



INSTITUTO POLITÉCNICO NACIONAL
ESFM-IPN, CIC-IPN



Circuitos cuánticos variacionales usando pennylane

III Mini Workshop on High Performance Computing in Science and Engineering

Alejandro Monroy Azpeitia
Arturo Alejandro Arvizu Velazques
Roberto Navarro Arenas

Julio 30, 2024

Aprendizaje de máquina supervisado

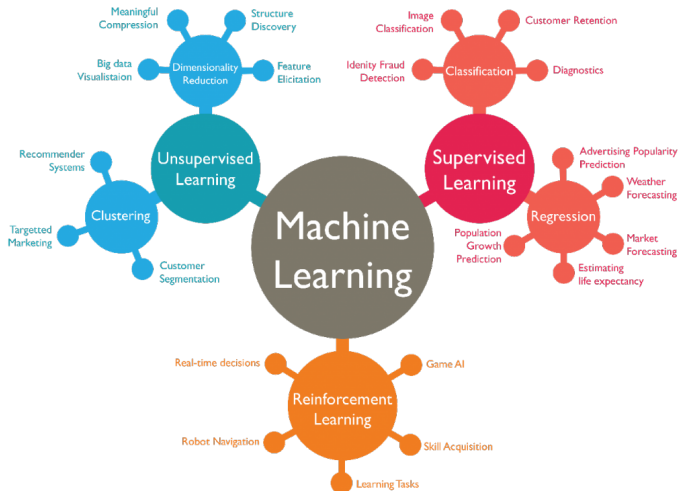
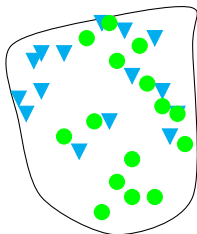


Figura 1: Categorías del aprendizaje de máquina: supervisado, no supervisado y aprendizaje por refuerzo

El aprendizaje de máquina *supervisado* considera datos como un conjunto de vectores-etiquetas

$$\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}.$$

Consideramos una tarea de *clasificación binaria*, i.e. las etiquetas se toman en un conjunto $\{a, b\}$.



$$\mathcal{D} = \mathcal{X} \times \mathcal{Y} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\} \times \{\pm 1\}$$

Figura 2: Conjunto de datos-etiquetas

Se busca una función o *modelo*, $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$, con un parámetro θ para el cual esperamos encontrar un buen θ^* que ajuste bien los datos con las etiquetas, es decir,

$$f(\mathbf{x}_i, \theta^*) \approx y_i, \quad \text{para cada } \mathbf{x}_i \in \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}.$$

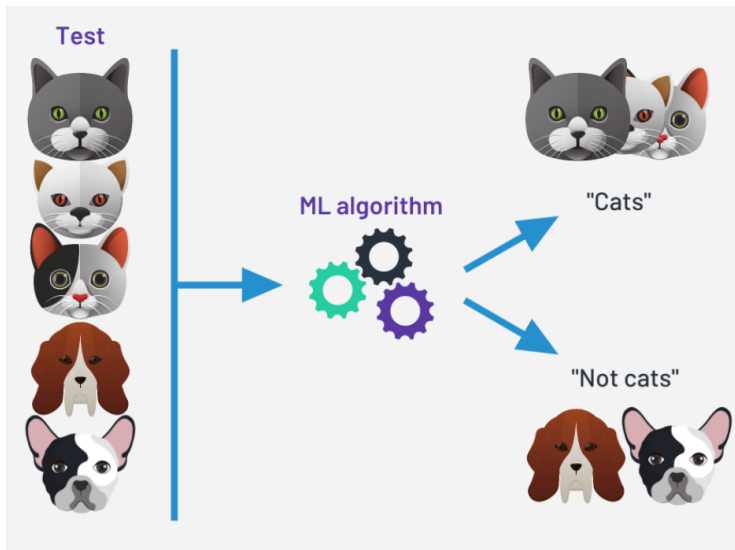


Figura 3: Características de un modelo de clasificación.

La función de costo $\ell(y_n, \hat{y}_n)$ toma como argumentos la etiqueta original y_n y la predicción hecha por el modelo \hat{y}_n y regrese un número real no-negativo el cual representa cuánto error se ha producido.

Medimos el promedio de la función de coste como:

$$\mathbb{L}(f, \mathcal{X}, \mathcal{Y}) = \frac{1}{N} \sum_{n=1}^N \ell(y_n, \hat{y}_n),$$

.

Máquinas de Soporte Vectorial

Para un conjunto de datos $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, una máquina de soporte vectorial es un clasificador que mediante un hiperplano, separa los datos en las 2 clases del conjunto de datos.

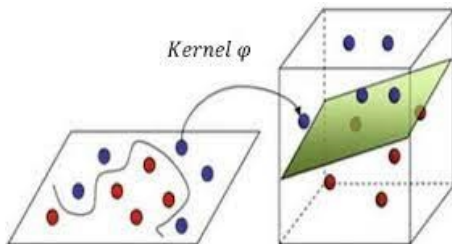


Figura 4: Máquinas de soporte vectorial (idea geométrica)

Observe que los datos no necesariamente admiten una separación mediante un hiperplano, por lo cual requerimos definir un mapeo que facilite tal separación.

¿Qué es un Kernel (M.L)?

Un *kernel* es un mapeo $\mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ para el cuál existe un espacio de Hilbert \mathcal{H} y un mapeo $\phi : \mathcal{X} \rightarrow \mathcal{H}$, usualmente llamado **Feature Map**, tal que:

$$K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{H}} \quad (1)$$

...

Es una función de similitud entre dos datos $\mathbf{x}_i, \mathbf{x}_j$ que involucra «representar» dichos datos en un espacio matemático posiblemente más abstracto y complejo, para usar la estructura de dicho espacio en el cálculo de la similitud.

Dado un data set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ y un modelo, se verifica¹ que si la función de costo \mathbb{L} cumple ciertas características, entonces cualquier función $f^* \in \mathcal{R}_K$ ² que minimiza a \mathbb{L} puede ser escrita como

$$f^*(\mathbf{x}) = \sum_{i_s} \alpha_{i_s} K(\mathbf{x}, \mathbf{x}_{i_s})$$

para ciertos parámetros $\alpha_i \in \mathbb{R}$.

Los vectores \mathbf{x}_{i_s} son llamados *vectores de soporte*.

¹Representator theorem

² $\mathcal{R}_K = \{f : \mathcal{X} \rightarrow \mathbb{C} | f(x) = \langle w, \phi(x) \rangle_{\mathcal{H}}, \forall x \in \mathcal{X}, w \in \mathcal{H}\}$

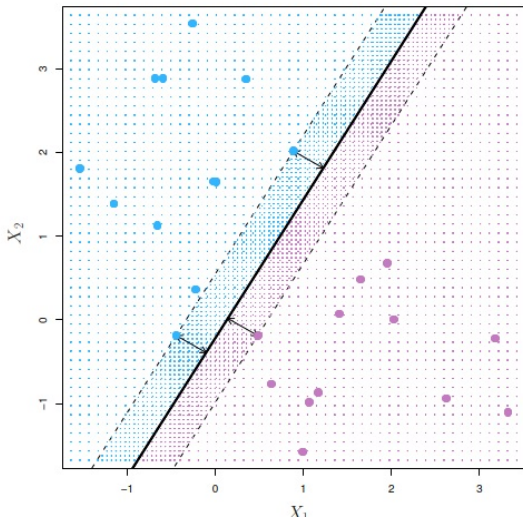


Figura 5: Vectores de soporte y margen de una SVM

¿Cómo podemos saber si elegimos un buen kernel?

El *kernel ideal* es el kernel K^{id} con entradas

$$K_{ij}^{id} = \begin{cases} 1 & \text{si } y_i = y_j \\ -1 & \text{si } y_i \neq y_j \end{cases}$$

La *alineación con el kernel ideal* mide la similitud entre kernels mediante un producto interno.

Circuito Cuántico Variacional

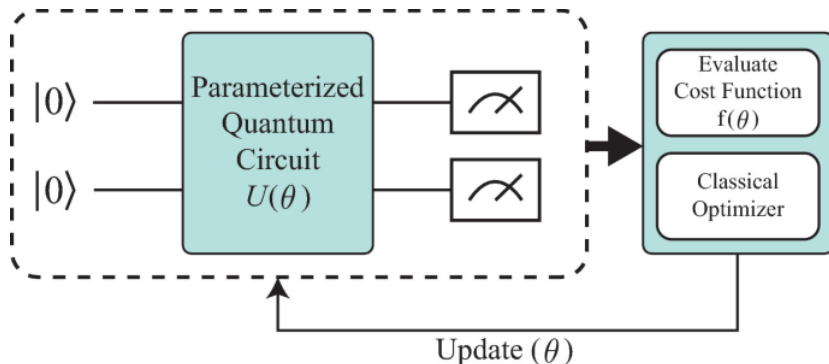


Figura 6: Un circuito cuántico variacional es entrenado por algoritmos clásicos de optimización para encontrar los parámetros θ adecuados.

Clasificador Cuántico Variacional

Un clasificador cuántico variacional se compone de tres partes:

- Codificación
- Evolución
- Medición

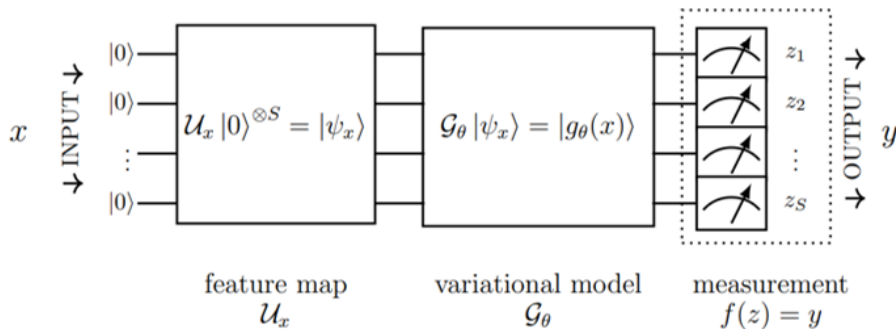


Figura 7: Un clasificador cuántico se compone de tres partes: codificación, parametrización y medición.

Clasificador Cuántico Variacional: Codificación

Los datos se codifican mediante un *quantum feature map*, el cual toma datos $x \in \mathbb{R}^D$ y los transforma a un espacio de Hilbert de D qubits.

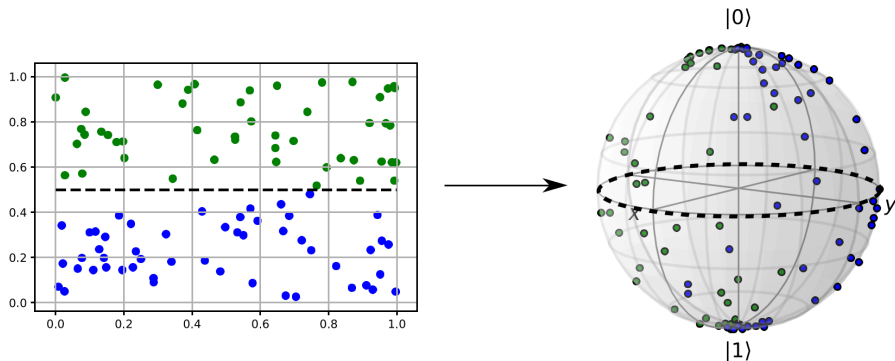


Figura 8: El feature map transforma los puntos de un espacio euclidiano a un espacio de Hilbert mediante un circuito cuántico.

La codificación se realiza mediante un circuito cuántico dependiente de los datos y posiblemente parametrizado:

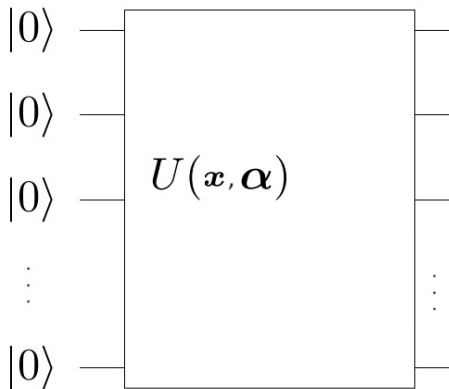


Figura 9: Circuito cuántico (variacional) codificador

¿Qué es una buena codificación?

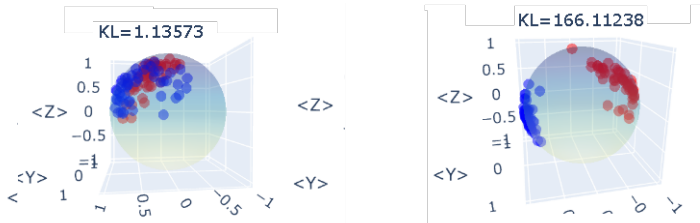


Figura 10: Mala codificación (izquierda) vs buena codificación (derecha). Intuitivamente y para los fines de clasificación una buena codificación hace más separables las clases.

Codificaciones por base y en amplitud

Dos ejemplos de codificaciones son la codificación por base y en amplitud.

Encoding	Input	Data	Qubit
Basis	Binario	$x^{(m)} = b_1, \dots, b_N$	$ x^{(m)}\rangle = b_1, \dots, b_N\rangle$
Amplitude	Vector	$\mathbf{x} = (x_1, \dots, x_N)$	$ \psi_x\rangle = \sum_{i=1}^N x_i i\rangle$

Cuadro 1: Codificaciones de base y amplitud para datos clásicos.

Clasificador Cuántico Variacional: Evolución y Medición

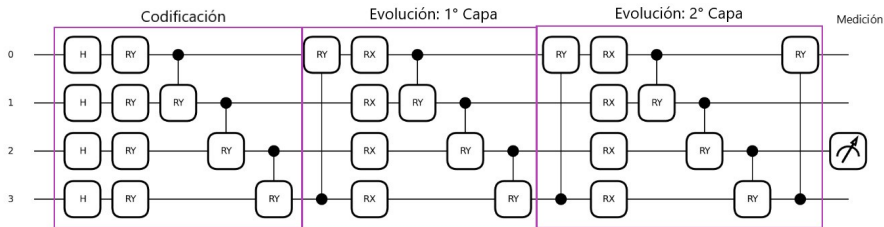


Figura 11: Ejemplo de modelos de evolución y medición de un clasificador variacional cuántico.

Quantum Long Short Term Memory (Q-LSTM)

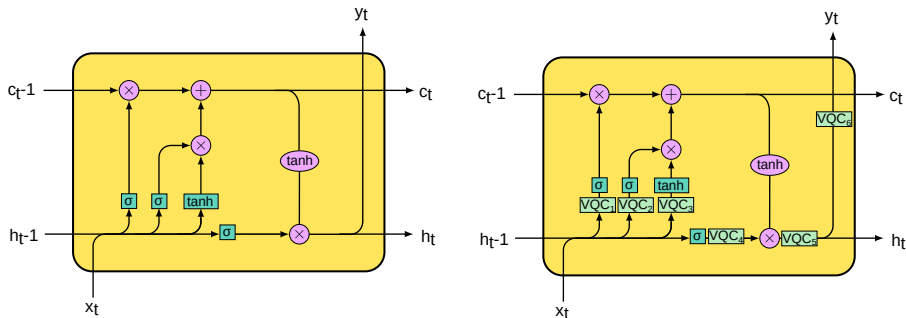


Figura 12: LSTM clásica (izquierda) y versión cuántica QLSTM (derecha).

Quantum Convolutional Neural Network (Q-CNN)

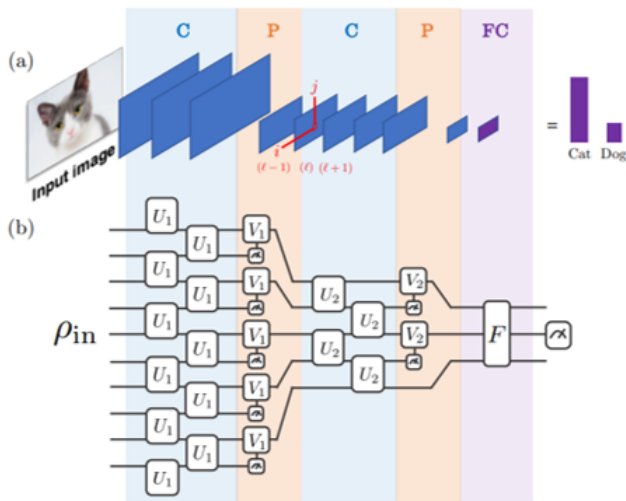


Figura 13: Red Neuronal Convolutiva cuántica.[14]

Quantum Support Vector Machines (Q-SVM)

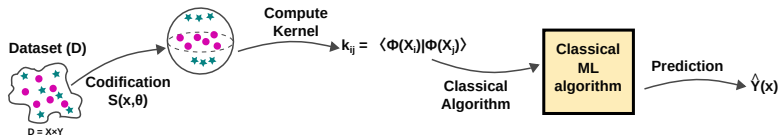


Figura 14: Diagrama versión cuántica máquina de soporte vectorial.

Optimizando la codificación

Parametrizando la codificación:

$$\hat{U}(x, \alpha) |0\rangle^{\otimes S} = |\phi(x, \alpha)\rangle \quad (2)$$

El producto interno entre dos datos codificados es una *medida de similitud* entre dos clases:

$$K_{ij}(\alpha) = \langle \phi(x_i, \alpha) | \phi(x_j, \alpha) \rangle \quad (3)$$

Formalmente es un Kernel. Podemos optimizar ese Kernel intentando que se parezca al *Kernel ideal*:

$$K_{ij}^{ideal} = y_i y_j = \begin{cases} 1 & \text{si } y_i = y_j \\ -1 & \text{si } y_i \neq y_j \end{cases} \quad (4)$$

Intentamos maximizar la alineación kernel objetivo:

$$A(K(\alpha), K^{id}) = \langle K(\alpha) | K^{id} \rangle \quad (5)$$

Optimizando la codificación

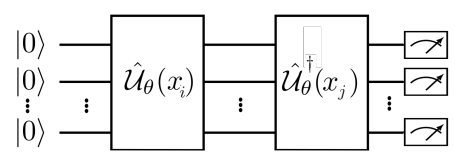


Figura 15: Circuito para el cálculo del producto interno entre dos datos codificados.

Los métodos de QML son clasificadores lineales

Codificación + Evolución (a.k.a Parametrización) + Medición = Método Lineal

Si al codificar dato + parametrización:

$$|\Psi(\mathbf{x}, \theta)\rangle = \hat{W}_\theta |\Psi(\mathbf{x})\rangle = \hat{W}_\theta \hat{U}(\mathbf{x}) |0\rangle^k \quad (6)$$

Al considerar la medición \hat{M} :

$$f_\theta(\mathbf{x}) = \langle \Psi(\mathbf{x}, \theta) | \hat{M} | \Psi(\mathbf{x}, \theta) \rangle \quad (7)$$

Formalmente **los clasificadores variacionales son métodos lineales**:

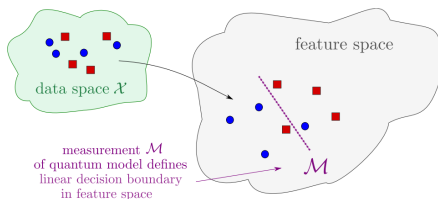


Figura 16: Imagen tomada de *Quantum machine learning models are kernel method*, M. Schuld et al.

Los métodos de QML (variacional) son métodos de Kernel

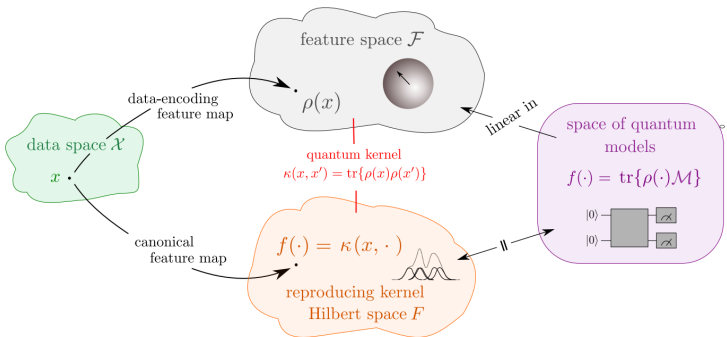
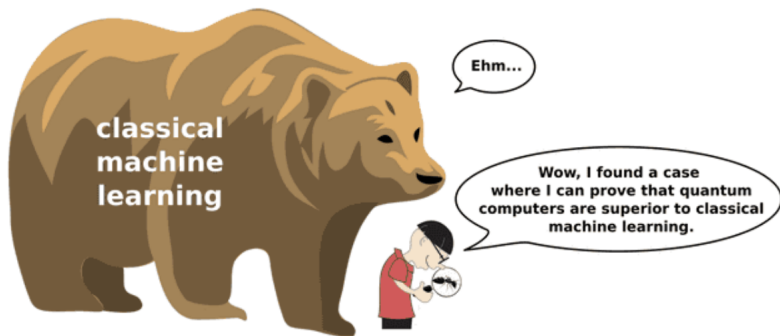


Figura 17: Relación entre QML variacionales y los métodos de Kernel

¿Por qué aprendizaje de máquina cuántico?



Algoritmos de aprendizaje de máquina clásico vs algoritmos de aprendizaje de máquina cuánticos.³

³Why measuring performance is our biggest blind spot in quantum machine learning , PennyLane post.

Aprendizaje de máquina cuántico: Problemas actuales y realidades

- Barren Plateaus: *Problemas en el optimizar*
- Exponential Concentration: *Modelos que tienden a no hacer nada*
- ¿Qué es una buena codificación?
- ¿Cuál es un buen circuito variacional?
- **NISQ era**: Aún no tenemos suficientes qubits de *buena calidad*

Referencias

- 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 Huang, HY., Broughton, M., Mohseni, M. et al. Power of data in quantum machine learning. Nat Commun 12, 2631 (2021)
- 6 Bernhard Schölkopf and Alexander J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. The MIT Press (2018)
- 7 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).

- 8 T. Hastie, R. Tibshirani and J. Friedman, An Introduction to Statistical Learning, Springer Texts in Statistics (2013).
- 9 Bishop, C. M., Pattern recognition and machine learning (Information science and statistics). Springer-Verlag New York, Inc (2006)
- 9 Learning with kernels
- 10 Michael A. Nielsen & Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge 10th ed (2010)
- 11 Ryan LaRose and Brian Coyle , Robust data encodings for quantum classifiers, Phys. Rev. A 102, 032420 (2020)
- 12 Thomas M. Cover & Joy A. Thomas, Elements of Information Theory, John Wiley & Sons, 2nd ed (2006)
- 13 Mark M. Wilde, Quantum Information Theory, Cambridge University Press 2nd edition (2017)
- 14 Cong, I., Choi, S. Lukin, M.D. Quantum convolutional neural networks. Nat. Phys. 15, 1273–1278 (2019).

Club de Computación Cuántica ESFM - ESCOM

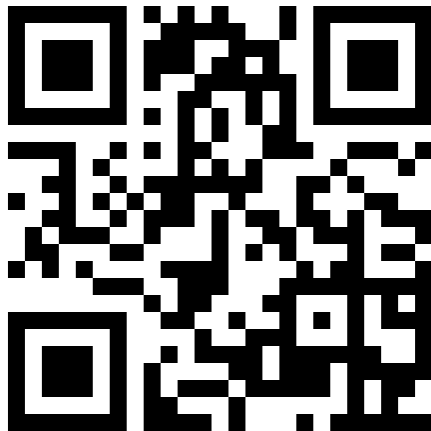


Figura 18: Invitación al servidor en Discord