training and test datasets with %80 and %20 of the total patterns, respectively. Moreover, in order to obtain statistical significance results, we carry out 100 experiments for each dataset, where the splitting is randomly taken each time.

We summarize our results in the Table 1.

Let us notice that the ACC of the QNMC is significantly greater than the ACC of the NMC for all the datsets, except for the Cancer dataset. In particular, this improvement is even greater for the 2-feature datasets. The Cancer dataset deserves a particular attention: although we do not obtain an improvement on the ACC with respect to the NMC, surprisingly enough the QNMC is able to perfectly detect malignant tumors (without any classification error).

Table 1 Average results for NMC and QNMC classifiers (in %) and their standard deviations

Datasets		NMC			QNMC		
Name	$\#\mathcal{S}_{\text{tr}}/\#\mathcal{S}_{\text{ts}}$	ACC	TPR	TNR	ACC	TPR	TNR
Banana	4240/1060(2)	55.0 ± 1.8	57.1 ± 2.5	53.4 ± 2.2	**71.0 ± 1.2**	**70.5 ± 1.6**	**71.4 ± 1.8**
Gaussian	160/40(2)	55.5 ± 7.7	61.4 ± 10.7	49.7 ± 11.4	**76.2 ± 5.6**	**70.9 ± 8.1**	**81.8 ± 8.7**
Moon	160/40(2)	77.9±5.7	78.7 ± 9.2	76.9 ± 9.2	**88.9 ± 4.4**	**94.3 ± 5.8**	**83.4 ± 7.9**
Diabetes	614/154(8)	63.4 ±3.9	**73.5 ± 4.6**	44.9 ± 7.2	**68.7 ± 3.2**	69.5 ± 4.1	**67.1 ± 5.1**
Cancer	546/137 (10)	**96.4 ± 1.4**	**97.9 ± 1.3**	93.5 ± 3.0	93.7 ± 1.9	90.4 ± 2.9	**100.0 ± 0.0**
Liver	463/116(10)	53.8 ± 4.2	39.3 ± 5.9	**89.3 ± 5.6**	**64.3 ± 3.5**	**59.6 ± 4.2**	75.9 ± 6.9
Ionosphere	280/71(34)	72.9 ± 4.5	**74.0 ± 7.6**	72.3 ± 6.4	**83.7 ± 4.3**	58.6 ± 9.9	**98.4 ± 1.6**

$\#\mathcal{S}_{\text{tr}}$ cardinality of the training dataset; $\#\mathcal{S}_{\text{ts}}$ $=$ cardinality of the test set; (d) $=$ number of features of each element of its respective dataset

The entries with bold emphasizes the classifier (NMC or NMC) with the best statistical measure (ACC, TPR and TNR)

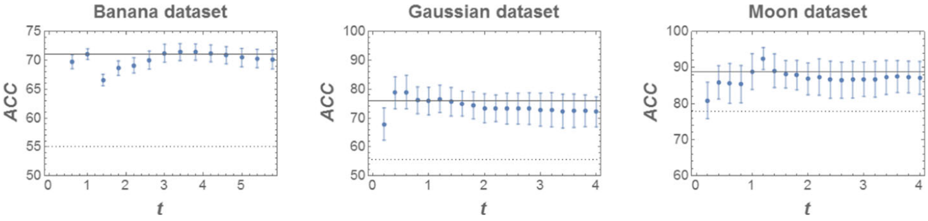
5 The Rescaling Factor

A key difference between NMC and QNMC regards the invariance under rescaling. Let us suppose that each pattern of the training and test sets is multiplied by the same *rescaling factor* t, *i.e.* $\mathbf{x}_m \mapsto t\mathbf{x}_m$ and $\mathbf{y}_{m'} \mapsto t\mathbf{y}_{m'}$ for any m and m'. Then, the (classical) centroids change according to $\mathbf{x}_+ \mapsto t\mathbf{x}_+$ and $\mathbf{x}_- \mapsto t\mathbf{x}_-$ and the classification problem of each pattern of the rescaled test set becomes

$$\arg\min_{i\in\{+,-\}} d(t\mathbf{y}_{m'}, t\mathbf{x}_i) = \sqrt{t}\,\arg\min_{i\in\{+,-\}} d(\mathbf{y}_{m'}, \mathbf{x}_i), \tag{6}$$

which has the same solution of the unrescaled problem, *i.e.* $t = 1$.

On the contrary, the QNMC turns out to be not invariant under rescaling. Far from being a disadvantage, this allows us to introduce a "free" parameter, *i.e.* the *rescaling factor*, that could be useful to get a further improvement of the classification performance as it is shown in Fig. 1 for the 2-feature datasets. The pictures in Fig. 1 represent the experimental results where we repeated the same experiments described in the previous Section by rescaling (within a small range) the coordinates of the initial dataset. The picture of the Fig. 1 shows


Banana dataset

Gaussian dataset

Moon dataset

Fig. 1 Accuracy vs rescaling factor for Banana (left), Gaussian (medium) and Moon (right) datasets. For a fixed t, each point corresponds to an average of the accuracy under 20 experiments and its standard deviation is represented by the corresponding error bar. In each plot, we fix the y-axis to the case $t = 1$ (no rescaling). The dotted line represents the average accuracy reached by the NMC

that for each dataset there is an interval (I_t) of the rescaling factor t such that for any $t \in I_t$ the average values of the accuracy are slightly greater than the accuracy values of the respective unrescaled cases. Unfortunately, we do not find a significative improvement with the other datasets.

Even though, in principle, it is not possible to establish an *a priori* superiority of QNMC with respect to the NMC according to the well known *No Free Lunch Theorem* [6], the method we introduced allows us to get some relevant improvements of classification when we have an *a priori* knowledge about the distribution of the dataset we have to deal with. In other words, if we need to classify an unknown pattern, looking at the distribution of the training dataset, we can know (before the classification) if: *i)* for that kind of distribution the QNMC performs better than the NMC and *ii)* what is the suitable rescaling has to be applied to the original dataset in order to get a further improvement of the accuracy.

6 Concluding remarks

In this paper we introduced a quantum-inspired version of the Nearest Mean Classifier, obtained as a quantum counterpart of the three fundamental ingredients of the NMC: *i)* a definition of quantum pattern, *ii)* a definition of quantum centroid, *iii)* a distance between quantum patterns. Unlike the proposal presented in [20], this new version of the QNMC can be easily generalized to d-feature patterns and provides more relevant improvements of the accuracy. Anyway, this investigation is far to be concluded. A further development will be focused on finding out the encoding (from real vectors to density operators) that guarantees the *optimal* improvement (at least for a finite class of datasets) in terms of the classification process accuracy. On this basis, it will be useful to compare the *optimal* QNMC also with other commonly used classical classifiers.

Let us finally remark that, in general, the set of all correctly classified patterns obtained by the NMC is not a subset of the set of all correctly classified patterns obtained by the QNMC. On this basis, it could be interesting to investigate a suitable method to mix both classifiers in order to reach a further improvement of the classification accuracy.

Acknowledgements This work is supported by the Sardinia Region Project ”Modeling the uncertainty: quantum theory and imaging processing”, LR 7/8/2007. RAS CRP-59872 and by the Firb Project ”Structures and Dynamics of Knowledge and Cognition” [F21J12000140001]. GMB acknowledges support from CONICET and UNLP (Argentina).

References

1. Aerts, D., Sozzo, S.: Quantum structure in cognition: Why and how concepts are entangled. Lect. Notes Comput. Sci **7052**, 116–127 (2011)
2. Aerts, D., Sozzo, S., Gabora, L., Veloz, T.: Quantum Structure in Cognition: Fundamentals and Applications. In: Privman, V., Ovchinnikov, V. (eds.) ICQNM 2011: The Fifth International Conference on Quantum, Nano and Micro Technologies (2011)
3. Aïmeur, E., Brassard, G., Gambs, S.: Machine learning in a quantum world. In: Conference of the Canadian Society for Computational Studies of Intelligence Springer Berlin Heidelberg (2006)
4. Dalla Chiara, M.L., Giuntini, R., Leporini, R., Sergioli, G.: Holistic logical arguments in quantum computation. Mathematica Slovaca **2**, 66 (2016)
5. Dalla Chiara, M.L., Giuntini, R., Leporini, R., Negri, E., Sergioli, G.: Quantum information, cognition and music. Front. Psychol. **6**, 1583 (2015)

6. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley Interscience, 2nd edition (2000)
7. Eisert, J., Wilkens, M., Lewenstein, M.: Quantum games and quantum strategies. Phys. Rev. Lett. **83**(15), 3077 (1999)
8. Eldar, Y.C., Oppenheim, A.V.: Quantum signal processing. IEEE Signal Process. Mag. **19**(6), 12–32 (2002)
9. Gambs, S.: Quantum classification, arXiv:0809.0444v2 [quant-ph] (2008)
10. Helstrom, C.W.: Quantum Detection and Estimation Theory, Academic Press (1976)
11. Holik, F., Sergioli, G., Freytes, H., Plastino, A.: Pattern Recognition in non-Kolmogorovian Structures Foundations of Science (2017)
12. Lloyd, S., Mohseni, M., Rebentrost, P.: Quantum algorithms for supervised and unsupervised machine learning. arXiv:1307.0411[quant-ph] (2013)
13. Lloyd, S., Mohseni, M., Rebentrost, P.: Quantum principal component analysis. Nat. Phys. **10**(9), 631–633 (2014)
14. Manju, A., Nigam, M.J.: Applications of quantum inspired computational intelligence: a survey. Artif. Intell. Rev. **42**(1), 79–156 (2014)
15. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information - 10th Anniversary Edition. Cambridge university press, Cambridge (2010)
16. Piotrowski, E.W., Sladkowski, J.: An invitation to quantum game theory. Int. J. Theor. Phys. **42**(5), 1089–1099 (2003)
17. QP-PQ: Quantum Probability and White Noise Analysis: Volume 21. Quantum Bio-Informatics II, From Quantum Information to Bio-Informatics, World Scientific (2008)
18. Schuld, M., Sinayskiy, I., Petruccione, F.: An introduction to quantum machine learning. Contemp. Phys. **56**(2), 172–185 (2014)
19. Schuld, M., Sinayskiy, I., Petruccione, F.: The quest for a Quantum Neural Network. Quantum Inf. Process **13**(11), 2567–2586 (2014)
20. Sergioli, G., Santucci, E., Didaci, L., Miszczak, J.A., Giuntini, R.: A quantum inspired version of the NMC classifier. Soft Computing (forthcoming) (2017)
21. Schuld, M., Sinayskiy, I., Petruccione, F.: An introduction to quantum machine learning. Contemp. Phys. **56**(2) (2014). arXiv:1409.3097
22. Wittek, P.: Quantum machine learning: What quantum computing means to data mining academic press (2014)