

# Desafíos en el desarrollo e implementación de análogos cuánticos directos a las redes neuronales artificiales y avances significativos en el área

Andrey Gonzalez

*Facultad de ciencias, Universidad Nacional de Colombia sede Medellín*

## Resumen

Se analizan los desafíos y avances en el desarrollo de análogos cuánticos de las redes neuronales artificiales (RNAs), explorando la viabilidad de modelos como los circuitos cuánticos variacionales (CCV), las máquinas cuánticas de Boltzmann (MCB) y las redes neuronales convolutivas cuánticas (RNCC). Se presenta una comparación entre RNAs clásicas y modelos cuánticos, destacando dificultades como la falta de no linealidad en sistemas cuánticos, la imposibilidad de aplicar retropropagación directa y las limitaciones del hardware cuántico actual. Además, se implementa un experimento con un CCV y una RNA clásica, evaluando su desempeño en una tarea de clasificación binaria. Finalmente, se concluye que, aunque aún no existe un análogo cuántico directo de las RNAs, los avances en computación cuántica y aprendizaje automático cuántico (ACM) podrían permitir desarrollos prometedores en el futuro.

## 1. Introducción

Dentro del vasto mundo del aprendizaje de máquina, los modelos de redes neuronales artificiales (RNA) gozan de especial atención, en cuanto a que, por diseño, exhiben una naturaleza no lineal que les hace particularmente útiles en tareas de clasificación de patrones, desde el reconocimiento facial hasta artillería asistida por inteligencia artificial (IA).

Debido a esto y muchos otros factores, la investigación en aprendizaje profundo y el uso de RNAs ha ganado gran interés tanto en sectores especializados como para el público general. Grandes compañías y centros de investigación destinan enormes recursos para la mejora de sus sistemas asistidos por estas herramientas, tanto para el desarrollo de modelos más eficientes, como la infraestructura y centros de cómputo adecuados para la ejecución de estas rutinas.

Si estos modelos han permitido un gran paso en el procesamiento de los datos y obtención de información desde los mismo, una hazaña

similar se da de manera paralela con el advenimiento de la computación cuántica y su potencial para mejorar las capacidades de cómputo actuales y superar la respuesta de un computador clásico a problemas de complejidad superior.

Gigantes tecnológicos han desarrollado procesadores que, inspirados en los resultados de la mecánica cuántica, han logrado avances significativos en la computación, incluyendo la corrección de errores y la realización de tareas complejas en tiempos récord, según Neven (2024), refiriéndose al procesador Willow de Google.

Es de esperar que al tener RNAs cada vez más exigentes con los recursos computacionales disponibles, a la par de computadores cuánticos que superan los límites de la computación clásica, un híbrido de estos dos sea buscado. No obstante, las propiedades intrínsecas de estos enfoques, sumado al hecho que ambas son aun disciplinas en crecimiento supone retos que aun plantan cara a los mejores científicos e ingenieros y en el presente artículo, se exploran

algunos de estos junto con algunas alternativas prometedoras que han visto luz en los últimos años.

## 2. Redes neuronales y circuitos cuánticos

La red neuronal mas simple corresponde al perceptrón de Rosenblatt introducido en 1958 el cual consiste en una neurona artificial que recibe múltiples entradas, las promedia mediante pesos a cada una, lo que produce una salida regulada por una función de activación, cuya tarea es compactar todo el posible rango de salidas de la neurona, a un pequeño intervalo de la recta numerica comprendido entre cero 1 uno. Matemáticamente, la salida  $y$  de un perceptrón puede expresarse como:

$$y = f \left( \sum_{i=1}^n \omega_i x_i + b \right)$$

Con  $x_i, w_i$  las entradas y sus respectivos pesos,  $b$  es el sesgo y  $f$  es la función de activación.

El perceptrón está limitado por no ser apto para la resolución de problemas no lineales, por su estructura simple y la elección de la función de escalón como función de activación, que producía una salida binaria (Rosenblatt, 1958). Para solventar esto, surgieron las redes neuronales multicapa (RNM), que consisten en múltiples capas de neuronas: una capa de entrada, una o más capas ocultas y una capa de salida. En la figura 1 se muestra un modelo sencillo de una red de este tipo.

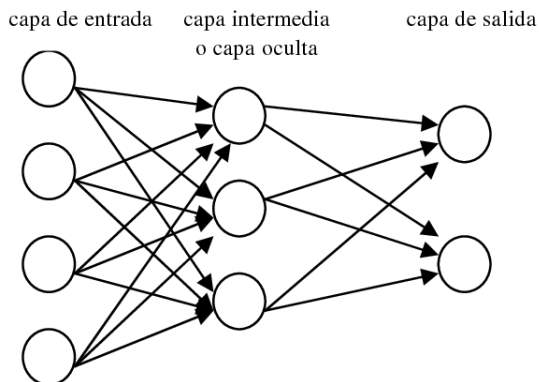


Figura 1. Ejemplo de perceptrón multicapa. Fuente: Botti, V., & Carrascosa, C. (2002). Introducción a la inteligencia artificial. Universidad Politécnica de Valencia.

Cada neurona en una capa se conecta a todas las de la siguiente capa, formando una red, y en el caso de este tipo de redes, se prefiere

la elección de funciones de activación no lineales, como la sigmoide o la ReLU (Rectified Linear Unit), permite que la red aprenda representaciones de patrones más complejos y resuelva problemas no lineales (Goodfellow, Bengio, & Courville, 2016).

Por su parte, un circuito cuántico se constituye por secuencia de operaciones cuánticas (puertas cuánticas) aplicadas a un conjunto de cúbits, que son las unidades básicas de información en la computación cuántica. Diferentes a los bits clásicos, que pueden estar en estados 0 o 1, los cúbits pueden estar en una superposición de ambos estados simultáneamente, lo que permite una capacidad de procesamiento exponencialmente mayor (Nielsen & Chuang, 2010). Se tiene entonces sistemas análogos a los encontrados en la mecánica cuántica, cuya interacción entre si es en general de naturaleza no trivial y expresada en forma de estados entrelazados, coexistiendo mutuamente. La estructura básica de un circuito cuántico incluye:

- **Cúbits de entrada:** Representan el estado inicial del sistema cuántico.
- **Puertas cuánticas:** Representan operaciones unitarias que manipulan el estado de los cúbits. Ejemplos comunes incluyen la puerta Hadamard (H), que crea superposiciones, y la puerta CNOT, que introduce entrelazamiento entre cúbits.
- **Medición:** El proceso de observar el estado de los cúbits, obteniendo un estado clásico definido.

Un ejemplo de circuito cuántico simple es el que genera un estado de Bell, que es una superposición de máximo entrelazamiento entre dos cúbits es aplicando una compuerta Hadamard al primer cúbit para crear una superposición, seguida de una compuerta CNOT, utilizando el primer cúbit como control y el segundo como objetivo, para entrelazar ambos cúbits. En la figura 2, este y otros circuitos basicos para generar estados entrelazados.

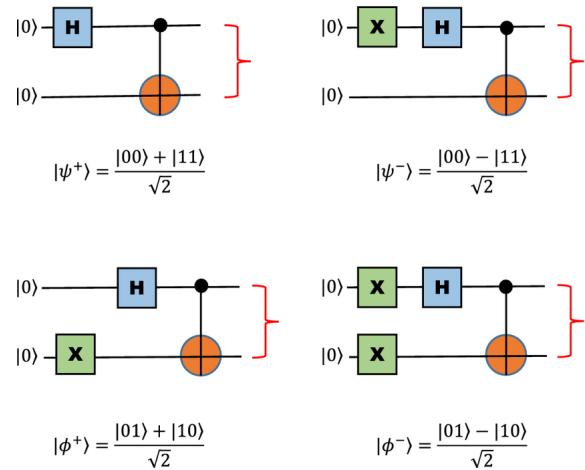


Figura 2. Circuito cuántico para generar un estado de Bell. Fuente: Acharya, R., Abanin et al (2024 ). Quantum circuits for Bell states preparation by using quantum gates. Nature

## 3. Redes neuronales cuánticas y sus retos

Actualmente, no existe un análogo cuántico directo de las RNAs que replique exactamente su funcionamiento usando herramientas de computación cuántica. No obstante, se han desarrollado varios enfoques en el campo del aprendizaje cuántico de maquina (ACM) que intentan usar la computación cuántica para realizar tareas similares a las RNAs, pero usando principios distintos, propios de la mecánica cuántica.

Uno de los principales retos corresponde a linealidad de los sistemas. Las redes neuronales clásicas deben su capacidad de modelar funciones complejas a la no linealidad introducida por funciones de activación como ReLU, Sigmoid o Tanh. Sin embargo, los sistemas cuánticos evolucionan de manera unitaria y lineal según la ecuación de Schrödinger. Esto significa que no hay una manera natural de incluir estas no linealidades en los circuitos cuánticos sin realizar mediciones intermedias, lo cual colapsaría el estado cuántico y limitaría la ventaja cuántica (Schuld et al., 2014).

Se suma a esto que en IAS RNAs los pesos se ajustan mediante retropropagación del error (backpropagation), donde los gradientes se recorren desde la salida a la entrada de la red usando la regla de la cadena del cálculo diferencial para ello, pero en un sistema cuántico, la información no puede clonarse debido al teorema de no clonación, lo que impide la implementación directa del algoritmo de retropropagación clásico. Existen aproximaciones, una de ellas es el método de parameter shift rule (Mitarai et al., 2018).

La naturaleza de los datos también tiene un papel importante, pues en el caso de los sistemas cuánticos pensados para el análisis de datos clásicos, los datos deben codificarse en estados cuánticos antes de ser procesados, lo que introduce una sobrecarga computacional adicional. Además, extraer información del sistema cuántico requiere mediciones, lo que introduce ruido y colapso del estado cuántico (Biamonte et al., 2017).

Cabe destacar igualmente el problema de la disponibilidad de hardware para la ejecución de algoritmos cuánticos, puesto que las RNAs pueden ejecutarse en hardware clásico utilizando GPUs y TPUs optimizadas para acelerar el entrenamiento, sin embargo, las arquitecturas cuánticas aún están en etapas tempranas y están limitadas por la cantidad de qubits, la coherencia del sistema y la tasa de error, lo que dificulta la implementación práctica de modelos equivalentes a ANN en hardware cuántico actual (Preskill, 2018).

## 4. Alternativas al problema

A pesar de estas diferencias fundamentales, hay varios enfoques en el campo del ACM que buscan desempeñar un rol similar a las RNAs en el campo cuántico. Algunos de estos modelos son:

- **Circuitos cuánticos variacionales (CCV):** estos son circuitos cuánticos con puertas parametrizadas que pueden ser optimizadas para resolver problemas específicos, donde los parámetros de las puertas cuánticas tienen un rol análogo a los pesos de una red neuronal, y se optimizan mediante algoritmos clásicos como el descenso de gradiente, siendo su diferencia clave el que no pueden realizar funciones no lineales sin medición. (Mitarai et al, 2018).
- **Máquinas cuánticas de Boltzmann (MCB):** Son un análogo cuántico de las Máquinas de Boltzmann, que son modelos probabilísticos utilizados en redes neuronales profundas, los cuales utilizan una función de energía y un proceso de aprendizaje para encontrar distribuciones de probabilidad óptimas. No obstante, no están basadas en unidades neuronales tradicionales, sino en principios de mecánica cuántica como estados de superposición (Amin et al., 2018).
- **Redes neuronales convolutivas cuánticas (RNCC):** Son una arquitectura cuántica inspirada en las redes neuronales convolutivas y utilizadas principalmente en visión artificial. Estas

se componen de estructuras jerárquicas con capas de procesamiento cuántico que extraen características de los datos, aun así, esto lo hacen aprovechando la compresión cuántica en lugar de convoluciones tradicionales de su contraparte clásica.

Los CCV, MCB y las RNCC son modelos cuánticos que buscan desempeñar roles similares al aprendizaje profundo, aunque con principios distintos debido a la naturaleza de los fenómenos cuánticos y al procesamiento de información de los mismos. Si se logra desarrollar una forma eficiente de introducir no linealidades en sistemas cuánticos sin mediciones intermedias, entonces sería más viable construir una red neuronal cuántica que equipare realmente a las RNAs clásicas.

## 5. Un hola mundo a los circuitos variacionales

Vale la pena observar un poco en la práctica estos sistemas. Si bien el ACM aún está en sus primeras etapas y con muchos retos por superar, ya hay importantes aportes al estudio de esta área. Compañías como PennyLane ofrecen al público librerías bastante completas de computación cuántica, donde el aprendizaje profundo bajo este paradigma también tiene lugar.

Así pues, se puede explorar el comportamiento de modelos sencillos de CCV, siendo los modelos más próximos a las RNAs clásicas usando los paquetes de circuitos cuánticos de PennyLane disponibles para su importación en lenguajes de programación altamente usados como python.

En esta sección se comparte un desarrollo sencillo, nada refinado y solo de naturaleza exploratoria para comparar como se desempeñan ambos planteamientos. El módulo Keras de TensorFlow y scikit learn de python, junto con PennyLane y otras librerías científicas adicionales hacen la tarea.

```
import numpy as np
from tensorflow import keras
from sklearn.datasets import make_moons
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
import pennylane as qml
import matplotlib.pyplot as plt
import pennylane.numpy as pnp
```

Con las librerías a usar ya definidas, se puede generar un conjunto de datos sintéticos para poner a prueba los modelos

```
def generate_data():
    X, y = make_moons(n_samples=200,\
                      noise=0.1, random_state=42)

    X = StandardScaler().fit_transform(X)
    y = y.reshape(-1, 1)

    return train_test_split(X, y,\
                          test_size=0.2, random_state=42)
```

```
X_train, X_test, y_train, y_test = generate_data()
```

Hecho esto, se define una red neuronal clásica, usando las capacidades de TensorFlow para entrenar la misma con los datos artificiales.

```
def create_ann_model():
    model = keras.Sequential([
        keras.Input(shape=(2,)),
        keras.layers.Dense(10, activation='relu'),
        keras.layers.Dense(10, activation='relu'),
        keras.layers.Dense(1, activation='sigmoid')
    ])
```

```

model.compile(optimizer='adam',\
loss='binary_crossentropy', metrics=['accuracy'])

return model

```

```

ann_model = create_ann_model()
history = ann_model.fit(X_train, y_train,\
epochs=50, batch_size=10, verbose=0,\
validation_data=(X_test, y_test))

ann_acc = ann_model.evaluate(X_test,\
y_test, verbose=0)[1]

```

PennyLane permite la creación de compuertas cuánticas parametrizadas, con las cuales se puede establecer un CCV con dos cúbits como entrada del sistema y los respectivos pesos mediados por las operaciones de las compuertas cuánticas. El proceso de entrenamiento sigue una estructura similar a la del entrenamiento de una RNA clásica.

```

n_qubits = 2
dev = qml.device("default.qubit", wires=n_qubits)

```

```

@qml.qnode(dev)
def quantum_circuit(weights, x):
    qml.AngleEmbedding(x, wires=range(n_qubits),\
rotation="Y")

```

```

    qml.StronglyEntanglingLayers(weights,\
wires=range(n_qubits))

```

```

    return qml.expval(qml.PauliZ(0))

```

```

weights_shape = (3, n_qubits, 3)

```

```

weights = pnp.random.randn(*weights_shape,\
requires_grad=True)

```

```

opt = qml.GradientDescentOptimizer(stepsize=0.1)

```

```

def cost(weights, X, y):
    loss = 0
    for i in range(len(X)):
        prediction = quantum_circuit(weights, X[i])
        loss += (prediction - y[i]) ** 2
    return loss / len(X)

```

```

vqc_losses = []
vqc_val_losses = []

```

```

for epoch in range(50):
    weights = opt.step(lambda w: cost(w,\
X_train, y_train), weights)

    train_loss = cost(weights, X_train,\
y_train).item()

    val_loss = cost(weights, X_test,\
y_test).item()

    vqc_losses.append(train_loss)
    vqc_val_losses.append(val_loss)

    if epoch % 10 == 0:

```

```

        print(f'Epoch_{epoch},\
Train_Loss:{train_loss:.4f},\
Validation_Loss:{val_loss:.4f}')

```

```

def predict(X):
    return np.array([1 if quantum_circuit(weights, x) > 0\
else 0 for x in X])

```

```

y_pred = predict(X_test)
vqc_acc = np.mean(y_pred == y_test.flatten())

```

En la figura 3 se comparan las perdidas en el proceso de entrenamiento y en la etapa de validación del modelo, mientras que la figura 4 compara sus precisiones:

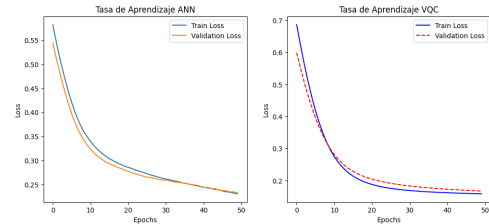


Figura 3. Funcion de perdida vs Epoca de entrenamientos.

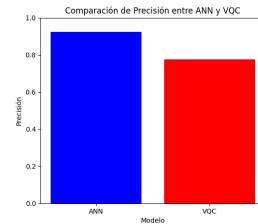


Figura 4 Precision: RNA vs CCV

Se observa que la tasa de aprendizaje de ambos modelos, o sea, la minimización de la pérdida en función del número de épocas es similar en ambos modelos, aunque ha de tenerse en cuenta que el número de cúbits por capa haciendo el papel de neuronas es mucho menor que en la red neuronal clásica, implicando igualmente un número menor de parámetros a ajustar para el entrenamiento de la red, y con una precisión al momento de la validación, si bien superior para las RNAs, aun así para nada despreciable en el caso de los CCV, lo cual permite vislumbrar su potencial para optimizar problemas en campos como la clasificación a medida que se refinan los modelos.

## 6. Conclusiones

No existe un análogo cuántico exacto de las RNAs, debido a la falta de no linealidad en la mecánica cuántica, la imposibilidad de clonar información y la diferencia en los procesos de entrenamiento. Pero se han llevado a cabo avances notorios en el campo con la implementación de modelos como los CCV, las RNCC y las MCB que permiten desempeñar tareas similares inspirados en las RNAs.

El ACM aun es un arte de estudio en desarrollo temprano, y de los avances independientes que se den en cada una de sus áreas componentes, RNAs y computación cuántica dependerá su evolución y posible aplicación a problemas de importancia actual.

## Referencias

- [1] Acharya, R., Abanin, D. A., Aghababaie-Beni, L., Aleiner, I., Andersen, T. I., Ansmann, M., ... & Neven, H. (2024). *Quantum circuits for Bell states preparation by using quantum gates*. Nature.
- [2] Amin, M. H., Andriyash, E., Rolfe, J., Kulchytskyy, B., & Melko, R. G. (2018). Quantum Boltzmann machine. *Physical Review X*, 8(2), 021050. <https://doi.org/10.1103/PhysRevX.8.021050>
- [3] Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., & Lloyd, S. (2017). Quantum machine learning. *Nature*, 549(7671), 195–202. <https://doi.org/10.1038/nature23474>
- [4] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- [5] Mitarai, K., Negoro, M., Kitagawa, M., & Fujii, K. (2018). Quantum circuit learning. *Physical Review A*, 98(3), 032309. <https://doi.org/10.1103/PhysRevA.98.032309>
- [6] Neven, H. (2024). *Meet Willow, our state-of-the-art quantum chip*. Recuperado de <https://blog.google/technology/ai/meet-willow-our-state-of-the-art-quantum-chip/>
- [7] Nielsen, M. A., & Chuang, I. L. (2010). *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press.
- [8] Preskill, J. (2018). Quantum computing in the NISQ era and beyond. *Quantum*, 2, 79. <https://doi.org/10.22331/q-2018-08-06-79>
- [9] Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386-408. <https://doi.org/10.1037/h0042519>
- [10] Schuld, M., Sinayskiy, I., & Petruccione, F. (2014). The quest for a quantum neural network. *Quantum Information Processing*, 13(11), 2567–2586. <https://doi.org/10.1007/s11128-014-0861-2>