



UNIVERSIDAD VERACRUZANA

**Estudio de algoritmos cuánticos  
para la resolución de problemas de  
aprendizaje supervisado.**

**T E S I S**

Como requisito para obtener el grado de

**Maestría en Inteligencia Artificial**

P R E S E N T A

**Emmanuel Isaac Juárez Caballero**

Asesor:

Dr. Horacio Tapia McClung

Xalapa Enríquez, Veracruz

Junio 2023



*“We are just an advanced breed of monkeys on a minor planet of a very average star. But we can understand the Universe. That makes us something very special.”*

— **S. Hawking**

## Agradecimientos

- A Rosa, mi mamá, que no pudo ver ninguno de los logros académicos más importantes que tuve, todo esto es por ti Ma, a ti que sigues aquí ayudando y permitiendo que todo esto sea posible.
- A mis abuelas, Josefina y Guadalupe, quienes no faltaron nunca, siempre me apoyaron y que sin ellas nada de esto hubiera sido posible, gracias por todo su esfuerzo, gracias por acompañarme en este proceso.
- A mi tía, Marisol, mi segunda mamá, gracias por estar ahí siempre, por todo el cariño, por no dejarme solo en ningún momento.
- A Pao e Irvin, por ser la razón de que siga, por no dejar que me rinda, por pese a todo intentar seguir con la vida, gracias.
- A Ro y Sari, por ser mis hermanas de otra madre, gracias por todo.
- A mi director de tesis, Horacio Tapia; un gran ser humano y director de tesis, por compartir el conocimiento de una forma tan amena y por impulsarme siempre a convertirme en un mejor estudiante y científico.
- A Sergio Lerma, gracias por permitirme llevar a cabo la estancia de investigación contigo, gracias por impulsarme a descubrir otras áreas del conocimiento y por darme la oportunidad de trabajar en conjunto, gran parte de este trabajo también fue gracias a todo ese apoyo.
- A Emmanuel, por intentar estar, ojalá en el futuro todo sea diferente, aún así, gracias por todo.
- A Sol, por ser una gran persona y maestra, por acompañarme desde la licenciatura y apoyarme de mil y un maneras.
- A mis amigos y amigas, a Azalea; por ser una persona increíble y acompañarme en todo momento, a Jen porque fue mi compañera de maestría e hizo que este viaje fuese

más llevadero, a Lilibeth; que a la distancia se preocupaba por mi vida estudiantil y estuvo presente, a Daniel, David, Josué y Manuel porque desde la licenciatura me han acompañado y dejaron de ser colegas para ser compañeros de vida.

- A mis alumnos, que me proveían de un lugar en donde despejarme cuando las cosas se ponían complicadas, por enseñarme muchas más cosas de las que yo pude enseñarles a ellos y por siempre mantener un buen ánimo.
- A todos los que de alguna manera estuvieron presentes pero que por extensión me es imposible agradecer, siempre, siempre estarán en mi y en cada pedacito que haga.



# Resumen

En este trabajo se presenta un análisis experimental de los algoritmos cuánticos (VQC, QNN y QSVC) en la resolución de problemas de clasificación mediante el uso de aprendizaje supervisado mediante la comparación del valor de precisión en presencia y ausencia de ruido de Pauli. Se obtuvo que la acción de ruido de Pauli tiene una repercusión distinta según el conjunto de datos, teniendo un impacto distinto en conjuntos hechos a medida que en conjuntos de datos del estado del arte. Se observó que, en general, el clasificador variacional cuántico se adaptó de mejor manera a las ejecuciones, en cambio los enfoques híbridos tuvieron un funcionamiento deficiente en los conjuntos de datos usados y la presencia de ruido no es significativa respecto del valor de precisión. Se observó que en el caso del VQC el ruido tendía a mantener o mejorar el valor de precisión en un promedio que oscila entre 2% y 8% .

**Palabras clave:** Quantum Machine Learning, Quantum Computing, Quantum Information.





# Índice general

Índice de figuras	XI
Índice de tablas	XV
Antecedentes	1
<b>1 Fundamentos preliminares: Teoría de la información y computabilidad.</b>	<b>11</b>
1.1 Representación de la información . . . . .	13
1.1.1 Errores, corrección de errores, mitigación de errores. . . . .	14
1.1.2 Entropía. . . . .	16
1.2 Modelo de compuertas lógicas . . . . .	19
1.2.1 Compuertas lógicas . . . . .	21
1.2.2 Máquinas de Turing: algoritmos computables y no computables. . . . .	26
1.2.3 Complejidad de los algoritmos . . . . .	28
<b>2 Fundamentos de la computación cuántica</b>	<b>31</b>
2.1 ¿Qué es una computadora cuántica? . . . . .	32
2.2 Elementos previos. . . . .	33
2.2.1 Álgebra lineal . . . . .	34
2.2.2 Postulados de la mecánica cuántica [27] . . . . .	37
2.3 Sistemas de dos niveles y computación. . . . .	40
2.4 Compuertas cuánticas . . . . .	42
2.4.1 Complejidad de un algoritmo cuántico . . . . .	49
<b>3 Quantum Machine Learning.</b>	<b>51</b>

3.1	Algoritmos de QML . . . . .	52
3.1.1	Circuitos cuánticos variacionales. . . . .	52
3.2	Componentes de un modelo de QML. . . . .	55
3.2.1	Ansatz . . . . .	55
3.2.2	Feature Map . . . . .	56
3.3	Algoritmos variacionales cuánticos de aprendizaje supervisado. . . . .	57
3.3.1	QNN y Clasificador variacional cuántico (VQC). . . . .	58
3.3.2	Clasificador de soporte vectorial cuántico (QSVC) . . . . .	60
3.4	Ruido y técnicas de mitigación. . . . .	61
3.4.1	Ruido de Pauli . . . . .	61
3.4.2	Métodos de mitigación de ruido . . . . .	63
3.4.3	Uso del sampler en Qiskit. . . . .	63
<b>4</b>	<b>Configuración del experimento y resultados.</b>	<b>65</b>
4.0.1	Procesamiento de datos y conjuntos de datos usados. . . . .	65
4.0.2	Funcionamiento del Script . . . . .	66
4.0.3	Parámetros usados en los algoritmos. . . . .	67
4.0.4	Metodología de análisis . . . . .	67
4.1	VQC . . . . .	69
4.1.1	Conjunto de datos de Iris. . . . .	73
4.1.2	Conjunto de datos sintéticos. . . . .	76
4.1.3	Conjunto de datos de crédito. . . . .	78
4.2	Enfoques híbridos . . . . .	81
4.2.1	QNN . . . . .	81
4.2.2	QSVC . . . . .	83
	<b>Conclusiones</b>	<b>87</b>
4.3	Trabajo a futuro. . . . .	89
	<b>Referencias</b>	<b>91</b>

# Índice de figuras

1.1	Esquema de la transmisión de información de un punto a otro mediante el uso de un canal ruidoso. El mensajero envía un mensaje $\varphi$ , el receptor recibe un mensaje $\varphi'$ , esperando que los mensajes sean semejantes o exactamente iguales.	12
1.2	Diagrama que muestra la transición de cada elemento binario en función de la probabilidad de que el ruido haga un intercambio o no del valor del bit.	15
1.3	Esquema que ilustra el comportamiento de la acción de los circuitos bajo sus dos operaciones elementales, la suma y la multiplicación, como fue descrito en (1.1)	22
1.4	Esquema que ilustra la acción de la compuerta Identidad ante una entrada.	22
1.5	Esquema que ilustra la acción de la compuerta NOT ante una entrada.	23
1.6	Esquema que ilustra la acción de la compuerta AND ante la entrada de dos proposiciones.	23
1.7	Esquema que ilustra la acción de la compuerta OR ante la entrada de dos proposiciones.	24
1.8	Esquema que ilustra la acción de la compuerta XOR ante la entrada de dos proposiciones.	24
1.9	Resumen de la acción de las compuertas lógicas sobre alguna entrada.	25
1.10	Ejemplificación del tiempo de ejecución en función de la cantidad de operaciones que debe realizar un algoritmo.	29
1.11	Esquema que ilustra el comportamiento de los problemas en los conjuntos P y NP, en el caso de que $P \neq NP$ y $P = NP$ .	30

2.1	Parametrización de la esfera de Bloch, el vector $\mathbf{P}$ se conoce como vector de Bloch, se muestran los ángulos correspondientes a la parametrización. Imagen tomada de Grynberg, Aspect, Fabre et al.[28] . . . . .	42
2.2	Representación de la esfera de Bloch en términos de los estados base de un qubit y de los estados $ \pm\rangle,  \pm i\rangle$ que resultan como una combinación lineal de $ 0\rangle,  1\rangle$ . Imagen tomada de Wong[29] . . . . .	43
2.3	Representación de la acción de la compuerta de Pauli en X sobre la esfera de Bloch. Imagen tomada de Wong[29] . . . . .	44
2.4	Representación de la acción de la compuerta de Pauli en Y sobre la esfera de Bloch. Imagen tomada de Wong[29] . . . . .	45
2.5	Representación de la acción de la compuerta de Pauli en Z sobre la esfera de Bloch. Imagen tomada de Wong[29] . . . . .	45
2.6	Representación de la acción de la compuerta de Hadamard sobre la esfera de Bloch. Imagen tomada de Wong[29] . . . . .	46
2.7	Equivalencia de las compuertas clásicas con compuertas cuánticas. Nótese que no todas las compuertas mostradas son reversibles y que la aridad en el caso clásico y el cuántico no coincide. Imagen tomada de Wong[29] pp. 170. . . . .	48
2.8	Estructura de un circuito cuántico, las líneas grises denotan los cables o rieles, las compuertas cuánticas actúan en cada qubit. . . . .	50
3.1	Ejemplo de un mapa de características ZZ (a la izquierda) usado para codificar cuatro características, requiere 4 qubits para llevar a cabo este proceso, se observa esta dependencia de los valores de las características $x_i$ . A la derecha se muestra el Ansatz Two Local, se observa la dependencia de los parametros $\theta_i$ y la expresión compacta. . . . .	57
3.2	Representación de la estructura de una red neuronal cuántica. Imagen tomada de Kwak, Yun, Jung et al.[37] . . . . .	60
4.1	Diagrama de la metodología de análisis. . . . .	68
4.2	Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos ad_hoc mediante la acción de ruido en el mapa de características. . . . .	70

---

4.3	Distribución por cuartiles del valor de precisión del modelo en ausencia de ruido en el mapa de características. . . . .	70
4.4	Distribución por cuartiles del valor de precisión del modelo en presencia de ruido en el mapa de características. . . . .	71
4.5	Distribución de la precisión del modelo en presencia y en ausencia de ruido en el ansatz para el conjunto de datos ad_hoc. . . . .	72
4.6	Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos ad_hoc en ausencia de ruido en el mapa de características. . . . .	73
4.7	Distribución por cuartiles del valor de precisión del modelo en presencia de ruido en el mapa de características. . . . .	73
4.8	Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos de Iris mediante la acción de ruido en el mapa de características. . . . .	74
4.9	Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos Iris mediante la acción de ruido en el ansatz. . . . .	75
4.10	Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos sintéticos generados con Sklearn mediante la acción de ruido en el mapa de características. . . . .	77
4.11	Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos sintéticos mediante la acción de ruido en el mapa de características. . . . .	78
4.12	Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos de Credit mediante la acción de ruido en el mapa de características. . . . .	79
4.13	Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos de crédito mediante la acción de ruido en el ansatz. . . . .	80



# Índice de tablas

1.1	Equivalencia entre el cálculo de proposiciones y la interpretación de circuitos de Shannon, tabla tomada de Shannon[21]. . . . .	21
2.1	Equivalencias de notación en el álgebra tradicional con la notación de dirac. La notación de Dirac siempre considera que la acción de un operador es hacia	38
4.1	Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos ad.hoc con la comparación de la acción de ruido en el mapa de características (véase 4.6). . . . .	69
4.2	Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido. . . . .	69
4.3	Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos ad.hoc con la comparación de la acción de ruido en el mapa de características (véase 4.5). . . . .	71
4.4	Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido. . . . .	71
4.5	Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de Iris con la acción de ruido en el mapa de características. (véase 4.8) . . . . .	73
4.6	Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido. . . . .	74
4.7	Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de Iris. (véase 4.9) . . . . .	75
4.8	Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido. . . . .	75

4.9	Distribución del valor de precisión en el conjunto de datos sintéticos (véase 4.9)	76
4.10	Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido. . . . .	76
4.11	Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de datos sintéticos generados con sklearn(véase 4.11) . . . . .	77
4.12	Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido. . . . .	77
4.13	Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de credito con la acción de ruido en el mapa de características. (véase 4.12) . . . . .	79
4.14	Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido. . . . .	79
4.15	Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de credit. (véase 4.13) . . . . .	80
4.16	Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido. . . . .	81
4.17	Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos de Iris. . . . .	81
4.18	Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos de ad_hoc. . . . .	82
4.19	Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos de credit. . . . .	82
4.20	Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos sintéticos. . . . .	83
4.21	Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QSVC para el conjunto de datos Iris. . . . .	83
4.22	Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QSVC para el conjunto de datos ad_hoc. . . . .	84
4.23	Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos Credit. . . . .	84



---

4.24	Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos sintéticos. . . . .	85
------	---	----



# Antecedentes

Una de las problemáticas históricas del cómputo radica en establecer medidas que garanticen que la información sea transmitida adecuadamente, este tipo de problemáticas dan origen a lo que se conoce como teoría de la información. En el caso del cómputo tradicional los elementos desarrollados por Shannon [1] son usados para estimar la precisión de la información transmitida y para combatir estos errores en el cómputo tradicional se suelen usar metodologías conocidas como correcciones de errores como la conocida codificación de Hamming [2].

En las últimas décadas el surgimiento del cómputo cuántico, un modelo para llevar a cabo operaciones aprovechando la naturaleza cuántica del universo ha tomado relevancia debido a muestras experimentales de su funcionamiento en donde bajo condiciones específicas pareciera superar en rapidez a su contraparte clásica. La idea de llevar a cabo cómputo modelándolo mediante las mismas reglas que sigue cualquier sistema es de particular interés, no obstante esta aproximación a la computación no es ideal.

Cuando se trata de implementar cómputo cuántico sucede algo análogo al tradicional, la transmisión de la información no está garantizada por una diversidad de motivos que generan ruido en la ejecución, dentro de los cuales destaca la decoherencia cuántica de los elementos, el tipo de computador cuántico, etc.

Esta problemática es abordada de diversas maneras, una de ellas consiste en llevar a cabo modificaciones en el hardware de la computadora cuántica de manera que los errores generados por la misma sean disminuidos, sin embargo, esta es una limitante en las computadoras cuánticas de cómputo general pues estos pueden ser originados por una diversidad de fuentes; mediciones, procesos térmicos que afectan al cómputo cuántico, entre otros.

Una de las alternativas utilizadas para poder tratar los errores consiste en hacer modificaciones a nivel de implementación con protocolos clásicos y cuánticos de manejo de la información entre los que destacan (M3, ZNE, PEC) [3-5], estas técnicas son conocidas como **mitigación de errores** y son utilizadas para tratar de prevenir y lidiar los errores que pueden ocurrir durante el procesamiento de la información en una computadora cuántica. El funcionamiento de estas es variado, no obstante, algunas de las más destacadas consisten en: realizar un análisis probabilístico clásico de la medición (resultado del cómputo) [4], considerar un análisis estadístico de las mediciones y amplificar estas para que comparativamente destaquen los resultados de cómputo más probables [5], entre otras.

Una anotación importante es mencionar que estas técnicas no permiten llevar a cabo una corrección de errores, pues la diferencia esencial entre corregir y mitigar es la acción de estas, la aplicación de una de estos métodos de corrección permite garantizar que no suceden errores al final de la transmisión cuando la información llegue al receptor, en cambio, la mitigación permite llevar a cabo comunicación con algún error, tratando de garantizar que en la mayoría de las ejecuciones el resultado computado es correcto [6].

Para poder llevar a cabo la medición de la eficacia de estas técnicas es conveniente llevar a cabo comparación con observables que de alguna manera son análogas a las propuestas para el cómputo clásico en el área de teoría de la información, en el caso del cómputo cuántico algunas observables de interés son: la distancia de la traza, una medida que permite verificar como un estado físico se diferencia de otro al estar en un sistema físico; la fidelidad, una medida fuertemente convexa que caracteriza el comportamiento de un estado respecto a la evolución temporal del mismo; la fidelidad del entrelazamiento, medida que permite cuanti-

ficar el entrelazamiento a lo largo de un proceso mecánico-cuántico (algoritmo cuántico) que involucra dos sistemas[7].

La importancia de la presencia del error en algoritmos de QML no ha sido descrita a profundidad, artículos recientes como el propuesto por Du, Hsieh, Liu et al.[8] muestran que los errores que actúan pueden llevar a cabo mejoras en algoritmos cuánticos dedicados al aprendizaje máquina, no obstante, está bien documentado que la presencia de ruido puede afectar el funcionamiento de determinados elementos pertenecientes a un circuito cuánticoUrbanek, Nachman, Pascuzzi et al. [9], de manera que los algoritmos implementados pueden variar y consecuentemente no llevar a cabo el proceso deseado de aprendizaje.

## Hipótesis

- El valor de precisión de los algoritmos de machine learning cuántico es susceptible a la presencia de ruido.
- El comportamiento del valor de precisión tiene un impacto distinto si el ruido se aplica en el mapa de características o en el ansatz.

## Objetivos Generales

- Estudiar el comportamiento de la mitigación de errores mediante el uso de un Sampler, estimando la cuasi-distribución de probabilidad para llevar a cabo el ajuste de los modelos de QML y obtener valores de precisión.
- Describir el comportamiento del valor de precisión ante la presencia de ruido de Pauli en el mapa de características y en el ansatz.

## Objetivos específicos.

- Realizar una revisión bibliográfica del estado del arte del cómputo cuántico, los algoritmos cuánticos y sus fuentes de error al ser aplicados en problemas de ML y su comparación contra los algoritmos tradicionales.
  - Actividades: Lectura de los artículos comparativos e implementación de los mismos en un entorno adecuado (Qiskit Runtime).
- Proponer mejoras relacionadas al valor de precisión en una variedad de algoritmos cuánticos para problemas específicos.
- Comprobar el funcionamiento de los algoritmos con sus modificaciones en problemas de referencia en ML.
  - Actividades: Implementar los algoritmos para su funcionamiento en datasets de benchmark.

## Estructura de la tesis

- Capítulo 1. **Fundamentos preliminares.** Se presenta a modo de síntesis los elementos del cómputo clásico, justificándose en los fundamentos de la teoría de la información y la computabilidad.
- Capítulo 2: **Fundamentos de la computación cuántica.** Se presenta un desglose de los conceptos físicos más importantes en el área del cómputo cuántico, comparando mediante elementos análogos la equivalencia entre los fundamentos del cómputo clásico al cuántico. Se presenta una revisión del tipo de computadoras cuánticas, se muestra su clasificación por tecnología haciendo énfasis en las computadoras cuánticas de cómputo general y el modelo de implementación por circuitos cuánticos.
- Capítulo 3. **Algoritmos cuánticos de ML y modelos de mitigación.** Se muestra una recopilación de los algoritmos usados para llevar a cabo la implementación del aprendizaje máquina y búsqueda en ambos tipos de cómputo, se desglosa detalladamente los análogos cuánticos con su contra-parte cuántica. Se explican las técnicas de mitigación de errores más populares.
- Capítulo 4. **Configuración del experimento y análisis de resultados.** En este capítulo se detalla las modificaciones a los algoritmos usados. Además, se muestra la configuración del experimento y la delimitación de los alcances.  
En este capítulo se explican las comparaciones encontradas así como las métricas utilizadas para el análisis.
- Capítulo 5. **Discusión, conclusiones y trabajo a futuro.**

## Motivación y justificación

A lo largo del desarrollo histórico de las ciencias de la computación se han encontrado problemáticas a la vez que se encontraban soluciones algorítmicas a problemas diversos. En particular en últimos años una sub-área que ha surgido prometiendo dar soluciones a pro-

blemas complejos como las búsquedas, la clasificación/segmentación de datos, la predicción a partir de datos, entre otros fenómenos, es la que recibe por nombre *Inteligencia Artificial* y que ha tenido un fuerte desarrollo durante años recientes [10], en donde han surgido una diversidad de enfoques para llevar a cabo la solución de las problemáticas mencionadas con anterioridad, dentro de los que resaltan: el uso de bases de conocimiento, redes bayesianas informadas, entre otras como lo son las conocidas redes neuronales y sus variaciones. [11]

Dentro de la *inteligencia artificial* existen una diversidad sub-ramas dentro de las cuales resalta por sus aplicaciones a la ciencia básica y aplicada el área de **machine learning** o **aprendizaje máquina** por su traducción literal al español, esta área forma un cruce entre diversas ramas de la ciencia, pues puede considerarse un área perteneciente a las matemáticas, la física, ciencias computacionales y la inteligencia artificial.

En general, el área del aprendizaje máquina pretende llevar a cabo la construcción de algoritmos que favorezcan la posibilidad de que una computadora aprenda mediante diversas técnicas matemáticas, mismas que se ahondarán de manera progresiva a lo largo de este texto [12]. Dentro del área de ML<sup>1</sup> se proponen una diversidad de algoritmos que dan solución a problemas varios en los cuales se incluyen: la clasificación de conjuntos de datos, la segmentación de datos, la predicción de alguna(s) variable(s) en función de un conjunto de datos, entre otras.

Sin embargo, una de las limitantes con las que el área ha combatido han sido los alcances del cómputo actual, pues para la implementación de diversos algoritmos se requieren recursos que no están al alcance de todos.[13]

Las limitantes anteriores suponen un área de oportunidad para las ciencias computacionales en favor de la mejora y creación de algoritmos más óptimos.

---

<sup>1</sup>Abreviaremos como ML al aprendizaje máquina haciendo referencia a las siglas en inglés de Machine Learning, pues es la notación usual en la bibliografía especializada.



## ¿Cómo computamos?

A lo largo de los párrafos anteriores se ha mencionado de manera repetitiva el concepto de ciencias computacionales sin ahondar en los detalles que existen detrás del mismo. Si bien, las ciencias computacionales se entienden en la actualidad como una sub-área de las matemáticas discretas esto no ha sido históricamente así, pues la predominancia de lo que ahora conocemos como computadoras digitales sobre las computadoras analógicas no sucedió hasta el surgimiento de los modelos teóricos y aplicados generados a partir de los desarrollos de Turing, Church [14, 15].

Dicho esto es importante ahora desarrollar con detalle el concepto de las ciencias computacionales, sus alcances y sobre todo su área de aplicabilidad, de esta manera podemos decir que en general existen dos áreas principales de las ciencias computacionales, las cuales son: la computación teórica; misma que se dedica al estudio de los modelos de computación de manera abstracta y la computación aplicada; la cual explora la aplicabilidad de los conceptos teóricos partiendo de que en la realidad los recursos son finitos.

Respecto a la computación teórica podemos decir que primordialmente se compone por las áreas de autómatas; donde se llevan a cabo estudios teóricos acerca de la composición algorítmica de los dispositivos que son capaces de computar, teoría de la computabilidad; que de manera concisa se entiende como una extensión de la teoría de autómatas sobre las máquinas de Turing y, finalmente, teoría de la complejidad; que de manera resumida hace referencia al estudio de la complejidad en algoritmos.

En cuanto a la computación aplicada, podemos referirnos a ella como la colección de algoritmos *computables* que pueden ser implementados en los modelos teóricos, destacan de manera particular los compiladores, los analizadores sintácticos, los lenguajes de programación, etc.[16]

## ¿Podemos computar de mejor manera?

A partir de lo descrito anteriormente es posible que nos preguntemos ¿Cómo podemos mejorar la forma de llevar a cabo el cómputo? La respuesta a esta pregunta puede ser ambigua, pues desde el punto de vista teórico, la máquina de Turing y sus equivalentes son el modelo de cómputo más completo y robusto que conocemos actualmente, sin embargo, la pregunta puede ser válida si nos preguntamos como mejorar la eficiencia en tiempo y/o espacio de un algoritmo computable.

Si nos decidimos a investigar la segunda de las opciones podemos encontrar una variedad de alternativas para mejorar la complejidad de espacio y de tiempo de los algoritmos, en particular una alternativa que ha tenido auge en años recientes ha sido el llevar a cabo el cómputo de una manera esencialmente distinta, la cual consiste en usar la descripción de sistemas físicos, en particular sistemas mecánico-cuánticos de dos estados para emular el comportamiento clásico de un computador que a nivel más elemental consiste de operaciones de dos estados(binarias).

En la actualidad el surgimiento del cómputo cuántico ha supuesto un cambio de perspectiva en la computación teórica, pues gracias a los avances recientes [17] se han desarrollado algoritmos en estos sistemas de cómputo que prometen demostrar superioridad en la velocidad de ejecución y que debido a la construcción de los dispositivos de cuánticos ofertan una mejora en la capacidad de acceso de memoria. *A priori* esto puede suponer un cambio de paradigma, principalmente para todas aquellas áreas en donde el cómputo requiere de una cantidad gigantesca de recursos como lo es el Machine Learning, de esta manera han surgido aproximaciones para solucionar diversos problemas del área de ML a partir de la equivalencia con sistemas físicos los cuales pueden ser implementados en dispositivos de cómputo cuántico, los cuales serán abordados a detalle en el resto del texto.

No obstante, la implementación de este tipo de algoritmos requiere de una alta precisión en los resultados obtenidos, sin embargo, esta es una de las principales limitantes de los

sistemas de cómputo cuántico pues los sistemas de cómputo cuántico por naturaleza son sistemas que tienen errores asociados a la transmisión de la información y que por ende son menos *confiables* para llevar a cabo cómputo. En el caso de la implementación de un algoritmo cuántico en una computadora clásica (simulación de una computadora cuántica) muchas veces estos errores intrínsecos de la naturaleza cuántica del cómputo pasan inadvertidos.

Sin embargo, cuando las ejecuciones de los algoritmos se realizan directamente en dispositivos cuánticos la existencia de fenómenos físicos como la superposición y la naturaleza probabilística de la mecánica cuántica surgen y caracterizan el comportamiento de los resultados de los algoritmos. Existen una gran variedad de factores que condicionan el funcionamiento adecuado y de los algoritmos y que permiten a estos ser más confiables en cuanto al valor computado; algunas técnicas usadas para aumentar la precisión de un resultado consisten en: mejorar la tecnología a nivel de hardware, establecer técnicas de mitigación, supresión de errores, etc.



# 1

## Fundamentos preliminares: Teoría de la información y computabilidad.

Un aspecto importante para poder llevar cómputo de manera óptima consiste en que la información que fluye a lo largo de un sistema de cómputo lo haga de manera adecuada,<sup>1</sup> para esto existe una sub-área interdisciplinaria que tiene como origen la física, las matemáticas y la teoría de la comunicación, y que ahora conforma una parte importante de la aplicabilidad de las ciencias computacionales, el área que se encarga de llevar a cabo lo anterior se conoce como *teoría de la información*.

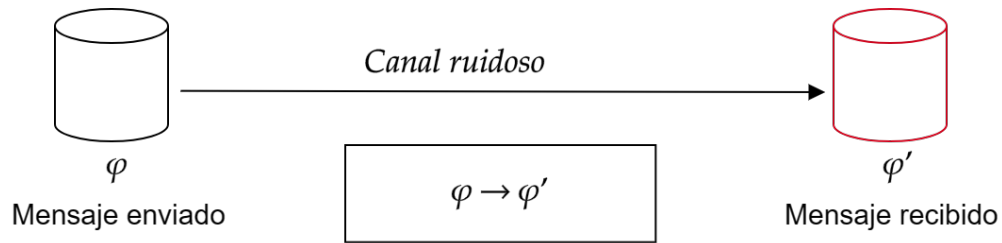
De manera más adecuada es posible referirse a la *teoría de la información* como el área que busca mejorar la compresión de la información cuando fluye a través de un medio y hallar la capacidad de transmitir información a partir de un canal de transmisión[18, 19]

El problema principal entonces consiste en encontrar un método tal que un mensaje creado en un punto sea reproducido de manera parcial o exacta en otro punto [1](véase 1.1), las implicaciones que tiene esto sobre la forma de hacer cómputo no son pocas y conforman un punto fundamental en la toma de decisiones para la construcción de algoritmos, no obstante,

---

<sup>1</sup>No es necesario que haya flujo activo de información, esta puede ser almacenada. Un caso particular es el caso de una memoria, en donde la información se guarda para posteriormente acceder a ella.

también es útil para poder llevar a cabo el diseño de dispositivos de cómputo.



**Figura 1.1.** Esquema de la transmisión de información de un punto a otro mediante el uso de un canal ruidoso. El mensajero envía un mensaje  $\varphi$ , el receptor recibe un mensaje  $\varphi'$ , esperando que los mensajes sean semejantes o exactamente iguales.

El origen de este tipo de errores en la transmisión no es único y se remonta a los procesos físicos involucrados en el proceso de comunicación, una forma de evidenciar la presencia de un canal ruidoso es considerar un caso en donde dos personas tratan de comunicarse mediante dispositivos de radio, usualmente las señales de radio se ven afectadas por el fenómeno de interferencia, pues las ondas que transmiten la información interactúan con otras en la misma región del espectro electromagnético, por consiguiente supongamos que la primera persona trata de comunicarle a otra que acaba de avistar un automóvil con placas ( $\varphi \rightarrow ABC$ ), debido a la presencia de interferencia y, en general del canal ruidoso puede que la persona no reciba el mensaje adecuado ( $\varphi' \rightarrow ABS$ ), de esta manera la comunicación no será eficaz, esto puede ser problemático o no dependiendo de la situación.

En el caso de que la comunicación sea entre personas meramente curiosas por saber las placas de vehículos el hecho de que la comunicación no sea efectiva no representa un problema de importancia, en cambio, suponiendo que esta interacción sea para la identificación de un automóvil de parte de las autoridades este error puede suponer un cambio importante en

la toma de decisiones.

Este tipo de problemas a la hora de llevar a cabo la comunicación genera nuevas alternativas para evitar la posibilidad de riesgo de error, una de las soluciones simples a esta problemática es considerar enviar más información para evitar el error, por ejemplo, podemos considerar decir  $\varphi \rightarrow ABC$  como  $\varphi_2 \rightarrow \text{alpha} - \text{beta} - \text{charlie}$ , este tipo de transformación es conocida como codificación y permite “sobredeterminar” el mensaje por transmitir, entre otras cosas este mensaje codificado nos permite reducir el error de transmisión, no obstante, a su vez requiere de que la transmisión de información sea mayor[19].

Es posible preguntarse si existe alguna forma de optimizar la comunicación de manera que la transmisión se realice con la mínima cantidad de información y de error de manera que el flujo de la misma sea rápido y confiable, una forma de abordar este problema es considerar la razón de cambio entre la probabilidad de que dada una cadena de entrada algún bit cambie de valor bajo la presencia de ruido, esta cantidad es distinta para cada modelo de comunicación.<sup>2</sup> En general, los códigos que resultan óptimos son determinados por propiedades estadísticas como la entropía, mismas que se describirán posteriormente a lo largo del texto.

## 1.1. Representación de la información

De acuerdo a lo expuesto con anterioridad una pregunta natural consiste en preguntarse cómo podemos representar la información en el cómputo. Para responder a lo anterior es importante recordar los orígenes de los dispositivos que funcionan a partir del cómputo digital; los cuales están implementados sobre dispositivos electrónicos digitales y que funcionan a partir de impulsos eléctricos con un valor de umbral el cual delimita la activación o no de un dispositivo, de esta manera que pueden suceder dos casos, el primero en donde el valor de umbral es superado y su complemento.

Esta cualidad de los sistemas digitales que tienen una salida con dos opciones se conoce como

---

<sup>2</sup>En el caso de la transmisión de información en un canal simétrico, el cual será discutido brevemente se tiene que el código de repetición óptimo es el conocido como  $\mathbf{R}_3$

un sistema binario, algo importante por resaltar es que uno puede nombrar a estas opciones de una manera arbitraria, por ejemplo, designando como  $a$  al caso en el que el dispositivo supera el umbral y por  $b$  al caso en donde no lo hace, no obstante una convención clásica por su utilidad es usar un uno en lugar de la  $a$  ( $1 \rightarrow a$ ) y un cero en lugar de la  $b$  ( $0 \rightarrow b$ ).

El uso del sistema numérico binario para llevar a cabo la *codificación* de la información no es casualidad, pues en 1854 Boole [20] presentó un desarrollo matemático que presentaba lo que hoy conocemos como el *álgebra de Boole* como parte de un sistema lógico y que fue precursor de avances como los propuestos posteriormente por Shannon [21].

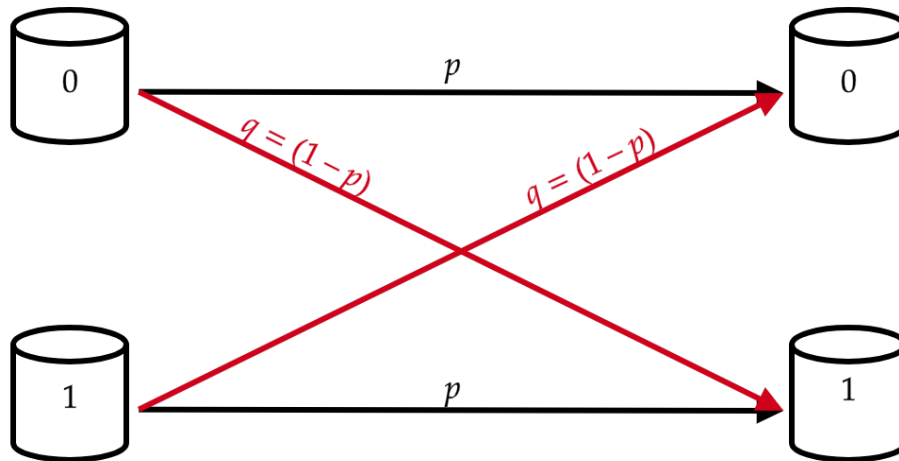
### 1.1.1. Errores, corrección de errores, mitigación de errores.

Como se comentó con anterioridad la presencia de un canal ruidoso puede generar que el mensaje original que intenta transmitirse no sea el recibido, por ejemplo, es posible considerar a una cadena binaria  $s$  compuesta por elementos  $\{0, 1\}$ .

Denominemos a la cadena  $s = \varphi$  como la cadena enviada por el emisor antes de ser transmitida por el canal ruidoso, y denominemos como  $\varphi'$  a la cadena transmitida, es decir, aquella que puede ser recibida por el receptor (véase 1.1), cada elemento de  $\varphi$  tiene una probabilidad de que el valor del elemento cambie, el cambio está modelado por un diagrama de transición binario, descrito por la probabilidad de que el elemento recibido sea igual al transmitido  $p$  y por la probabilidad de que el elemento recibido no sea igual transmitido  $q = (1 - p)$  [19], este modelado en particular es conocido como canal de transmisión simétrico.

El modelado de una transición binaria permite llevar a cabo el ajuste de distribuciones de probabilidad que describan al error en el cambio de bits en una cadena  $\varphi$  de longitud  $|\varphi| = n$ , debido a que cada elemento de la cadena  $s$  es binario es posible observar que la cantidad de errores está determinada por la suma de contribuciones de una distribución de probabilidad





**Figura 1.2.** Diagrama que muestra la transición de cada elemento binario en función de la probabilidad de que el ruido haga un intercambio o no del valor del bit.

binomial en función del ruido ( $p$ ):

$$P_T(p) = \sum_{k=0}^n \binom{n}{k} p^{n-k} (1-p)^k = \sum_{k=0}^n \binom{n}{k} p^{n-k} q^k \quad (1.1)$$

Es importante además notar que en la expresión ((1.1)) se considera el caso de que haya error en al menos un bit de la secuencia  $\varphi$ , no obstante, esta expresión se modifica si se considera algún tipo de codificación, por ejemplo, en el caso de la codificación de  $R_3$  o la determinada por los conocidos *códigos de Hamming* la expresión de la probabilidad de error cambia en función de estos, pues en ellos se consideran entre otros elementos los conocidos como *bits de paridad*.

No obstante, como se comentó previamente esta es solo una representación de un caso en particular en donde se considera un canal simétrico con probabilidades de error determinadas, en la realidad este modelo no permite llevar a cabo la descripción adecuada en los casos no simétricos o donde el ruido no esté determinado *a priori*.

### 1.1.2. Entropía.

Cuando consideramos la comunicación a lo largo de un canal ruidoso podemos llevar a cabo un enfoque probabilista para poder entender el comportamiento de las cadenas transmitidas, así como el de los posibles errores que tendrá nuestro mensaje, una forma común y válida de hacerlo es considerar al mensaje recibido como una variable aleatoria, es decir, la cadena que recibamos puede ser cualquier combinación de valores de nuestra codificación que mantenga la estructura (longitud, codificación) de la cadena de entrada. Bajo esta consideración podemos llevar a cabo comparativas entre las cadenas emitidas y las recibidas, esto es de particular importancia, por ejemplo, cuando se quiere observar en qué medida la cadena emitida y la recibida varían.

Para llevar a cabo un análisis completo del comportamiento probabilista de una variable aleatoria se suele llevar a cabo la consideración de una cantidad conocida como **ensamble**, de manera informal se puede definir a un ensamble como una colección de elementos que tienen una probabilidad determinada.

**Definición: Ensamble.**

Un ensamble  $X$  es una tripleta de elementos  $(x, \mathcal{A}_X, \mathcal{P}_X)$  donde:

- $x$  es el valor de la salida de una variable aleatoria,
- $\mathcal{A}_X$  es el alfabeto del ensamble compuesto por los elementos  $\mathcal{A}_X = \{a_1, a_2, a_3, \dots\}$ ,
- $\mathcal{P}_X$  es el conjunto de las probabilidades correspondientes a los elementos del alfabeto.

Es importante mencionar que un ensamble cumple con los axiomas de Kolmogorov [22]. Podemos suponer que el conjunto de salidas del ensamble nos arrojarán determinada información respecto del sistema de estudio, bajo esta consideración, podemos suponer que la información será una medida de la probabilidad con la que sucede un evento, es decir  $I(P(x = a_i)) = I(p_i)$ , en esta apreciación puede considerarse que el valor  $x$  es una “señal” o

un “mensaje” transmitido[22].

Una apreciación importante es que la información  $I(p)$  debe cumplir con ciertas propiedades que sean congruentes con la naturaleza de la probabilidad, de esta manera se consideran:

- La información  $I(p)$  debe ser aditiva si se consideran dos eventos independientes
 
$$I(p_1 p_2) = I(p_1) + I(p_2)$$
- La información  $I(p)$  debe ser no negativa,  $I(p) \geq 0$
- La información  $I(p)$  debe ser monótona, es decir, si  $p_1 > p_2$  se cumple que  $I(p_1) > I(p_2)$

Dadas estas propiedades es posible tratar de considerar funciones de “prueba” que cumplan con estas condiciones, es evidente que una función que se ajusta a la perfección es la función determinada por el inverso multiplicativo del logaritmo

$$I(p) = A \frac{1}{\log_a(p)} = -A \log_a(p),$$

donde  $A$ ,  $a$  son constantes que se determinarán según el contexto y en donde la única relevante es  $a$ , pues determinará el tipo de medida de información.<sup>3</sup> Es importante resaltar que esta definición de la información permite el estudio en dos casos límite:

- $p \rightarrow 0$ , en este caso  $I(p)$  se indetermina, pues diverge a infinito ( $\lim_{p \rightarrow \infty} I(p) \rightarrow \infty$ ), de esta manera, podemos considerar que un evento improbable nos brinda una gran cantidad de información.
- Otro caso importante es considerar el caso donde  $p = 1$ , en este caso  $I(p = 1) = 0$ , es decir, la información arrojada por un evento certero es nula.

Con estas consideraciones es posible y apropiado preguntarnos qué pasa si se consideran una colección de eventos que arrojan información respecto del sistema y que suceden con una probabilidad igual a la que es evaluada en la función de información. La respuesta a esta pregunta lleva el nombre de **entropía** y podemos definirla como la ponderación de la probabilidad de que suceda un evento dentro de un ensamble y la información transmitida.

---

<sup>3</sup>si se considera  $a = e$  los elementos medidos por la información reciben el nombre de “nats”, en cambio, si se usa  $a = 2$ , reciben el nombre de “bits”

**Entropía: Caso discreto**

La entropía de un ensamble  $X$  está determinada por el promedio ponderado de la colección de elementos del alfabeto y los valores de probabilidad que toman los mismos.

$$H(p_1, p_2, \dots, p_n) = \sum_{i=1}^n p_i I(p_i) = - \sum_{i=1}^n p_i \log_a(p_i)$$

En el caso discreto las probabilidades están determinadas para cada elemento no continuo del alfabeto, sin embargo, si se considera una aproximación en términos de una función de distribución de probabilidad para la variable aleatoria  $p$  se puede llevar a cabo una definición equivalente.

**Entropía: Caso continuo**

Sea una función de distribución de probabilidad  $f(x)$  que caracteriza el comportamiento de una variable  $x$  de un ensamble  $X$ , la entropía de la variable aleatoria está determinada por:

$$H(x) = - \int_{-\infty}^{\infty} f(x) \log_a f(x) dx$$

Una apreciación relevante consiste en analizar la interpretación de la entropía, comúnmente se hace referencia a que la entropía mide la cantidad de “desorden” de un sistema, no obstante definir las palabras “orden” o “desorden” es subjetivo y consecuentemente esta interpretación es errónea. En el contexto de las ciencias físicas la entropía cuantifica la relación de micro estados accesibles del sistema con un macro estado, por ejemplo, puede medir el aumento de la cantidad de micro estados energéticos mediante el aumento de temperatura de un sistema.

En el contexto de la teoría de la información la definición apropiada describe el grado de **incertidumbre**, bajo esta apreciación la entropía de un sistema en el contexto de teoría de la información permite determinar la certeza de un mensaje transmitido. Existen una variedad de métricas que permiten llevar a cabo la comparación directa entre cadenas (variables aleatorias) mediante el uso de entropía, una de las más comunes es conocida como la métrica de Kuller-Back [22].

## 1.2. Modelo de compuertas lógicas

A lo largo de la introducción al capítulo se describió la importancia de los resultados de Boole [20] quien en términos simples llevó a cabo una equivalencia a la lógica proposicional mediante la creación de un sistema formal en el cual los valores lógicos de verdadero/falso toman ahora los valores  $\{1, 0\}$  a partir de operaciones definidas formalmente, no obstante, el trabajo consecuente de Shannon[21] propuso el diseño de circuitos lógicos, en donde las operaciones descritas por Boole de manera abstracta ahora podían ser implementadas en dispositivos físicos.

Un hecho importante es que el trabajo de Boole al ser equivalente a la lógica proposicional puede expresarse de manera adecuada en lo que comúnmente recibe el nombre de tablas de verdad, además cumple con las operaciones lógicas definidas en la lógica proposicional: **identidad, disyunción, conjunción, negación, disyunción exclusiva, conjunción negada, etc.**<sup>4</sup>

Una anotación importante consiste en visualizar al menos de manera descriptiva el comportamiento de las operaciones la partir del número de argumentos,<sup>5</sup> que necesita una operación lógica para poder ser llevada a cabo. Por ejemplo, para llevar a cabo la negación de una proposición únicamente se requiere el operador de negación y una proposición, por lo que la cantidad de argumentos que toma es uno, por otro lado, si se considera la operación **AND** es posible notar que para que esta sea llevada a cabo necesita de al menos dos proposiciones, por lo que la operación es de aridad dos.

Estas observaciones son importantes pues para llevar a cabo la traducción del álgebra de Boole a dispositivos electrónicos las propiedades descritas con anterioridad deben mantenerse, de manera que se proponen de manera tal que respeten la aridad y el comportamiento del álgebra de Boole.

---

<sup>4</sup>En general nos referiremos a estas operaciones por sus nombres anglosajones, pues es lo usual en la literatura: **identity, OR, AND, NOT, XOR, NAND, etc.**

<sup>5</sup>En el contexto formal esto recibe el nombre de *aridad* y hace referencia al número de argumentos que puede tomar una función.

## Álgebra de Boole

Para poder introducir de manera adecuada el concepto de las compuertas lógicas es importante entender con anterioridad los conceptos del álgebra de Boole, en el presente texto se llevará a cabo un desglose informal acerca de los conceptos del álgebra de Boole, únicamente con la intención de que sea ameno a la lectura y evidente la equivalencia con las compuertas lógicas propuestas por Shannon. Para esto se utilizará una notación moderna y alternativa a la usada por Boole.

### **Definición: Álgebra de Boole.**

Un álgebra de Boole o de lógica binaria está definida como un conjunto de acciones sobre un conjunto  $\mathcal{B} = \{0, 1\}$  de tal manera que las operaciones  $\{\neg, +, \cdot\}^a$  están definidas de forma que la acción de alguna de las operaciones sobre el conjunto  $\mathcal{B}$  sea un elemento de  $\mathcal{B}$

<sup>a</sup>De manera alternativa e indistinta se suele usar la notación  $\{\sim, \oplus, \odot\}$  o una mezcla de ellas.

En donde las operaciones reciben los nombres de complemento, suma interna y producto interno respectivamente. Algo importante a notar es que para que la acción de las operaciones sea tal que el resultado quede en el mismo conjunto de partida la acción de las operaciones no es la usual, de hecho, se dice que la lógica binaria está construida sobre la aritmética de módulo dos, es decir que la suma sobre  $\mathcal{B}$  es tal que  $\oplus \rightarrow (+)_{mod_2}$  y el producto interno  $\odot \rightarrow (\cdot)_{mod_2}$ .

Bajo la notación presentada con anterioridad es posible establecer la relación del álgebra booleana con la lógica proposicional, tal que sean  $p, q$  proposiciones con valores de verdad  $\{V, F\}$  y de manera equivalente, sean  $a, b \in \mathcal{B}$  con valores  $\{1, 0\}$ , se sigue que

- $\neg p$  es equivalente a  $\sim a$ ,
- La operación  $p \wedge q$  es equivalente a  $a \odot b$ ,
- La operación  $p \vee q$  es equivalente a  $a \oplus b$ ,

y a partir de las anteriores es posible definir de manera adecuada las operaciones **XOR**, **NAND**.

### 1.2.1. Compuertas lógicas

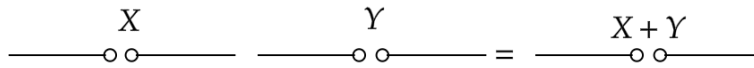
Para poder definir adecuadamente el funcionamiento de las compuertas lógicas, es conveniente definir adecuadamente las propiedades previas que deben cumplir los circuitos que conformados por la colección de las compuertas, se debe considerar que estos están únicamente conformados por *relays* y *switches*, además de que a cualquier tiempo dado el circuito entre dos terminales del mismo el circuito deberá estar abierto o cerrado, bajo estas premisas es posible definir las equivalencias como se muestra en (1.1,1.3).

Símbolo	Interpretación en términos de circuitos	Interpretación en términos de la lógica proposicional
$X$	El circuito $X$	La proposición $X$
$0$	El circuito es cerrado	La proposición es falsa
$1$	El circuito es abierto	La proposición es verdadera
$X + Y$	La conexión entre los circuitos $X$ y $Y$ es en serie	La proposición es verdadera si $X$ o $Y$ es verdadera.
$XY$	La conexión es en paralelo entre los circuitos $X$ y $Y$	La proposición es verdad si $X$ y $Y$ son verdad.
$X'$	El circuito que es abierto cuando $X$ es cerrado y abierto cuando $X$ es cerrado	La contradicción de la proposición $X$
$=$	Los circuitos son abiertos y cerrados simultáneamente	Cada proposición implica la otra. height

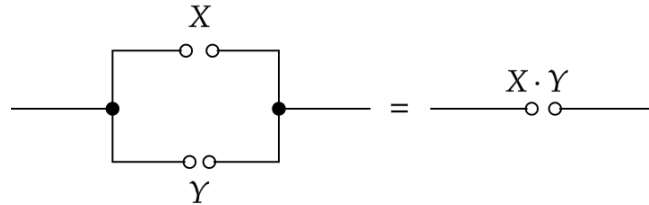
**Tabla 1.1.** Equivalencia entre el cálculo de proposiciones y la interpretación de circuitos de Shannon, tabla tomada de Shannon[21].

Sin embargo, esta forma de visualizar la creación de compuertas lógicas no es trivial, por consecuente, en la actualidad existen una serie de descripciones provenientes de la parte teórica descrita con anterioridad las cuales sirven para llevar a cabo implementaciones físicas.

Es importante mencionar que las compuertas lógicas no son más que una colección de circuitos arreglados de manera tal que cumplan la acción equivalente de un operador de la lógica proposicional, la estructura elemental de las mismas está dada por elementos de entrada y de salida, a los elementos de entrada se les conocerá como *argumentos* y la cantidad de ellos como aridad. La lectura de las mismas es de izquierda a derecha.



Interpretación de la adición de circuitos

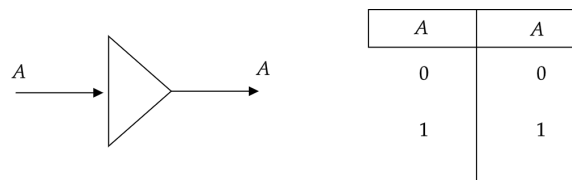


Interpretación de la multiplicación de circuitos

**Figura 1.3.** Esquema que ilustra el comportamiento de la acción de los circuitos bajo sus dos operaciones elementales, la suma y la multiplicación, como fue descrito en (1.1)

### La compuerta Identidad

La primer compuerta por definir es una compuerta que toma como argumento una proposición, y mantiene los valores de verdad o falsedad de la misma, es decir, mantiene “intacta” la proposición introducida.

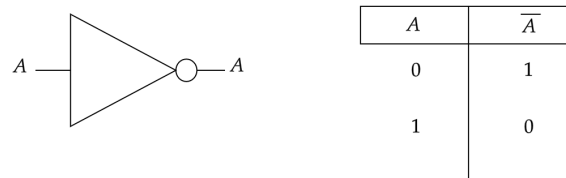


**Figura 1.4.** Esquema que ilustra la acción de la compuerta Identidad ante una entrada.

### Compuerta NOT

La compuerta **NOT** es una compuerta lógica que toma como argumento el valor de una proposición y devuelve como salida el complemento de la misma, es decir, si el valor de la proposición dada es verdadera la salida de la compuerta será falsa y viceversa.

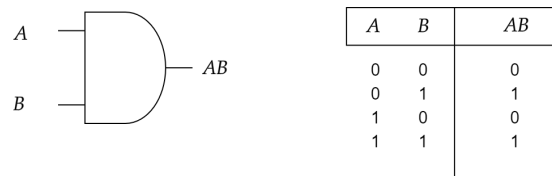




**Figura 1.5.** Esquema que ilustra la acción de la compuerta NOT ante una entrada.

### Compuerta AND

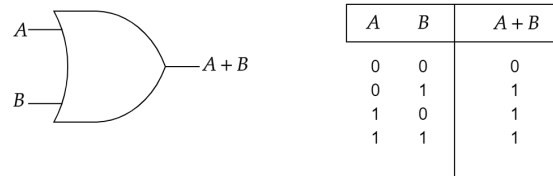
La compuerta **AND** es una compuerta lógica que toma como argumento el valor de dos proposiciones y devuelve como salida la conjunción de las dos proposiciones.



**Figura 1.6.** Esquema que ilustra la acción de la compuerta AND ante la entrada de dos proposiciones.

### Compuerta OR

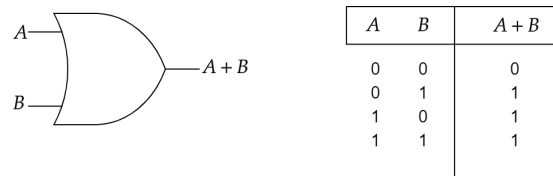
La compuerta **OR** es una compuerta lógica que toma como argumento el valor de dos proposiciones y devuelve como salida la disyunción de las dos proposiciones.



**Figura 1.7.** Esquema que ilustra la acción de la compuerta OR ante la entrada de dos proposiciones.

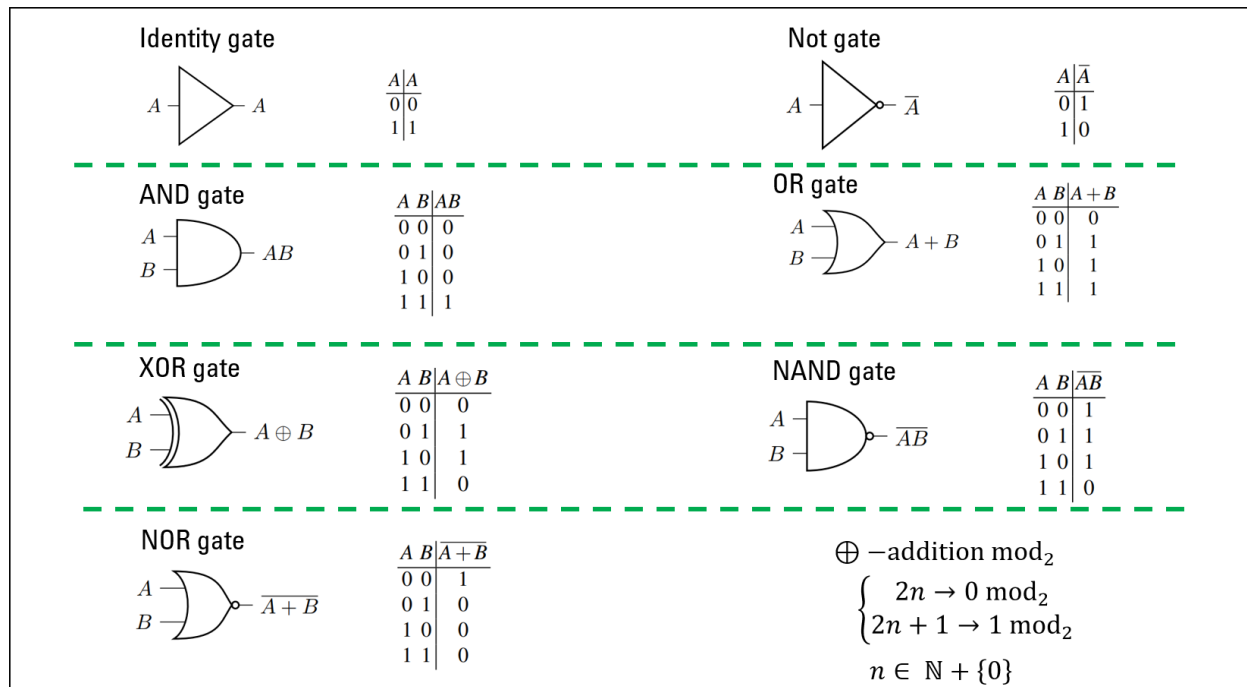
### Compuerta XOR

La compuerta **XOR** es una compuerta lógica que toma como argumento el valor de dos proposiciones y devuelve como salida la disyunción de las dos proposiciones, algo importante a notar es que la forma de calcular la salida de la compuerta es haciendo la suma en módulo dos como se discutió con anterioridad.



**Figura 1.8.** Esquema que ilustra la acción de la compuerta XOR ante la entrada de dos proposiciones.

De la misma manera es posible definir las operaciones de **NAND**, **NOR**, de forma que la descripción completa de las compuertas lógicas puede resumirse a partir de la figura (1.9)



**Figura 1.9.** Resumen de la acción de las compuertas lógicas sobre alguna entrada.

De esta manera a partir del modelo de compuertas, es posible llevar a cabo la implementación algorítmica para la solución de un problema, por ejemplo, es posible construir circuitos que puedan llevar a cabo la suma de dos números, la diferencia, el producto, así como la implementación de funciones computables más complejas.

Hasta el momento hemos descrito que la representación binaria es la adecuada para poder llevar a cabo cómputo y por eso conforma la cantidad de información mínima de un sistema y recibe el nombre de *bit*, además, es posible construir máquinas que puedan llevar a cabo el cómputo a partir de una cadena de bits como es descrito en Turing [14] y Penrose [23] y la implementación de las mismas puede ser tal que se haga en un entorno digital como se demostró por Shannon [21].

De esta manera ahora tenemos varios modelos equivalentes para poder computar y las limitaciones son las mismas, en el modelo teórico propuesto por Turing se supone la existencia de una cinta infinita que permite el almacenamiento de información infinito y que permite que cualquier algoritmo computable se detenga en una serie finita de pasos, no obstante, es evidente a partir de la interpretación digital de Shannon notar que no es físicamente posible

llevar a cabo el desarrollo de un sistema que permita almacenar información infinita.

### 1.2.2. Máquinas de Turing: algoritmos computables y no computables.

La teoría de las máquinas de Turing es un campo de la informática y la matemática que se ocupa del estudio de los algoritmos y su capacidad para ser ejecutados por máquinas. Esta teoría fue desarrollada por Alan Turing en la década de 1930 [14] y ha sido fundamental en el desarrollo de la informática moderna. Un algoritmo es un conjunto de pasos bien definidos que se sigue para resolver un problema o realizar una tarea. Una máquina de Turing es un modelo matemático que simula la ejecución de un algoritmo. En esta teoría, se clasifican a los algoritmos en dos categorías: computables y no computables [16].

Los algoritmos computables son aquellos que pueden ser ejecutados por una máquina de Turing, mientras que los algoritmos no computables son demasiado complejos o no tienen una solución definida y, por lo tanto, no pueden ser ejecutados por una máquina de Turing. En este tema, profundizaremos en la teoría de las máquinas de Turing y cómo se utilizan para clasificar a los algoritmos en estas dos categorías. Una máquina de Turing tiene propiedades complementarias a las de otros modelos de cómputo, podemos de manera simple definir a una máquina de Turing en términos de las siguientes propiedades:

- Una máquina de Turing está compuesta por una cinta infinita dividida en celdas, cada una de las cuales puede contener un símbolo de un alfabeto finito.
- La máquina de Turing tiene un cabezal de lectura y escritura y un conjunto finito de estados accesibles.
- La máquina sigue un conjunto de reglas o instrucciones para leer y escribir símbolos en la cinta y cambiar de estado hacia la izquierda o derecha mediante el movimiento del cabezal.
- Los estados especiales como el de rechazo y el de aceptación toman efecto de manera inmediata.

Formalmente la máquina puede definirse en términos de una colección de conjuntos, estos conjuntos son:  $Q$ ; el conjunto de estados accesibles por la máquina de Turing;  $\Sigma$ , es el alfabeto de entrada de una máquina de Turing *a grosso modo* podemos considerar al alfabeto como el conjunto que contiene a todas las cadenas posibles que no contienen al símbolo en blanco., está usualmente determinado por una colección binaria<sup>6</sup> de elementos que no contiene el símbolo en blanco  $\sqcup$ ;  $\Gamma$  es la cinta del alfabeto que puede comprenderse como la colección de todas las cadenas admisibles en la máquina de Turing que sí contienen al símbolo en blanco. El funcionamiento de la máquina está determinado por lo que se conoce como una función de transición la cual puede entenderse como una función que permite llevar a cabo el movimiento de la cinta en función de los valores de entrada, el conjunto de estados accesibles y los movimientos de la cinta.

**Definición: Máquina de Turing.**

Una máquina de Turing es una siete-tupla,  $\mathcal{M} \rightarrow (Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}})$  donde los conjuntos  $Q, \Sigma, \Gamma$  son conjuntos finitos.

- $Q$  es el conjunto de estados accesibles por la máquina  $\mathcal{M}$ ,
- $\Sigma$  es la entrada del alfabeto que no contiene al símbolo “blanco”  $\sqcup$ ,
- $\Gamma$  es la cinta del alfabeto, donde  $\sqcup \in \Gamma$  y  $\Sigma \subseteq \Gamma$ ,
- $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$  es la función de transición,
- $q_0 \in Q$  es el estado inicial,
- $q_{\text{accept}} \in Q$  es el estado de aceptación, y
- $q_{\text{reject}} \in Q$  es el estado de rechazo, donde  $q_{\text{reject}} \neq q_{\text{accept}}$ .

A partir de la definición anterior es normal que surjan algunas preguntas importantes, por ejemplo, podríamos preguntarnos, ¿siempre hay un estado de aceptación o de rechazo? ¿existe alguna cadena de entrada que nos permita no llegar a un estado de aceptación o

---

<sup>6</sup>Existen máquinas de Turing de más de una cinta, por lo que el alfabeto no necesariamente debe ser una colección binaria.

rechazo? Estas preguntas han sido estudiadas a detalle y no son sencillas de responder, sin embargo, son el parte-aguas para un análisis más profundo del funcionamiento de una máquina de Turing, en particular esta discusión genera lo que se conoce como el *halting problem* y da origen de manera conjunta a la discusión de si existen problemas que son solubles, es decir, existe un algoritmo (cadena de entrada en una máquina  $\mathcal{M}$  que tiene un estado de aceptación o rechazo) o no.

Una observación importante es que las máquinas de Turing pueden ser de dos tipos; deterministas, es decir, la función de transición está bien determinada; no deterministas, la función de transición no está definida, este enfoque de la función de transición puede ser probabilista o heurístico. La presencia de una función no determinista genera que el comportamiento de la cinta varíe aún con la misma entrada, es decir, puede o no causar que la cinta termine en un estado de aceptación o rechazo.

### 1.2.3. Complejidad de los algoritmos

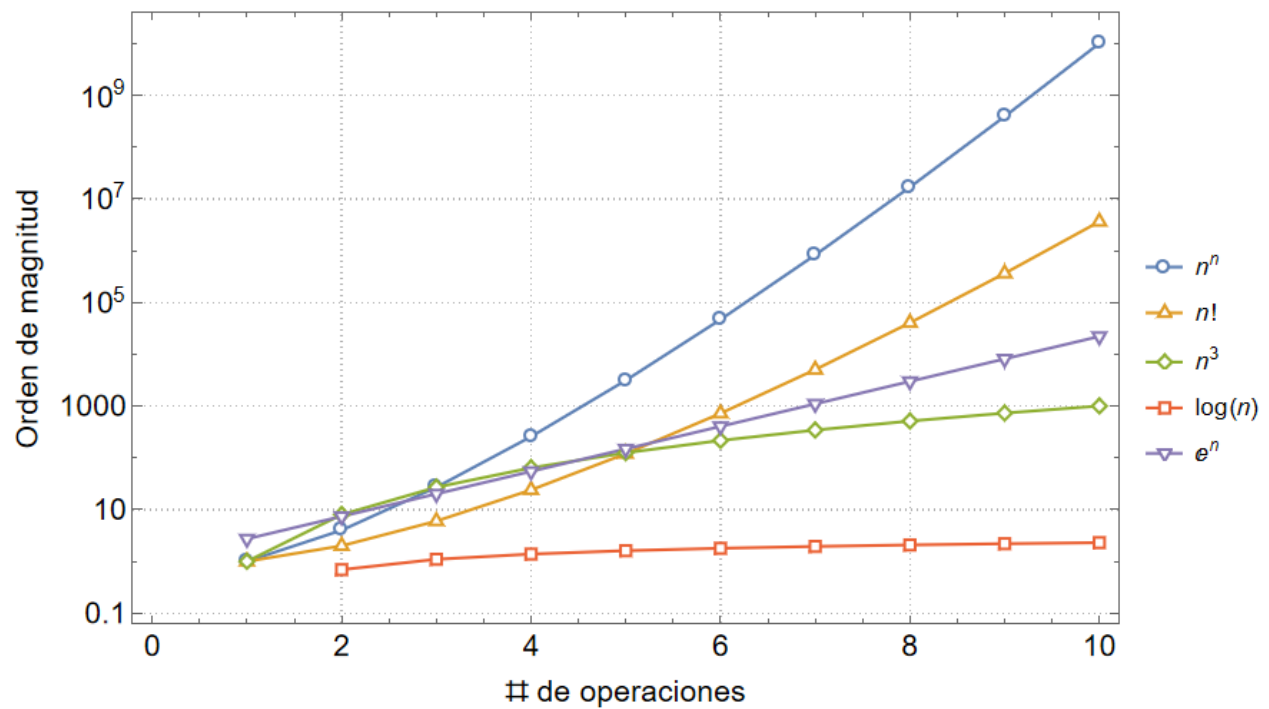
Previamente se describió una de las principales problemáticas existentes en el desarrollo del cómputo a nivel teórico y experimental, pues, pese a que un algoritmo sea computable en un tiempo finito de pasos, es posible que la cantidad de memoria necesaria para llevar a cabo el cómputo sea físicamente imposible de almacenar, por otro lado puede suceder lo contrario, puede que se cuente con una memoria capaz de almacenar el cómputo resultante de un algoritmo, sin embargo, puede que la cantidad de pasos para poder llevar a cabo la corrida del mismo sea muy grande.

Al análisis de estas limitantes se le conoce como *análisis de la complejidad* y en síntesis trata de hallar mejoras a algoritmos existentes de manera que ocupen menor cantidad de memoria en la ejecución y que el tiempo de ejecución sea menor. Al análisis de la cantidad de memoria usada se le conoce como *complejidad espacial*, de manera correspondiente al análisis de los tiempos de ejecución de un algoritmo se le conoce como *complejidad temporal*.<sup>[24]</sup>

La forma de llevar a cabo el análisis de los algoritmos es a partir de los tipos de estruc-

turas de datos y los métodos programáticos que se implementen. En particular es de interés analizar la complejidad temporal, misma que se describe en términos de  $\mathcal{O}(n, t)$ , donde la letra  $\mathcal{O}$  denota el orden de crecimiento del algoritmo en función del tiempo  $t$  y el número de ejecuciones  $n$ . Sin embargo, la notación usualmente se suele describir como  $\mathcal{O}(n)$ .

Es posible mostrar que el comportamiento de un algoritmo puede ser delimitado por cotas, de manera que así se puede encontrar una expresión óptima para el funcionamiento del mismo y las variaciones que buscan mejorar el comportamiento. Por ejemplo, es posible analizar de manera gráfica (véase: 1.10 el comportamiento de la complejidad de un algoritmo en función del orden de complejidad que siguen).



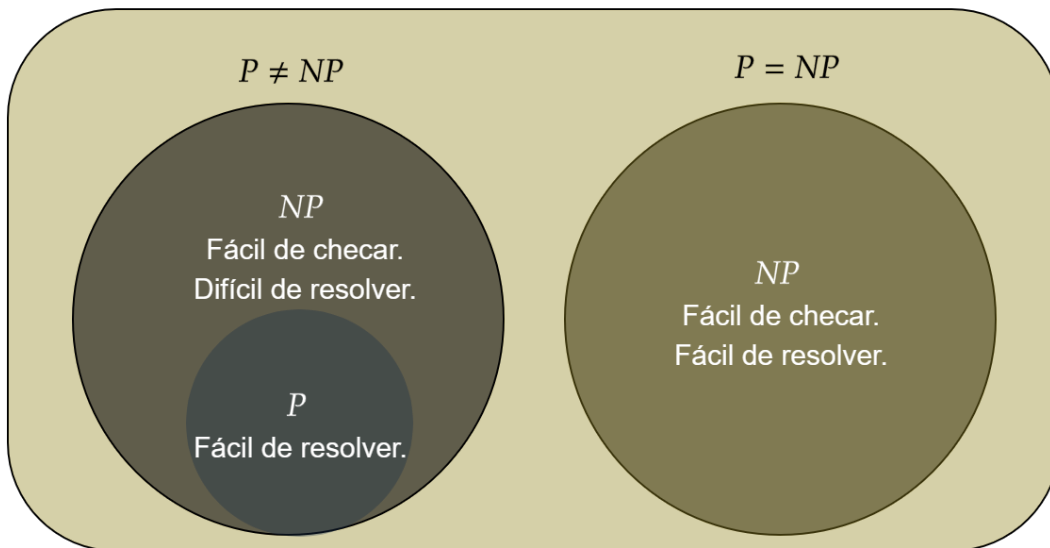
**Figura 1.10.** Ejemplificación del tiempo de ejecución en función de la cantidad de operaciones que debe realizar un algoritmo.

Es importante notar que el orden asintótico de la complejidad está acotado por problemas que crecen exponencialmente, este tipo de problemas son considerados como costosos por su alta demanda de tiempo de cómputo y de almacenamiento.

## Clasificación según complejidad

Los comportamientos asintóticos definidos con anterioridad permiten definir parcialmente clases de problemas computables las cuales están compuestas por problemas que pueden ser resueltos en un tiempo polinomial (**Problemas P**) y problemas que a priori no pueden ser resueltos en un tiempo polinomial (**Problemas NP**). Esta clasificación pese a que es útil en la actualidad tiene serios problemas, el principal es que a día de hoy no hay una demostración formal de que los problemas NP no se puedan convertir a P, ni lo opuesto, no hay ninguna prueba formal de que los problemas NP se puedan convertir a P.

Actualmente se supone que no todos los problemas pueden ser polinomiales, es decir, se supone que  $P \neq NP$ . Además de esta clasificación P y NP respectivamente se clasifican de propiamente según algunas características como el tipo de máquina de Turing (deterministas o no deterministas) que puede o no resolver un algoritmo.



**Figura 1.11.** Esquema que ilustra el comportamiento de los problemas en los conjuntos P y NP, en el caso de que  $P \neq NP$  y  $P = NP$ .



# 2

## Fundamentos de la computación cuántica

El descubrimiento de las máquinas de Turing y sus equivalentes las máquinas de POST y el cálculo- $\lambda$  de Church fue un parte aguas en el desarrollo de los modelos de cómputo que acompañado del formalismo de circuitos propuesto por Shannon permitió que las ciencias computacionales evolucionaran de manera importante a lo largo de décadas, pues en conjunto estas herramientas sirvieron como soporte teórico y tecnológico para el desarrollo de máquinas capaces de llevar a cabo cómputo de manera eficaz.

Uno de los resultados más importantes descritos por Turing y por Church [1, 15] es la conocida **tesis fuerte de Church-Turing**.

### **Tesis fuerte de Church-Turing**

Cualquier proceso algorítmico puede ser simulado de manera eficiente mediante el uso de una máquina de Turing ( $\mathcal{M}$ )

La palabra clave en el enunciado previo es “eficiente”, esta palabra se usa para indicar y resaltar el comportamiento de una máquina de Turing, no obstante, y pese a que pareciera que esta tesis es válida, esta es confrontada por computadoras analógicas, pues bajo la presencia de ruido, algoritmos que *a priori* son computables en el sentido de Turing debido a la presencia de ruido no pueden ser implementados[7].

**Tesis fuerte de Church-Turing (modificada)**

Cualquier proceso algorítmico puede ser simulado de manera eficiente mediante el uso de una máquina de Turing probabilista ( $\mathcal{M}_p$ ).

Cuando se considera en cambio una máquina de Turing probabilista la tesis de Church-Turing parece ser un modelo de cómputo que puede sobreponerse a las dificultades de su semejante determinista, esta situación fue analizada por David Deutsch, quien considerando la naturaleza probabilista del mundo cuántico propuso un modelo de cómputo universal considerando la naturaleza cuántica del universo al cuál llamó computadora cuántica universal [25].

## 2.1. ¿Qué es una computadora cuántica?

La idea descrita en la sección anterior y propuesta por Deutsch acerca de modelos de cómputo que siguen las reglas cuánticas de la naturaleza es innovadora *per se*, no obstante, tiene como inspiración una discusión previa generada por el premio Nobel de Física Richard Feynman quien en una de sus conocidas clases comentó acerca de las limitantes de llevar a cabo simulaciones del mundo cuántico mediante computadoras clásicas [26], en donde deja entrever que para poder llevar a cabo una simulación de la naturaleza es necesario que la máquina capaz de simularla siga las mismas reglas, es decir, la mecánica cuántica.

Bajo esta premisa Deutsch propuso un modelo de cómputo construido con análogos propuestos por Turing previamente. La propuesta de Deutsch (1985) describía las ventajas que podría tener un modelo de cómputo sustentado únicamente en principios mecánico-cuánticos, una de sus principales observaciones fue establecer una formulación física de la tesis de Church-Turing.

**Tesis de Church-Turing, formulación física.**[25]

Cualquier sistema físico finito realizable puede ser perfectamente simulado por un modelo de cómputo universal operado en un número finito de pasos.

Esta formulación permite entre otras cosas definir en términos de procesos físicos el cómputo, esto es en particular conveniente, pues permite establecer analogías con sistemas dinámicos bien estudiados.

Los modelos de cómputo deterministas son inherentemente clásicos, es posible conocer el valor del cómputo a un instante dado, no obstante en la presencia de la mecánica cuántica es sabido que esto no es posible, pues no existe ningún estado físico que pueda tener determinada simultáneamente la posición y el valor de una observable asociada a un tiempo dado.

En palabras simples podemos resumir el concepto de una computadora cuántica como un modelo de cómputo análogo al propuesto por Turing que obedece las leyes de la mecánica cuántica, mismas que se describirán posteriormente.

## 2.2. Elementos previos.

La mecánica cuántica es una rama de la física que se encarga de estudiar procesos físicos que suceden principalmente a pequeñas escalas, esta rama es de particular importancia para la descripción de procesos que la física clásica<sup>1</sup> no es capaz de explicar.

El origen de la mecánica cuántica se remonta al problema de la *catástrofe ultravioleta*, el cual en términos simples consistía en dar una explicación a la energía emitida por distintas longitudes de onda, de manera clásica todas las aproximaciones habían fallado y no fue hasta que en 1900 Max Planck propuso la idea de que la naturaleza en realidad estaba cuantizada, es decir, la naturaleza estaba compuesta de pequeñas partículas a las cuales llamó *cuantos*. Esta descripción permitía establecer que la energía, que según la mecánica clásica se consi-

---

<sup>1</sup>Nos referimos a la física clásica a aquella que describe el comportamiento únicamente de objetos macroscópicos y para los cuales la teoría de Newton y/o Einstein es suficiente.

deraba como una observable continua era en realidad una observable discreta que se encontraba descrita en función del número de *cuantos*. Fue hasta 1905 donde Einstein describió un problema complementario conocido como el efecto fotoeléctrico y en donde describió el comportamiento de la energía emitida por una fuente lumínica y los *cuantos* mediante una proporción que usaba la hipótesis propuesta por Planck[27].

De esta manera el uso de la palabra *cuanto* dio pauta al estudio de nuevos fenómenos que obedecían una descripción en términos de la hipótesis de Planck, al estudio de estos fenómenos se les denominó como mecánica cuántica, pues tenía como objetivo de igual manera que la mecánica clásica describir el comportamiento de las entidades físicas.

Los fundamentos de la mecánica cuántica y la demostración de su validez universal no han sido dadas, no obstante, el uso de esta teoría como metodología de descripción del mundo que nos rodea es precisa y ha sido verificada mediante una gran colección de experimentos en donde se evidencia la naturaleza cuántica del mundo, de manera consecuente a lo largo de este texto evitaremos discutir los orígenes teóricos e implicaciones a nivel fundamental y nos centraremos en llevar a cabo un análisis *operacional* de las herramientas de la mecánica cuántica para la resolución de problemas relacionados al cómputo cuántico.

### 2.2.1. Álgebra lineal

El lenguaje por excelencia de la mecánica cuántica es el álgebra lineal,<sup>2</sup> de manera consecuente a continuación se describirán y definirán algunos conceptos importantes que facilitarán el tratado posterior de operaciones y definiciones correspondientes a la mecánica cuántica. En el estudio de la M.C. es de particular importancia del estudio de unas estructuras matemáticas que son conocidas como *espacios vectoriales*, un espacio vectorial se puede definir como una estructura que tiene definidas dos operaciones, la suma interna y el producto escalar, los elementos de un espacio vectorial son conocidos como vectores.

---

<sup>2</sup>La mecánica cuántica tiene formulaciones equivalentes, una propuesta por Schrödinger, la cual está descrita en términos de funciones de onda y otra propuesta por Heisenberg, la cual usa conceptos relacionados al álgebra lineal, por su simplicidad y conveniencia es común usar la interpretación de Heisenberg.

**Espacio vectorial lineal.**

Sean  $\psi, \phi, \xi, \dots$  un conjunto de vectores pertenecientes a un espacio vectorial  $V$  y sean  $\{a, b, c, \dots\} \in \mathbb{C}$  un conjunto de escalares, se dice que el espacio vectorial es lineal si se cumplen las siguientes propiedades:

➤ Las reglas de adición (suma interna) cumplen con las propiedades de un grupo abeliano.

- Si  $\psi, \varphi \in V \implies \psi + \varphi \in V$
- Conmutatividad:  $\psi + \varphi = \varphi + \psi$
- Asociatividad:  $(\psi + \varphi) + \xi = \psi + (\varphi + \xi)$
- Existe el elemento neutro de la adición:  $\psi + \mathbf{0} = \psi$
- Existe el inverso aditivo:  $(\psi + (-\psi)) = \mathbf{0}$

➤ El producto por un escalar está definido:

- Cerradura de la multiplicación:  $a\psi \in V$
- Distributividad:  $a(\psi + \varphi) = a\psi + a\varphi$
- Asociatividad:  $a(b\psi) = ab(\psi)$
- Elemento neutro multiplicativo:  $I(\psi) = \psi$
- Escalar nulo:  $0(\psi) = \mathbf{0}$

La definición previa permite comprender las operaciones que se suscitan dentro de un espacio vectorial, esto es de particular importancia pues permite describir una colección de espacios vectoriales que cumplen con ser lineales.

Los espacios vectoriales tienen una serie de definiciones complementarias que son útiles y que permiten describir de manera adecuada las propiedades de los mismos, algunas de ellas son: el concepto de dimensión y base de un espacio vectorial.

**Conjunto de vectores linealmente independiente.**

Un conjunto de vectores  $\varphi_1, \varphi_2, \dots, \varphi_N$  se dice que es linealmente **si y solo si** la solución a la ecuación

$$\sum_i^N a_i \varphi_i = 0$$

es  $a_1 = a_2 = a_3 = \dots = a_N = 0$

**Dimensión de un espacio vectorial.**

La dimensión de un espacio vectorial está determinada por el número máximo de vectores linealmente independientes que puede tener el espacio vectorial. Un espacio vectorial con  $N$  vectores linealmente independientes se dice que es  $N$ -dimensional.

Estas propiedades de los espacios vectoriales son particularmente importantes pues permiten describir lo que se conoce como **principio de superposición**. Dado que se tienen a lo más  $N$  vectores linealmente independientes entonces es posible proponer que cualquier vector del espacio vectorial es una combinación lineal de los vectores que generan al espacio vectorial, de esta manera cualquier estado puede ser descrito en términos de la base de los vectores linealmente independientes de  $V$ .

**Principio de superposición**

Sea  $\{\phi_N\}$  el conjunto de la máxima cantidad de vectores linealmente independientes que pertenecen a un espacio vectorial  $V$ . Cualquier vector  $\psi \in V$  puede ser expresado como:

$$\psi = \sum_i^N a_i \phi_i$$

**Espacios de Hilbert.**

Un tipo de espacios vectoriales de interés son los conocidos como **espacios de Hilbert**, un espacio de Hilbert  $\mathcal{H}$  consiste en un conjunto de vectores y un conjunto de escalares que cumplen las propiedades:

➤  $\mathcal{H}$  es un espacio vectorial lineal.

- $\mathcal{H}$  es un espacio vectorial con producto interno que es estrictamente positivo.
- $\mathcal{H}$  es un espacio vectorial separable.
- $\mathcal{H}$  es completo.

Los espacios de Hilbert son de dimensión infinita, no obstante, pueden usarse representaciones de dimensión finita, la presencia de transformaciones lineales en el mismo genera vectores que viven en el mismo espacio de Hilbert. Una propiedad de los espacios vectoriales lineales es que tienen asociado un “dual”, en términos no formales se puede pensar como un espacio que “refleja” a los elementos de un espacio vectorial mediante el proceso de cálculo de la transpuesta y la acción del cálculo del complejo conjugado.

### 2.2.2. Postulados de la mecánica cuántica [27]

La mecánica cuántica no relativista puede resumirse en cinco postulados que describen las propiedades más generales de un sistema cuántico, es aquí en donde las propiedades del álgebra lineal descritas previamente toman relevancia, pues los cinco postulados están descritos en el lenguaje del álgebra lineal.

- Primer Postulado:

A un tiempo fijo  $t_0$ , el estado de un sistema aislado está definido por la especificación de un ket  $|\psi(t_0)\rangle = |\psi_0\rangle$  el cual pertenece al espacio  $\mathcal{H}$

- Segundo Postulado:

Cualquier cantidad física medible  $\mathcal{A}$  está descrita por un operador  $\hat{A}$  actuando en  $\mathcal{H}$ ; este operador es denominado una observable.

- Tercer Postulado:

El único resultado posible de la medición de una cantidad física  $\mathcal{A}$  es uno de los valores propios de la observable correspondiente  $\hat{A}$

- Cuarto Postulado:

Cuando una cantidad física  $\mathcal{A}$  es medida en un sistema actuando sobre un estado

normalizado  $|\psi\rangle$ , la probabilidad de hallar a ese estado en uno de los valores propios no degenerados  $a_n$  está dada por:

$$\mathcal{P}(a_n) = \|\langle u_n | \psi \rangle\|^2$$

➤ Quinto Postulado:

La evolución temporal de un estado físico está determinada por la acción de la ecuación de Schödinger:

$$i\hbar \frac{d}{dt} |\psi_t\rangle = H(t) |\psi_t\rangle$$

donde  $H$  es la observable asociada con la energía total de un sistema (Hamiltoniano)

Es importante hacer una mención clara respecto a la notación usada, la cual es conocida como **notación de Dirac** (véase la tabla 2.1) y es usada ampliamente en la literatura relacionada a la mecánica cuántica y al cómputo cuántico.

**Tabla 2.1.** Equivalencias de notación en el álgebra tradicional con la notación de dirac. La notación de Dirac siempre considera que la acción de un operador es hacia

Descripción	Notación Típica	Notación de Dirac
Un vector	$\psi$	$ \psi\rangle$
El dual de un vector	$\psi^*$	$\langle\psi $
Un operador	$A$	$\hat{A}$
Producto interno.	$(\psi, \varphi)$	$\langle\psi \varphi\rangle$
Acción de un operador/Transformación Lineal	$A\psi$	$\hat{A} \psi\rangle$

De manera simplista podemos considerar a estos postulados como “reglas” que deben seguir los objetos cuánticos y de alguna manera análogas a las leyes del movimiento usadas ampliamente para la descripción de sistemas físicos en el área de mecánica clásica.

En particular para el área del cómputo cuántico es de importancia el entender claramente lo que es un operador; en general, podemos decir que un operador en mecánica cuántica es una transformación **lineal** de un estado físico.



## Un estado físico

En general, un estado físico en mecánica cuántica se puede escribir como una superposición de vectores propios correspondientes a una observable, de esta manera, un estado  $|\psi\rangle$  se puede escribir como

$$|\psi\rangle = \sum_{k=0}^n c_k |\phi\rangle_k, \quad (2.1)$$

donde  $\{|\phi\rangle_k\}$  es el conjunto de vectores propios correspondientes a una observable.

Los operadores que actúan sobre los estados físicos pueden ser interpretados como transformaciones lineales de un estado, es decir, considerando que un operador  $\hat{A}$  actúa sobre un estado  $|\psi\rangle$  de la forma  $\hat{A}|\psi\rangle$  el resultado de esta operación será un nuevo vector  $|\psi\rangle'$  que vive en el espacio de Hilbert.

Un caso particular de esto es cuando el estado  $|\psi\rangle$  se encuentra en la misma base de vectores propios del operador que actúa sobre el, cuando esto sucede la relación de la transformación se escribe como  $\hat{A}|\psi\rangle = a|\psi\rangle$ , en este caso la relación es una relación de escala, pues como el estado estaba en la misma base de vectores propios de la observable se obtiene lo que usualmente se conoce como la ecuación de autovalores.

En general, la dinámica de un sistema se describe como se mencionó en el postulado cinco, mediante la acción de la ecuación de Schrödinger, para esto es necesario hacer uso de un operador que es de particular importancia y se conoce como el **operador de evolución temporal**, este operador como su nombre lo indica permite caracterizar el comportamiento de un sistema físico caracterizado por un estado  $|\psi\rangle$  a un tiempo dado, la forma general del operador de evolución está en función del operador Hamiltoniano; el cual como se comentó brevemente caracteriza las contribuciones energéticas en el sistema. Es posible definir al operador de evolución según

$$\hat{U} = \exp\left\{-i\frac{\hat{H}t}{\hbar}\right\}, \quad (2.2)$$

la acción del operador evolución sobre un estado se escribe como  $\hat{U}|\psi\rangle = |\psi_t\rangle = |\psi(t)\rangle$ , esto es, la forma que tendrá un estado determinado al tiempo  $t$ , este tipo descripciones a un tiem-

po dado permiten describir la dinámica del sistema mediante la ecuación de Schrödinger. En general existe una variedad amplia de propiedades que cumplen los estados y las observables de un sistema físico, no obstante, las necesarias para el cómputo cuántico serán descritas conforme se vayan necesitando, una referencia explícita y central en el área de la mecánica cuántica es Cohen-Tannoudji, Diu y Laloe [27] y una descripción profunda de los conceptos relacionados al cómputo e información cuántica está dada en Nielsen y Chuang[7].

### 2.3. Sistemas de dos niveles y computación.

Una vez descritas algunas de las propiedades más importantes de los sistemas de la física cuántica debemos entrar en materia. Como se comentó previamente Deutsch [25] planteó que la naturaleza cuántica podría llevar a cabo cómputo de manera óptima y equivalente a los modelos de Turing, de esta manera es posible comenzar a preguntarse acerca de cómo llevar a cabo esto, la principal problemática es ahora encontrar un sistema de codificación que permita replicar lo que hacen las computadoras digitales de uso tradicional, bajo esta premisa debemos considerar algún sistema físico que permita tener acceso a dos valores, como en el caso del cómputo tradicional.

En la física existen una gran variedad de sistemas físicos compuestos por dos niveles, por su relevancia histórica destaca el experimento de Stern-Gerlach, a partir del cual se descubrió la existencia del espín de las partículas. Las computadoras cuánticas están compuestas por sistemas de dos niveles de distinta procedencia, algunos de los sistemas más de dos niveles que son usados en el cómputo cuántico son: trampas de iones, átomos fríos, puntos cuánticos, superconductores, etc.

Si bien es importante considerar el tipo de sistema físico en el que está sustentado una computadora cuántica es aún más relevante entender cómo llevar a cabo la codificación directa de la información, a partir de estos sistemas. A partir de esto podemos definir el concepto de un **qubit** o **cúbit**, que de manera simple diremos que es el análogo cuántico a los bits, de igual manera que en el clásico en donde se trabaja con bits, diremos que un qubit puede tomar dos valores, los cuales estarán etiquetados por estados  $|0\rangle, |1\rangle$ , no obstante, un qubit se encuentra, en general, en lo que se conoce como una superposición cuántica.

**Qubit**

Un qubit ( $|\psi\rangle$ ) es un sistema de dos niveles descrito por una superposición de estados propios  $|0\rangle, |1\rangle$ , y constantes arbitrarias  $\alpha, \beta \in \mathbb{C}$  tales que:

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle, \quad (2.3)$$

y donde se cumple además que  $|\alpha|^2 + |\beta|^2 = 1$  de manera que el estado/qubit se encuentre normalizado.

Como es posible observar el qubit se encuentra en una superposición de estados, de manera que contrario al caso de un bit en donde el valor del bit se encontraba perfectamente definido, en este caso la “localización” del qubit no está determinada, por lo que la naturaleza de los qubits es determinista.

Siguiendo el cuarto postulado mencionado con anterioridad es posible notar que la probabilidad de localizar al qubit en el estado  $|0\rangle$  es proporcional a  $\alpha^2$  y de manera complementaria, la probabilidad de hallar al qubit en el estado  $|1\rangle$  es proporcional a  $\beta^2$ .

sin embargo, como se puede observar de la definición de qubit (2.3) los valores de  $\alpha, \beta$  son en general complejos, de esta manera poder visualizar la “localización” de un estado es complicado, no obstante la visualización de un qubit puede llevarse a cabo mediante una parametrización que es conocida como la esfera de Bloch (véase 2.1) o esfera de Poincaré, esta parametrización se define en términos de dos variables angulares.

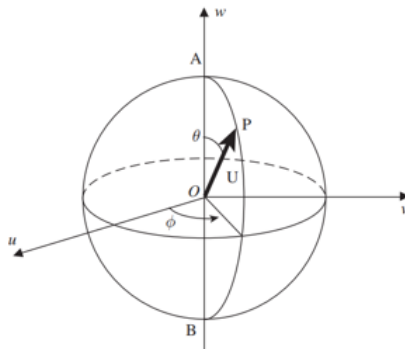
**Parametrización de la esfera de Bloch**

La esfera de Bloch es la representación pictórica que surge de considerar una parametrización de un qubit definida por:

$$\alpha = \cos \frac{\theta}{2}, \quad \beta = \exp\{-i\phi\} \sin \frac{\theta}{2}$$

de forma que el estado que representa al qubit ahora está dado según:

$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + \exp\{-i\phi\} \sin \frac{\theta}{2} |1\rangle$$



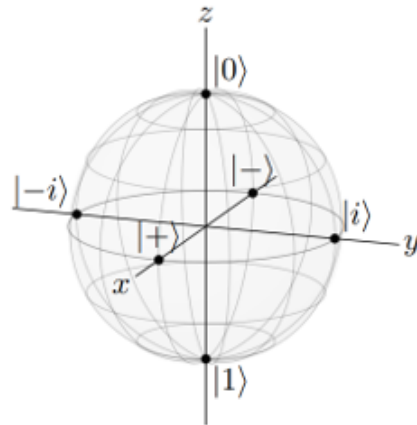
**Figura 2.1.** Parametrización de la esfera de Bloch, el vector  $\mathbf{P}$  se conoce como vector de Bloch, se muestran los ángulos correspondientes a la parametrización. Imagen tomada de Grynberg, Aspect, Fabre et al.[28]

## 2.4. Compuertas cuánticas

En el cómputo tradicional se había abordado la necesidad de proponer modelos de cómputo que pudieran ser construidos a partir de elementos físicos, uno de los resultados más importantes relacionado a la construcción de dispositivos de cómputo fue el obtenido por Shannon[21], en donde mediante la equivalencia de la lógica de Boole encontró que estos dispositivos podían generar cómputo, a los dispositivos los llamó compuertas lógicas (véase la sección 1.2). De manera que es adecuado preguntarnos si existe alguna manera de proponer compuertas cuánticas que puedan replicar la idea propuesta por Shannon, afortunadamente la respuesta a la pregunta es positiva.

Las compuertas cuánticas tienen la particularidad de actuar sobre un qubit como las compuertas lógicas sobre los bits, no obstante tienen algunas propiedades interesantes entre las que destacan:

- Las compuertas cuánticas son mapeos lineales (operadores)  $\mathcal{U}$  que actúan sobre los estados  $|\psi\rangle$  de la forma  $|\psi\rangle = \mathcal{U}\alpha|0\rangle + \mathcal{U}\beta|1\rangle$
- Las compuertas cuánticas son **unitarias**, es decir, si se tiene un estado normalizado  $|\psi\rangle \implies \langle\psi|\psi\rangle = 1$ , de manera que la acción de  $\mathcal{U}$  sobre  $|\psi\rangle \implies \langle\psi|\mathcal{U}^\dagger|\mathcal{U}\psi\rangle$
- Las compuertas lógicas reversibles son compuertas cuánticas válidas.



**Figura 2.2.** Representación de la esfera de Bloch en términos de los estados base de un qubit y de los estados  $|\pm\rangle, |\pm i\rangle$  que resultan como una combinación lineal de  $|0\rangle, |1\rangle$ . Imagen tomada de Wong[29]

➤ Todas las compuertas cuánticas son reversibles.

Gracias a la geometría de las compuertas cuánticas  $\mathcal{U}$  es posible representar la acción de ellas sobre un qubit como rotaciones sobre la esfera de Bloch (véase 2.2), y si bien esta definición puede quedarse corta cuando se considera más de un qubit es útil para evidenciar la acción de las compuertas. La acción de las compuertas se escribe mediante la acción en los vectores base de un qubit.

## Compuerta identidad $\mathcal{I}$

La compuerta identidad  $\mathcal{I}$  es la compuerta que cumple con dejar a un qubit en el mismo estado en el que se encontraba inicialmente.

➤  $\mathcal{I}|0\rangle = |0\rangle$

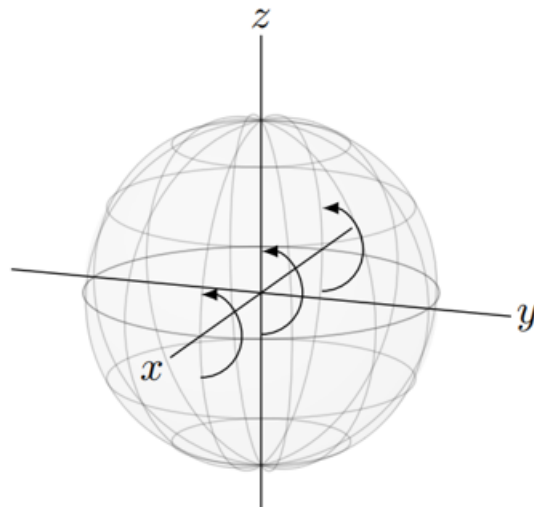
➤  $\mathcal{I}|1\rangle = |1\rangle$

## Compuerta Pauli-X, NOT

La compuerta Pauli en  $X$  es una compuerta que efectúa una rotación alrededor del eje  $X$  de  $180^\circ$  (véase 2.3)

$$\triangleright X |0\rangle = |1\rangle$$

$$\triangleright X |1\rangle = |0\rangle$$



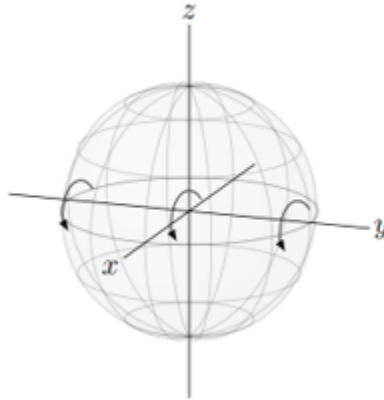
**Figura 2.3.** Representación de la acción de la compuerta de Pauli en X sobre la esfera de Bloch. Imagen tomada de Wong[29]

## Compuerta Pauli-Y

La compuerta Pauli en Y  $Y$  es una compuerta que efectúa una rotación alrededor del eje  $Y$  de  $180^\circ$  (véase 2.4)

$$\triangleright Y |0\rangle = i |1\rangle$$

$$\triangleright Y |1\rangle = -i |0\rangle$$



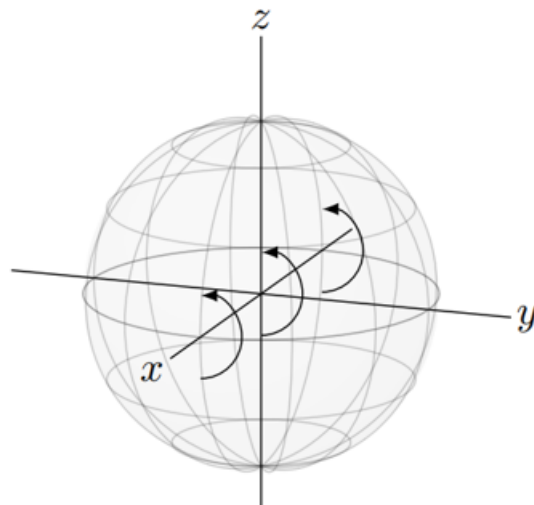
**Figura 2.4.** Representación de la acción de la compuerta de Pauli en Y sobre la esfera de Bloch. Imagen tomada de Wong[29]

## Compuerta Pauli-Z, NOT

La compuerta Pauli en Z es una compuerta que efectúa una rotación alrededor del eje Z de  $180^\circ$  (véase 2.5)

➤  $Z |0\rangle = |0\rangle$

➤  $Z |1\rangle = -|1\rangle$



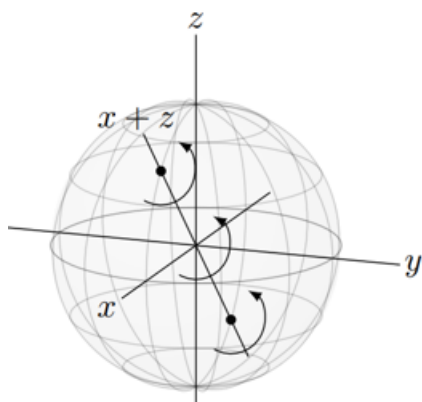
**Figura 2.5.** Representación de la acción de la compuerta de Pauli en Z sobre la esfera de Bloch. Imagen tomada de Wong[29]

## Compuerta de Hadamard

La compuerta identidad  $H$  es una compuerta que efectúa una rotación alrededor del eje  $X + Z$  de  $180^\circ$  (véase 2.6)

$$\triangleright H |0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle$$

$$\triangleright H |0\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle$$



**Figura 2.6.** Representación de la acción de la compuerta de Hadamard sobre la esfera de Bloch. Imagen tomada de Wong[29]

Es importante destacar que hasta este momento se han hecho uso solo de compuertas que actúan sobre un qubit, la acción de más qubits es un caso de estudio en particular que se abordará a continuación, sin embargo, es prudente hacer una anotación importante sobre la forma de los operadores  $\mathcal{U}$ , en general, se tiene un número infinito de ellas, pues cualquier rotación en la esfera de Bloch es una compuerta cuántica válida, las mostradas previamente solo son una referencia a compuertas útiles que se ocupan comúnmente en el área del cómputo cuántico.

En el caso clásico es común que contemos con al menos una cadena  $s$  que contenga una secuencia binaria y a partir de la cual llevemos a cabo el cómputo, en el caso cuántico se puede llevar a cabo el mismo proceso mediante la consideración de una colección de qubits, no obstante el tratamiento no es tan trivial como en el caso clásico, para operar una cadena de qubits es necesario introducir el concepto de **producto tensorial**, de manera intuitiva



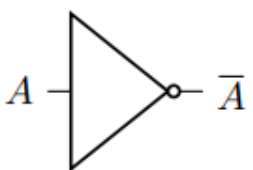
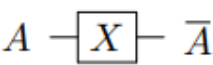

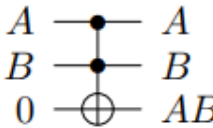

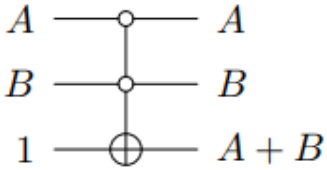

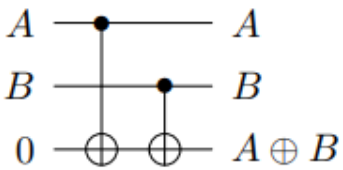

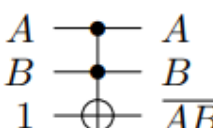
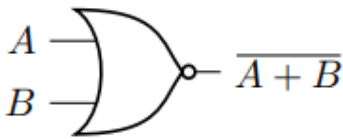
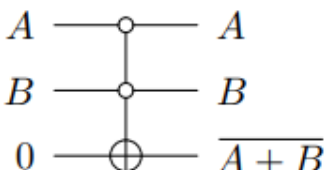
podemos definirlo como una operación que genera un “espacio” en donde los dos qubits toman relevancia.

De manera más adecuada podemos decir que si un sistema cuántico  $S_1$  interactúa con un sistema cuántico  $S_2$  para llevar a cabo el estudio total del sistema se requiere del cálculo del producto tensorial de ambos sistemas  $S_1 \otimes S_2$

Podemos hacer uso de esta notación para denotar una cadena de qubits compuesta por  $\{|q\rangle_i\}$ , de esta manera una cadena  $|s\rangle = \otimes_i^n |q\rangle_i$ , es importante hacer mención que conforme la cardinalidad de la cadena aumenta, es decir, el número de qubits involucrados en la cadena crezca el espacio necesario para poder llevar a cabo la representación de los mismos crecerá exponencialmente a razón de  $2^{|s|}$ , este hecho es de particular importancia en el caso que queramos llevar a cabo la simulación de algún proceso cuántico en una computadora clásica, notaremos rápidamente que la cantidad de memoria necesaria para emular un proceso cuántico es extremadamente alta, no obstante, en el caso de hacer la ejecución en una computadora cuántica esto no será relevante.

En general existen una diversidad de compuertas cuánticas que actúan sobre los qubits, el significado geométrico de cada una es en muchos casos evidente, sin embargo, generalmente es conveniente obviar esta interpretación y buscar análogos clásicos (véase 2.7) que sean de utilidad para hacer un análogo con el modelo de circuitos propuesto por Shanon, bajo esta premisa es posible llevar a cabo la construcción de circuitos cuánticos, los cuales están estructurados conforme al diagrama mostrado en 2.8, en donde la acción de las compuertas se efectúa sobre los cableados de cada qubit.

Es importante mencionar que no existe una metodología definida para la creación de circuitos cuánticos, si bien se trata de explotar la analogía con los circuitos tradicionales es importante notar que la naturaleza propia de los sistemas cuánticos hacen que la traducción de algoritmos tradicionales no sea factible pues como se discutió con anterioridad en el caso cuántico las compuertas lógicas son reversibles, algo que no sucede en todas las compuertas clásicas.

	Classical		Reversible/Quantum
NOT		X-Gate	
AND		Toffoli	
OR		anti-Toffoli	
XOR		CNOTs	
NAND		Toffoli	
NOR		anti-Toffoli	

**Figura 2.7.** Equivalencia de las compuertas clásicas con compuertas cuánticas. Nótese que no todas las compuertas mostradas son reversibles y que la aridad en el caso clásico y el cuántico no coincide. Imagen tomada de Wong[29] pp. 170.

### 2.4.1. Complejidad de un algoritmo cuántico

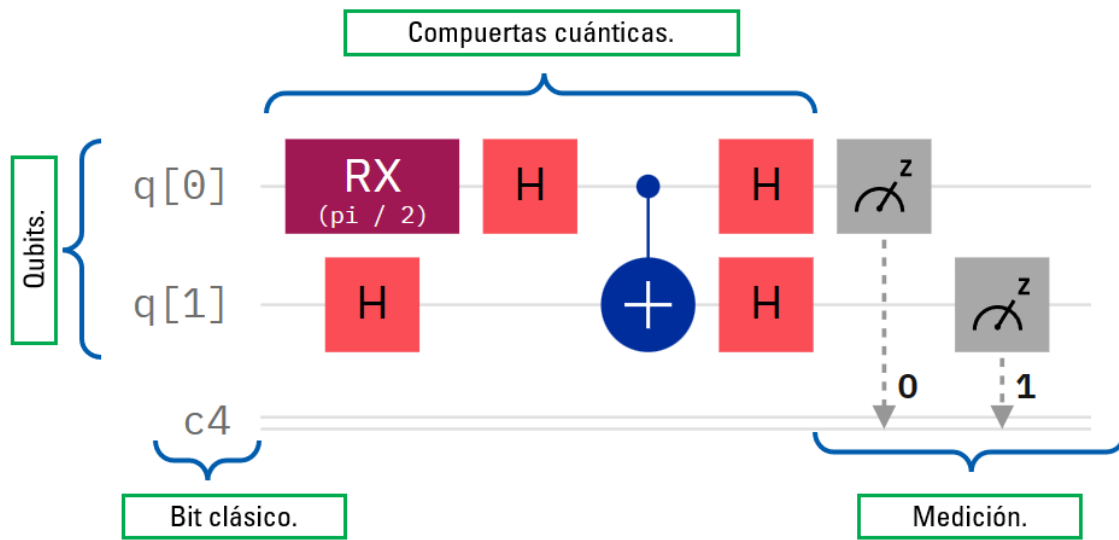
Uno de los retos de la computación cuántica en la actualidad radica en la necesidad de adaptarse a las limitantes experimentales actuales. Como se ha discutido previamente la existencia de ruido en los canales de transmisión de información cuánticos suponen un problema de cara a la ejecución de algoritmos cuánticos en computadoras reales.

**Complejidad temporal de un algoritmo cuántico** Sea  $\mathcal{C}$  un circuito cuántico, se puede definir a la complejidad temporal del algoritmo como la cantidad de bloques de compuertas cuánticas presentes en el circuito.

**Complejidad espacial de un algoritmo cuántico** Sea  $\mathcal{C}$  un circuito cuántico, se puede definir a la complejidad espacial de un algoritmo cuántico como la cantidad de qubits presentes en el circuito cuántico.

Para evidenciar lo discutido previamente consideremos el circuito representado en la figura 2.8, en ese caso en particular la complejidad espacial del algoritmo corresponde a dos qubits, por otro lado la complejidad temporal del circuito representado corresponde a la acción de cuatro bloques de compuertas cuánticas, un hecho importante por destacar es que la medición no suele ser considerada como un bloque que añade complejidad[30].

Entender estos conceptos es extremadamente útil para poder definir las limitantes de los circuitos y consecuentemente de los algoritmos cuánticos en un entorno de simulación o de ejecución real.



**Figura 2.8.** Estructura de un circuito cuántico, las líneas grises denotan los cables o rieles, las compuertas cuánticas actúan en cada qubit.

# 3

## Quantum Machine Learning.

Como se ha discutido con anterioridad el uso del cómputo cuántico comienza a suponer un cambio de paradigma en la aplicación de algoritmos y técnicas computacionales por los recientes avances en la temática.

En particular el uso del modelo de circuitos cuánticos y su analogía previamente explicada con los circuitos propuestos por Shannon para la resolución de problemas lógicos han permitido que el avance sea considerablemente rápido, pues al tener análogos la creación de algoritmos se ha llevado a cabo siguiendo el “camino” clásico.

Pese al auge de algoritmos clásicos que pueden ser transportados al lenguaje de la mecánica cuántica, adquieren relevancia los algoritmos que se aprovechan de las ventajas computacionales del cómputo cuántico como la posibilidad de computar de manera natural de forma paralela[29] y reducen la complejidad de sus contra partes clásicas.

## 3.1. Algoritmos de QML

El cómputo cuántico al igual que su contra-parte clásica ha tenido un desarrollo exponencial, partiendo desde la resolución de problemas que para un humano son esencialmente triviales como la suma o resta de dos números hasta la implementación de algoritmos complicados que pueden llevar a cabo tareas que un humano simplemente no podría en un tiempo finito. Es así como el origen del QML tiene como antecedente la existencia de algoritmos cruciales que han ayudado al desarrollo de esta área, por consiguiente a lo largo de este capítulo se explicarán una serie de algoritmos cuánticos que tienen una alta importancia histórica dentro del área y que son fundamentales para el desarrollo de algoritmos como el aprendizaje máquina cuántico.

### 3.1.1. Circuitos cuánticos variacionales.

El soporte principal del aprendizaje máquina cuántico se sostiene en lo que se conocen como circuitos variacionales o circuitos parametrizados, esto es particularmente utilizado dentro del contexto de cómputo cuántico a corto plazo<sup>1</sup> debido a la versatilidad presente dentro del modelo de circuitos cuánticos.

En general podemos definir a un circuito cuántico variacional como aquel modelo compuesto por una colección de compuertas cuánticas que se encuentra descrito por una colección de parámetros  $\theta_i$ , con el objetivo de hallar un conjunto de parámetros que minimicen el valor esperado del circuito cuántico.

Sea  $\hat{C}(x; \theta)$  un operador que represente una colección de compuertas cuánticas controladas y consideremos a un operador  $\hat{A}$  el cual esté por simplicidad pero sin pérdida de generalidad en la base de vectores propios de  $\hat{C}(x; \theta)$  es posible definir a una función escalar  $f$  que represente el valor esperado del operador  $\hat{A}$  bajo la acción de una colección de parámetros

---

<sup>1</sup>Este modelo es también conocido como Near Term Computing en la literatura especializada.

$\{\theta_i\}$  como

$$f(x; \theta) = \langle 0 | \hat{C}(x; \theta)^\dagger \hat{A} \hat{C}(x; \theta) | 0 \rangle, \quad (3.1)$$

a  $f$  usualmente se le conoce como la función objetivo o función de costo.

### Tarea de aprendizaje supervisado.

En el contexto del QML se puede definir a una tarea de aprendizaje supervisado [31] como aquella que dado un conjunto de valores de entrada  $\mathcal{X}$  y donde sea un conjunto  $\mathcal{Y}$  de valores de salida, si existe un conjunto  $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$  con  $x_i \in \mathcal{X}$  y de manera correspondiente  $y_i \in \mathcal{Y}$  se define al conjunto de pares  $(x_i, y_i)$  como miembros del espacio del producto cartesiano de los conjuntos  $\mathcal{X} \times \mathcal{Y}$ , de esta manera la tarea de aprendizaje supervisado se puede resumir como la posibilidad de hallar una función de distribución de probabilidad  $\rho(x, y)$  tal que los datos no clasificados del conjunto  $\mathcal{X}$  puedan ser utilizados para hallar su correspondiente valor de  $\mathcal{Y}$ .

La resolución de una tarea de aprendizaje supervisado entonces se lleva a cabo mediante la definición de un conjunto de funciones  $\{f\}$  las cuales actúan sobre los valores de entrada  $\mathcal{X}$  a valores del conjunto  $\mathcal{T}$ .<sup>2</sup> En el aprendizaje supervisado la definición de las funciones que pretenden llevar a cabo el mapeo entre los conjuntos se lleva a cabo mediante un proceso conocido como minimización de riesgos, la idea detrás del mismo es crear una función de pérdida  $L$  que pretende disminuir la disparidad promedio de los valores predichos.

La elección del conjunto de funciones  $f$  que pretende llevar a cabo la predicción en el aprendizaje clásico es de particular importancia pues define el tipo de aprendizaje que se llevará a cabo. La elección del modelo puede ser de dos tipos, si existe una función  $f : \mathcal{X} \rightarrow \mathcal{Y}$  se dice que el modelo de aprendizaje es determinista. En cambio si la forma de la función  $f$  es tal que cumpla con las propiedades de distribución de cuasi-probabilidad  $\rho : \mathcal{X} \times \mathcal{Y} \rightarrow [0, 1]$  se dice que el modelo es probabilístico.

---

<sup>2</sup>En el lenguaje de ML se suele denominar al conjunto  $\mathcal{X}$  como el conjunto de características y al conjunto  $\mathcal{Y}$  como el conjunto de clases o conjunto de regresión según la tarea de aprendizaje.

### Modelos de aprendizaje cuántico deterministas.

Definir de manera adecuada el concepto de aprendizaje determinista o probabilista en términos clásicos es sencillo, pues la propia naturaleza de la propuesta de solución determina el tipo de modelo.

Se dice que un modelo  $\mathcal{M}$  de aprendizaje cuántico es determinista si dada una colección de valores de entrada  $\mathcal{X}$  y donde sea  $\hat{C}(x; \{\theta_i\})$  un circuito cuántico que depende de los parámetros de entrada, si consideramos además un operador hermítico  $\hat{A}$  y un estado de preparación  $|\psi(x, \theta) = \hat{C}(x; \theta)\rangle$  entonces la función a optimizar se define como

$$f_\theta(x) = \langle \psi(x, \theta) | \hat{A} | \psi(x, \theta) \rangle, \quad (3.2)$$

es posible hallar que si la medida se hace en la base diagonal del operador  $\hat{A} = \sum a_i |a_i\rangle \langle a_i|$  la función característica del modelo se puede escribir como

$$f_\theta = \sum_i a_i | \langle a_i | \psi(x, \theta) \rangle |^2 = \sum a_i p(a_i),$$

en donde se observa que el valor resultante coincide con el valor esperado de hallar al estado en una determinada clase.

### Modelos cuánticos probabilísticos.

una vez comprendida la tarea del aprendizaje determinista en el área de QML es natural preguntarnos por los modelos no deterministas, es decir, aquellos que tienen una naturaleza probabilística, en general podemos clasificar a los modelos de este tipo según:

- Modelos de aprendizaje supervisado probabilistas sustentados en una distribución condicional.
- Modelos de aprendizaje no supervisado probabilistas.



Con fines de atender las limitantes de este texto bastará con definir el primer concepto.

Se dice que un modelo de aprendizaje supervisado cuántico es aquel que dado un conjunto de entrada  $\mathcal{X}$  y un conjunto de valores de salida  $\mathcal{Y}$ , considerando un circuito cuántico  $\hat{C}(x, \theta)$  que depende exclusivamente de los parámetros y del conjunto de entradas de manera que se cumple con  $|\psi(x, \theta)\rangle = U(x, \theta) |0\rangle$ . Se asocia a cada valor propio o salida de la medición del observable con un posible valor de salida  $y$ , de manera que  $\hat{A} = \sum_{y \in \mathcal{Y}} y |y\rangle \langle y|$ , de esta manera la función de distribución condicional está definida según

$$p_{\theta}(y|x) = |\langle Y|\psi(x, \theta)\rangle|^2 \quad (3.3)$$

dado que  $\{|y_i\rangle\}$  es una base el proyector  $P_{|\psi\rangle} = \sum_i |y_i\rangle \langle y_i| = \hat{I}$ .

## 3.2. Componentes de un modelo de QML.

Como se ha visto a lo largo de este capítulo el uso de los circuitos variacionales en el área del aprendizaje máquina cuántico es fundamental, pues tareas de regresión o clasificación requieren de que el modelo sea ajustado y medido en términos de una observable para poder llevar a cabo la predicción o clasificación de un determinado conjunto de datos. Si bien la estructura de un circuito parametrizado como se comentó previamente es simplemente una colección de compuertas cuánticas que poseen unos parámetros que tienden a ajustarse según una observable el uso de ellos es amplio, pues suponen parte fundamental de un modelo de aprendizaje, en general un modelo de aprendizaje tiende a tener dos circuitos variacionales, los cuales son conocidos como *ansatz* y *feature map*.<sup>3</sup>

### 3.2.1. Ansatz

El *ansatz* es un circuito cuántico variacional que sirve como propuesta de solución para una tarea específica, puede considerarse como una suposición inicial parametrizada y que

---

<sup>3</sup>La traducción al español de estos dos términos sería “propuesta de solución” y “mapa de características” respectivamente, sin embargo, es más común referirnos por los términos en inglés pues la literatura especializada usa el concepto anglosajón.

cumple con la necesidad de ser adaptable a una colección de problemas diversos. El funcionamiento del ansatz es tal que va ajustando los parámetros siguiendo el proceso discutido en la ecuación (3.1), una de las propiedades que debe tener el ansatz además de su adaptabilidad es que debe ser lo suficientemente simple para que la optimización de la función de costo sea eficiente.

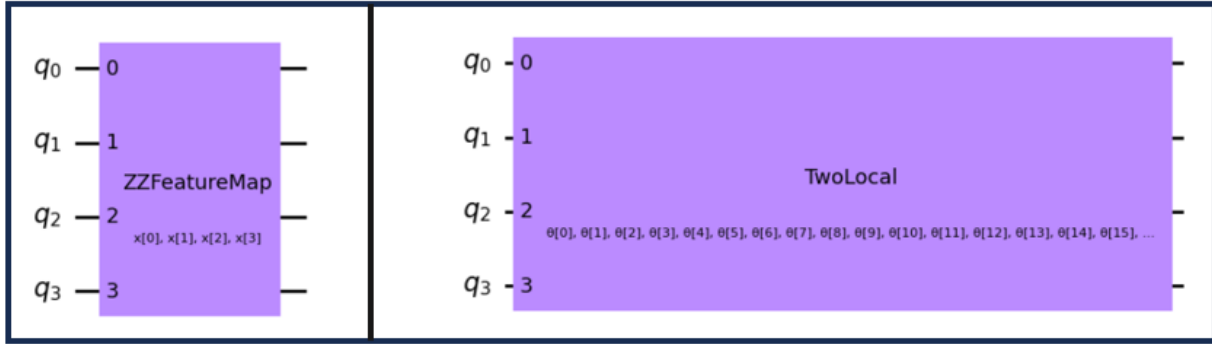
Existen una amplia diversidad de propuestas, una de las más comunes es la encontrada por Dennis, Kitaev, Landahl et al. [32], la cual está compuesta dos capas de compuertas cuánticas parametrizadas, de las cuales la primera actúa únicamente sobre cada qubit del circuito y la segunda actúa sobre pares de qubits. Una de las ventajas de este modelo de ansatz es que está diseñado para ser implementado en computadoras cuánticas de corto término.

### 3.2.2. Feature Map

El mapa de características o *feature map* es uno de los elementos claves para la resolución de problemas de aprendizaje con el uso de QML, el mapa de características al igual que en el caso clásico se trata de un elemento que codifica las características de un conjunto de datos y que sirve como codificador de información para poder llevar a cabo la clasificación. En el caso de QML un Q-FeatureMap es en esencia un circuito variacional de igual manera que lo es un ansatz, sin embargo, cuenta con un objetivo distinto, en este caso el circuito variacional tiene la intención de llevar a cabo un mapeo de los datos clásicos a un estado cuántico. En esencia el mapa de características cuántico está diseñado para capturar la estructura de los datos clásicos.

Como es de suponerse debido a que el mapa de características es el encargado de codificar la información clásica tiene un desempeño importante en el proceso de aprendizaje, pues aunque tiene la posibilidad de adaptarse la naturaleza del mapa de características es tal que cada conjunto de datos tendrá un mejor ajuste según el mapa de características [33].

Uno de los mapas de características más usados es el conocido como ZZ, el cual se en-



**Figura 3.1.** Ejemplo de un mapa de características ZZ (a la izquierda) usado para codificar cuatro características, requiere 4 qubits para llevar a cabo este proceso, se observa esta dependencia de los valores de las características  $x_i$ . A la derecha se muestra el Ansatz Two Local, se observa la dependencia de los parámetros  $\theta_i$  y la expresión compacta.

carga de llevar a cabo la codificación del conjunto de datos de entrada como el producto tensorial de rotaciones de cada qubit y un cambio de fase  $U(\varphi) |\psi_i\rangle = e^{i\varphi} |\psi_i\rangle$  (phase shifting ZZ), un proceso también usado en algoritmos como el algoritmo de estimación de fase.

### 3.3. Algoritmos variacionales cuánticos de aprendizaje supervisado.

Como es de esperar el interés en encontrar algoritmos cuánticos que repliquen el funcionamiento de uno de los estandartes de la inteligencia artificial como lo son las redes neuronales y sus diversas estructuras como: ANN, CNN y Adversarial-NN es alto, debido a esto el desarrollo de estructuras de red que imiten el comportamiento de su contra parte clásica ha tenido un auge en los últimos años, principalmente acompañado del avance experimental en donde recientemente se han llevado a cabo ejecuciones de algoritmos en computadoras cuánticas cada vez más potentes y robustas [34-36].

### 3.3.1. QNN y Clasificador variacional cuántico (VQC).

Las redes neuronales cuánticas (QNN), el VQC junto con sus semejantes pueden considerarse como algoritmos de aprendizaje máquina cuántico, no obstante, es importante enfatizar que el concepto de las QNN proviene de una equivalencia en la capacidad de resolución de problemas, más allá de el análogo biológico, sin embargo, por similitud y conceptualización se suelen ocupar los mismos nombres para las características que los modelos clásicos y cuánticos tienen en común.

La estructura típica de una QNN está definida por tres componentes (véase Figura 3.2), el mapa de características, el cual es una representación cuántica de los datos clásicos, la observable; que determina la estructura de la clasificación, el bloque variacional; el cual se encarga de tratar los datos de entrada de manera que el observable pueda obtener soluciones adecuadas. [37]

El componente clave de los algoritmos es el mapa de características, pues su función es la más importante para poder llevar cabo el proceso de aprendizaje, debido a que en este componente se lleva a cabo la codificación de los elementos a clasificar. Un mapa de características adecuado debe ser tal que la representación de las clases corresponda a elementos que sean esencialmente distintos, de esta manera el bloque variacional podrá encontrar soluciones óptimas al operador y de esta manera hallar resultados de clasificación con alto porcentaje de eficacia.

Sin embargo, es posible encontrar mapas de características que no sean adecuados para la representación de los datos, es decir, que el mapa de características no provea una codificación adecuada para el problema. Cuando esto sucede es posible llevar a cabo la variación del mapa de características mediante la introducción de un conjunto de parámetros auxiliares  $\theta_i$ , bajo la introducción de estos parámetros los operadores que componen al bloque variacional pueden llevar a cabo modificaciones respecto a ellos y así “ajustarse” para que el mapa de características corresponda a una codificación adecuada [38, 39].

El proceso de encontrar un mapa de características adecuado es inherentemente complicado y se han ocupado una diversidad de técnicas como las utilizadas en los algoritmos cuánticos variacionales (VQC<sup>4</sup>)[40] los cuales tienen una gran similitud con las QNN debido a que ambos aproximan una función mediante el aprendizaje de parámetros. Debido a que en el cómputo cuántico todas las operaciones son reversibles, pues las operaciones son en esencia operadores lineales unitarios, la estructura de una QNN y de un clasificador variacional cuántico tiene elementos análogos a los de las ANN con diferencias marcadas, por ejemplo, en lugar de tener una serie de funciones de activación como en el caso de las ANN, en las QNN usan capas de entrelazamiento para que de esta manera haya estructuras multicapa.

Para poder evaluar el rendimiento de un mapa de características ( $\hat{F}(x_i)$ ) en particular se hace uso comúnmente de la siguiente métrica que caracteriza al error del mapa de características

$$Error(\hat{F}) = - \sum_{i,j} y_i y_j \langle 0 | \hat{F}(x_i)^\dagger \hat{F}(x_j) | 0 \rangle, \quad (3.4)$$

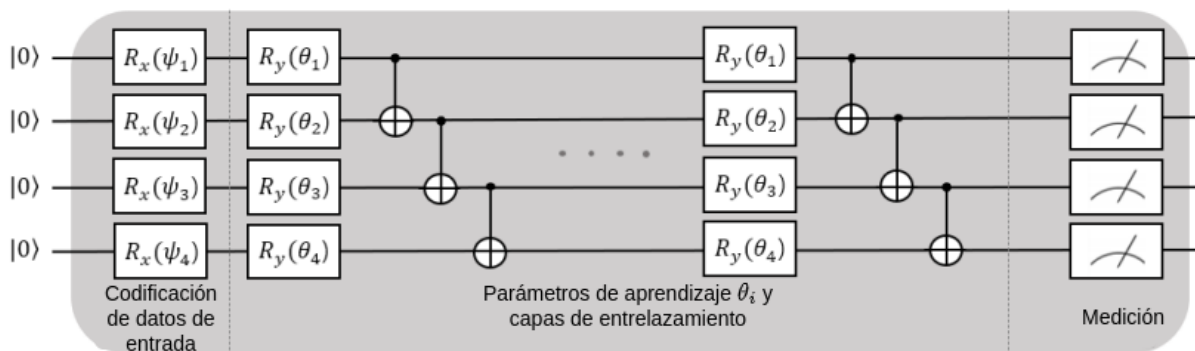
esta métrica garantiza que el mapa de características funciona adecuadamente, cuando esto sucede se espera que el error sea una cantidad cercana a cero [33].

En esencia el concepto de un clasificador variacional cuántico y una QNN es prácticamente equivalente con la salvedad de la estructura de la función de costo, en el caso de que la función de costo<sup>5</sup> a optimizar por un VQC sea una función de entropía o paridad se dice que son completamente equivalentes, sin embargo, este tipo de funciones de costo por minimizar son usadas dentro del contexto de las QNN, una de las ventajas de las QNN es que además es posible llevar a cabo codificaciones explícitas para la representación de las mismas, en el caso de los algoritmos variacionales cuánticos pese a que es posible no es una práctica común. Otra variación común es que, como es en el caso de la experimentación comentada

---

<sup>4</sup>Suele usarse el mismo acrónimo que para los clasificadores variacionales cuánticos pues su funcionamiento depende de los algoritmos cuánticos variacionales, no obstante, es importante mencionar que un clasificador variacional cuántico es un algoritmo variacional, sin embargo un algoritmo variacional no es necesariamente un clasificador variacional.

<sup>5</sup>En el caso del ML clásico la función de costo a minimizar es comúnmente la función de pérdida o la desviación cuadrática media, este concepto proviene de una analogía física con la minimización de la energía de un sistema.



**Figura 3.2.** Representación de la estructura de una red neuronal cuántica. Imagen tomada de Kwak, Yun, Jung et al.[37]

más adelante las QNN pueden tener un enfoque híbrido y funcionar con optimizadores no necesariamente diseñados para circuitos cuánticos.

### 3.3.2. Clasificador de soporte vectorial cuántico (QSVC)

El clasificador cuántico de soporte vectorial (QSVC) es una adaptación de una máquina de soporte vectorial clásica SVC que en la librería de Qiskit recibe el nombre de QSVC. A diferencia de los clasificadores variacionales cuánticos y las QNN que están compuestos principalmente por dos bloques variacionales en el caso del QSVC se tiene que el único elemento cuántico es la posibilidad de llevar a cabo la implementación del SVC con un kernel basado en un circuito cuántico, es decir, permite llevar a cabo la clasificación de conjuntos de datos de manera clásica a partir de un mapa de características cuántico, en este caso el QSVC por razones obvias no cuenta con una propuesta de solución o ansatz pues el proceso de solución es clásico. El hecho de que el proceso de clasificación sea clásico permite evaluar la capacidad de los mapas de características cuánticos y ver si suponen una ventaja en expresabilidad respecto a sus contrapartes clásicas, además implementaciones como las presentadas por Shalev-Shwartz, Singer y Cotter [41] permiten mediante procesos de optimización llevar a cabo pruebas en conjuntos numéricos con una gran cantidad de instancias, algo que es una limitante conocida dentro del cómputo cuántico.

## 3.4. Ruido y técnicas de mitigación.

Una de las principales adversidades de la computación cuántica es que en la actualidad solo es posible llevar a cabo implementaciones de corto término, esto debido a que entre más tiempo consume la ejecución la interacción del sistema experimental con el ambiente tiende a generar ruido que afecta el funcionamiento de los algoritmos de cómputo cuántico. Lo descrito anteriormente está bien estudiado en algoritmos cuánticos deterministas, es decir, en algoritmos como el Deutsch-Josza [42] que evalúa el comportamiento de una función binaria, sin embargo, el estudio de las implicaciones del ruido en algoritmos cuánticos no deterministas no está bien comprendido, sobre todo de cara a la realización de tareas de aprendizaje máquina[43]. En [8] se presenta un estudio en donde se sugiere que la presencia de ruido puede favorecer a algoritmos de ML en casos donde la clasificación de datos presente datos no etiquetados de manera adecuada.

### 3.4.1. Ruido de Pauli

Existen una variedad de modelos de ruido utilizados en el cómputo cuántico, desde modelos de relajación térmica que caracterizan el proceso de interacción con el ambiente hasta modelos que caracterizan el comportamiento de los nodos de un circuito cuántico experimental, sin embargo, uno de los modelos de ruido más sencillos de simular, pero no por eso menos importante es el conocido como ruido de Pauli.

El ruido de Pauli recibe este nombre debido a la característica de presentarse precisamente en la acción de las compuertas de Pauli,<sup>6</sup> este tipo de error se caracteriza por realizar una rotación extra sobre un qubit, supongase que se intentaba llevar a cabo una rotación parametrizada a lo largo del eje x de la forma  $R_x(\theta)$ , en este caso el ruido de Pauli genera una rotación de la forma  $R_x(\theta + \varepsilon)$  donde  $\varepsilon$  es un factor de ruido, este factor de ruido provoca que a la hora de calcular el valor esperado de un estado sobre una observable el valor del mismo varíe un poco respecto del resultado esperado. Es importante notar que además, dado

---

<sup>6</sup>En el contexto del cómputo cuántico se les llama compuertas de Pauli, en el contexto de la mecánica cuántica nos referimos simplemente a la posibilidad de que haya una rotación según  $\sigma_i(\varphi)$

la linealidad de los operadores en mecánica cuántica la expresión

$$R_x(\theta + \varphi) = R_x(\theta)R_x(\varphi)^7$$

La idea de llevar a cabo correcciones sobre este tipo de errores ha sido una constante incógnita en el pasado reciente, estudios como el presentado en [44] muestran evidencia empírica de que es posible aprender este tipo de ruido.

Es importante además comentar que en el estado del arte se presentan estudios como el desarrollado por Wu, Yu y Segal[45] en donde se lleva a cabo una discusión detallada de la dinámica de un sistema de qubits en interacción en un baño térmico, si bien este tipo de resultados analíticos son de utilidad para el desarrollo teórico, así como para la comprensión del comportamiento de un sistema físico en particular, analogías directas o resultados analíticos sobre circuitos cuánticos parametrizados no son posibles de hallar, pues el análisis requiere justamente de una estructura bien definida del operador, esta en el caso de los circuitos cuánticos no está presente hasta que se determinan los mismos.

Pese a la imposibilidad de hallar una descripción analítica o experimental sin el acceso directo al hardware de una computadora cuántica una buena estimación del error en estudios como [46, 47] se presenta una descripción detallada de los cambios en los qubits según la variación de temperatura.

De la misma forma es importante recalcar que en este trabajo no se llevará a cabo una discusión explícita del ruido de Pauli en función de la temperatura, a lo largo de este texto el ruido será considerado como proveniente de causas no explícitas, es decir, simplemente será el análogo a la colección de todas las contribuciones de ruido, para discusiones más detalladas es posible observar [46] en la figura 4.

---

<sup>7</sup>Esta expresión es válida únicamente cuando las rotaciones son en la misma dirección, de igual manera es conveniente denotar que el significado de estas rotaciones hacen referencia a “aplicaciones” sucesivas de estos operadores y no al concepto típico de multiplicación



### 3.4.2. Métodos de mitigación de ruido

La necesidad de llevar a cabo cómputo en ambientes ruidosos ha llevado a la creación de métodos de mitigación de ruidos, es decir, de técnicas que plantean reducir la frecuencia de errores, es importante diferenciar el concepto de mitigación de errores del de corrección de errores, pues son esencialmente distintos.

Los métodos de mitigación de errores más comunes son los conocidos como M3, ZNE y PCE, es importante mencionar que este conjunto de métodos tiene aplicaciones determinadas. En particular M3 [4] es un método de mitigación de errores encargado de llevar a cabo el proceso de mitigación en las mediciones de la probabilidad mediante el método de reducción de la dimensionalidad.

ZNE es el acrónimo de Zero Noise Extrapolation, este método de mitigación funciona de manera completamente distinta a M3, ZNE se encarga de extrapolar el posible ruido presente en la ejecución del circuito cuántico y de esta manera estimar el impacto del ruido y de esta manera disminuir su impacto.

### 3.4.3. Uso del sampler en Qiskit.

El uso de las computadoras cuánticas reales para la resolución de ciertos tipos de problemas computacionales es todavía una limitante, pues pese a los grandes avances experimentales la ausencia de un equivalente tradicional a la memoria cuántica plantea limitantes en la implementación, no obstante, teniendo estas limitantes en mente se han desarrollado entornos de ejecución que simulan con alta precisión el comportamiento de algoritmos en entornos reales.

Existen dentro del entorno de Qiskit, librería desarrollada por IBM para el desarrollo de cómputo cuántico mediante el modelo de circuitos cuánticos, dos tipos de entornos de ejecución para los algoritmos, ambos con objetivos esencialmente distintas, una conocida como *estimador* y otra conocida como *sampler*, la diferencia principal entre estas rutinas de ejecu-

ción recae en la utilización y objetivo de la ejecución, el uso de un estimador es útil cuando se trata de tareas que tienen un comportamiento determinado estimando el valor esperado de una colección de observables para una colección de parámetros de entrada, por otro lado, en tareas como las que se llevan a cabo en el área de QML el uso del sampler es particularmente importante pues se encarga de llevar a cabo una estimación de cuasiprobabilidad con mitigación de ruido incorporada.

El hecho de que el sampler nos otorgue una mitigación de errores es de particular importancia para este trabajo de investigación, pues nos permite verificar si esta estimación del ruido que tiene como sustento por default el uso del algoritmo M3 es útil ante la presencia de ruido de Pauli en circuitos cuánticos que se usarán para tareas de aprendizaje.

# 4

## Configuración del experimento y resultados.

Para poder evaluar el desempeño de los algoritmos mencionados previamente (VQC, QNN, QSVC) se llevó a cabo la implementación de un script que permite llevar a cabo la experimentación para evaluar el comportamiento de los algoritmos. El script está diseñado para probarse en tareas de clasificación mediante el uso de aprendizaje supervisado, es decir, toma un conjunto de datos de entrenamiento y aprende para poder llevar a cabo la clasificación de valores de prueba..<sup>1</sup>

### 4.0.1. Procesamiento de datos y conjuntos de datos usados.

El procesamiento de los datos para la realización de la clasificación se llevó a cabo considerando las limitantes computacionales de simulación y de posible migración a backends reales, por lo que de momento el script solo puede tomar como entrada conjuntos a lo más cuatro características, en caso de que el conjunto de datos no cumpla con esta condición se llevará a cabo una reducción de las mismas por medio de la aplicación del algoritmo de

---

<sup>1</sup>El script que lleva a cabo la experimentación está disponible en <https://github.com/EisaacJC/QMLTests/tree/main>

PCA, dejando únicamente las cuatro características más representativas del conjunto de datos. Este paso es en particular importante, pues existirán datos en donde las características más relevantes no sean suficientes para llevar a cabo clasificación.

Para poder llevar a cabo la experimentación se usaron cuatro conjuntos de datos, de los cuales dos corresponden a los conjuntos típicos del estado del arte “Iris” y “Credit”, de manera complementaria se usaron dos conjuntos, uno conocido como “ad\_hoc”, el cual es un modelo estándar para la evaluación de algoritmos de QML, pues el mapa de características, así como el ansatz funcionan de manera adecuada para la clasificación de este conjunto de datos. Finalmente el último conjunto de prueba consistió en tomar un conjunto de datos sintéticos generados por la librería Sklearn, el cual cumple con las mismas condiciones que los anteriores.

#### 4.0.2. Funcionamiento del Script

El script está compuesto por dos funciones principales, una dedicada a la ejecución del experimento y otra relacionada con la implementación del ruido (Ruido de Pauli). La función encargada de simular el ruido consta de la adición de compuestas cuánticas parametrizadas  $(Rx(r_1), Ry(r_2), Rz(r_3))$  con un valor aleatorio ( $r_i$ ) que oscila entre  $[0, 1]$ , lo que corresponde a una rotación aleatoria que genera un error dentro del intervalo de  $[0\%, 15.91\%]$ , es importante mencionar que para la experimentación se usó como base únicamente el mapa de características conocido como “ZZFeatureMap” con la elección de entrelazamiento lineal y 2 repeticiones, de igual manera el ansatz elegido fue el conocido como “TwoLocal”.

La función principal “experimentación” permite seleccionar el algoritmo a evaluar, las opciones para este segmento de código son: VQC, Torch y QSVC, las cuales corresponden a los algoritmos VQC, QNN(vía Torch) y QSVC. Una vez elegido el algoritmo se hace uso de la opción “noise\_method” que toma como entradas “feature” o “ansatz”, esta opción permite elegir el lugar en donde se llevará a cabo el ruido tomando como opciones aplicarlo sobre el mapa de características o sobre el ansatz.

El script está diseñado para que sea ejecutado una cantidad determinada de veces y otorgue

los resultados en un documento separado por comas para su posterior procesado y análisis con el uso de Wolfram Mathematica.

### 4.0.3. Parámetros usados en los algoritmos.

Con el objetivo de buscar que la comparación entre los algoritmos fuese lo más justa posible se llevó a cabo la elección de las propuestas de ansatz y mapas de características más justas posibles.

- Ansatz: Se eligió para los casos en donde era posible usar un ansatz TwoLocal con bloques de rotación controlados mediante las matrices de Pauli  $R_x(\theta_i), R_y(\theta_j)$  (Torch, VQC) y con bloques de entrelazamiento  $Cz$ .<sup>2</sup>
- FeatureMap: El mapa de características usado a lo largo de los tres algoritmos fue el “ZZFeatureMap” con entrelazamiento lineal.
- Además para el procesamiento de los datos se usó el 80 % del conjunto para llevar a cabo pruebas y 20 % como conjunto de entrenamiento.
- Parámetro de ruido: para realizar comparaciones que puedan emular de mejor manera el comportamiento de los algoritmos en un ambiente real se llevaron a cabo la elección de parámetros  $r_i \in [0, 1]$  aleatorios a cada qubit del circuito de ansatz o feature map.
- Número de ejecuciones: Se eligieron 27 ejecuciones en cada conjunto de datos (VQC y QNN)<sup>3</sup> para tener suficiencia estadística y poder llevar a cabo comparaciones utilizando a la desviación estándar como una métrica confiable.

### 4.0.4. Metodología de análisis

Para llevar a cabo la evaluación de los experimentos se consideró comparar la distribución de los valores de precisión, f1-score y recall de cada ejecución comparando las distribuciones con ruido de Pauli y en ausencia de ruido de Pauli para cada conjunto de datos.

---

<sup>2</sup>Estas opciones son las usadas típicamente en el uso de algoritmos de QML.

<sup>3</sup>Para la prueba de QSVC solo llevaron 27 ejecuciones en el conjunto de Iris y ad.hoc debido a los altos tiempos de compilación.

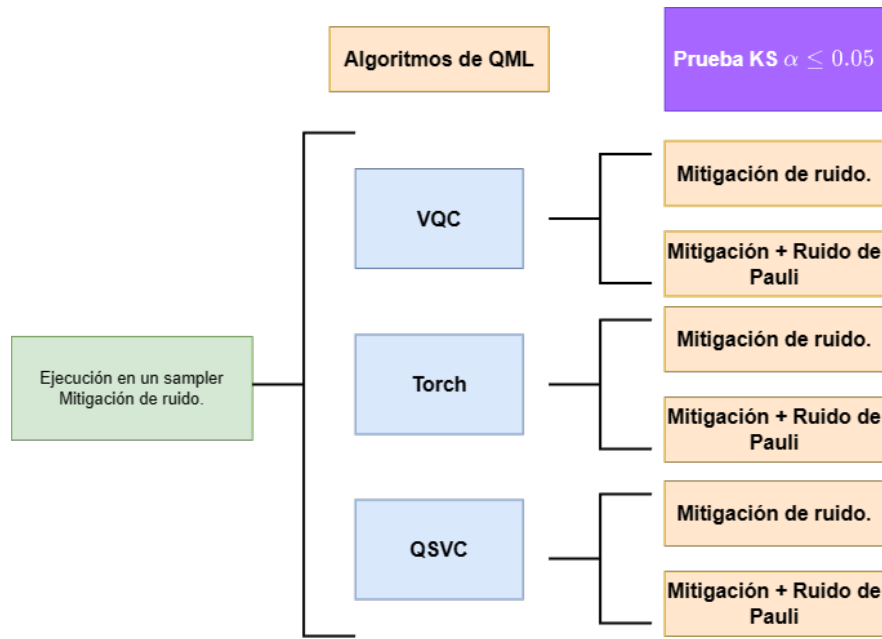


Figura 4.1. Diagrama de la metodología de análisis.

Con el fin de comparar si las distribuciones resultantes ejecutadas en el ambiente mitigado son distintas se realiza una prueba de hipótesis, en donde se define como  $\mathcal{H}$  a la hipótesis que considera que **“la distribución de las métricas del QML es estadísticamente semejante con la distribución del algoritmo en ausencia de Ruido de Pauli.”** y como  $\mathcal{H}_0$  la hipótesis nula, es decir, que hay una diferencia estadística significativa entre ambas distribuciones.

Para llevar a cabo la prueba de hipótesis se consideró llevar a cabo la prueba de Kolmogorov-Smirnov, una prueba no paramétrica que compara que tan semejantes son dos distribuciones empíricas mediante la comparación de su CDF.

La prueba de Kolmogorov-Smirnov se selecciona con un parámetro de tolerancia  $\alpha \leq 0.05$ , es decir, si se cumple con este valor se considera que ambas distribuciones son estadísticamente semejantes.

## 4.1. VQC

Conjunto de datos Ad\_hoc El conjunto de datos “ad\_hoc” como su nombre lo dice es un conjunto de datos estándar que se ocupa para verificar el funcionamiento de los algoritmos cuánticos pues se trata de un conjunto diseñado para que funcione de manera adecuada con el mapa de características y con el ansatz propuestos. La realización de los experimentos con este conjunto de datos se llevó a cabo considerando que el mismo consta de dos características y dos clases, de manera que el aprendizaje es rápido en los algoritmos probados.

### Ruido en el mapa de características.

Bajo la acción de ruido de Pauli en el mapa de características es posible observar (véase 4.6) que la distribución de los resultados de la precisión son semejantes a simple vista, sin embargo, es necesario llevar a cabo una inspección de los parámetros estadísticos para describir de manera adecuada el comportamiento de las distribuciones. A continuación se muestran los resultados estadísticos de las ejecuciones. En este caso pese a que el valor de

**Tabla 4.1.** Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos ad\_hoc con la comparación de la acción de ruido en el mapa de características (véase 4.6).

	Valor mínimo.	Mediana.	Promedio.	Desviación Estándar.	25 %	75 %	V. Máximo.
Ruido de Pauli	0.66	0.85	0.89	0.07	0.95	0.92	0.95
Sin Ruido	0.6	0.88	0.84	0.09	0.77	0.92	0.92

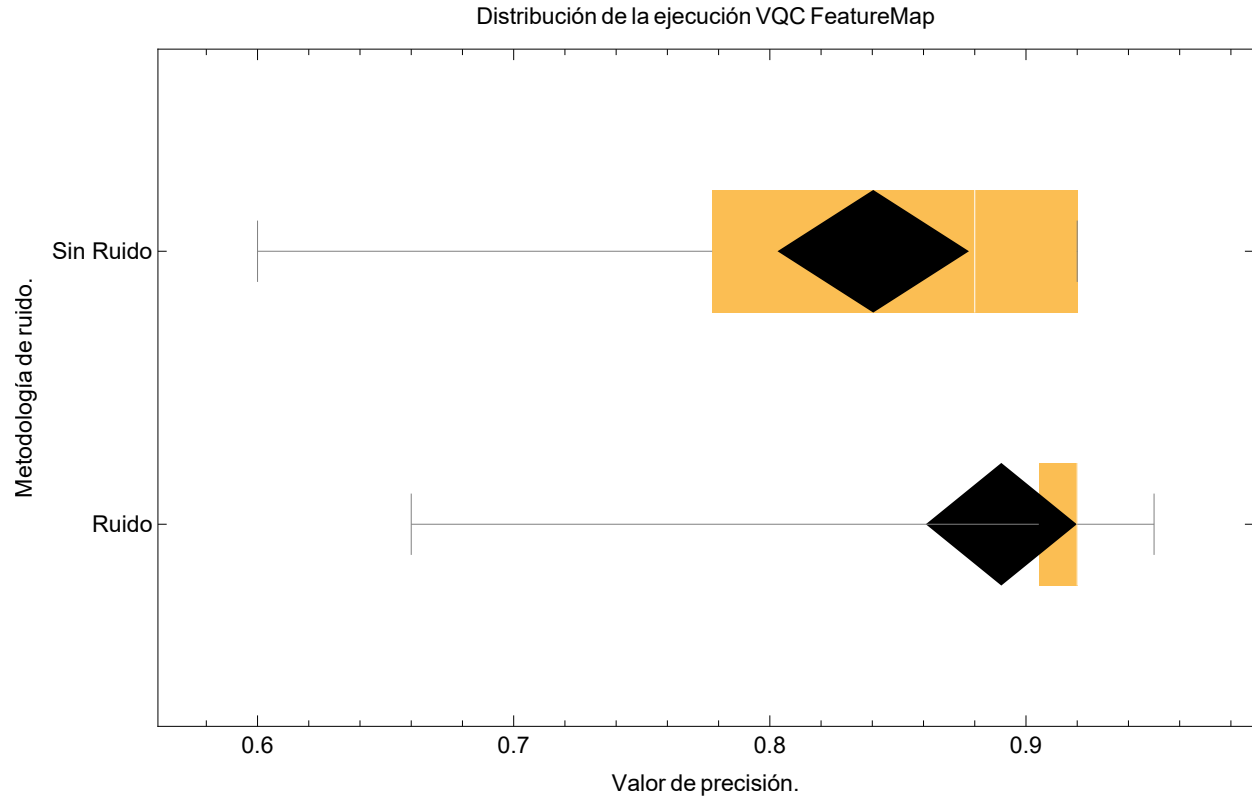
p es considerablemente cercano a cero no se puede asegurar que los datos no provienen de la misma distribución,<sup>4</sup> esto es particularmente interesante pues supone que podría ser una simple coincidencia que el comportamiento de las distribuciones sea semejante, no obstante es importante mencionar que el hecho de llevar a cabo más estadísticas para evaluar los

**Tabla 4.2.** Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido.

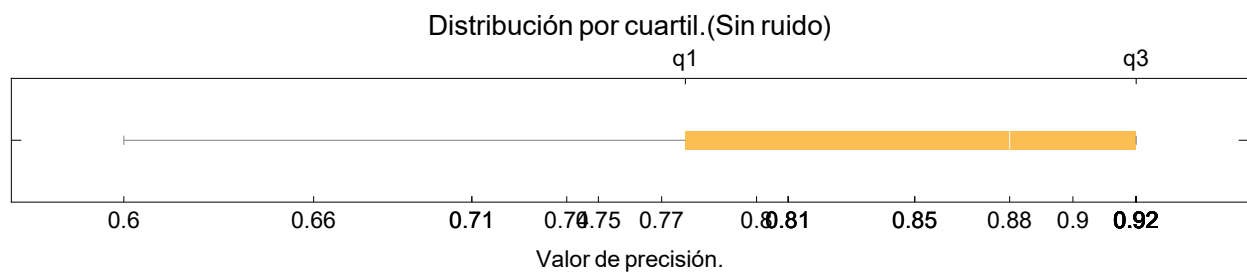
	Estadístico	P-Value
Kolmogorov -Smirnov	0.296296	0.0795925

<sup>4</sup>Considerando un parámetro de significancia  $\alpha = 0.05$ .

resultados si bien puede ser útil puede llevar a la inclusión de sesgo, pues es bien sabido que si el número de muestras aumenta el valor del factor  $p$  tiende a disminuir, lo que es usualmente conocido como “p-hacking” o “data-dredging”[48].

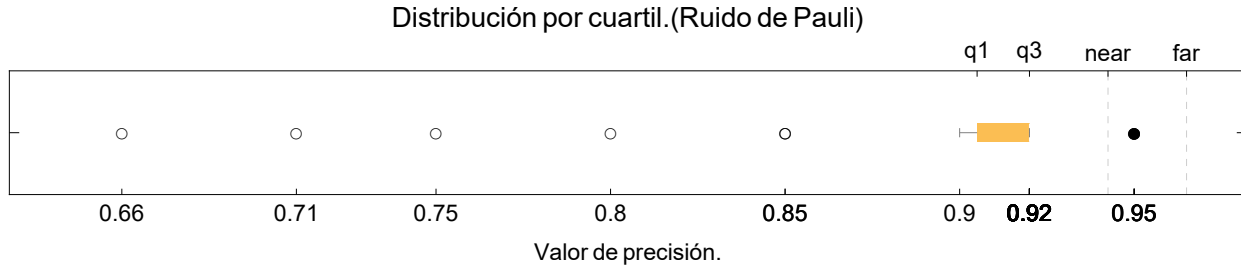


**Figura 4.2.** Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos ad\_hoc mediante la acción de ruido en el mapa de características.



**Figura 4.3.** Distribución por cuartiles del valor de precisión del modelo en ausencia de ruido en el mapa de características.





**Figura 4.4.** Distribución por cuartiles del valor de precisión del modelo en presencia de ruido en el mapa de características.

### Ruido en el ansatz.

Cuando se efectúa la acción de ruido de Pauli sobre el ansatz es posible observar (véase 4.9) que la distribución de los valores de precisión parecieran tener un mayor grado de variabilidad que la ejecución sin ruido de Pauli. A continuación se muestran los resultados de las ejecuciones con sus estadísticas descriptivas (véase 4.7).

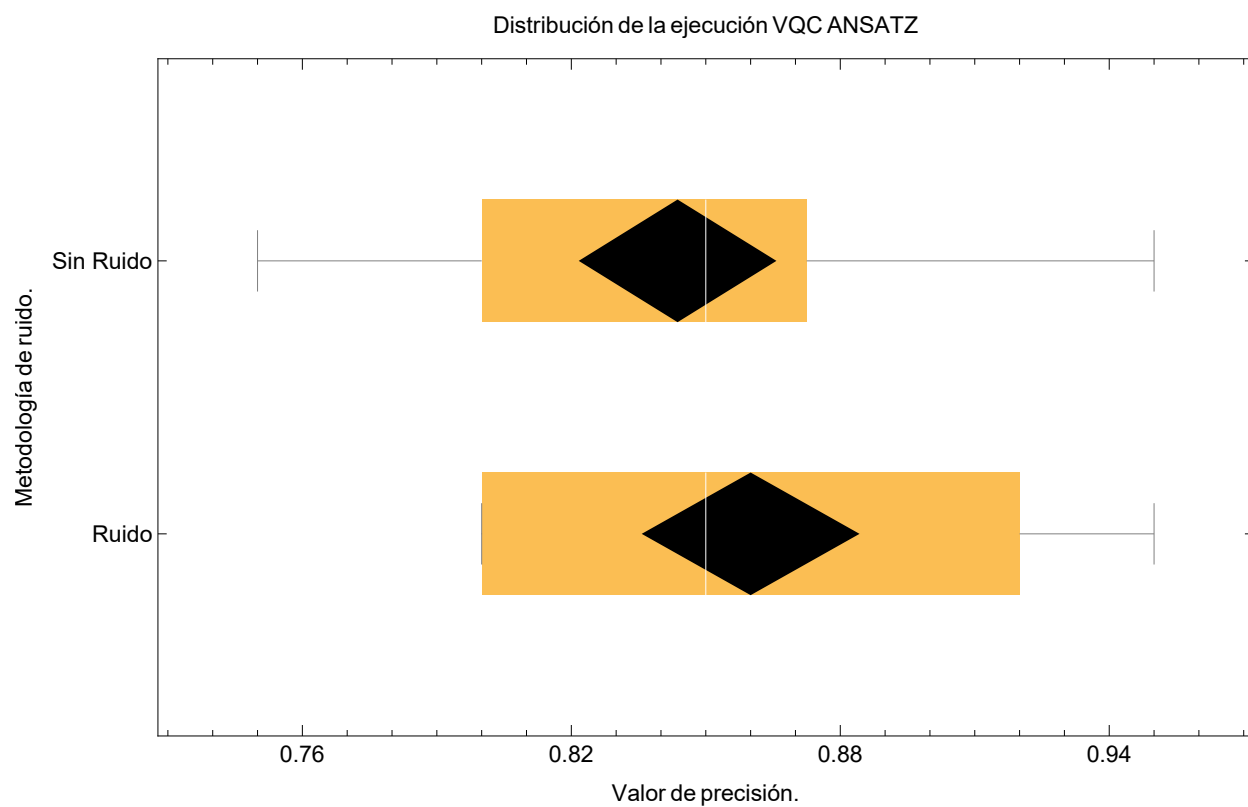
**Tabla 4.3.** Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos ad.hoc con la comparación de la acción de ruido en el mapa de características (véase 4.5).

	Valor mínimo.	Mediana.	Promedio.	Desviación Estándar.	25 %	75 %	V. Máximo.
Ruido de Pauli	0.8	0.85	0.84	0.05	0.8	0.87	0.95
Sin Ruido	0.75	0.85	0.86	0.09	0.8	0.92	0.95

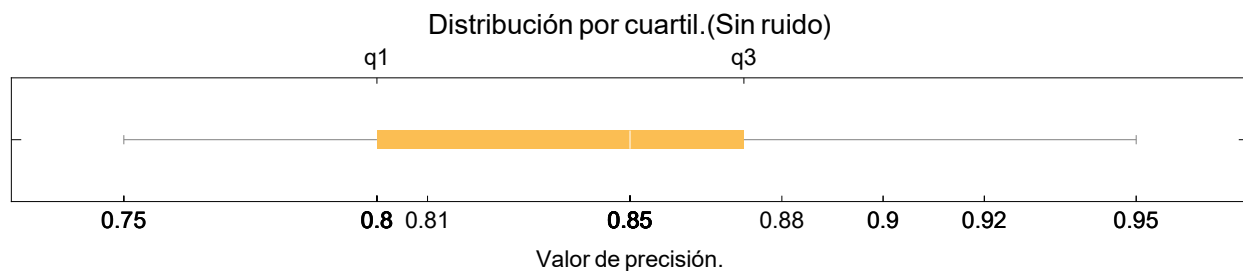
**Tabla 4.4.** Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido.

	Estadístico	P-Value
Kolmogorov -Smirnov	0.185185	0.412838

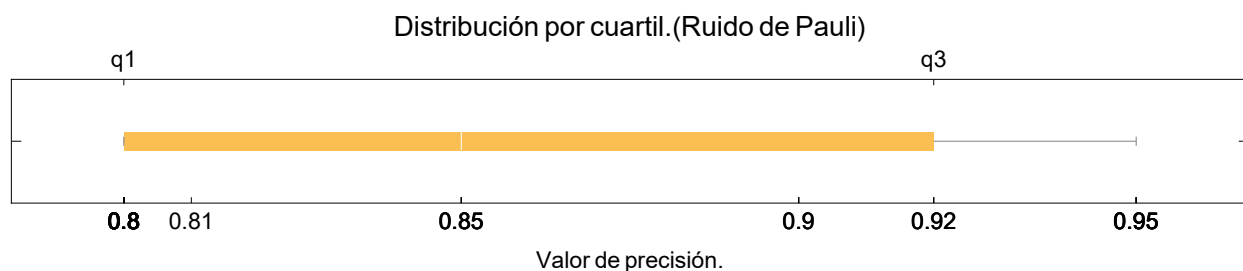
En este caso es posible observar que el valor de  $p$  difiere significativamente de cero, por lo que en esta ocasión no hay evidencia suficiente para decir si existe o no diferencia significativa entre la distribución con ruido y ausencia de ruido sobre el ansatz.



**Figura 4.5.** Distribución de la precisión del modelo en presencia y en ausencia de ruido en el ansatz para el conjunto de datos `ad_hoc`.



**Figura 4.6.** Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos ad\_hoc en ausencia de ruido en el mapa de características.



**Figura 4.7.** Distribución por cuartiles del valor de precisión del modelo en presencia de ruido en el mapa de características.

### 4.1.1. Conjunto de datos de Iris.

#### Ruido en el mapa de características.

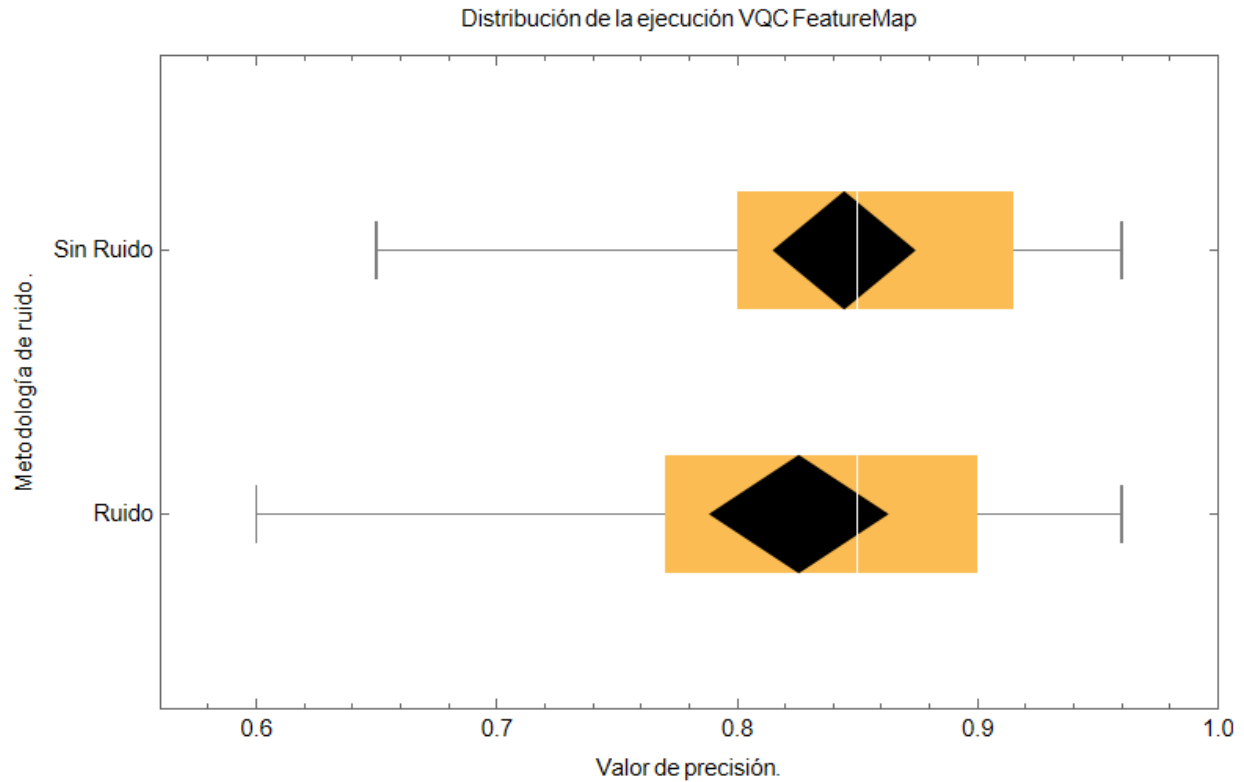
Bajo la acción de ruido de Pauli en el mapa de características es posible observar (véase 4.8) que la distribución de los resultados de la precisión son semejantes a simple vista, sin embargo, es necesario llevar a cabo una inspección de los parámetros estadísticos para describir de manera adecuada el comportamiento de las distribuciones. A continuación se muestran los resultados estadísticos de las ejecuciones.

**Tabla 4.5.** Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de Iris con la acción de ruido en el mapa de características. (véase 4.8)

	Valor mínimo.	Mediana.	Promedio.	Desviación Estándar.	25 %	75 %	Valor máximo.
Ruido de Pauli	0.6	0.85	0.82	0.09	0.8	0.9	0.96
Sin Ruido	0.65	0.85	0.84	0.07	0.8	0.91	0.96

**Tabla 4.6.** Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido.

	Estadístico	P-Value
Kolmogorov -Smirnov	0.148148	0.746643

**Figura 4.8.** Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos de Iris mediante la acción de ruido en el mapa de características.

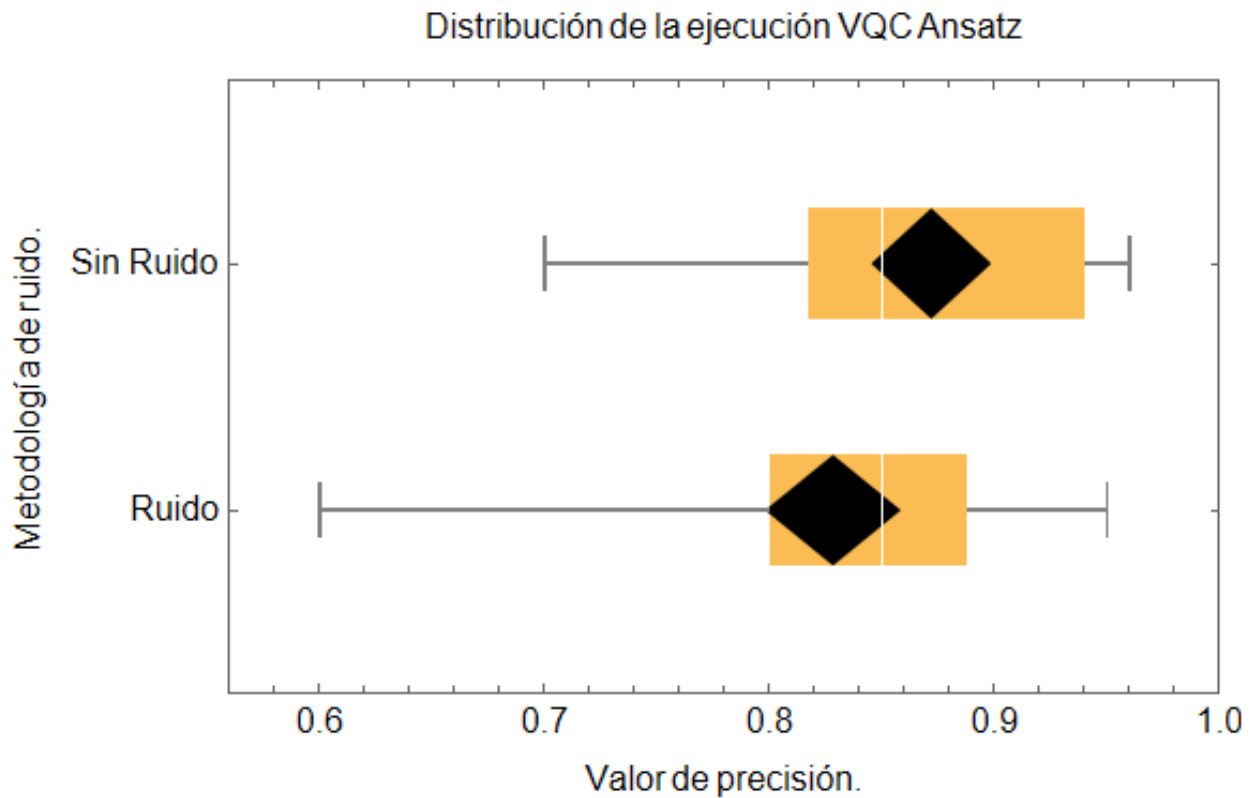
### Ruido en el ansatz.

Cuando se efectúa la acción de ruido de Pauli sobre el ansatz es posible observar (véase 4.9) que la distribución de los valores de precisión parecieran tener un mayor grado de variabilidad que la ejecución sin ruido de Pauli. A continuación se muestran los resultados de las ejecuciones con sus estadísticas descriptivas (véase 4.7).

Con motivo de indagar si los resultados obtenidos con ruido de Pauli y en ausencia del mismo provenían de la misma distribución se llevó a cabo la ejecución de la prueba de hipótesis de Kolmogorov-Smirnov teniendo como resultados: Donde debido a que el valor de  $p$  es cercano a cero es poco probable que ambos datos provengan de la misma distribución

**Tabla 4.7.** Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de Iris. (véase 4.9)

	Valor mínimo.	Mediana.	Promedio.	Desviación Estándar.	25 %	75 %	V. Máximo.
Ruido de Pauli	0.6	0.85	0.82	0.07	0.8	0.88	0.95
Sin Ruido	0.7	0.85	0.87	0.06	0.81	0.94	0.96



**Figura 4.9.** Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos Iris mediante la acción de ruido en el ansatz.

**Tabla 4.8.** Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido.

	Estadístico	P-Value
Kolmogorov -Smirnov	0.259259	0.166355

(véase 4.16).

### 4.1.2. Conjunto de datos sintéticos.

#### Ruido en el ansatz.

Bajo la acción de ruido de Pauli en el ansatz es posible observar (véase 4.10) que la distribución de los resultados de la precisión son semejantes a simple vista, sin embargo, es necesario llevar a cabo una inspección de los parámetros estadísticos para describir de manera adecuada el comportamiento de las distribuciones. A continuación se muestran los resultados estadísticos de las ejecuciones.

**Tabla 4.9.** Distribución del valor de precisión en el conjunto de datos sintéticos (véase 4.9)

	Valor mínimo.	Mediana.	Promedio.	Desviación Estándar.	25 %	75 %	V. Máximo.
Ruido de Pauli	0.48	0.62	0.60	0.07	0.56	0.69	0.75
Sin Ruido	0.48	0.62	0.61	0.07	0.56	0.64	0.75

**Tabla 4.10.** Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido.

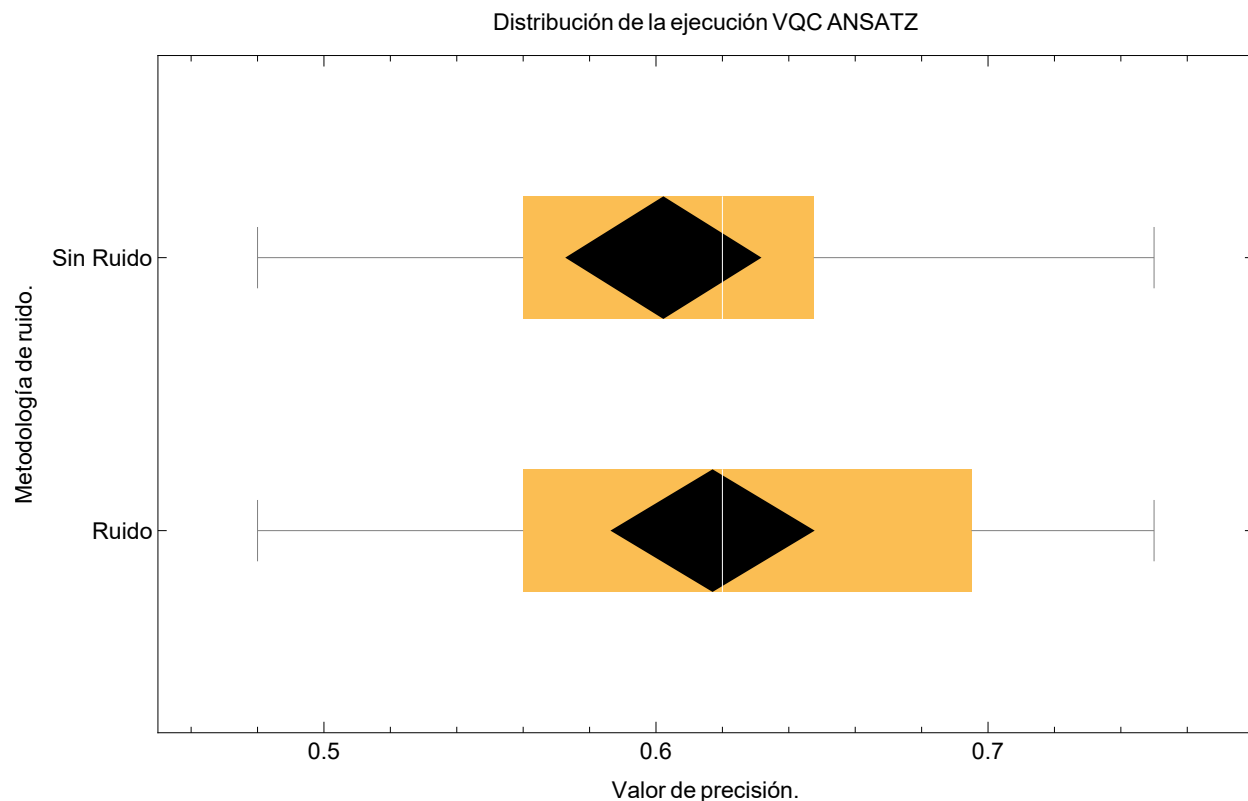
	Estadístico	P-Value
Kolmogorov -Smirnov	0.148148	0.74243

En esta ocasión el análisis estadístico nos indica que existe cierta semejanza entre las distribuciones, sin embargo, el resultado no es suficientemente robusto para llevar a cabo aseveraciones, no obstante es notorio según la tabla 4.9 que el comportamiento de los primeros momentos estadísticos así como los de sus cuartiles son semejantes.

#### Ruido en el mapa de características.

Cuando se efectúa la acción de ruido de Pauli sobre el mapa de características es notorio que posible observar (véase 4.11) que la distribución de los valores de precisión parecieran tener un mayor grado de variabilidad que la ejecución sin ruido de Pauli. A continuación se muestran los resultados de las ejecuciones con sus estadísticas descriptivas (véase 4.11).

Con motivo de indagar si los resultados obtenidos con ruido de Pauli y en ausencia del mismo provenían de la misma distribución se llevó a cabo la ejecución de la prueba de



**Figura 4.10.** Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos sintéticos generados con Sklearn mediante la acción de ruido en el mapa de características.

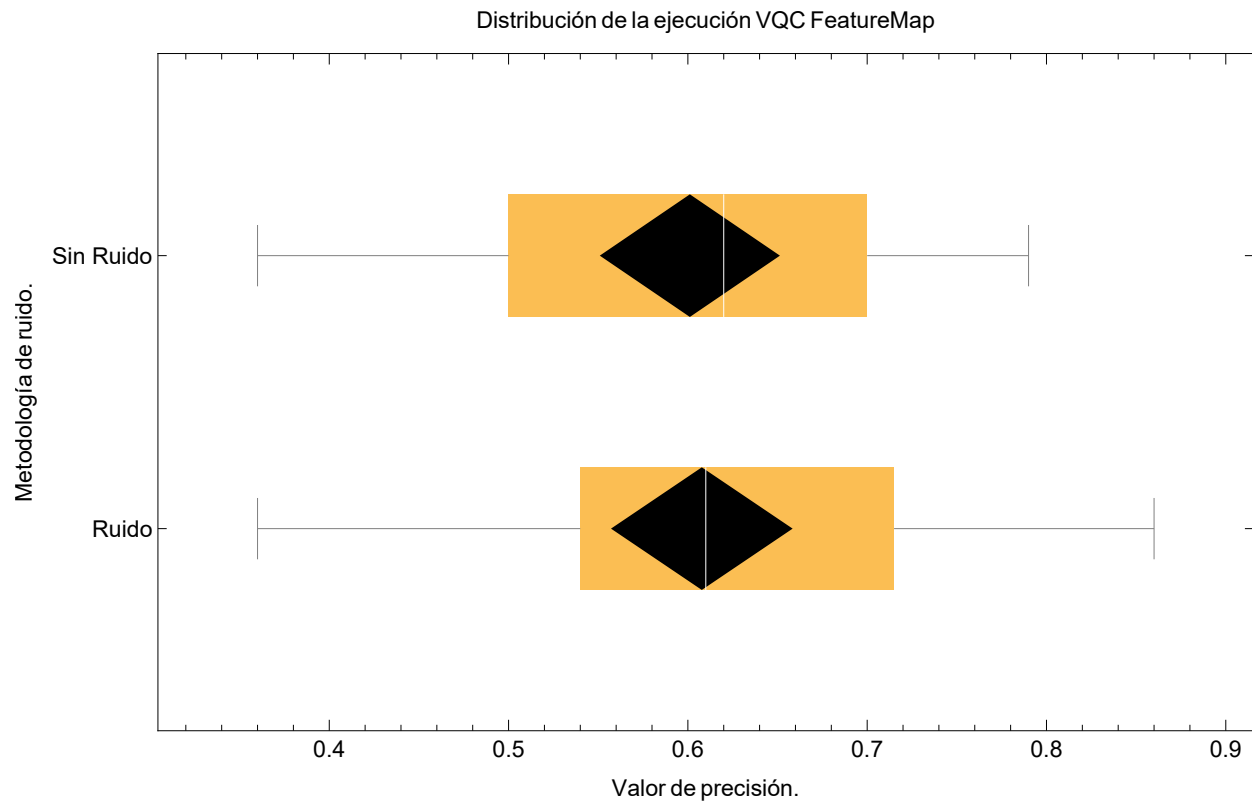
**Tabla 4.11.** Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de datos sintéticos generados con sklearn(véase 4.11)

	Valor mínimo.	Mediana.	Promedio.	Desviación Estándar.	25 %	75 %	V. Máximo.
Ruido de Pauli	0.6	0.61	0.607	0.127	0.54	0.71	0.79
Sin Ruido	0.36	0.62	0.601	0.128	0.5	0.71	0.86

**Tabla 4.12.** Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido.

	Estadístico	P-Value
Kolmogorov -Smirnov	0.148148	0.880872

hipótesis de Kolmogorov-Smirnov teniendo como resultados: Donde el valor de  $p$  cercano a 1 indica que existe una posibilidad de que las distribuciones de datos sean semejantes, esto puede evidenciarse a partir de los datos estadísticos presentados previamente en donde las medidas de los primeros momentos de la distribución eran bastante semejantes entre la



**Figura 4.11.** Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos sintéticos mediante la acción de ruido en el mapa de características.

ejecución con ruido y en ausencia del mismo.

### 4.1.3. Conjunto de datos de crédito.

#### Ruido en el mapa de características.

Bajo la acción de ruido de Pauli en el mapa de características es posible observar (véase 4.12) que la distribución de los resultados de la precisión son semejantes a simple vista, sin embargo, es necesario llevar a cabo una inspección de los parámetros estadísticos para describir de manera adecuada el comportamiento de las distribuciones. A continuación se muestran los resultados estadísticos de las ejecuciones.

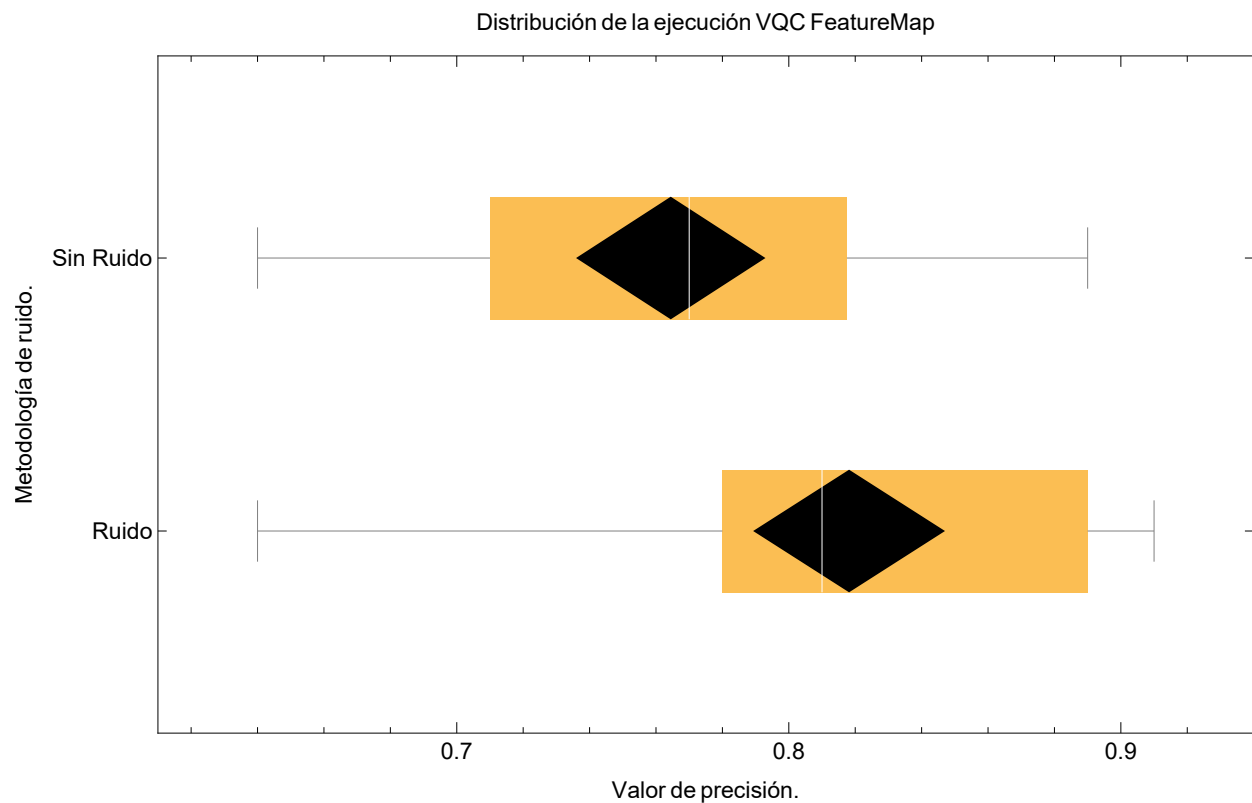


**Tabla 4.13.** Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de credito con la acción de ruido en el mapa de características. (véase 4.12)

	Valor mínimo.	Mediana.	Promedio.	Desviación Estándar.	25 %	75 %	Valor máximo.
Ruido de Pauli	0.6	0.81	0.64	0.07	0.8	0.89	0.91
Sin Ruido	0.64	0.77	0.66	0.09	0.71	0.81	0.89

**Tabla 4.14.** Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido.

	Estadístico	P-Value
Kolmogorov -Smirnov	0.296296	0.122418



**Figura 4.12.** Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos de Credit mediante la acción de ruido en el mapa de características.

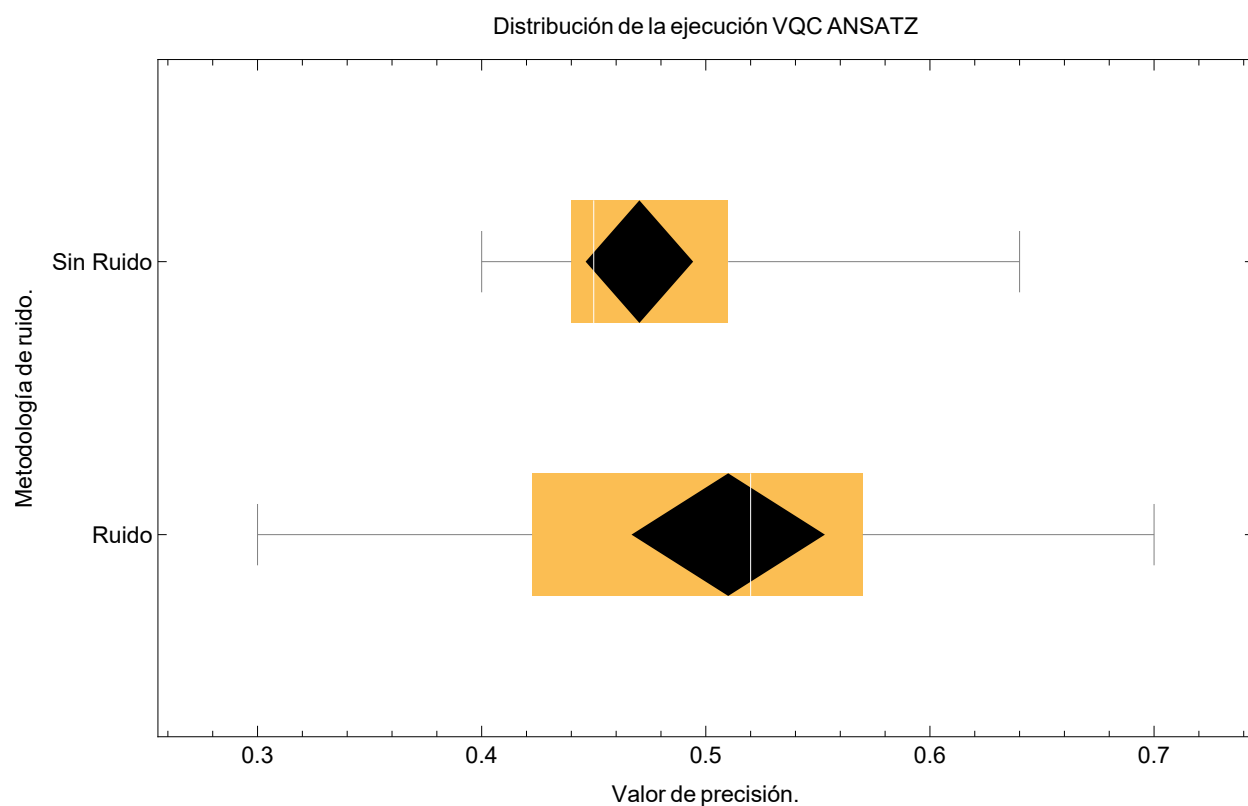
### Ruido en el ansatz.

Cuando se efectúa la acción de ruido de Pauli sobre el ansatz es posible observar (véase 4.9) que la distribución de los valores de precisión parecieran tener un mayor grado de

variabilidad que la ejecución sin ruido de Pauli. A continuación se muestran los resultados de las ejecuciones con sus estadísticas descriptivas (véase 4.15).

**Tabla 4.15.** Estadísticas descriptivas correspondientes a la ejecución del VQC en el conjunto de datos de credit. (véase 4.13)

	Valor mínimo.	Mediana.	Promedio.	Desviación Estándar.	25 %	75 %	V. Máximo.
Ruido de Pauli	0.3	0.52	0.51	0.10	0.42	0.57	0.70
Sin Ruido	0.4	0.45	0.47	0.06	0.44	0.51	0.64



**Figura 4.13.** Resultados de la distribución de valores para el algoritmo VQC en el conjunto de datos de crédito mediante la acción de ruido en el ansatz.

Con motivo de indagar si los resultados obtenidos con ruido de Pauli y en ausencia del mismo provenían de la misma distribución se llevó a cabo la ejecución de la prueba de hipótesis de Kolmogorov-Smirnov teniendo como resultados: Donde debido a que el valor de  $p$  es cercano a cero es poco probable que ambos datos provengan de la misma distribución (véase 4.16).

**Tabla 4.16.** Significancia estadística para la comparación de ruido de Pauli y ausencia de ruido.

	Estadístico	P-Value
Kolmogorov -Smirnov	0.37037	0.03289981

## 4.2. Enfoques híbridos

Para la realización de pruebas en los algoritmos híbridos, es decir, aquellos que usan únicamente de manera parcial el cómputo cuántico se consideró llevar a cabo un enfoque de análisis distinto, principalmente debido a la posibilidad de replicar los resultados mediante una semilla para la configuración de números aleatorios, por lo que en contraste a la implementación del VQC se realizó una ejecución en ausencia de los dos tipos de ruido y 27 ejecuciones con ruido, únicamente para ver la variabilidad de las ejecuciones con ruido respecto de la ausencia de ruido.

### 4.2.1. QNN

El uso de QNN permitió que se llevaran a cabo ejecuciones con ruido y sin ruido en ambas modificaciones tal cual se presentó en el VQC, sin embargo, es importante hacer énfasis en que el diseño de esta estructura es híbrido, es decir, pese a que el proceso de ajuste de parámetros es cuántico la inferencia de las clases es clásica mediante una implementación conectada a PyTorch.

## Resultados IRIS

**Tabla 4.17.** Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos de Iris.

	Sin Ruido	0.95				
	Valor mínimo	Valor Promedio	Desviación Estándar	Valor máximo	25 %	75 %
<b>Ruido Ansatz</b>	0.8	0.91	0.05	0.98	0.92	0.95
<b>Ruido FM</b>	0.96	0.92	0.03	0.98	0.9	0.95

## Resultados ad\_hoc

**Tabla 4.18.** Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos de ad\_hoc.

	Sin Ruido	0.98				
	Valor mínimo	Valor Promedio	Desviación Estándar	Valor máximo	25 %	75 %
Ruido Ansatz	0.93	0.95	0.1	1	0.95	0.97
Ruido FM	0.97	0.98	0.01	1	0.98	0.99

## Resultados Credit.

En el caso de la ejecución del algoritmo sobre el conjunto de datos se obtuvieron valores francamente anómalos en comparación con los presentados previamente en el VQC por este mismo algoritmo, obteniendo valores de clasificación por debajo del 50 % para ambas estrategias de ruido tanto en el ansatz como en el mapa de características, una de las posibles razones es que debido a la mayor complejidad de este conjunto de datos puede que el número de iteraciones efectuadas por el algoritmo híbrido no se haya llegado a un resultado óptimo. Además se observó que no hubo diferencias significativas en la forma de las distribuciones de los valores de precisión en la metodología de ruido.

**Tabla 4.19.** Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos de credit.

	Sin Ruido	0.65				
	Valor mínimo	Valor Promedio	Desviación Estándar	Valor máximo	25 %	75 %
Ruido Ansatz	0.37	0.54	0.06	0.65	0.5	0.6
Ruido FM	0.42	0.56	0.12	0.68	0.6	0.66

## Resultados datos sintéticos.

Los resultados del conjunto de datos sintéticos obtenidos fueron congruentes con los obtenidos para el algoritmo de VQC, teniendo variaciones dentro de lo normal, esto es importante, pues sugiere que los valores de precisión obtenidos en el conjunto de datos Credit fue una anomalía.

**Tabla 4.20.** Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos sintéticos.

	Sin Ruido		0.74			
	Valor mínimo	Valor Promedio	Desviación Estándar	Valor máximo	25 %	75 %
<b>Ruido Ansatz</b>	0.54	0.69	0.12	0.73	0.58	0.66
<b>Ruido FM</b>	0.57	0.63	0.14	0.76	0.6	0.64

### 4.2.2. QSVC

El uso de QSVC como se comentó con anterioridad hace uso de la estrategia de clasificación con máquinas de soporte vectorial clásicas, de manera que la única diferencia con respecto de las máquinas de soporte vectorial clásicas es que el mapa de características sobre el cuál se hace la clasificación se estima haciendo un ajuste de parámetros análogo al usado en el VQC y discutido previamente en (3.1). A continuación se presentan los resultados para la distribución de los valores de precisión de los distintos conjuntos de datos, dado que en el caso del QSVC solo se ocupa un mapa de características no es posible comparar el ruido más que con el valor de precisión obtenido sin ruido. Para esta sección de experimentación se decidió de igual manera fijar una semilla para la ejecución del algoritmo y llevar a cabo comparaciones de las 27 ejecuciones bajo la presencia de ruido.

### Resultados Iris

Los resultados correspondientes a la base de datos de Iris indican un comportamiento en el que no hay variaciones extremas a la hora de llevar a cabo una ejecución con ruido dentro del mapa de características, se observa apenas una diferencia del 5% en el valor promedio de porcentaje de clasificación correcta respecto de la ejecución sin ruido.

**Tabla 4.21.** Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QSVC para el conjunto de datos Iris.

	Sin Ruido		0.89			
	Valor mínimo	Valor Promedio	Desviación Estándar	Valor máximo	25 %	75 %
<b>Ruido FM</b>	0.78	0.83	0.08	0.87	0.8	0.85

### Resultados Ad\_hoc

Los resultados correspondientes a la base de datos de Iris indican un comportamiento en el que no hay variaciones extremas a la hora de llevar a cabo una ejecución con ruido dentro del mapa de características, se observa apenas una diferencia del 0.05 % en el valor promedio de porcentaje de clasificación correcta respecto de la ejecución sin ruido. En particular este conjunto de datos demuestra nuevamente ser extremadamente robusto a la presencia de ruido, esto puede ser debido a que el diseño del mapa de características es hecho a medida para este tipo de conjunto de datos.

**Tabla 4.22.** Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QSVC para el conjunto de datos ad\_hoc.

	Sin Ruido	0.95				
	Valor mínimo	Valor Promedio	Desviación Estándar	Valor máximo	25 %	75 %
Ruido FM	0.9	0.95	0.02	1	0.95	0.95

### Resultados Credit

Los resultados correspondientes a la base de datos de Credit indican un comportamiento en el que no hay variaciones extremas a la hora de llevar a cabo una ejecución con ruido dentro del mapa de características, se observa apenas una diferencia del 8 % en porcentaje de clasificación respecto de la ejecución sin ruido.

**Tabla 4.23.** Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos Credit.

	Sin Ruido	0.78				
	Valor mínimo	Valor Promedio	Desviación Estándar	Valor máximo	25 %	75 %
Ruido FM	0.65	0.7	0.09	0.75	0.7	0.72

### Resultados datos sintéticos.

Los resultados correspondientes a la base de datos de datos sintéticos indican un comportamiento en el que no hay variaciones extremas a la hora de llevar a cabo una ejecución con

ruido dentro del mapa de características, se observa apenas una diferencia del 1 % en el valor promedio de porcentaje de clasificación correcta respecto de la ejecución sin ruido. Pese a que la diferencia porcentual sea mínima es importante observar que existen variaciones claras respecto de la ejecución original, pudiendo en este caso empeorar significativamente (14 %) el porcentajen de precisión y en este caso la mejora podría ser únicamente de como máximo el 5 %.

**Tabla 4.24.** Datos estadísticos correspondientes a la ejecución del enfoque híbrido de QNN para el conjunto de datos sintéticos.

	Sin Ruido	0.69				
	Valor mínimo	Valor Promedio	Desviación Estándar	Valor máximo	25 %	75 %
Ruido FM	0.55	0.7	0.04	0.75	0.64	0.7





# Conclusiones

En el capítulo anterior, además de presentar los resultados, se realizó una discusión breve pero detallada sobre el comportamiento del algoritmo de clasificación cuántica variacional. En particular, se examinó en detalle el impacto del ruido en el conjunto de datos especialmente diseñado para esta tarea.

Se observó que cuando el ruido se aplicaba al mapa de características, se lograban mejoras significativas en la precisión de la clasificación. Esto es especialmente relevante, ya que el conjunto de datos analizado fue diseñado específicamente para ajustarse al mapa de características ZZ.

Por otro lado, al analizar la influencia del ruido en el *ansatz*, se encontró que no tenía un efecto significativo en el problema de clasificación. Esto se evidencia en el valor de  $p$  reportado por la prueba de Kolmogorov-Smirnov.

Aunque podría suponerse que este comportamiento se repetiría en otros conjuntos de datos,

se evidencia un patrón opuesto al analizar el conjunto de datos de Iris. En este caso, los resultados de las ejecuciones del algoritmo revelan que la influencia del ruido en el algoritmo es mayor que su efecto en el mapa de características.

En el análisis de los conjuntos de datos generados sintéticamente por Sklearn, no se encontró ningún impacto significativo debido a la presencia de ruido, tanto en la acción sobre el mapa de características como en la acción sobre la propuesta de solución.

Sin embargo, al examinar la clasificación de la base de datos de crédito, se obtuvo nuevamente la conclusión de que el ruido en el *ansatz* tiene un impacto mayor que en el caso del mapa de características.

Estas observaciones nos permiten hacer varias afirmaciones. Por ejemplo, en el caso del conjunto de datos *ad\_hoc* diseñado exclusivamente para ser completamente separable mediante el mapa de características elegido, se observa que el valor de precisión presenta mayores variaciones cuando se introduce ruido en este tipo de conjuntos de datos. Por otro lado, en conjuntos de datos que no son diseñados a medida, se obtienen mejores resultados al perturbar la propuesta de solución que al aplicar ruido al mapa de características.

Durante las ejecuciones con el algoritmo de QNN, se observó que las perturbaciones en el mapa de características y en la propuesta de solución no tuvieron un impacto significativo en el valor de precisión, en comparación con el caso en el que se aplicó ruido de Pauli. Sin embargo, este modelo obtuvo los peores resultados en términos de precisión, a pesar de que la función objetivo a minimizar se encontraba cerca de un mínimo. Esto puede deberse a varios factores, como la elección de los datos o el número de iteraciones en el proceso de optimización. Sin embargo, no se obtuvieron conclusiones definitivas debido a estos resultados.

En el caso del algoritmo de QSVC, se utilizó una metodología diferente, ya que solo es posible implementar ruido sobre el mapa de características. A pesar de esto, debido al funcionamiento del algoritmo sustentado en el kernel que verifica la fidelidad del estado, la acción del ruido no tuvo un impacto significativo. Para los cuatro conjuntos de datos, se encontró que la presencia de ruido en este algoritmo generalmente mejoraba el valor de precisión en aproximadamente un 5% en términos de clasificación.

Es posible considerar a estos resultados como contra intuitivos, pues la presencia de ruido en general está relacionada con errores, no obstante en este trabajo se observa que en el caso del QML esta observación intuitiva no coincide con los resultados obtenidos, pues se evidencia que existe una tendencia a mejorar el valor de precisión de los algoritmos de aprendizaje cuando existe presencia de ruido. Este resultado sugiere, pese a no ser completamente concluyente que el ruido tiene un impacto positivo sobre el valor de precisión.

### 4.3. Trabajo a futuro.

Una de las limitantes principales de este trabajo fue la dificultad de ingresar a una computadora cuántica real para poder llevar a cabo ejecuciones que fuesen estadísticamente semejantes, teniendo en cuenta eso, uno de las mejoras a futuro es llevar a cabo implementaciones con un tiempo reservado de cómputo cuántico en alguna de las plataformas de cómputo cuántico disponibles.

Además de lo discutido previamente la infinidad de posibles conjuntos para llevar a cabo clasificación hacen que este estudio empírico no sea concluyente, pues debido a que los circuitos variacionales son adaptables el mapa de características así como el ansatz tienen la posibilidad de mutar para comportarse de manera distinta en cada conjunto de datos.

El estudio de modelos de ruido distintos al propuesto también es interesante pues en este caso únicamente se llevó a cabo una serie de perturbaciones aleatorias en el mapa de

características y en el ansatz, modelos de ruido más completos que consideren interacción térmica pueden ser útiles para describir el comportamiento de estos algoritmos en un entorno real. De igual manera estudios comparativos en ausencia de mitigación de errores pueden ser llevados a cabo.

Es bien sabido que en la actualidad las redes neuronales y en general el área de aprendizaje máquina clásico suele trabajar con conjuntos de datos de grandes dimensiones, es decir, en tareas de clasificación multiclase con una colección grande de características, proponer estrategias para llevar a cabo esto con cómputo cuántico es un reto, pues los modelos presentados en este trabajo requieren de una cantidad de qubits proporcional al número de características presentes en el conjunto de datos, esto compone una limitante experimental pues pese a que haya procesos de escalado dimensional como PCA en muchas ocasiones el tratamiento de las características no genera un conjunto de datos lo suficientemente útil para poder llevar a cabo aprendizaje.

# Referencias

- [1] C. E. Shannon, “A Mathematical Theory of Communication”, The Bell System Technical Journal **27**, 623-656 (1948).
- [2] R. W. Hamming, *Error detecting and error correcting codes*, from Richard W. Hamming collection (NPS-018), Naval Postgraduate School. Papers and Speeches., 1982-08.
- [3] K. Temme, S. Bravyi y J. M. Gambetta, “Error Mitigation for Short-Depth Quantum Circuits”, *Phys. Rev. Lett.* **119**, 180509 (2017).
- [4] P. D. Nation, H. Kang, N. Sundaresan y J. M. Gambetta, “Scalable Mitigation of Measurement Errors on Quantum Computers”, *PRX Quantum* **2**, 040326 (2021).
- [5] T. Giurgica-Tiron, Y. Hindy, R. LaRose, A. Mari y W. J. Zeng, “Digital zero noise extrapolation for quantum error mitigation”, en *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)* (2020), páginas 306-316.
- [6] IBM, *Differences in error suppression, mitigation, and correction — IBM Research Blog*.
- [7] M. A. Nielsen e I. Chuang, *Quantum computation and quantum information*, 2002.
- [8] Y. Du, M.-H. Hsieh, T. Liu, D. Tao y N. Liu, “Quantum noise protects quantum classifiers against adversaries”, *Phys. Rev. Res.* **3**, 023153 (2021).

- 
- [9] M. Urbanek, B. Nachman, V. R. Pascuzzi, A. He, C. W. Bauer y W. A. de Jong, “Mitigating Depolarizing Noise on Quantum Computers with Noise-Estimation Circuits”, *Physical Review Letters* **127**, [10.1103/physrevlett.127.270502](https://doi.org/10.1103/physrevlett.127.270502) (2021).
- [10] S. Tobin, B. Jayabalasingham, S. Huggett y M. de Kleijn, “A brief historical overview of artificial intelligence research”, *Information Services and Use* **39**, 291-296 (2019).
- [11] G. Y. C. S. Brian McGuire Ting Huang, “The History of Artificial Intelligence”, (2006).
- [12] S. Marsland, “Machine Learning : An Algorithmic Perspective”, [10.1201/9781420067194](https://doi.org/10.1201/9781420067194) (2011).
- [13] N. C. Thompson, K. Greenewald, K. Lee y G. F. Manso, *The Computational Limits of Deep Learning*, 2020, arXiv: [2007.05558](https://arxiv.org/abs/2007.05558).
- [14] A. M. Turing, “On Computable Numbers, with an Application to the Entscheidungsproblem”, *Proceedings of the London Mathematical Society* **s2-42**, 230-265 (1937).
- [15] A. Church, “A note on the Entscheidungsproblem”, *Journal of Symbolic Logic* **1**, 40-41 (1936).
- [16] M. Sipser, “Introduction to the Theory of Computation”, (2012).
- [17] J. Preskill, “Quantum Computing and the entanglement frontier”, (2012), arXiv: [1203.5813](https://arxiv.org/abs/1203.5813).
- [18] T. Cover y J. Thomas, *Elements of Information Theory* (Wiley, 2012).
- [19] D. MacKay, D. Kay y C. U. Press, *Information Theory, Inference and Learning Algorithms* (Cambridge University Press, 2003).
- [20] G. Boole, *An Investigation of the Laws of Thought: On Which Are Founded the Mathematical Theories of Logic and Probabilities*, Cambridge Library Collection - Mathematics (Cambridge University Press, 2009).
- [21] C. E. Shannon, “A symbolic Analysis of Relay and Switching Circuits.”, *Transactions American Institute of Electrical Engineers* **57**, 38-80 (1948).
- [22] S. Širca, *Probability for Physicists*, Graduate Texts in Physics (Springer International Publishing, 2016).

- 
- [23] R. Penrose, *The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics*, Oxford landmark science (Oxford University Press, 2016).
- [24] R. Sedgewick y P. Flajolet, *Introduction to the Analysis of Algorithms*.
- [25] D. Deutsch, "Quantum theory, the Church–Turing principle and the universal quantum computer", [Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences](#) **400**, 97-117 (1985).
- [26] R. P. Feynman, "Simulating physics with computers", [International Journal of Theoretical Physics](#) **21**, 467-488 (1982).
- [27] C. Cohen-Tannoudji, B. Diu y F. Laloe, "Quantum Mechanics, Volume 1", *Quantum Mechanics* **1**, 898 (1986).
- [28] G. Grynberg, A. Aspect, C. Fabre y C. Cohen-Tannoudji, *Introduction to Quantum Optics: From the Semi-classical Approach to Quantized Light* (Cambridge University Press, 2010).
- [29] T. G. Wong, "Introduction to classical and quantum computing", Rooted Grove (2022).
- [30] M. A. Nielsen e I. L. Chuang, *Quantum computation and quantum information*, en, 10th anniversary ed (Cambridge University Press, Cambridge ; New York, 2010).
- [31] M. Schuld y F. Petruccione, *Machine Learning with Quantum Computers*, en, Quantum Science and Technology (Springer International Publishing, Cham, 2021).
- [32] E. Dennis, A. Kitaev, A. Landahl y J. Preskill, "Topological quantum memory", [Journal of Mathematical Physics](#) **43**, arXiv:quant-ph/0110143, 4452-4505 (2002).
- [33] S. Altares-López, A. Ribeiro y J. J. García-Ripoll, "Automatic design of quantum feature maps", en, [Quantum Science and Technology](#) **6**, 045015 (2021).
- [34] S. Bravyi, O. Dial, J. M. Gambetta, D. Gil y Z. Nazario, "The Future of Quantum Computing with Superconducting Qubits", en, [Journal of Applied Physics](#) **132**, arXiv:2209.06841 [quant-ph], 160902 (2022).
- [35] K. Wan, W. J. Huggins, J. Lee y R. Babbush, *Matchgate Shadows for Fermionic Quantum Simulation*, en, arXiv:2207.13723 [physics, physics:quant-ph], jul. de 2022.

- [36] D. W. Berry, Y. Su, C. Gyurik, R. King, J. Basso, A. D. T. Barba, A. Rajput, N. Wiebe, V. Dunjko y R. Babbush, *Quantifying Quantum Advantage in Topological Data Analysis*, en, arXiv:2209.13581 [quant-ph], ene. de 2023.
- [37] Y. Kwak, W. J. Yun, S. Jung y J. Kim, “Quantum Neural Networks: Concepts, Applications, and Challenges”, en, en [2021 Twelfth International Conference on Ubiquitous and Future Networks \(ICUFN\)](#) (ago. de 2021), páginas 413-416.
- [38] M. Schuld, I. Sinayskiy y F. Petruccione, “The quest for a Quantum Neural Network”, [Quantum Information Processing](#), [10.1007/S11128-014-0809-8](#) (2014).
- [39] K. Beer, D. Bondarenko, T. Farrelly, T. J. Osborne, R. Salzmann, D. Scheiermann y R. Wolf, “Training deep quantum neural networks”, en, [Nature Communications](#) **11**, [808](#) (2020).
- [40] R. J. Lipton y K. W. Regan, *Quantum algorithms via linear algebra: a primer*, en (The MIT Press, Cambridge, Massachusetts, 2014).
- [41] S. Shalev-Shwartz, Y. Singer y A. Cotter, “Pegasos: Primal Estimated sub-GrAdient Solver for SVM”, en,
- [42] D. Deutsch y R. Jozsa, “Rapid Solution of Problems by Quantum Computation”, [Proceedings of the Royal Society of London Series A](#) **439**, [553-558](#) (1992).
- [43] K. Mitarai, M. Negoro, M. Kitagawa y K. Fujii, “Quantum circuit learning”, [Physical Review A](#), [10.1103/PHYSREVA.98.032309](#) (2018).
- [44] S. Y.-C. Chen, C.-M. Huang, C.-W. Hsing, H.-S. Goan e Y.-J. Kao, “Variational quantum reinforcement learning via evolutionary optimization”, en, [Machine Learning: Science and Technology](#) **3**, Publisher: IOP Publishing, [015025](#) (2022).
- [45] L.-A. Wu, C. X. Yu y D. Segal, “Exact dynamics of interacting qubits in a thermal environment: Results beyond the weak coupling limit”, [New Journal of Physics](#) **15**, [arXiv:1207.6995 \[quant-ph\]](#), [023044](#) (2013).



- 
- [46] C. H. Yang, R. C. C. Leon, J. C. C. Hwang, A. Saraiva, T. Tanttu, W. Huang, J. Camirand Lemyre, K. W. Chan, K. Y. Tan, F. E. Hudson, K. M. Itoh, A. Morello, M. Pioro-Ladrière, A. Laucht y A. S. Dzurak, “Operation of a silicon quantum processor unit cell above one kelvin”, en, [Nature](#) **580**, Number: 7803 Publisher: Nature Publishing Group, 350-354 (2020).
- [47] L. Petit, H. G. J. Eenink, M. Russ, W. I. L. Lawrie, N. W. Hendrickx, S. G. J. Philips, J. S. Clarke, L. M. K. Vandersypen y M. Veldhorst, “Universal quantum logic in hot silicon qubits”, en, [Nature](#) **580**, Number: 7803 Publisher: Nature Publishing Group, 355-359 (2020).
- [48] M. L. Head, L. Holman, R. Lanfear, A. T. Kahn y M. D. Jennions, “The Extent and Consequences of P-Hacking in Science”, en, [PLOS Biology](#) **13**, Publisher: Public Library of Science, e1002106 (2015).

