

EFICIENCIA COMPARATIVA ENTRE ALGORITMOS CLÁSICOS Y CUÁNTICOS EN UN PROBLEMA DE CLASIFICACIÓN

Autores/as Osenda, Omar; Santana Perón, Sofía y Legnazzi, Nicolás

Eje temático: Investigación en disciplinas básicas y aplicadas

unc | campus virtual | innovación y vinculación

2024 - ARGENTINA

Eficiencia comparativa entre algoritmos clásicos y cuánticos en un problema de clasificación

Sofía Perón Santana, Nicolás Legnazzi, Omar Osenda

[1] *Facultad de Matemática, Astronomía, Física y Computación, Universidad Nacional de Córdoba e Instituto de Física Enrique Gaviola (CONICET-UNC). Av. Medina Allende s/n, Ciudad Universitaria, CP:X5000HUA Córdoba, Argentina .*
**Correo Electrónico (autor de contacto): sofia.peron@mi.unc.edu.ar*

RESUMEN
La computación cuántica ofrece una perspectiva diferente para atacar problemas tradicionales de la computación ya que la clase de complejidad es distinta a la de las computadoras tradicionales. Si bien formular algoritmos usando herramientas cuánticas no es simple, al menos en el área de aprendizaje automático hay claros avances en problemas de regresión y clasificación a tal punto que la comparación entre eficiencia de recursos y rendimiento se vuelve necesaria. En este trabajo se compara la eficiencia clasificatoria de un circuito cuántico variacional que usa muy pocos recursos cuánticos con ejemplos estándar de clasificadores disponibles en paquetes de software dedicados a éste tipo de problemas .

ABSTRACT
Quantum computing offers a different perspective for tackling traditional computing problems since its complexity class is different from that of traditional computers. Although the formulation of algorithms using quantum tools is not simple, at least in the Machine Learning field there has been a huge breakthrough in solving regression and classification problems to the point that the comparison of performance and resource efficiency has become a necessity. In this work, we compare the classification efficiency of a quantum variational circuit using very few quantum resources with standard classifiers available in dedicated software packages.

Palabras clave: aprendizaje automático, procesamiento cuántico, clasificación, benchmarks

Introducción

Dado el creciente avance en el diseño experimental de hardware cuántico, resulta necesario considerar cómo adaptar la algoritmia actual para ser implementada en este tipo de sistemas. Es decir, dado que la mayoría de los algoritmos desarrollados al día de la fecha se basan en la lógica de compuertas clásicas y datos codificados en bits binarios, surge la necesidad de diseñar algoritmos que respondan a la lógica particular de la computación cuántica.

El Quantum Circuit Learning es un marco de trabajo híbrido para implementar Machine Learning en circuitos de baja profundidad (1). Se dice que es híbrido ya que combina la implementación de un circuito cuántico parametrizado con procesamiento y optimización clásicos. Por otro lado, se pone el foco en circuitos de baja profundidad ya que esto implica un bajo número de compuertas aplicado a cada qubit lo que permitiría implementaciones a corto plazo (2), a diferencia de otros algoritmos cuánticos.

El método consiste en optimizar los parámetros de un circuito cuántico, por ejemplo, los ángulos asociados a compuertas de rotación, para que el circuito resuelva alguna tarea de aprendizaje clásico como clasificación o regresión. Para un conjunto de datos de entrada $\mathbf{x_i}$ con sus correspondientes etiquetas $\mathbf{f(x_i)}$, el objetivo será que el circuito produzca un output $\mathbf{y_i = y(x_i,\{\theta\})}$ que se aproxime a las etiquetas $\mathbf{f(x_i)}$. Para ello, es necesario encontrar los valores de θ que minimicen alguna función costo. Puede pensarse a este marco de trabajo como una implementación física de una red neuronal donde los qubits representan a las neuronas clásicas y los ángulos de rotación y demás parámetros del sistema son análogos a los pesos.

Este trabajo analiza uno de los experimentos planteados en el trabajo fundacional de esta técnica (1) usando simulaciones. El mismo consiste en un problema clásico estándar de clasificación binaria. Para demostrar la efectividad de este marco de trabajo, se compara con algoritmos de clasificación clásicos tomados como *benchmark* (regresión logística, gradient boosting y support vector classification) a través del cálculo de varias métricas (3).

Materiales y Métodos

En general, para un problema de aprendizaje supervisado se cuenta con un conjunto de datos de entrada asociados a sus correspondientes etiquetas **{f(x$_i$)}** y se busca entrenar un modelo que permita obtener una salida para cada dato de entrada que aproxime lo suficientemente bien dicha etiqueta. En el Quantum Circuit Learning, este modelo estará dado por un circuito cuántico variacional (1) y los parámetros a optimizar serán aquellos asociados a las compuertas que lo componen. Por ejemplo, ángulos de rotación.

La Figura 1 muestra una representación esquemática de implementación de QCL. En primer lugar, es necesario codificar el dato de entrada $\mathbf{x_i}$ en un estado cuántico $\psi_{in}(\mathbf{x_i})$ aplicando una compuerta unitaria $U(\mathbf{x_i})$ a qubits inicializados en el estado $|0>$. Siguiendo el criterio del trabajo de referencia (1), se utiliza como compuerta de entrada un conjunto de compuertas de rotación dado por:

$$U_{in}(x_i) = \prod_{j=1}^{N} R_j^z(cos^{-1}(x^2_{i,j\ mod\ 2}) R_j^y(cos^{-1}(x_{i,j\ mod\ 2})), \qquad [1]$$

donde N es el número de qubits en el circuito.

Luego, se aplica la compuerta parametrizada o compuerta de salida (Figura 1b) dada por:

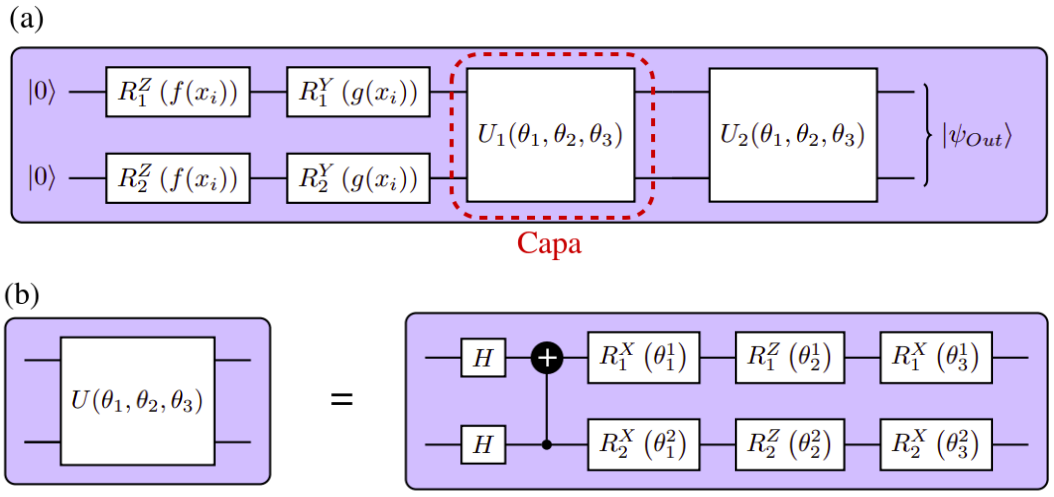
$$U_{out}(\{\theta\}) = \prod_{i=1}^{P} \left[\prod_{j=1}^{N} H_j U_{ent}(j, j-1). R_j^z(\theta^{(i)}_{j1}) R_j^X(\theta^{(i)}_{j2}) R_j^z(\theta^{(i)}_{j3}) \right], \qquad [2]$$

donde las matrices $R_j^{x,y,z}$ son rotaciones y la compuerta de entrelazamiento (U_{ent}) y la de Hadamard (H_j) están dadas por:

$$U_{ent}(j, j-1) = |0><0|_j \otimes I_{j-1} + |1><1|_j \otimes \sigma^x_{j-1}, \quad H_j = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

El subíndice j en las compuertas de Hadamard y en las rotaciones indica sobre qué qubit actúa cada una de ellas. La operación unitaria escrita entre corchetes en la ecuación [2] es una de las capas del circuito (ver Figura 1). El circuito usa P de estas capas, que es igual a la profundidad del circuito.

Figura 1: Representación esquemática de la implementación de circuito variacional cuántico. (a) Circuito para dos qubits y dos capas que contienen los parámetros a aprender. El diagrama muestra cómo se aplican las distintas compuertas secuencialmente. Los qubit se inicializan en el estado de referencia |0>. Las compuertas de rotación preparan el estado correspondiente a un dato de entrada x_i. (b) Representación esquemática explícita de todas las compuertas cuánticas que contiene una capa. De izquierda a derecha se tiene dos compuertas de Hadamard, una compuerta CNOT que provee entrelazamiento, y compuertas de rotación de un qubit, cada una de ellas dependen de un parámetro a aprender.



Para comprender las ecuaciones [1] y [2] es importante destacar que en sistemas de múltiples qubits, los operadores que actúan sobre distintos qubits se combinan usando el producto tensorial denotado con $\otimes$ (6). Las rotaciones del qubit j-ésimo sobre la dirección D = x,y o z están dadas por:

$$R_j^D(\theta) = cos(\theta)\, I_j + sin(\theta)\sigma_j^D \;,\; \text{donde } \sigma_j^x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \sigma_j^y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \sigma_j^z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix},$$

son las matrices de Pauli usuales actuando sobre el j-ésimo qubit e I_j es la matriz identidad 2x2.

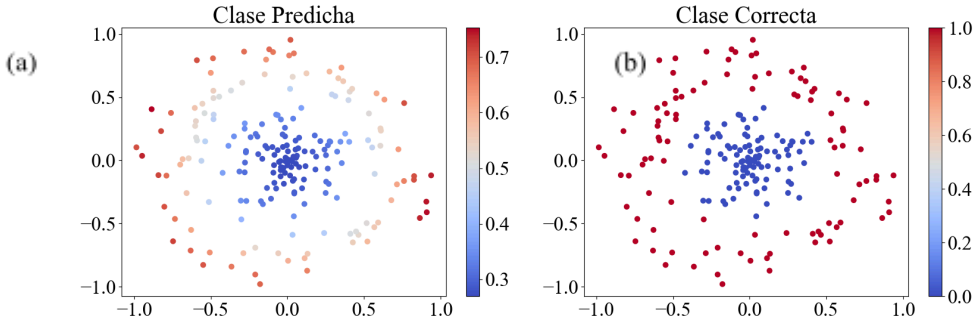
Como consecuencia de cómo se construye el circuito, se tienen 3 ✕ N ✕ P parámetros a optimizar. Para este trabajo, se simuló un circuito de 6 qubits y profundidad 6, resultando en un total de 108 parámetros.

Una vez aplicada esta compuerta se mide un conjunto de valores de expectación. Para este problema, se miden los valores de expectación de la magnetización en Z asociados al primer y segundo qubit. Luego de medir, el procedimiento se realiza de manera clásica. Dado que es un problema de clasificación, los valores medidos se transforman utilizando la función softmax **F** y se elige como función costo la función *cross entropy*. Con algún método de minimización clásico (en este caso se usa BFGS), se propone un nuevo conjunto de parámetros **{θ}** y se repite el procedimiento hasta que el método converge. Por último, es necesario evaluar la performance chequeando la función costo con respecto a un conjunto de testeo tomado independientemente del de entrenamiento. Cabe destacar que esta arquitectura no es única para cada problema, y, al ser un hiper-parámetro, no es trivial encontrar "la mejor arquitectura". Las simulaciones se realizaron utilizando la librería qulacs (4).

Resultados

La Figura 2a muestra las predicciones obtenidas para el conjunto de testeo en relación a las etiquetas correspondientes (Figura 2b). Las barras de color representan la probabilidad de pertenecer a cada clase (1: rojo, 0: azul). Se tomaron 100 muestras de entrenamiento y 100 de testeo. Los datos fueron generados artificialmente y de forma aleatoria.

Figura 2: Clase predicha por el algoritmo de QCL (a) comparada con la clase correcta (b) para 100 muestras de testeo.



Para corroborar la efectividad del método, se compara con 3 clasificadores clásicos estándar implementados usando scikit-learn (3): Regresión logística, *Gradient Boosting* y SVC (*support vector classification*) con kernel sigmoide. Los resultados obtenidos se muestran en la Figura 3.

Figura 3: Matriz de confusión para el algoritmo de QCL comparada con las asociadas a clasificadores clásicos típicos.

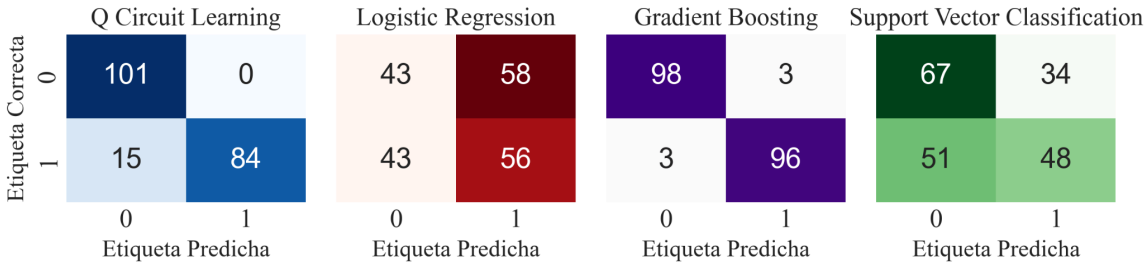


Tabla 1: Comparación de métricas para los distintos clasificadores

Clasificador	**ROC AUC**	**F1-score**	**Accuracy**
Quantum Circuit Learning	0.924242	0.918033	0.925
Regresión Logística	0.4957	0.525822	0.495
Gradient Boosting	0.969997	0.969697	0.97
Support Vector Classification	0.574107	0.530387	0.575

Discusión

La Figura 2 muestra que las predicciones sobre el conjunto de testeo se acercan considerablemente a las etiquetas correctas. Para los puntos umbral, el clasificador asigna 50% de probabilidad de pertenecer a cada clase. Es natural que estos sean los puntos más difíciles de clasificar.

Se puede ver en la Figura 3 que para 15 valores el clasificador falla y asigna la clase 0 a elementos de la clase 1. El error inverso no ocurre. De la Figura 3 y la Tabla 1, se observa que la eficiencia del algoritmo de QCL es superior a la lograda por SVC y regresión logística. Esto tiene sentido ya que los datos son altamente no lineales. Por otro lado, su performance es comparable a la del algoritmo de Gradient Boosting, logrando una precisión superior al 90%. Es importante señalar que no se busca mejorar el rendimiento de los algoritmos clásicos, cambiando el tipo de kernel u otros hiper-parámetros, sino mostrar que el algoritmo cuántico sin ninguna adaptación, con pocos qubits y pocas capas logra un rendimiento similar. Sería interesante estudiar qué tipo de mejoras son implementables en el setup cuántico.

Para el tipo particular de datos utilizado, una transformación no lineal, en el caso de regresión logística, o el uso de kernel polinomiales, en el caso de support vector classification, resultaría en una mejor performance. Otro encoding de los datos en el circuito cuántico, también. Esto se analizará posteriormente. Las posibles ventajas de métodos cuánticos de clasificación sobre los métodos clásicos son objeto de debate en la actualidad (7,8).

Conclusiones

Es posible implementar un algoritmo de clasificación para un problema clásico de aprendizaje supervisado en este framework con un desempeño equivalente al de clasificadores típicos. Más allá de las métricas, la riqueza de esta propuesta radica en la adaptación de algoritmos originalmente pensados para computación clásica a una lógica programable en procesadores cuánticos. Además, al estar planteado para desarrollarse en un número bajo de qubits y compuertas podría implementarse en el corto plazo. Actualmente, la mayor dificultad se encuentra en la sensibilidad al ruido de las computadoras cuánticas. Sin embargo, ya existen trabajos que ejecutan algoritmos similares experimentalmente (2,5).

Agradecimientos

Agradecemos a Martín Domínguez (Famaf-UNC) y a Federico Ribetto (Famaf-UNC) por sus sugerencias y comentarios acerca de este trabajo. También agradecemos al CCAD- UNC. Este trabajo forma parte del Proyecto “Aplicación de técnicas de aprendizaje automático en física cuántica y sistemas complejos”, SECYT-UNC 30820230200010CB. Agradecemos a los revisores por comentarios sobre cómo mejorar el rendimiento de los algoritmos clásicos

REFERENCIAS

1. Mitarai K, Makoto Negoro, Kitagawa, M. and Fujii, K. (2018). Quantum Circuit Learning. Physical Review A, 98(032309). doi: https://doi.org/10.1103/physreva.98.032309.
2. Kim, Y., Eddins, A., et al.. (2023). Evidence for the utility of quantum computing before fault tolerance. Nature, 618(7965), 500–505. doi: https://doi.org/10.1038/s41586-023-06096-3
3. Scikit-Learn. (2015). 3.4. Metrics and scoring: Quantifying the Quality of Predictions. [online] Disponible en: https://scikit-learn.org/1.5/modules/model_evaluation.html
4. Qulacs.org. (2020). About Qulacs — Qulacs Documentation. [online] Disponible en: https://docs.qulacs.org/en/latest/intro/0_about.html
5. Havlícek, V., Córcoles, A.D., Temme, K., Harrow, A.W., Kandala, A., Chow, J.M. and Gambetta, J.M. (2019). Supervised Learning with quantum-enhanced Feature Spaces. Nature, [online] 567, pp.209–212. doi: https://doi.org/10.1038/s41586-019-0980-2.
6. Nielsen, M.A. and Chuang, I.L. (2010). Quantum Computation and Quantum Information. Cambridge University Press.
7. Kumar, T., Kumar, D. and Singh, G. (2022). Performance Analysis of Quantum Classifier on Benchmarking Datasets. International Journal of Electrical and Electronics Research, 10(2), pp.375–380. doi:https://doi.org/10.37391/ijeer.100252.
8. Bowles, J., Ahmed, S. and Schuld, M. (2024). Better than classical? the Subtle Art of Benchmarking Quantum Machine Learning Models. [online] Available at: https://arxiv.org/pdf/2403.07059