Analysis and Design of Optimized Quantum Feature Maps for Hybrid Kernel Methods through Kullback-Leibler Divergence, Quantum Relative Entropy and Quantum Metrics

***Abstract*—A B S T R A C T**

***Index Terms*—Quantum embeddings, Quantum Kernels, Quantum machine learning.**

I. INTRODUCTION

Machine learning, a field of study within artificial intelligence, involves automatically identifying patterns in observed data to infer or predict and generalize to unseen data [13]. Given a dataset that we assume can be read by a computer and represented adequately in numerical format in an appropriate vector representation [15], machine learning utilizes algorithms that iteratively learn from problem-specific training data. This enables computers to uncover hidden insights and intricate patterns without explicit programming [13], [14]. Particularly, supervised learning considers data with discrete labels to design algorithms for classification tasks. In this context, given a dataset $\mathcal{D} = \mathcal{X} \times \mathcal{Y}$, we assume a model:

$$\mathcal{Y} = f(\mathcal{X}, \theta) + \epsilon$$

Here, the algorithm is encoded in the function f, θ represents its parameters, $\mathcal{X}$ is the input set, $\mathcal{Y}$ is the discrete output set, and ϵ is the irreducible error [12]. The specific form of our model f, the values of the θ parameters, and model hyperparameters are obtained after training on the test data. This process allows accurate classification of new inputs in a test set, known as generalization [14]. In recent decades, there have been significant advancements in machine learning, particularly with the evolution of artificial neural networks into more intricate structures exhibiting enhanced learning capabilities, commonly referred to as deep learning [13].

To generalize to unseen data points in supervised learning and establish notions of similarity, we aim to predict the corresponding $y \in \pm 1$ for a given $x \in \mathcal{X}$. In a statistical context, a Kernel function is a symmetric function between two data points that yields a real number characterizing their similarity [16]. Without assuming any particular mathematical structure in the input space, we must embed each data point into a Hilbert Space $\mathcal{H}$, where the inner product and geometric structure provide a means to quantify similarity. The freedom to choose the embedding map allows us to design various similarity measures and machine learning algorithms. Specifically, a Support Vector Machine (SVM) involves finding a hyperplane that linearly separates the data in the *feature* Hilbert space [15].

In the current NISQ era, where quantum devices lack sufficient qubits or tolerance to decoherence, the intersection of Quantum Computing and Machine Learning, referred to as Quantum Machine Learning (QML), is proposed and theorized to have the potential to accelerate supervised tasks with classical and quantum data, outperforming classical methods [6]. Formally, Quantum Machine Learning comprises Kernel models [4], and there is a clear similarity between their training and that of Artificial Neural Networks (ANN).

The main idea in QML in the implicit approach [1] is the implementation of parameterized quantum unitary gates on quantum states where classical data is encoded to make predictions for unseen data using a hybrid approach. Beyond the computational difficulty in calculating a kernel [10] or the potential to represent complex relation functions between inputs and outputs of a quantum circuit [11] as a way of discerning a quantum implementation, a real economic or performance reason for the use of QML instead of CML is still not very clear for real data. This is due to the momentary lack of knowledge, metrics, real cases, or rules to determine which method to use beyond accuracy. Furthermore, despite the existence of ways to optimize parameterized encoded maps via the Kernel Target Alignment (KTA) [4] or the Hilbert-Schmidt distance [3] to make corresponding classes more *separable*, there is no clear methodology on how to design quantum embeddings for specific datasets.

In this work, we present a new implementation of metrics using uniform ensembles [3] and probability density functions for projected and non-projected post-optimized encoded states with Pauli measurements, using the Kullback-Leibler divergence, the quantum relative entropy, the trace distance and the Hilbert-Schmidt distance as ways to design and analyze optimized Quantum Feature Maps. [1]

[1] *Observación:* La creación del estado mezcla al realizar un muestreo uniforme de los estados cuánticos mapeados NO ES UNÍVOCAMENTE DETERMINADA; hay una infinidad de formas de crear dicho estado mezcla con otros estados [17]

II. THEORY

A. Support Vector Machines and Kernel methods

The main idea behind a SVC for binary problems is to find a hyperplane, a linear classifier, that works as a boundary decision. The classifier for a datapoint $\mathbf{x} \in \mathcal{X}$ and $y \in \mathcal{Y} = \{\pm 1\}$:

$$y(x) = \text{sgn}\left(\langle \mathbf{w}, \mathbf{x} \rangle + \beta\right) \tag{1}$$

If the data is not linearly separable in its original feature space, the algorithm maps it into another Hilbert space where the data is, $\phi : \mathcal{X} \to \mathcal{H}$:

$$y(x) = \text{sgn}\left({}_{\mathcal{H}}\langle \mathbf{w}', \phi(\mathbf{x}) \rangle_{\mathcal{H}} + \beta\right) \tag{2}$$

The representation theorem states that we can write our decision function as:

$$y(x) = \text{sgn}\left(\sum_{i \in \mathcal{S}} \alpha_i {}_{\mathcal{H}}\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}) \rangle_{\mathcal{H}} + \beta\right) \tag{3}$$

where $\mathcal{S}$ indicates the collection of indexes of the support vectors. Kernels are defined by a map $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ for which there exists a Hilbert Space $\mathcal{H}$ and a map $\phi : \mathcal{X} \to \mathcal{H}$ such that:

$$K_{ij} = K(\mathbf{x}_i, \mathbf{x_j}) = {}_{\mathcal{H}}\langle \phi(\mathbf{x_i}), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}} \tag{4}$$

where for each kernel exists a unique Hilbert Space (RKHS). This Kernel function is a measure of the similarity between 2 input data; our goal is to achieve a feature map ϕ such that each class dataset is *similar* in $\mathcal{H}$, making it easier for linear separation.
The Kernel Matrix Alingmnet is the forbenius innner product between the normalized Kernels matrix

$$KTA(K, K') = \frac{\langle K, K' \rangle_F}{\|K\|_F \|K'\|_F} \tag{5}$$

It can be considered as a measure of similarity between Kernels, taking values in $[0, 1]$ due to the Schwarz inequality and the spectral theorem, since:

$$\begin{aligned} \langle K, K' \rangle_F &= \langle UU^T, U'U'^T \rangle_F = Tr(U^T U U' U'^T) \\ &= \|U^T U'\|_F^2 \geq 0 \end{aligned}$$

The KTA measures the similarity between a proposed Kernel considering a dataset $\{(\mathbf{x_i}, y_i)\}_{i=1}^n$ and the ideal Kernel K_Y calculated from the output labels:

$$K_Y = \vec{y}\ \vec{y}^T \tag{6}$$

having an increase in the KTA according to the similarity between these Kernels, or in other words, increasing its value depending on the K's ability to assimilate the similarity between the classes.

B. Quantum embedding kernels

In the context of Quantum Machine Learning (QML) for classical data, how do we encoded it into quantum states have a relationship with Kernel methods [1], [2], formally the codification can be considered as a feature map from the original input space $\mathcal{X}$ to the quantum Hilbert space of Qubits $\mathcal{H}_{Quantum}$ attributing a fundamental importance in QML, as it changes the structure of the data in a non-trivial way [3], [19], influencing the expressiveness of the models [20].

The way in which these data are encoded is through a quantum circuit characterized by the input space $\mathcal{X}$ and possibly by other parameters, applied to an initial state that is usually $|0\rangle^{\otimes N}$ Formally, the quantum feature map or circuit is:

$$\hat{\mathcal{U}} : \mathcal{X} \to \mathcal{H}_{\text{Quantum}} = \mathbb{C}^{2N} \tag{7}$$

With a parameter dependence of this map with $\theta \in \Theta$:

$$|0\rangle^{\otimes N} \to \hat{\mathcal{U}}_\theta(\mathbf{x}) |0\rangle^{\otimes N} \tag{8}$$

where we can optimize the parameters of $\hat{\mathcal{U}}_\theta$ with the idea of making the classes separable. There are ways to optimize and evaluate this map: either with the Hilbert-Schmidt or trace distance [3], or with the kernel-target alignment (KTA) [4]. Both of these approaches try to find the best parameters for the map. In this work, we consider the KTA as our optimization function. The associated kernel to (4):

$$K_{ij} = {}^{\otimes N}\langle 0| \hat{\mathcal{U}}_\theta^\dagger(\mathbf{x_i}) \hat{\mathcal{U}}_\theta(\mathbf{x_j}) |0\rangle^{\otimes N} \tag{9}$$

C. Kullback–Leibler divergence for analyzing optimized QFM

Following the definitions and concepts in [18], [22] and [23]; given a discrete random variable X distributed according p , and $q : X \to [0, \infty]$, the Kullback-Leibler divergence is defined as:

$$D_{KL}(p||q) = \mathbb{E}_X\left(\log \frac{p(x)}{q(x)}\right) \tag{10}$$

with the convention [22]:

$$0 \log \tfrac{0}{0} = 0 \quad , \quad p \log \tfrac{p}{0} = \infty \quad , \quad 0 \log \tfrac{q}{0} = 0$$

that means

$$D_{KL}(p||q) = \infty \quad if \quad supp(p) \not\subseteq supp(q) \tag{11}$$

with the support of a function with domain D_f defined as:

$$supp(f) \equiv \{x \in D_f : f(x) \neq 0\}$$

i,e $D_{KL}(p||q) = \infty$ if exists $x \in$ such that $p(x) \neq 0$ but $q(x) = 0$. Also:

$$D_{KL}(p||q) = 0 \quad \iff \quad p = q \tag{12}$$

Considering q as a probability distribution defined on X the D_{KL} can be regarded as a way to *measure* the similarity between both distributions. However, in a mathematical sense it is not a metric since it is not symmetric and does not satisfy the triangle inequality.

III. COMPUTATIONAL EXPERIMENTS

In this study, we implemented the methodologies discussed earlier. We employed the 'make_circles' dataset from the 'scikit-learn' library, consisting of data points arranged in two concentric circles centered at the origin. Using the Pennylane software, we optimized five variational quantum circuits through the KTA method (see Figure 3).

Subsequently, we analyzed the ensemble of quantum states in the final Hilbert space. Initially, we examined the Kullback-Leibler divergence by considering the entire Hilbert space of the tensor product of both qubits. Next, we traced out each qubit and analyzed the reduced density operator.

To visualize the probability distribution of the quantum states, we generated a discrete distribution using a technique akin to Kernel Density Estimation (KDE). Initially, the KDE was applied without imposing the physical constraints of the expected values of each Pauli operator. Subsequently, a discrete probability distribution was generated by creating a 3D grid on the Bloch sphere. Physical constraints corresponding to these estimations were enforced upon evaluating the KDE, followed by normalization.

We then computed the Kullback-Leibler divergence for each class of the data along with the accuracy of the corresponding Support Vector Machine by considering the kernel associated with the mapping of the reduced final circuit.

For each individual qubit, we conducted an additional analysis, focusing on the planes associated with expected values of Pauli observables X, Y, Z, set to zero. This step facilitated the creation of a probability distribution for the data through an initial KDE estimation. Following this, we applied the relevant physical constraints and normalized the dataset. Subsequently, we calculated the Kullback-Leibler divergence for each class. We designed the kernel for the Support Vector Machine by considering quantum states with an expected value of zero for the specific Pauli observable under consideration. A summary of these results, including the Kullback-Leibler divergence and the Test accuracy of the SVM, is presented in Table 1.

IV. RESULTS

V. DISCUSSION

VI. CONCLUSION

ACKNOWLEDGMENT

REFERENCES

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

[2] Schuld, M.: Quantum machine learning models are kernel methods (2021). arXiv preprint. arXiv:2101.11020

[3] S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, & N. Killoran, *Quantum embeddings for machine learning*, arXiv:2001.03622 (2020).

[4] T. Hubregtsen, D. Wierichs, E. Gil-Fuster, Peter-Jan H. S. Derks, P.K. Faehrmann, and J.J. Meyer, *Training quantum embedding kernels on near-term quantum computers*.

[5] M. Schuld and N. Killoran, Is Quantum Advantage the Right Goal for Quantum Machine Learning?, PRX Quantum Vol. 3 (2022)

[6] Cerezo, M., Verdon, G., Huang, HY. et al. Challenges and opportunities in quantum machine learning. Nat Comput Sci 2, 567–576 (2022)

[7] Huang, HY., Broughton, M., Mohseni, M. et al. Power of data in quantum machine learning. Nat Commun 12, 2631 (2021)

[8] Bernhard Schölkopf and Alexander J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. The MIT Press (2018)

[9] Marcello Benedetti1, Erika Lloyd1, Stefan Sack1 and Mattia Fiorentini, Parameterized quantum circuits as machine learning models ,Quantum Sci. Technol. 4 043001 (2019).

[10] Vojtěch Havlíček, Antonio D Córcoles, Kristan Temme, Aram W Harrow, Abhinav Kandala, Jerry M Chow, and Jay M Gambetta, Supervised learning with quantum-enhanced feature spaces, Nature 567, 209 (2019).

[11] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, Quantum Circuit Learning, Phys. Rev. A. 98, 032309 (2018.)

[12] T. Hastie, R. Tibshirani and J. Friedman, An Introduction to Statistical Learning, Springer Texts in Statistics (2013).

[13] Janiesch, C., Zschech, P. & Heinrich, K. Machine learning and deep learning. Electron Markets 31, 685–695 (2021)

[14] Bishop, C. M., Pattern recognition and machine learning (Information science and statistics). Springer-Verlag New York, Inc (2006)

[15] Marc Peter Deisenroth, A.Aldo Faisal & Cheng Soon Ong, Mathemathics for Machine Learning, y Cambridge University Press (2020).

[16] Learning with kernels.

[17] Leslie E. Ballentine, Quantum Mechanics a Modern Development, World Scientific 1st ed.1(2006)

[18] Michael A. Nielsen & Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge 10th ed (2010)

[19] Machine Learning with Quantum Computers.

[20] Maria Schuld, Ryan Sweke, and Johannes Jakob Meyer, Effect of data encoding on the expressive power of variational quantum-machine-learning models, Phys. Rev. A 103, 032430 (2021)

[21] Ryan LaRose and Brian Coyle , Robust data encodings for quantum classifiers, Phys. Rev. A 102, 032420 (2020)

[22] Thomas M. Cover $ Joy A. Thomas, Elements of Information Theory, John Wiley & Sons, 2nd ed (2006)

[23] Mark M. Wilde, Quantum Information Theory, Cambridge University Press 2nd edition (2017)