Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 12, n. 1, 2026.

Trabalho apresentado no XLIV CNMAC, Fundação Getulio Vargas - Rio de Janeiro - RJ, 2025.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Quantum Memory in Support Vector Machines: An Empirical Study

Caio N. Silva[1] Demerson N. Gonçalves, [2] Andrias M. M. Cordeiro[3]
CEFET/RJ, Petrópolis, RJ
João T. Dias[4]
CEFET/RJ, Rio de Janeiro, RJ
Tharso D. Fernandes[5]
UFES, Alegre, ES

Quantum machine learning (QML) uses quantum computing to improve the efficiency of learning. A key approach is *quantum memory*, which enables the coherent storage and reuse of quantum states during computation. Theoretically, quantum memory can exponentially reduce data requirements by preserving correlations between training instances [1, 3], but its empirical validation remains limited. This work examines its impact on a quantum support vector machine (QSVM), comparing its performance with classical and quantum baselines.

Quantum kernels are fundamental in QML, extending classical kernel methods into higher-dimensional quantum feature spaces. At their core is the *quantum feature map*, which implements the kernel trick in a quantum setting. A quantum feature map is a function $\phi : \mathcal{X} \to \mathcal{H}$ that embeds a classical data point $\mathbf{x}$ into a quantum state. The quantum kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$ measures similarity by computing the fidelity between feature states $|\phi(\mathbf{x}_i)\rangle$ and $|\phi(\mathbf{x}_j)\rangle$:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \left|\langle \phi(\mathbf{x}_i)|\phi(\mathbf{x}_j)\rangle\right|^2. \tag{1}$$

These quantum states are generated via a parameterized unitary transformation $\mathcal{U}_{\phi(\mathbf{x})}$ applied to an initial reference state, that is, $|\phi(\mathbf{x}_i)\rangle = \mathcal{U}_{\phi(\mathbf{x}_i)}|0\rangle$. A widely used quantum feature map in QSVM applications is the ZZFeatureMap, which introduces entanglement to encode classical inputs into quantum states [2].

To incorporate quantum memory, we modify the quantum feature map to allow state reuse across training instances. The quantum memory kernel is defined as:

$$K_{\text{memory}}(\mathbf{x}_i, \mathbf{x}_j) = |\langle \phi(\mathbf{x}_i))|U_{\text{memory}}|\phi(\mathbf{x}_j))\rangle|^2, \tag{2}$$

where U_{memory} is a quantum operator designed to retain correlations between previous quantum states. Inspired by [3], in our implementation, U_{memory} consists of an additional layer of parameterized R_Y rotations:

$$U_{\text{memory}} = \prod_q R_Y(x_q \pi), \tag{3}$$

which is applied both before and after a sequence of entangling operations. The parameter x_q corresponds to the q-th component of the classical input feature vector $\mathbf{x}$, which is mapped onto

[1]caio.silva.1@aluno.cefet-rj.br
[2]demerson.goncalves@cefet-rj.br
[3]andrias.cordeiro@aluno.cefet-rj.br
[4]joao.dias@cefet-rj.br
[5]tharso.fernandes@ufes.br

the quantum state space. Each qubit is rotated proportionally to its respective input feature, ensuring that the encoding preserves class-dependent correlations. Specifically, the transformation applied to qubit q is given by:

$$R_Y(x_q\pi)|0\rangle = \cos\left(\frac{x_q\pi}{2}\right)|0\rangle + \sin\left(\frac{x_q\pi}{2}\right)|1\rangle. \tag{4}$$

This guarantees that different components of the input data influence different qubits in a structured manner, while the memory mechanism ensures that previously processed quantum states remain partially correlated with subsequent states.

To empirically evaluate the effect of quantum memory, we trained three models on datasets of varying sizes (100, 300, and 500 samples). The performance comparison is shown below:

Table 1: Comparison of classification accuracy across dataset sizes.

Dataset Size	Classical SVM (RBF)	QSVM (ZZFeatureMap)	QSVM (Quantum Memory)
100	60.0%	40.0%	**63.3%**
300	47.8%	51.1%	**53.3%**
500	45.3%	48.0%	**51.3%**

Despite promising results, practical implementation challenges remain. Since this study relies on noiseless simulations, real-world performance may be overestimated, as decoherence and gate errors can impact quantum memory reliability [5]. However, the findings suggest that quantum memory may help mitigate the degradation of accuracy as data complexity grows, indicating its potential as a resource for QML.

Validating these results on real quantum hardware is an important step toward assessing the feasibility of quantum memory in practical applications. While synthetic datasets provide useful insights, testing in domains where quantum correlations naturally arise, such as quantum chemistry [4] or high-energy physics, will further clarify its advantages. Additionally, refining quantum feature maps and exploring alternative kernels could improve the scalability and robustness of memory-assisted QML models.

References

[1] S. Chen, J. Cotler, H.-Y. Huang, and J. Li. "Exponential Separations Between Learning With and Without Quantum Memory". In: **Proc. of the 62nd IEEE Annual Symp. on Foundations of Computer Science (FOCS)**. IEEE, 2021. DOI: 10.1109/FOCS52979.2021.00063.

[2] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta. "Supervised learning with quantum-enhanced feature spaces". In: **Nature** 567 (2019), pp. 209–212. DOI: 10.1038/s41586-019-0980-2.

[3] H.-Y. Huang, M. Broughton, J. Cotler, S. Chen, J. Li, M. Mohseni, H. Neven, R. Babbush, R. Kueng, J. Preskill, and J. R. McClean. "Quantum advantage in learning from experiments". In: **Science** 376.6598 (2022), pp. 1182–1186. DOI: 10.1126/science.abn7293.

[4] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan. "Quantum computational chemistry". In: **Reviews of Modern Physics** 92.1 (2020), p. 015003. DOI: 10.1103/RevModPhys.92.015003.

[5] J. Preskill. "Quantum computing in the NISQ era and beyond". In: **Quantum** 2 (2018), p. 79. DOI: 10.22331/q-2018-08-06-79.