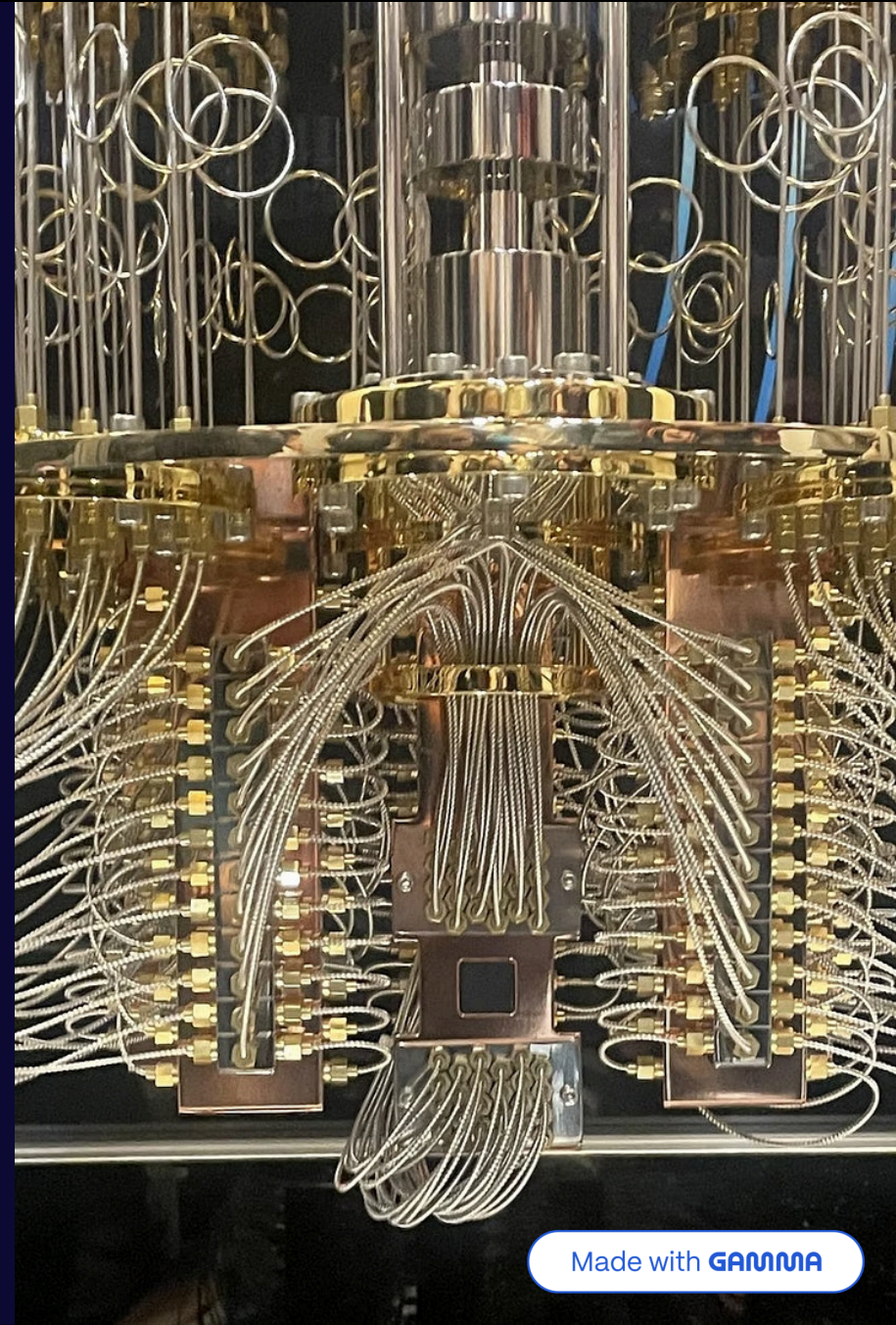


Clasificador Cuántico Variacional (VQC)



Descripción del Problema

1

Problema

- Clasificación binaria de datos no separables linealmente
- Requiere de capacidad de clasificación no lineal

2

Dataset

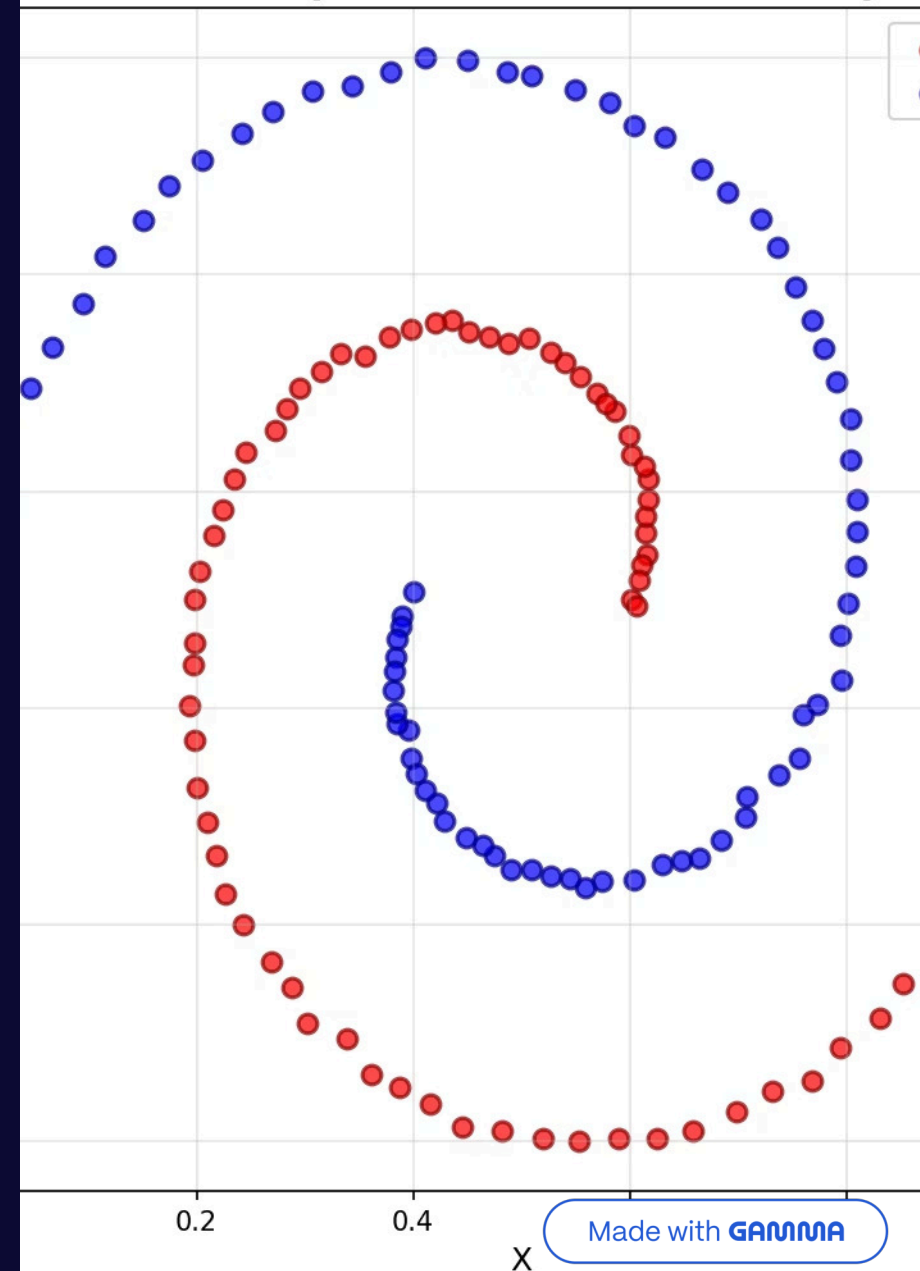
- Espirales entrelazadas rotando en direcciones opuestas
- 150 puntos, 2 clases
- Dataset típico de Scikit-Learn para ML

3

Matemática

- Ruido Gaussiano ($\sigma = 0.1$) + normalización $[0,1]$
- Clase 0: $r = 20 + \pi$
- Clase 1: $r = 20 - \pi$

Dataset de Espirales Entrelazadas (150 puntos)



Planteamiento Inicial

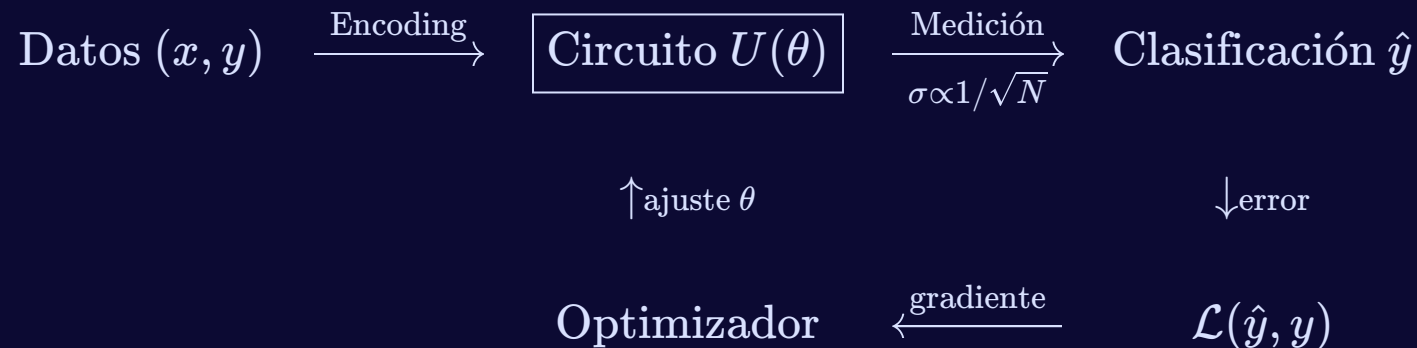
¿Qué es un VQC?

- Algoritmo **híbrido** que combina computación Clásica y Cuántica
- Circuitos cuánticos parametrizados para aprender patrones en datos
- Análogo cuántico a redes neuronales clásicas

Motivación

- Los computadores cuánticos pueden explorar espacios de características de alta dimensión
- Entrelazamiento cuántico = correlaciones no clásicas
- Potencial ventaja en problemas no lineales

Diagrama Conceptual Simple



Implementación Clásica - Componentes Híbridos

Preprocesamiento de Datos

- Generación del dataset
- Normalización min-max a rango [0, 1]
- Train/validation split 80/20
- Shuffle aleatorio

Función de Coste

- Basada en accuracy: $cost = 1 - accuracy$
- Evaluada en conjunto de validación
- Llamadas al circuito: aprox 4.6M de ejecuciones totales

Visualización y Métricas

- Fronteras de decisión
- Convergencia del entrenamiento
- Matrices de confusión, precisión y recall

1

2

3

4

5

Optimizador Clásico - COBYLA

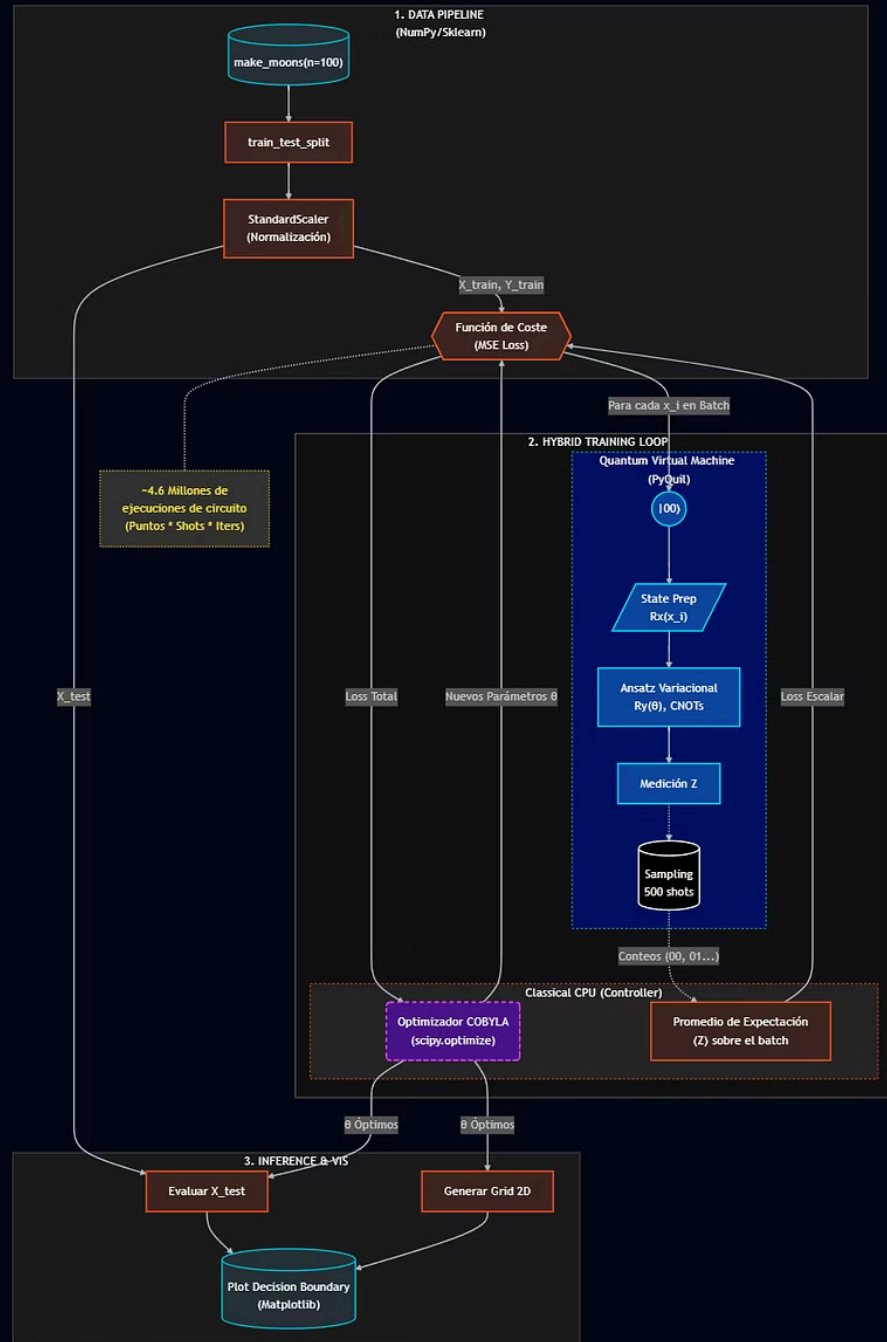
- Método: Constrained Optimization BY Linear Approximations
- Sin gradientes
- **Por qué:** robusto ante ruido cuántico:

$$\sigma_{\text{ruido}} \propto \frac{1}{\sqrt{N_{\text{shots}}}}$$

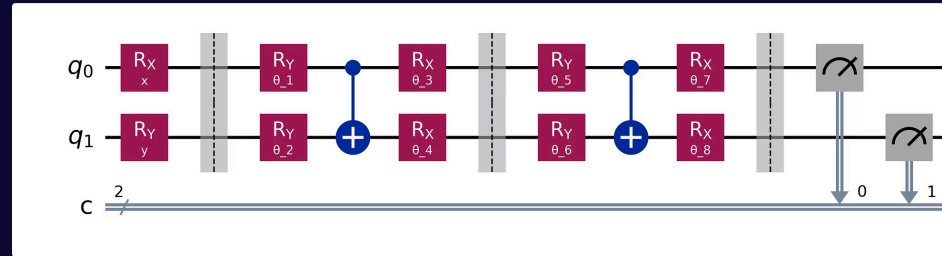
Training Loop Iterativo

Para cada iteración:

1. COBYLA propone nuevos parámetros θ
2. Circuito cuántico evalúa con $\theta \rightarrow$ predicciones
3. Calcular cost clásico (1 - accuracy)
4. COBYLA ajusta θ basado en cost
5. Repetir hasta convergencia o max_iter



Desarrollo Realizado - Circuito Cuántico



1

Encoding (Bloque Izquierdo)

Mapeo de Datos: Transforma el vector clásico (x, y) en un estado cuántico.

Estrategia: Dense Angle Encoding usando rotaciones ortogonales.

- q_0 : Aplica R_X proporcional a x
- q_1 : Aplica R_Y proporcional a y
- (Nota: Los inputs se escalan por 2π para cubrir la esfera de Bloch).

2

Ansatz Variacional (Bloque Central)

Estructura de 2 Capas: Diseñada para capturar relaciones no lineales.

Componentes:

- **Exploración:** Rotaciones entrenables $R_Y(\theta)$ y $R_X(\theta)$
- **Entrelazamiento:** Puertas CNOT (línea azul vertical) para correlacionar los qubits.

Eficiencia: Solo 8 parámetros ($\theta_1 \dots \theta_8$) y baja profundidad para evitar ruido (NISQ-friendly).

3

Medición (Bloque Derecho)

Readout: Medición en la base Z estándar.

Clasificación por Paridad:

- $|00\rangle, |01\rangle \rightarrow$ Clase 0
- $|10\rangle, |11\rangle \rightarrow$ Clase 1

Robustez: Votación mayoritaria sobre 500 shots (ejecuciones).

Desarrollo Realizado - Optimizaciones Clave

01

Full Bloch Sphere Encoding - Rotaciones 2π

- Antes: $R_X(\pi \cdot x)$
- Después: $R_X(2\pi \cdot x)$
- Impacto: mejora separación en espacio de características cuánticas
- Estándar en literatura 2024/25 (Qiskit)

02

Estrategia de Medición Multiqubit

- Antes: Solo medir q_0 desperdiciando información de q_1
- Después: combinar mediciones de ambos qbits:

$|00\rangle, |01\rangle \rightarrow \text{Clase 0}$

$|10\rangle, |11\rangle \rightarrow \text{Clase 1}$

- Impacto: explora espacio de Hilbert 4D completo (2^2 qbits)
- Beneficio: fronteras de decisión más expresivas

03

Optimización de Ruido - El desafío de la medición

- **Problema:** El ruido estadístico con bajo muestreo (150-300 shots) generaba gradientes inestables y fallos de convergencia.
- **Solución:** Identificación experimental de **500 shots** como el umbral crítico ("sweet spot") para suavizar la función de coste.
- **Impacto:** Permitió al optimizador encontrar el mínimo global, elevando el accuracy del 66% al **80% (+13.3 puntos)** en tiempo viable (~29 min).

Resultados Obtenidos - Accuracy & Métricas

Tabla 1: Métricas del Modelo Cuántico (VQC)

Métrica	Resultado
Validation Accuracy	80.00%
Training Accuracy	78.33%
Generalización (Gap)	-1.67% (Excelente)
Convergencia	77 iters (~29 min)

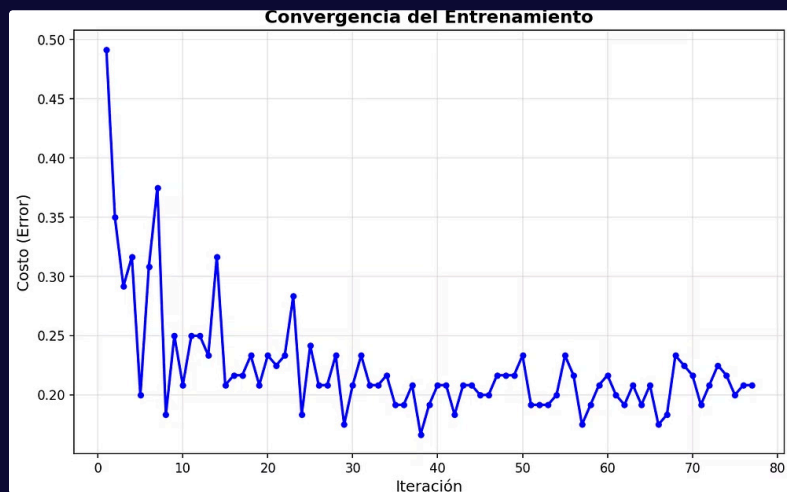
- Generalización Robusta:** El modelo no memoriza los datos (*overfitting* nulo), logrando mejor rendimiento en validación que en entrenamiento.
- Eficiencia de Entrenamiento:** Reducción del coste a 0.20 activando *early stopping* para evitar cómputo innecesario.

Tabla 2: Comparativa (Benchmark) vs Clásico

Modelo	Tipo	Accuracy
SVM (RBF Kernel)	Clásico (Baseline)	93.33%
VQC (Propio)	Cuántico (2 capas)	80.00%

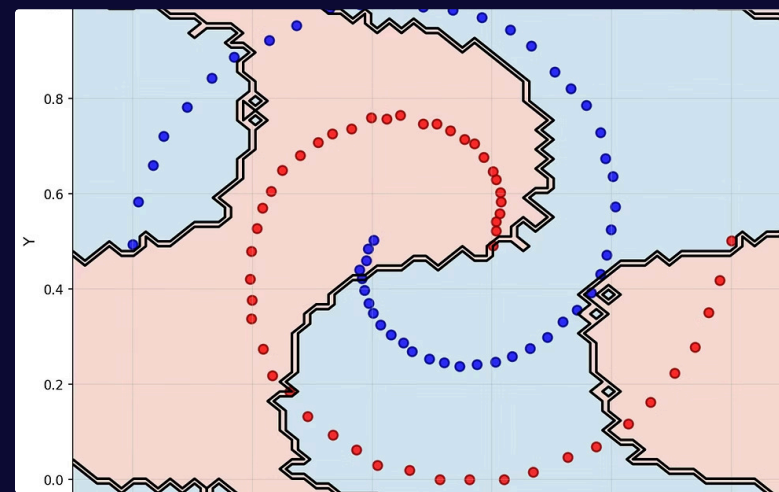
- Rendimiento Relativo:** El VQC alcanza el **86% de la capacidad** del clasificador clásico estándar (SVM).
- Competitividad NISQ:** Resultado sólido considerando las limitaciones actuales (ruido de disparo y circuito de baja profundidad) frente a un kernel matemático perfecto.

Resultados - Convergencia del Entrenamiento



Dinámica de convergencia

- **Descenso Robusto:** Reducción del coste de 0.42 a 0.20. La curva muestra cómo COBYLA logra minimizar el error a pesar del ruido inherente.
- **Estabilidad:** A diferencia de intentos fallidos con *Nelder-Mead*, aquí las oscilaciones son controladas gracias al muestreo de 500 shots.
- **Eficiencia:** El entrenamiento se detuvo automáticamente (Early Stopping) en la iteración 77, evitando cómputo innecesario.



Frontera de Decisión Cuántica

- **Resultados:** Se alcanza un **80% de Accuracy**, clasificando correctamente la mayoría de los puntos de validación.
- **No-Linealidad:** El circuito de solo 2 capas logra curvar el espacio para separar las dos "lunas", algo imposible para un clasificador lineal simple.
- **Generalización:** La frontera se ajusta a la forma de los datos sin memorizar el ruido (overfitting negativo de -1.67%), demostrando un aprendizaje real.

Conclusiones

1

Logros Principales

- **Viabilidad en No-Linealidad:** Éxito resolviendo el problema "Two Moons" (**80% Accuracy**) utilizando únicamente 2 capas y 2 qubits.
- **Competitividad Híbrida:** El modelo VQC alcanzó el **86% del rendimiento** de un SVM clásico (Kernel RBF), validando el enfoque NISQ.
- **Arquitectura Eficiente:** Se validó el *Hardware-Efficient Ansatz* como una solución robusta que maximiza la expresividad con el mínimo de puertas.

2

Hallazgos Clave

- **El "Sweet Spot" del Ruido:** Se demostró que **500 shots** es el umbral crítico que estabiliza los gradientes sin disparar el tiempo de cómputo.
- **Supremacía de COBYLA:** Los optimizadores *gradient-free* toleran mejor el "paisaje rugoso" de la función de coste cuántica que métodos como Nelder-Mead.
- **Poder Multiqubit:** La estrategia de medir la paridad de ambos qubits ($|00\rangle, |01\rangle$ vs $|10\rangle, |11\rangle$) fue vital para capturar correlaciones complejas.

3

Limitaciones Existentes

- **Coste Computacional (QVM):** La simulación en PyQuil requirió **~4.6 Millones de ejecuciones**, haciendo que 29 minutos para solo 150 puntos sea un desafío de escalabilidad.
- **Frontera "Rugosa":** La imagen de la frontera de decisión muestra bordes irregulares (pixelados) debido al *Shot Noise*, a diferencia de la suavidad matemática del SVM clásico.
- **Gap de Expresividad:** Un circuito de solo 2 capas (profundidad baja) limita la capacidad de suavizar completamente la curva de decisión en comparación con kernels clásicos infinitos.