

UCUENCA

Universidad de Cuenca

Facultad de Ingeniería

Carrera de Ingeniería en Telecomunicaciones


Clasificación automática de señales para comunicaciones inalámbricas basada en técnicas de inteligencia artificial

Trabajo de titulación previo a la obtención del título de Ingeniero en Telecomunicaciones


Autores:

Carlos Andrés Pardo Castro
Juan Pablo Pérez Vargas

Director:

Lizandro Damián Solano Quinde
ORCID:  0000-0001-7427-4889

Tutor:

Kenneth Samuel Palacio Baus
ORCID:  0000-0002-7318-8062

Cuenca, Ecuador

2025-09-24

Resumen

Este trabajo aborda la identificación de esquemas de modulación en señales digitales bajo condiciones de canal realistas —desvanecimiento por trayectos múltiples, ruido aditivo blanco gaussiano (AWGN) y desfase de portadora— considerando seis esquemas: BPSK, QPSK, 16-QAM, 32-QAM cruzada, 64-QAM y 128-QAM cruzada; además, se incluye un análisis complementario con 256-QAM para evaluar el desempeño del sistema ante modulaciones de mayor densidad. Para representar la variabilidad del entorno, se generaron escenarios con distintos niveles de relación señal-ruido (SNR), número de trayectos y valores de desfase utilizando el toolkit TorchSig; las señales en banda base se procesaron en formato en fase y cuadratura (I-Q) y se usaron como entrada de tres arquitecturas de redes neuronales entrenadas y evaluadas en tareas supervisadas. Finalmente, se propone un esquema de ensamble por votación que combina los k mejores modelos de cada red, logrando incrementos consistentes en la probabilidad de clasificación correcta (PCC) en todos los escenarios evaluados.

Palabras clave del autor: modulación digital, desvanecimiento multitrayecto, ruido aditivo blanco gaussiano, desfase de portadora, representación I-Q



El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Cuenca ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por la propiedad intelectual y los derechos de autor.

Repositorio Institucional: <https://dspace.ucuenca.edu.ec/>

Abstract

This work addresses the identification of digital modulation schemes under realistic channel conditions—multipath fading, additive white Gaussian noise (AWGN), and carrier phase offset—across six formats: BPSK, QPSK, 16-QAM, cross-32-QAM, 64-QAM, and cross-128-QAM; a complementary analysis with 256-QAM is also included to assess performance under higher constellation orders. To capture channel variability, scenarios with different signal-to-noise ratio (SNR) levels, numbers of propagation paths, and phase-offset values were generated using the TorchSig toolkit; the resulting baseband in-phase/quadrature (I–Q) signals fed three neural-network architectures trained and evaluated in supervised classification tasks. Finally, a voting-based ensemble that combines the top k models from each network is proposed, yielding consistent gains in correct classification probability (PCC) across all evaluated scenarios.

Author keywords: digital modulation, multipath fading, additive white Gaussian noise, carrier phase offset, I–Q representation



The content of this work corresponds to the right of expression of the authors and does not compromise the institutional thinking of the University of Cuenca, nor does it release its responsibility before third parties. The authors assume responsibility for the intellectual property and copyrights.

Institutional Repository: <https://dspace.ucuenca.edu.ec/>

Índice de contenido

1. Introducción	15
1.1. Problemática	15
1.2. Propuesta y justificación	16
1.3. Objetivos	16
1.3.1. Objetivo general:	16
1.3.2. Objetivos específicos:	17
1.4. Alcance del Trabajo	17
1.5. Estructura del documento	18
2. Marco teórico	19
2.1. Fundamentos de las comunicaciones digitales	19
2.1.1. Ventajas y retos de lo digital con respecto a lo analógico	19
2.1.2. Modelo genérico de transmisión y codificación	20
2.1.3. Adaptación al canal: Modulación y codificación adaptativa	20
2.2. Esquemas de modulación digital	21
2.3. Representación de señales I-Q	21
2.3.1. Transformación al dominio polar	22
2.3.2. Diagrama de constelación	22
2.3.3. Impacto de la densidad de símbolos	24
2.4. Modelos de canal e interferencia	25
2.4.1. Ruido blanco aditivo Gaussiano	25
2.4.2. Desvanecimiento por trayectoria múltiple	25
2.4.3. Interferencia co-canal y adyacente	26
2.5. Formulación del problema de clasificación automática de modulación	26
2.5.1. Estrategias de solución	27
2.6. Métodos clásicos de clasificación	27
2.6.1. Ventajas e inconvenientes	28
2.7. Fundamentos de aprendizaje automático y profundo	28
2.7.1. Arquitecturas básicas y perceptrón multicapa (MLP)	29
2.7.2. Redes neuronales convolucionales (CNN)	29
2.7.3. Redes residuales (ResNet)	30
2.7.4. Redes neuronales recurrentes (RNN) y Long Short-Term Memory (LSTM)	31
2.7.5. Redes neuronales profundas (DNN)	33
2.7.6. Mecanismos de atención y Transformer	33
2.7.7. Bloques Squeeze-and-Excitation (SE)	34
2.7.8. Hiperparámetros y técnicas de regularización	35

2.8. Resumen del capítulo.....	36
3. Estado del Arte	37
3.1. Clasificación automática de señales: métodos tradicionales	37
3.1.1. Pruebas de verosimilitud	37
3.1.1.1. Average Likelihood Ratio Test (ALRT).....	37
3.1.1.2. Generalized Likelihood Ratio Test (GLRT).....	38
3.1.1.3. Hybrid Likelihood Ratio Test (HLRT)	38
3.1.2. Métodos basados en momentos y cumulantes.....	38
3.1.3. Pruebas estadísticas no paramétricas	38
3.1.4. Visión condensada de métodos tradicionales	39
3.1.5. Comparación de métodos tradicionales.....	39
3.2. Métodos de inteligencia artificial para AMC.....	39
3.2.1. Métodos estadísticos avanzados	39
3.2.2. Redes neuronales en dominio polar	41
3.2.3. Modelos híbridos y atención.....	42
3.2.3.1. Metodología Jerárquico+algoritmo genético (GA).....	42
3.2.4. Robustez y seguridad: clasificación en presencia de ACI	43
3.2.5. Vulnerabilidades adversariales	43
3.2.5.1. Metodología	44
3.2.6. Visión condensada de métodos de inteligencia artificial	45
3.2.7. Comparación de métodos de inteligencia Artificial.....	45
3.2.8. Brechas y oportunidades en el contexto de la AMC	46
4. Metodología	48
4.1. Contexto y alcance.....	48
4.2. Recursos y herramientas	49
4.2.1. Entorno de desarrollo local.....	49
4.2.2. Entorno de cómputo en la nube.....	49
4.2.3. Automatización y Gestión en la Nube.....	49
4.3. Generación y preparación del conjunto de datos	50
4.3.1. Definición de esquemas de modulación	50
4.3.2. Parámetros de generación	50
4.3.3. Escenarios de canal	53
4.3.4. Conjuntos de Datos	55
4.3.5. Estructura de Almacenamiento y Etiquetado	56
4.4. Validación Temprana de Arquitecturas	57
4.4.1. Criterio de aceptación	57
4.5. Diseño de Arquitecturas Neuronales.....	58
4.5.1. Red CNN-BiLSTM-Attention	58
4.5.2. Red Inception-Res-SE-Attn.....	58
4.5.3. Red Res-TCN-SE	60
4.6. Entrenamiento y Validación Exhaustiva	63

	6
4.7. Selección del Modelo Final	64
5. Resultados y Discusión	66
5.1. Validación temprana de arquitecturas.....	66
5.1.1. Desempeño global frente a parámetros de canal	66
5.1.2. Análisis detallado por clase	70
5.1.2.1. Mapa de calor F1 por escenario y clase	70
5.1.2.2. Contribución al error global	71
5.2. Entrenamiento y validación exhaustiva.....	72
5.2.1. Desempeño global frente a parámetros de canal	72
5.2.2. Curvas porcentaje de clasificación correcta (PCC) por parámetro	72
5.2.3. Mapa de calor F1 por escenario y clase	76
5.2.4. Contribución al error global	77
5.2.5. Análisis de matrices de confusión	77
5.3. Experimento con siete modulaciones (inclusión de 256-QAM).....	81
5.3.1. PCC medio con IC 95% en el escenario base.....	81
5.3.2. Mapa de calor F1 por clase	82
5.3.2.1. Análisis.....	82
5.3.3. Contribución al error	83
5.3.3.1. Análisis	83
5.4. Sistema de votación por comité	84
5.4.1. Selección óptima del parámetro k	84
5.4.2. Comparación por parámetro univariable.....	85
5.5. Síntesis general.....	88
5.6. Resumen del capítulo.....	88
6. Conclusiones, recomendaciones y trabajos futuros	89
6.1. Conclusiones	89
6.2. Recomendaciones	90
6.3. Trabajos futuros	90
Referencias	92

Índice de figuras

2.1. Bloques esenciales de un sistema digital	20
2.2. Constelaciones de BPSK, QPSK, 16-QAM, 32-QAM, 64-QAM, 128-QAM y 256-QAM.....	24
2.3. Diagrama de un perceptrón con 5 entradas.....	29
2.4. Estructura de bloque residual.....	31
2.5. Diagrama de la arquitectura de una célula LSTM.....	32
4.1. Arquitectura de recepción con clasificador automático de señales	48
4.2. Metodología concisa para la generación, canalización y almacenamiento de señales moduladas	51
4.3. Impacto de la longitud de señal en la ocurrencia de símbolos para modulaciones de alto orden.	53
4.4. Constelación de BPSK, QPSK y 16-QAM (filas) en cuatro condiciones de canal (columnas): (1) señal ideal; (2) con ruido (SNR = 10dB); (3) con ruido y desvanecimiento (2 trayectos); (4) con ruido, desvanecimiento y desfase (45°)	54
4.5. Magnitud promedio de 6 taps en un canal multipropagación con perfil de potencia [1, 0.5, 0.25, 0.125, 0.06, 0.03], calculada con TorchSig sobre 4096 señales I-Q.	55
4.6. Esquema general de la arquitectura CNN-BiLSTM-Attention.	59
4.7. Esquema de la arquitectura del bloque Attention Pooling.....	59
4.8. Esquema general de la arquitectura Inception-Res-SE-Attn.	61
4.9. Esquema de la arquitectura del bloque Inception + ResNet + SE.....	62
4.10. Esquema general de la arquitectura Res-TCN-SE.....	63
4.11. Esquema de la arquitectura del bloque Transformer Encoder.....	64
4.12. Esquema del sistema de votación con múltiples modelos por arquitectura.	65
5.1. PCC medio con IC 95% vs. SNR (escenarios 1–5).	67
5.2. PCC medio con IC 95% vs. número de trayectos.	68
5.3. PCC medio con IC 95% vs. desfase.....	69
5.4. Mapa de calor de F1 por clase y escenario univariable	70
5.5. Distribución de la contribución al error (1–Recall) normalizada por clase. Todos los escenarios son considerados.....	71
5.6. PCC medio con IC 95% vs. SNR. Entrenamiento exhaustivo	73
5.7. PCC medio con IC 95% vs. número de trayectos. Entrenamiento exhaustivo	74
5.8. PCC medio con IC 95% vs. desfase. Entrenamiento exhaustivo.....	75
5.9. Mapa de calor de F1 por clase y escenario tras entrenamiento exhaustivo	76
5.10. Contribución al error después del reentrenamiento exhaustivo.....	77
5.11. Matriz de confusión promedio para escenarios con variación de SNR.	78

5.12. Matriz de confusión promedio para escenarios con variación en número de trayectos	79
5.13. Matriz de confusión promedio para escenarios con variación en desfase	80
5.14. Comparación de PCC con IC 95% (escenario 3) con seis vs. siete modulaciones.	81
5.15. Mapa de calor F1 por clase con siete modulaciones (escenario 3).....	82
5.16. Contribución al error con siete modulaciones	83
5.17. PCC promedio global vs. tamaño del subconjunto k	84
5.18. PCC vs. SNR: arquitecturas individuales vs. comité ($k = 3$).	85
5.19. PCC vs. número de trayectos: arquitecturas individuales vs. comité ($k = 3$).	86
5.20. PCC vs. desfase: arquitecturas individuales vs. comité ($k = 3$).	87

Índice de tablas

2.1. Esquemas de modulación digital considerados	21
3.1. Visión general de los métodos clásicos de AMC	39
3.2. Comparación de métodos tradicionales de AMC.....	39
3.3. Parámetros y resultados para el método Statistical Signal Characterization (SSC). 40	
3.4. Parámetros y resultados para el método basado en el Test de Kolmogorov-Smirnov	40
3.5. Parámetros y resultados clave para el método Polar+CNN.	41
3.6. Parámetros y resultados clave para el método CNN con CCN.	42
3.7. Parámetros y resultados clave para el clasificador jerárquico con GA.....	42
3.8. Parámetros y resultados clave para la clasificación bajo ACI con CNN.....	43
3.9. Parámetros y resultados clave para los ataques adversariales en clasificación de modulación	44
3.10. Visión general de las familias de métodos de IA para clasificación automática de modulación (AMC).	45
3.11. Comparación de métodos de IA para clasificación automática de modulación (AMC)	46
4.1. Símbolos únicos esperados y cobertura de constelación según tamaño y longitud de la señal.	52
4.2. Parámetros de configuración de señales moduladas	53
4.3. Escenarios univariados: se varía un solo parámetro mientras los otros se mantienen fijos.....	54
5.1. Resultados de validación temprana: exactitud (%) y pérdida (loss) media \pm desviación estándar, 20 corridas por condición univariable.....	66
5.2. Resultados de entrenamiento y validación exhaustiva: exactitud (%) y pérdida (loss) media \pm desviación estándar, 10 corridas por condición univariable	72

Dedicatoria

Dedico este trabajo a mi madre, Vilma, y a Emilio, por ser mis pilares incondicionales en cada etapa de este camino. Gracias por el esfuerzo, el amor y la confianza que depositaron en mí incluso en los momentos más difíciles. Su apoyo ha sido parte fundamental en mi crecimiento y siempre estaré agradecido por esto y mucho más.

También deseo dedicarlo a mis hermanos, por ser fuente constante de inspiración, compañía y motivación que me han llevado a donde estoy. Este logro también les pertenece.

Carlos Andrés Pardo Castro

Dedicatoria

Este trabajo refleja el esfuerzo, la dedicación y los valores que mis padres han sembrado en mí a través de su educación.

Lo dedico con profundo cariño a Deysi, mi mamá; a Pablo, mi papá; a Mafer, mi hermana; a Juan José, mi ñaño, y especialmente a Divina, mi compañera de largas noches, quien estuvo siempre conmigo durante mi vida universitaria sin importar la hora.

Juan Pablo Pérez Vargas

Agradecimientos

Queremos expresar nuestro más sincero agradecimiento a los docentes Lizandro Solano, PhD y Kenneth Palacio, PhD, por su valiosa guía y constante apoyo durante todo el desarrollo de este trabajo.

También extendemos nuestro reconocimiento a la universidad y a todas las personas que, de alguna manera, contribuyeron a nuestra formación académica y al logro de este trabajo de titulación.

Finalmente, agradecemos profundamente a nuestras familias y amigos por su apoyo incondicional y motivación durante todo este proceso.

Carlos Andrés Pardo Castro

Juan Pablo Pérez Vargas

Abreviaciones y acrónimos

- ACI** interferencia de canal adyacente. 26, 43
- ALRT** prueba de razón de verosimilitud promedio. 27, 37–39
- AMC** clasificación automática de modulación. 9, 15–18, 22, 26, 36, 46, 48, 90
- ANN** red neuronal artificial. 40
- AWGN** ruido blanco gaussiano aditivo. 2, 3, 17, 20, 25, 26, 36, 48, 89
- BPSK** *binary phase shift keying*. 19, 21, 24, 57, 89
- CCI** interferencia co-canal. 26
- CCN** Red de compensación de canal. 41, 42
- CDF** función de distribución acumulada. 40
- CNN** red neuronal convolucional. 15, 16, 27, 29, 30, 33, 41–43
- DNN** red neuronal profunda. 16, 33, 43
- ECDF** función de distribución acumulada empírica. 40
- FDP** función de densidad de probabilidad. 25, 37
- FGM** Fast Gradient Method. 44
- FPGA** matriz de compuertas programables en campo. 19
- GA** algoritmo genético. 5, 9, 42, 43
- GAP1D** Global Average Pooling One-Dimensional. 34
- GLRT** prueba de razón de verosimilitud generalizada. 27, 38, 39
- GPU** unidad de procesamiento gráfico. 30
- HLRT** prueba de razón de verosimilitud híbrida. 28, 38, 39
- HOC** cumulantes de orden superior. 38, 39
- I-Q** componentes en fase y en cuadratura. 7, 21, 22, 48–51, 54, 55, 58, 60
- IA** inteligencia artificial. 15, 16, 28, 46

- IoT internet de las cosas. 15
- KNN K vecinos más cercanos. 28
- KS Kolmogorov–Smirnov. 27, 38, 40, 45
- LDPC códigos de paridad de baja densidad. 19
- LoS línea de vista. 25
- LSTM memoria a largo y corto plazo. 27, 31, 34
- MLP perceptrón multicapa. 29, 42
- MSE error cuadrático medio. 35
- OTA transmisión por aire. 15
- PCA análisis de componentes principales. 44
- PCC porcentaje de clasificación correcta. 6–8, 38–40, 55, 57, 65–69, 72–75, 81, 84–87, 89
- PNR relación de perturbación a ruido. 44
- pp punto porcentual. 70, 75, 81, 87–89
- PSK *phase shift keying*. 20
- PSR relación de perturbación a señal. 44
- QAM *quadrature amplitude modulation*. 19–22, 24, 57, 89
- QPSK *quadrature phase shift keying*. 19, 21, 24, 57, 89
- RNN redes neuronales recurrentes. 31, 33
- SDR radio definida por software. 15, 91
- SE módulo squeeze-and-excitation. 15, 16, 34
- SGD descenso del gradiente estocástico. 35
- SIR relación señal-interferencia. 26
- SNR relación señal-ruido. 2, 3, 17, 20, 85, 89
- SSC Statistical Signal Characterization. 9, 39, 40, 45, 46
- SVM máquina de soporte de vectores. 27, 28
- VT-CNN2 visual transformer CNN versión 2. 44

1. Introducción

Las tecnologías inalámbricas han experimentado un crecimiento exponencial por ejemplo, en 2024, el tráfico mundial de datos móviles superó los 100 EB/mes, con un aumento anual cercano al 40 % [1]. En escenarios críticos —como comunicaciones de emergencia, internet de las cosas (IoT) industrial o redes 5G de baja latencia— la capacidad de identificar automáticamente el tipo de modulación resulta clave para optimizar el uso del espectro y garantizar la fiabilidad del enlace.

La clasificación automática de modulación (AMC) en los sistemas de comunicación, corresponde a un procedimiento que consiste en identificar y categorizar el tipo de modulación de una señal percibida en el receptor, constituyéndose como un problema fundamental en los sistemas de comunicaciones inalámbricas actuales.

La AMC tiene aplicaciones en modulación y codificación adaptativa, redes cognitivas, monitoreo de rendimiento, entre otras. En clasificación automática de señales, el receptor necesita identificar rápidamente la modulación para decodificar correctamente los datos, esto es esencial en redes cognitivas que detectan y aprovechan bandas espectrales libres sin causar interferencias.

A pesar de décadas de investigación en métodos clásicos, como los basados en funciones de verosimilitud o estadísticas de orden superior, todavía existen limitaciones importantes. En particular, estos enfoques suelen presentar alta complejidad computacional y pérdida de robustez en condiciones adversas, como bajo SNR, interferencia co-canal o modulaciones de orden elevado.

Los enfoques basados en inteligencia artificial (IA) han mostrado un gran potencial para resolver estos desafíos, especialmente mediante arquitecturas de aprendizaje profundo como las red neuronal convolucional (CNN). Sin embargo, aún persisten vacíos en cuanto a implementaciones prácticas en hardware de tipo radio definida por software (SDR) y transmisión por aire (OTA) y en la comparación sistemática entre arquitecturas híbridas —por ejemplo, aquellas que integran CNN, módulo squeeze-and-excitation (SE) y mecanismos de atención— frente a modelos clásicos o puramente convolucionales. Esta investigación busca contribuir en esa dirección, explorando la eficacia y robustez de arquitecturas en escenarios complejos y realistas.

1.1. Problemática

Actualmente, la clasificación automática de señales moduladas enfrenta desafíos derivados de la rápida evolución de las comunicaciones inalámbricas hacia señales más complejas y

variables. Los métodos tradicionales, basados en configuraciones estáticas y parámetros predefinidos, muestran limitaciones en entornos dinámicos (5G, IoT, radio cognitiva) y con bajo SNR [2]. La implementación de técnicas de IA mejora la adaptabilidad, sin embargo, aún faltan estudios para asegurar su desempeño en canales reales y condiciones adversariales.

1.2. Propuesta y justificación

Este proyecto propone desarrollar un sistema de AMC basado en red neuronal profunda (DNN), combinando bloques CNN, SE y mecanismos de atención.

La revisión del estado del arte revela dos deficiencias principales:

1. **Falta de estandarización en la generación de datos.** Muchos trabajos omiten detalles críticos (formatos, semillas, parámetros, frecuencias de muestreo) que impiden la replicación rigurosa de los conjuntos de muestras.
2. **Escenarios de canal poco realistas o incompletos.** Se tiende a descartar efectos clave como desvanecimiento por múltiples trayectos, desfase de portadora y modulaciones de orden elevado (128-QAM, 256-QAM), limitando la validez práctica de los resultados.

Nuestra propuesta incorpora:

- **Escenarios realistas y parametrizables:** Definición de curvas de SNR, número de trayectos y rangos de desfase que cubran condiciones validadas en la práctica.
- **Procedimiento de generación de datos replicable:** Documentación detallada de cada paso (configuración de herramientas, semilla de generadores, formatos de salida), garantizando trazabilidad y reutilización.
- **Análisis de modulaciones de alto orden:** Inclusión explícita de esquemas 128-QAM y 256-QAM para evaluar su impacto en el desempeño del clasificador y contrastar robustez frente a modulaciones de menor densidad.

1.3. Objetivos

1.3.1. Objetivo general:

Desarrollar y evaluar un sistema de clasificación automática de señales para comunicaciones inalámbricas, utilizando técnicas de inteligencia artificial.

1.3.2. Objetivos específicos:

1. Estudiar el estado actual y retos de la clasificación de señales en sistemas de comunicación mediante una revisión del estado del arte.
2. Establecer un escenario de evaluación apropiado para la clasificación de señales de un sistema de comunicaciones considerando diferentes esquemas de modulación digital y condiciones de canal.
3. Desarrollar e implementar un clasificador de señales para comunicaciones inalámbricas basado en técnicas de inteligencia artificial, considerando el escenario de evaluación establecido.
4. Evaluar el desempeño del clasificador desarrollado utilizando métricas de rendimiento estándar.

1.4. Alcance del Trabajo

Este trabajo aborda el problema de la AMC aplicada a señales digitales sintéticas generadas en entornos controlados utilizando la herramienta TorchSig.

El enfoque se centra en evaluar el desempeño de diferentes arquitecturas de redes neuronales ante variaciones de tres condiciones del canal:

- Presencia de ruido blanco gaussiano aditivo (AWGN).
- Desvanecimiento causado por múltiples trayectos.
- Desfase de portadora en un rango de 0 a 180 grados.

El conjunto de modulaciones digitales evaluado incluye:

- Esquemas principales: BPSK, QPSK, 16-QAM, 32-QAM cruzada, 64-QAM y 128-QAM cruzada.
- Análisis adicional: 256-QAM, considerado únicamente en el escenario base.

Las señales utilizadas en los experimentos se simulan bajo las siguientes condiciones:

- Representación en banda base mediante componentes I-Q.
- Valores de relación señal-ruido (SNR) entre 0 y 20 dB.
- Longitud fija de 512 símbolos por señal, con sobremuestreo de 8 muestras por símbolo.

Se evalúan tres arquitecturas de red: CNN-BiLSTM-Attention, Inception-Res-SE-Attention y Res-TCN-SE. Además, se propone un sistema de votación por comité, que combina varios modelos para mejorar la capacidad de clasificación frente a las condiciones del canal ya mencionadas.

1.5. Estructura del documento

Este documento se organiza de la siguiente forma:

- **Capítulo 1. Introducción:** motivación, gap, propuesta y objetivos.
- **Capítulo 2. Marco Teórico:** fundamentos de modulaciones digitales, representación I/Q, canales y bases de IA.
- **Capítulo 3. Estado del Arte:** métodos clásicos y de IA para AMC y brechas detectadas.
- **Capítulo 4. Metodología:** datos, preprocesamiento y arquitecturas propuestas.
- **Capítulo 5. Resultados y Discusión:** evaluación cuantitativa y comparación.
- **Capítulo 6. Conclusiones, recomendaciones y trabajos futuros:** aportes, limitaciones y líneas venideras.

2. Marco teórico

2.1. Fundamentos de las comunicaciones digitales

Las técnicas de comunicaciones digitales constituyen un pilar fundamental en los sistemas modernos de telecomunicaciones ya que permiten codificar la información en símbolos binarios, modular estos sobre señales portadoras de alta frecuencia y, posteriormente, procesarlos en dominios discretos [3, 4]. Este paradigma ofrece ventajas sustanciales frente a las señales analógicas, pero también plantea desafíos tanto en diseño como en operación.

2.1.1. Ventajas y retos de lo digital con respecto a lo analógico

Ventajas

- **Resistencia al ruido:** la regeneración bit a bit y los códigos de corrección de errores, como los convolucionales y códigos de paridad de baja densidad (LDPC), permiten mantener una baja tasa de error incluso en canales adversos.
- **Eficiencia espectral y flexibilidad:** la capacidad de cambiar dinámicamente entre esquemas de modulación, como *binary phase shift keying* (BPSK), *quadrature phase shift keying* (QPSK), *quadrature amplitude modulation* (QAM), permite maximizar la eficiencia espectral en función de la calidad del enlace.
- **Procesamiento digital:** los Procesadores Digitales de Señales y los dispositivos matriz de compuertas programables en campo (FPGA) pueden usarse para implementar filtros, ecualizadores y algoritmos de demodulación con alta precisión y adaptabilidad.

Retos

- **Complejidad de sincronización:** la recuperación precisa de portadora y reloj requiere algoritmos robustos para evitar deslizamientos (*desalineaciones en el sincronismo de portadora o reloj*) o pérdidas de símbolos.
- **Latencia y procesamiento:** el procesamiento de decodificación y corrección introduce retardo, crítico en aplicaciones de baja latencia.
- **Consumo energético:** los algoritmos digitales avanzados demandan potencia de cómputo, impactando el diseño de dispositivos móviles y embebidos.

2.1.2. Modelo genérico de transmisión y codificación

La Fig. 2.1 adaptada a partir del esquema propuesto por [5], muestra los componentes esenciales de un sistema de comunicación digital.

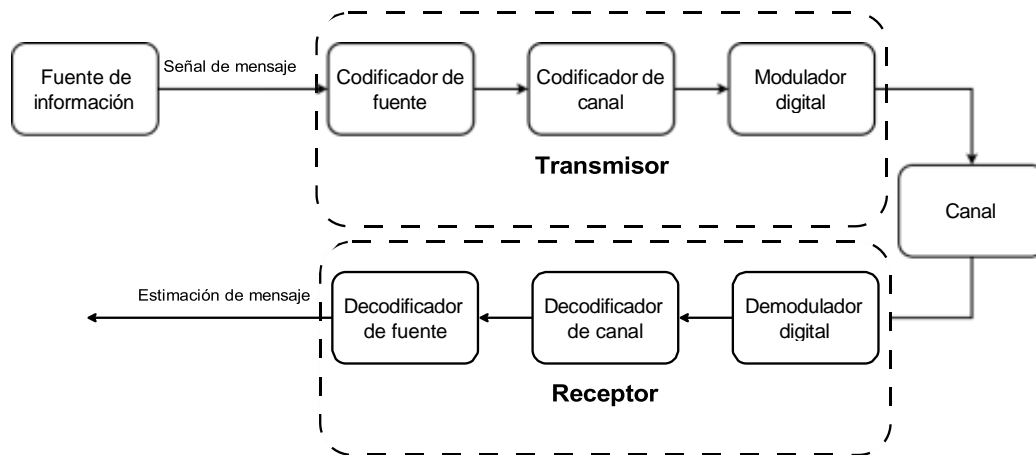


Figura 2.1: Bloques esenciales de un sistema de comunicación digital.

1. **Codificador de fuente:** convierte datos de alto nivel en un flujo binario optimizado para compresión y reconstrucción.
2. **Codificador de canal:** introduce redundancia controlada mediante códigos de bloque o convolucionales para corregir errores de transmisión.
3. **Modulador digital:** mapea el flujo binario en símbolos de constelaciones como PSK y QAM, adaptando amplitud, fase o frecuencia.
4. **Canal:** medio físico que impone atenuación, AWGN, desvanecimiento múltiple trayecto e interferencia.
5. **Demodulador digital, demodulador digital y decodificador de fuente:** recuperación de símbolos y aplicación de algoritmos de decodificación para reconstruir el flujo binario original.

2.1.3. Adaptación al canal: Modulación y codificación adaptativa

La Modulación y Codificación Adaptativa es un mecanismo dinámico que ajusta el esquema de modulación y la tasa de codificación según la estimación de la calidad del canal usando parámetros como SNR, desvanecimiento, interferencia [6]. En SNR altos, se seleccionan modulaciones de orden superior (p. ej., 64-QAM) para maximizar la tasa de bits; en SNR bajos,

se recurre a esquemas más robustos (BPSK, QPSK) para garantizar la fiabilidad. La clasificación automática de señales cierra el bucle al indicar al transmisor qué esquema de modulación usar para optimizar el enlace.

2.2. Esquemas de modulación digital

La modulación digital convierte secuencias binarias en formas de onda analógicas ajustando la amplitud, fase o frecuencia de una portadora, con el objetivo de transmitir $\log_2 M$ bits por símbolo y optimizar la eficiencia espectral [3]. Aquí, M representa la cantidad total de símbolos distintos que pueden ser transmitidos en cada modulación; es decir, el tamaño de la constelación. En este trabajo se emplean los siguientes esquemas:

Modulación	M	$\log_2 M$ bits/símbolo	Robustez
BPSK	2	1	Muy alta
QPSK	4	2	Alta
16-QAM	16	4	Media
32-QAM	32	5	Media-baja
64-QAM	64	6	Baja
128-QAM	128	7	Muy baja
256-QAM	256	8	Muy baja

Tabla 2.1: Esquemas de modulación digital considerados.

Aspectos destacados

- **BPSK:** dos fases ($0^\circ/180^\circ$), ideal para SNR bajos y entornos muy ruidosos.
- **QPSK:** cuatro fases equiespaciadas, duplica la tasa de BPSK manteniendo robustez.
- **16-QAM:** cuatro niveles de amplitud/fase, velocidad moderada y robustez media.
- **32-QAM:** cinco bits/símbolo, incremento de tasa sobre 16-QAM con ligera pérdida de robustez.
- **64-QAM:** seis bits/símbolo, alta eficiencia espectral a costa de mayor requisito de SNR.
- **128-QAM:** siete bits/símbolo, muy alta tasa, muy sensible al ruido e interferencia.
- **256-QAM:** ocho bits/símbolo, máxima eficiencia espectral, extremadamente susceptible al ruido.

2.3. Representación de señales I-Q

La representación de componentes en fase y en cuadratura (I-Q) descompone una señal modulada en dos componentes ortogonales, facilitando su procesamiento y análisis [3, 4]. Matemáticamente, una señal transmitida en el dominio del tiempo continuo $r(t)$ sobre una portadora

f_c se escribe como:

$$r(t) = I(t) \cos(2\pi f_c t) - Q(t) \sin(2\pi f_c t),$$

donde $I(t)$ y $Q(t)$ contienen la información modulada. Esta forma real se representa de manera más compacta como señal compleja:

$$r(t) = I(t) + jQ(t).$$

En el dominio discreto, cada muestra se almacena como el par $I[n], Q[n]$, lo cual es especialmente útil para el análisis digital y la implementación de algoritmos de decodificación.

2.3.1. Transformación al dominio polar

Para extraer características discriminativas en AMC, se convierte cada punto I-Q a coordenadas polares:

$$s = I + jQ = r e^{j\theta},$$

donde

$$r = |s| = \sqrt{I^2 + Q^2}, \quad \theta = \angle s = \tan^{-1} \frac{Q}{I}.$$

- **Magnitud r** : resalta variaciones de amplitud, clave en modulaciones QAM.
- **Ángulo θ** : indica desplazamientos de fase, esencial en modulaciones PSK.

Esta conversión potencia la separación de clases en el espacio de características, pues magnitud y fase suelen proporcionar información más robusta ante ruido y desvanecimiento.

En la clasificación automática, estas representaciones (cartesiana y polar) se utilizan como entrada para métodos estadísticos y modelos de aprendizaje profundo que aprenden a distinguir patrones de modulación directamente de las muestras I-Q.

2.3.2. Diagrama de constelación

Los diagramas de constelación son una herramienta visual utilizada para representar señales digitales moduladas. En estos diagramas cada símbolo transmitido se muestra como un punto en un plano cartesiano.

Este plano se conoce como plano I-Q, donde las siglas hacen referencia a las componentes I y Q de la señal. La componente I está en fase con la portadora, mientras que la componente Q está desfasada 90 grados.

El diagrama de constelación muestra así los símbolos modulados como puntos en este plano. A medida que M aumenta, la densidad de puntos crece y la distancia mínima entre símbolos disminuye, reduciendo la tolerancia al ruido [4].

Las modulaciones que emplearemos en esta investigación se centran en PSK y QAM, sus expresiones matemáticas se definen:

- **M-PSK:**

$$s_m = I_m + jQ_m = \sqrt{E_s} \exp \left[j \left(2\pi \frac{m}{M} + \varphi_0 \right) \right], \quad m = 0, 1, \dots, M-1, \quad (2.1)$$

donde

$$I_m = \sqrt{E_s} \cos \left(2\pi \frac{m}{M} + \varphi_0 \right), \quad (2.2)$$

$$Q_m = \sqrt{E_s} \sin \left(2\pi \frac{m}{M} + \varphi_0 \right), \quad (2.3)$$

E_s es la energía por símbolo y φ_0 es el desplazamiento de fase inicial.

- **Cuadratura M-QAM:** $M = L^2$.

$$s_{i,q} = a \left[(2i-1-L) + j(2q-1-L) \right], \quad i, q = 1, \dots, L, \quad (2.4)$$

donde

$$a = \frac{\sqrt{3E_s}}{2(M-1)} \quad (2.5)$$

garantiza que la energía promedio por símbolo se mantenga fija E_s .

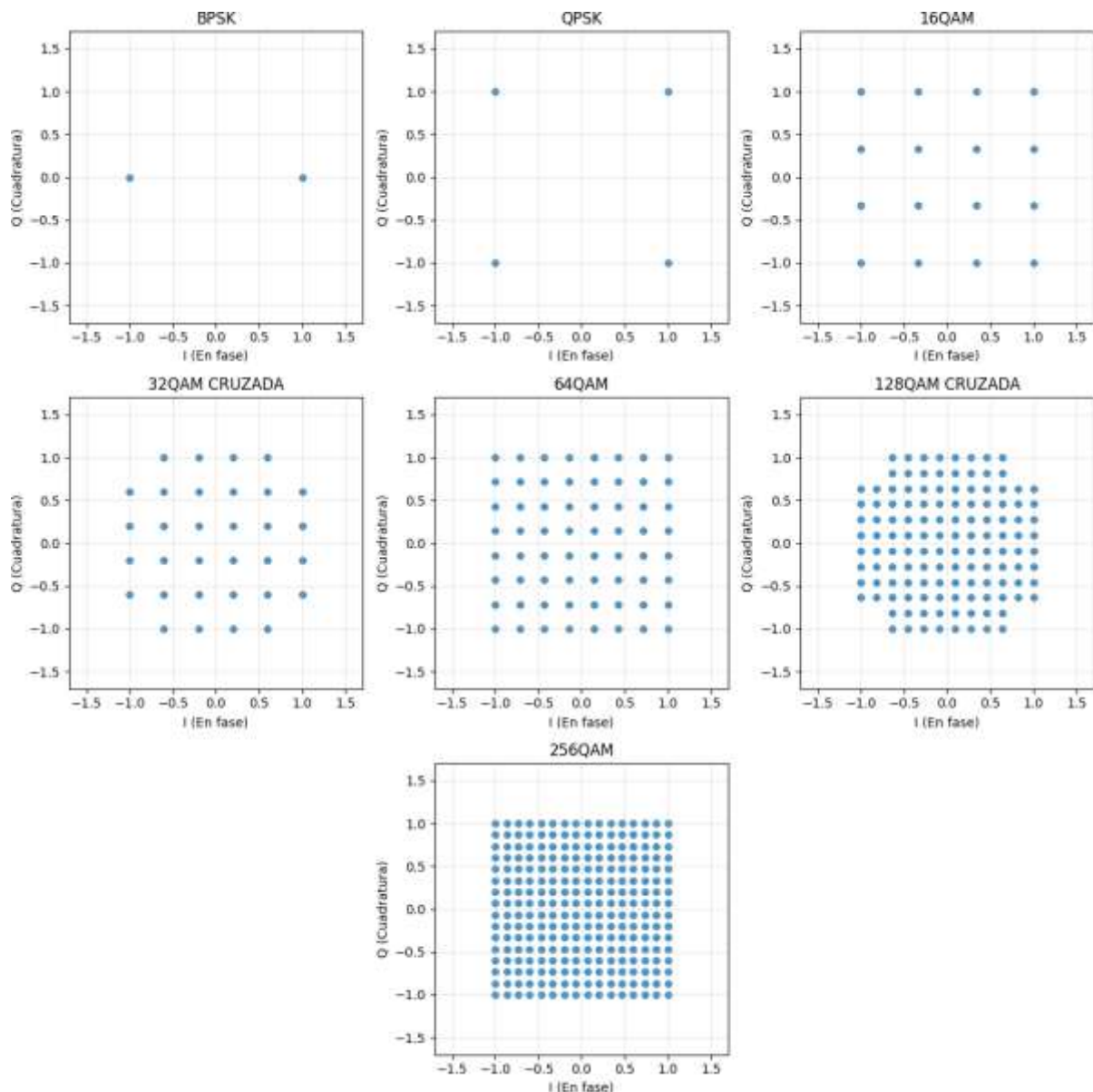


Figura 2.2: Constelaciones de BPSK, QPSK, 16-QAM, 32-QAM, 64-QAM, 128-QAM y 256-QAM.

2.3.3. Impacto de la densidad de símbolos

- **BPSK, QPSK (baja densidad):** puntos ampliamente separados, detección sencilla en presencia de ruido.
- **16-QAM a 256-QAM (alta densidad):** puntos muy cercanos, mayor eficiencia espectral pero alta sensibilidad al ruido, interferencia y desvanecimiento.

Esta diferencia en la densidad de símbolos implica que pequeñas perturbaciones del canal pueden acercar los puntos (símbolos) lo suficiente como para que sean confundidos y provoquen errores de clasificación, especialmente en modulaciones de orden superior que requieren técnicas de reconocimiento más sofisticadas [2].

2.4. Modelos de canal e interferencia

La señal modulada sufre alteraciones durante la transmisión inalámbrica, debidas a ruido, desvanecimiento y fuentes de interferencia. Estos efectos deben modelarse adecuadamente para diseñar y evaluar sistemas de clasificación automática de modulaciones.

2.4.1. Ruido blanco aditivo Gaussiano

El ruido térmico $w[n]$ se modela como un proceso Gaussiano de media cero y varianza σ^2 . Se asume además que las muestras $w[n]$ son independientes e idénticamente distribuidas (i.i.d.) y que poseen densidad espectral plana. En notación discreta:

$$r[n] = s[n] + w[n], \quad w[n] \sim CN(0, \sigma^2).$$

Este modelo de canal, denominado AWGN, es el punto de partida para el análisis de rendimiento y la comparación de esquemas de modulación [3].

2.4.2. Desvanecimiento por trayectoria múltiple

La propagación a través de múltiples trayectorias provoca variaciones de amplitud y fase en la señal recibida, debido a la interferencia constructiva y destructiva entre las copias retardadas de la señal. Estas variaciones se conocen como desvanecimiento y se modelan estadísticamente mediante una función de densidad de probabilidad (FDP) $f(a)$, que describe cómo se distribuye la amplitud a de la señal afectada por el canal. Los dos modelos más comunes para modelos el desvanecimiento son los siguientes:

- **Rayleigh:** sin componente línea de vista (LoS); la amplitud a sigue la siguiente función de densidad de probabilidad:

$$f(a) = \frac{a}{\sigma^2} e^{-a^2/(2\sigma^2)}, \quad a \geq 0.$$

donde σ^2 controla qué tan dispersos son los valores de amplitud. En otras palabras, está relacionado con la potencia promedio de la señal recibida cuando existen múltiples trayectorias.

- **Rician:** con componente dominante LoS; la amplitud a tiene distribución

$$f(a) = \frac{a}{\sigma^2} e^{-(a^2+s^2)/(2\sigma^2)} I_0 \left(\frac{as}{\sigma^2} \right),$$

donde s es la amplitud de la componente LoS e I_0 es la función de Bessel modificada de primer tipo [7].

En forma discreta, el canal de trayectorias múltiples se incorpora mediante convolución:

$$r[n] = \sum_{k=0}^L h[k] s[n-k] + w[n],$$

donde

$$h[k] = a_k e^{j\varphi_k},$$

describe la amplitud a_k y el desfase φ_k de la k -ésima trayectoria, y L es el número total de trayectorias. El término $w[n]$ representa el ruido aditivo blanco gaussiano en la muestra n [8].

2.4.3. Interferencia co-canal y adyacente

Además del ruido y el desvanecimiento (fading), la señal puede verse afectada por interferencia:

- **interferencia co-canal (CCI):** emisiones de otros transmisores que operan en la misma frecuencia, reduciendo la relación señal-interferencia (SIR).
- **interferencia de canal adyacente (ACI):** solapamiento de bandas contiguas por filtrado imperfecto de transmisor o receptor.

El modelo discreto con interferencia $i[n]$ corresponde a:

$$r[n] = \sum_k h[k] s[n-k] + i[n] + w[n],$$

y su impacto se analiza a través de la densidad espectral de potencia $S_i(f)$, que describe la distribución de energía del interferente sobre la banda [6].

2.5. Formulación del problema de clasificación automática de modulación

La AMC se plantea como un problema de decisión múltiple entre K hipótesis de modulación [2, 9]. En notación discreta, la señal recibida $r[n]$ se modela como:

$$H_k : r[n] = \sum_{m=0}^L h[m] s_k[n-m] + w[n] + i[n], \quad k = 1, \dots, K,$$

donde:

- $s_k[n]$ es la secuencia de muestras de la señal bajo la hipótesis H_k (por ejemplo, BPSK para $k = 1$, QPSK para $k = 2$, 16-QAM para $k = 3$, etc.).
- $h[m]$ son los coeficientes de la respuesta al impulso del canal (multipath).
- $w[n] \sim CN(0, \sigma^2)$ representa ruido AWGN con media cero, varianza σ^2 e idénticamente distribuido.

- $i[n]$ modela interferencia co-canal o adyacente.

En dominio continuo:

$$H_k : r(t) = \int_{-\infty}^{\infty} s_k(\tau)h(t - \tau) d\tau + w(t) + i(t).$$

El objetivo es, dada la observación $r[n]$, escoger la hipótesis H_k que maximiza la probabilidad de modulación utilizada.

2.5.1. Estrategias de solución

Las principales estrategias para resolver este problema son:

- **Métodos clásicos:** que extraen manualmente características estadísticas y espectrales de la señal para luego aplicar técnicas de clasificación basadas en pruebas estadísticas o algoritmos de aprendizaje supervisado tradicionales, como máquina de soporte de vectores (SVM).
- **Modelos de aprendizaje profundo:** arquitecturas de extremo a extremo tales como CNN, memoria a largo y corto plazo (LSTM), Transformadores, que aprenden directamente de las muestras I-Q o de sus representaciones polares, sin etapa previa de extracción de características [10, 11].

2.6. Métodos clásicos de clasificación

Los enfoques clásicos de clasificación automática de modulaciones siguen un esquema secuencial de dos etapas [2]:

1. **Extracción de características:** cálculo de métricas estadísticas y espectrales a partir de las muestras recibidas, tales como:
 - *Cumulantes de alto orden* para capturar información de asimetría y curtosis.
 - *Estadísticas ciclostacionarias* que aprovechan propiedades de periodicidad inherentes a la señal.
 - *Medidas de densidad espectral* basadas en la transformada de Fourier.
 - *Pruebas no paramétricas* como la prueba de Kolmogorov–Smirnov (KS).
2. **Clasificación estadística:** aplicación de pruebas de hipótesis o clasificadores de máquina:
 - **Pruebas de razón de verosimilitud:** métodos estadísticos como prueba de razón de verosimilitud promedio (ALRT), prueba de razón de verosimilitud generalizada

(GLRT) y prueba de razón de verosimilitud híbrida (HLRT) que evalúan la probabilidad de cada modulación bajo enfoques bayesianos o de máxima verosimilitud para la toma de decisión.

- **Clasificadores basados en separación en el espacio de características:** algoritmos como SVM y K vecinos más cercanos (KNN) que identifican fronteras óptimas para discriminar entre clases de modulación.

2.6.1. Ventajas e inconvenientes

- **Ventajas:**

- A diferencia de los métodos basados en IA, los métodos clásicos no requieren grandes conjuntos de datos para entrenamiento.
- Las características utilizadas en métodos clásicos tienen una interpretación directa y representan propiedades reales de la señal, lo que permite un mejor entendimiento y control sobre el proceso de clasificación. En contraste, las redes neuronales actúan como cajas negras donde sólo se conoce la arquitectura, pero no el detalle de qué está aprendiendo internamente.

- **Limitaciones:**

- Dependencia de conocimiento previo del canal y parámetros de la señal.
- Complejidad computacional elevada en pruebas de verosimilitud para modulaciones de alto orden.
- Sensibilidad a condiciones de canal no modeladas en la etapa de extracción.

Para un análisis detallado de estos métodos, sus variantes y desempeño en diferentes escenarios de SNR y canal, véase el Capítulo 3.

2.7. Fundamentos de aprendizaje automático y profundo

El aprendizaje automático y, en particular, el aprendizaje profundo, proporcionan herramientas potentes para la clasificación automática de señales al permitir que los modelos aprendan directamente de los datos sin necesidad de extracción manual de características [12]. A continuación, se presentan las arquitecturas y conceptos clave empleados en el desarrollo de este documento.

2.7.1. Arquitecturas básicas y perceptrón multicapa (MLP)

El perceptrón multicapa (MLP) es la extensión de la neurona artificial a redes con una o más capas ocultas. Cada neurona calcula:

$$y = f \sum_{i=1}^n w_i x_i + b ,$$

donde f corresponde a una función de activación (como ReLU, Sigmoid o Tanh), que introduce no linealidad en la salida de la neurona. En la expresión, x_i representa la entrada número i , w_i es el peso que define la importancia de esa entrada, y b es un término de sesgo que permite ajustar el umbral de activación de la neurona. La suma ponderada $\sum w_i x_i + b$ refleja cómo cada entrada contribuye a la decisión de la neurona. Los MLPs permiten aprender relaciones no lineales entre los datos de entrada y la salida deseada, pero no explotan la estructura espacial o temporal de las señales [13]. A continuación se muestra el esquemático de un perceptrón con 5 entradas.

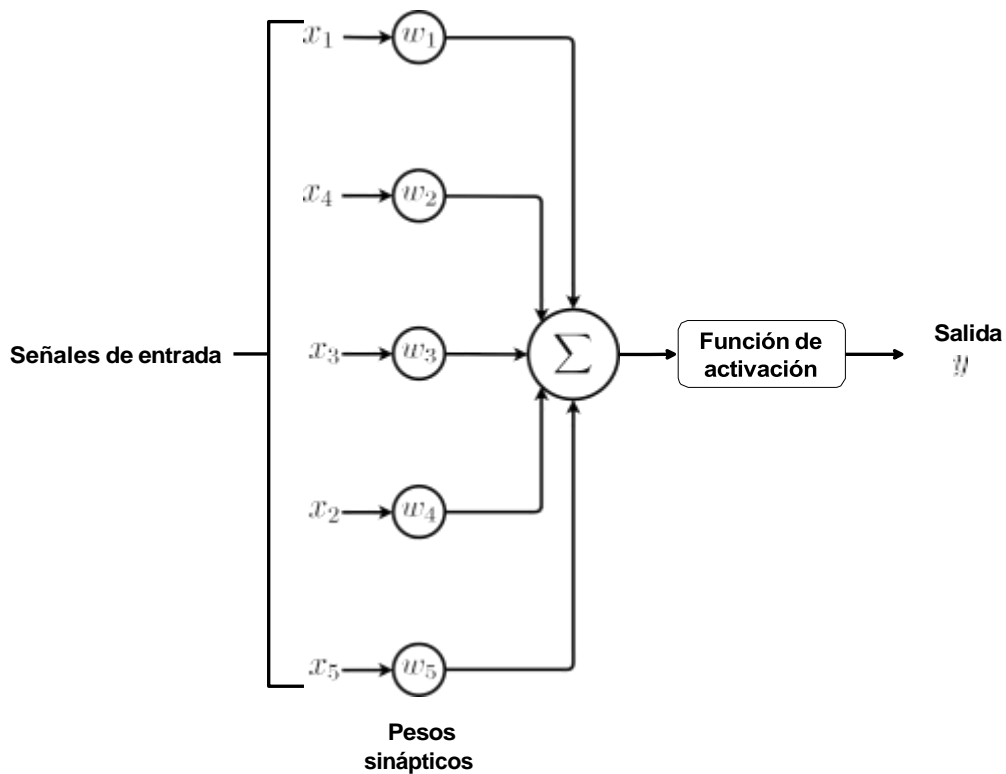


Figura 2.3: Diagrama de un perceptrón con 5 entradas.

2.7.2. Redes neuronales convolucionales (CNN)

Las CNN aplican filtros convolucionales que comparten pesos para extraer características locales, por ejemplo desde las representaciones I-Q o polares de una señal de comunicaciones. Una capa convolucional produce la salida:

$$y_{i,j} = \sum_{m,n} W_{m,n} x_{i+m,j+n} + b,$$

donde:

- $x_{i+m,j+n}$ es la entrada en la posición desplazada $(i + m, j + n)$, dentro de la región que cubre el filtro.
- $W_{m,n}$ representa el valor del filtro (o kernel) en la posición (m, n) , compartido para toda la imagen o mapa de características.
- b es un sesgo añadido al resultado de la convolución.
- $y_{i,j}$ es la salida (o activación) en la posición (i, j) del mapa de características generado por la capa.

Esta operación suele ir seguida de una etapa conocida como *pooling*, que reduce la dimensionalidad espacial y mejora la invariancia a pequeñas traslaciones. Entre las arquitecturas clásicas que siguen este esquema se encuentran las siguientes:

- **LeNet-5** [14], una de las primeras CNN exitosas aplicada a reconocimiento de dígitos manuscritos.
- **AlexNet** [15], que popularizó las CNN profundas y el uso masivo de unidad de procesamiento gráfico (GPU) para entrenamiento, ganando el concurso ImageNet 2012.
- **VGG** [16], caracterizada por el uso de filtros pequeños (3x3) y redes profundas, que mejoraron el desempeño en tareas de clasificación.

Las arquitecturas mencionadas cuentan con versiones preentrenadas, es decir, modelos cuyas redes ya fueron entrenadas previamente en grandes bases de datos de imágenes como ImageNet. Estas redes preentrenadas capturan características visuales generales en sus primeras capas, lo que permite reutilizarlas para tareas distintas a la original mediante técnicas de transferencia de aprendizaje. Esta estrategia reduce significativamente el tiempo de entrenamiento y la necesidad de grandes volúmenes de datos, mejorando la eficiencia y precisión en problemas específicos, como la clasificación de señales I-Q mediante representaciones visuales.

2.7.3. Redes residuales (ResNet)

Para entrenar redes muy profundas, los bloques residuales introducen conexiones de atajo que suman la entrada x al bloque de convolución $F(x)$:

$$y = F(x) + x.$$

Este tipo de red surgió para resolver el problema que aparece al aumentar la profundidad de una red, donde el gradiente tiende a disminuir a medida que se propaga hacia las capas iniciales. Dado que el gradiente es la señal que guía la actualización de los pesos durante el

entrenamiento, su desaparición impide que las capas aprendan adecuadamente. Las redes residuales permiten mantener esta señal activa gracias a las conexiones directas, lo que mejora la capacidad de aprendizaje en arquitecturas muy profundas [17]. La Figura 2.4 muestra la estructura básica de un bloque de conexión residual.

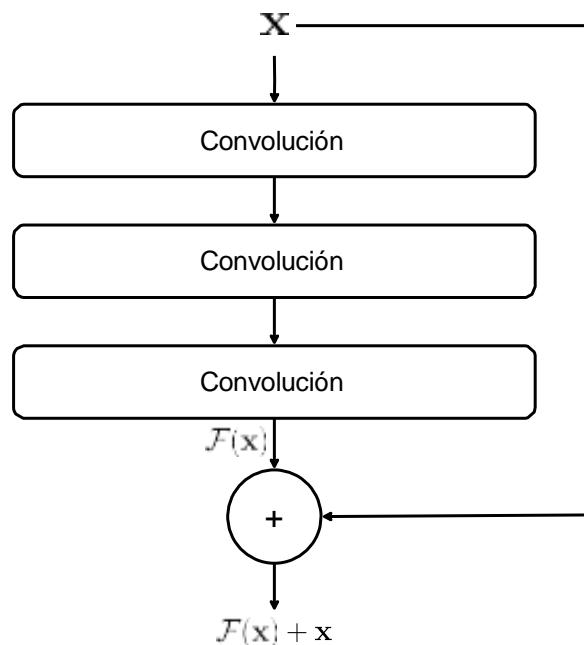


Figura 2.4: Estructura de bloque residual.

2.7.4. Redes neuronales recurrentes (RNN) y Long Short-Term Memory (LSTM)

Las redes neuronales recurrentes (RNN) modelan dependencias temporales al mantener un estado interno. Sin embargo, experimentan el desvanecimiento del gradiente en secuencias largas. Las LSTM introducen puertas de entrada, olvido y salida para regular el flujo de información y preservar gradientes [18]. La célula mínima LSTM es un bloque fundamental en redes neuronales recurrentes, diseñada para aprender dependencias a largo plazo en secuencias, mitigando el problema del desvanecimiento del gradiente. La Figura 2.5 muestra sus componentes principales:

- C_{t-1} : estado de la célula en el instante anterior, que representa la memoria a largo plazo.
- h_{t-1} : salida oculta en el instante anterior, memoria a corto plazo.
- x_t : entrada actual en el instante t .

El funcionamiento de la célula está regulado por tres puertas que controlan el flujo de información mediante funciones de activación:

1. **Puerta de olvido f_t** : utiliza la función sigmoide (σ) para decidir qué información del

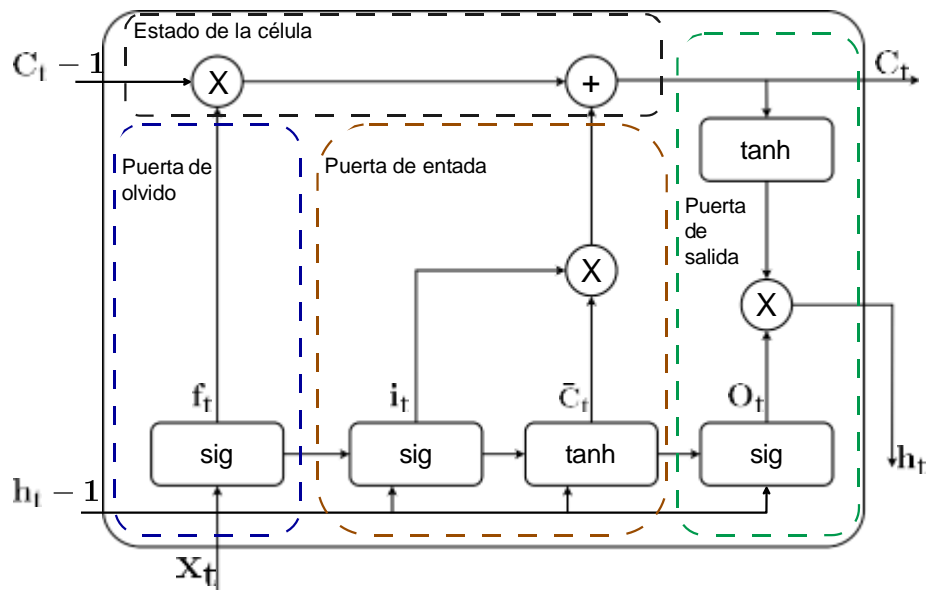


Figura 2.5: Diagrama de la arquitectura de una célula LSTM.

estado previo C_{t-1} debe mantenerse o eliminarse:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

2. **Puerta de entrada i_t y estado candidato \tilde{C}_t :** la puerta de entrada, también con función sigmoide, determina qué información nueva se añade al estado, mientras que el estado candidato se calcula con la función tangente hiperbólica (\tanh):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \quad \tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

3. **Actualización del estado c_t :** combina la información olvidada y la nueva entrada mediante una suma ponderada elemento a elemento:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$

4. **Puerta de salida O_t :** determina la salida oculta h_t aplicando sigmoide y \tanh al estado actualizado:

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o), \quad h_t = O_t \odot \tanh(C_t)$$

En estas expresiones, $\sigma(\cdot)$ representa la función de activación sigmoide que limita la salida entre 0 y 1, y $\tanh(\cdot)$ es la función tangente hiperbólica que limita la salida entre -1 y 1. El operador \odot denota el producto elemento a elemento.

Esta estructura permite a la célula LSTM seleccionar y mantener información relevante a lo largo del tiempo, facilitando el aprendizaje de dependencias temporales largas en las secuencias

de datos.

2.7.5. Redes neuronales profundas (DNN)

Una DNN comprende una extensión del perceptrón multicapa que incorpora múltiples capas ocultas para aprender representaciones jerárquicas de los datos. Cada capa l transforma su entrada $\mathbf{h}^{(l-1)}$ según:

$$\mathbf{h}^{(l)} = f(\mathbf{W}^{(l)} \mathbf{h}^{(l-1)} + \mathbf{b}^{(l)}),$$

donde:

- $\mathbf{h}^{(0)} = \mathbf{x}$ es el vector de entrada original (x_i son sus componentes).
- $\mathbf{W}^{(l)} \in \mathbb{R}^{n_l \times n_{l-1}}$ es la matriz de pesos de la capa l .
- $\mathbf{b}^{(l)} \in \mathbb{R}^{n_l}$ es el vector de sesgos de la capa l .
- f es la función de activación (ReLU, Sigmoid, Tanh, etc.).
- $\mathbf{h}^{(l)} \in \mathbb{R}^{n_l}$ es la activación de la capa l , con n_l neuronas.

Las DNN permiten:

- **Aprendizaje de características de alto nivel:** las primeras capas capturan patrones simples, y las sucesivas abstraen información más compleja.
- **Flexibilidad arquitectónica:** se pueden añadir técnicas como *dropout*, *batch normalization* o *skip connections* para mejorar la convergencia y regularización.

Limitaciones principales:

- **Requieren grandes volúmenes de datos** y potencia de cómputo para evitar el sobreajuste.
- **Gradientes inestables** en redes muy profundas, mitigados con mecanismos de normalización y optimizadores avanzados.
- **No explotan explícitamente** la estructura espacial o temporal de las señales, a menos que se combinen con otros bloques especializados (CNN, RNN, Transformer).

2.7.6. Mecanismos de atención y Transformer

Los mecanismos de atención permiten que una red neuronal enfoque selectivamente diferentes partes de una secuencia de entrada, en lugar de tratar toda la información por igual. Esto mejora la capacidad del modelo para capturar dependencias largas y contextos relevantes, lo cual es especialmente útil en tareas secuenciales como traducción automática o modelado de lenguaje.

El modelo Transformer elimina la necesidad de recurrencia (como en LSTM) y basa su funcionamiento en un mecanismo llamado *atención escalada por producto punto* [19], definido como:

$$\text{Attention}(Q, K, V) = \text{softmax} \frac{QK^T}{d_k} V.$$

donde:

- Q (queries), K (keys) y V (values) son matrices que se obtienen mediante proyecciones lineales de la secuencia de entrada.
- d_k es la dimensión de las claves (K), usada para escalar el producto punto y estabilizar el gradiente.
- softmax normaliza las similitudes para obtener pesos de atención que determinan la importancia relativa de cada elemento.
- El resultado es una combinación ponderada de los vectores V , donde los pesos dependen de la similitud entre Q y K .

El Transformer utiliza múltiples “cabezas” de atención de forma paralela para capturar distintos tipos de relaciones contextuales. A esto se le conoce como *atención multi-cabeza*, formalmente:

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V), \quad \text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O.$$

donde:

- W_i^Q, W_i^K, W_i^V son matrices de pesos que proyectan las entradas hacia el subespacio de cada cabeza.
- h es el número de cabezas de atención.
- W^O es una matriz de pesos que proyecta la concatenación de todas las cabezas hacia la salida final.

Para una representación visual de este mecanismo, véase la Figura 2 de la referencia [19].

2.7.7. Bloques Squeeze-and-Excitation (SE)

Los módulos SE integran un bloque de *squeeze*, junto a una capa de Global Average Pooling One-Dimensional (GAP1D) y uno de *excitation* (red de dos capas) para recalibrar dinámicamente la respuesta de cada canal convolucional [20]. Esto mejora la capacidad de la red para enfocarse en características discriminativas.

2.7.8. Hiperparámetros y técnicas de regularización

Los hiperparámetros configuran el comportamiento del entrenamiento y deben ajustarse cuidadosamente:

- **Tasa de aprendizaje:** determina el tamaño de los pasos de actualización de pesos.
- **Batch size y epochs:** controlan la granularidad y duración del entrenamiento.
- **Optimizadores:** descenso del gradiente estocástico (SGD) [21], Adam [22] y sus variantes influyen en la convergencia.
- **Función de pérdida (loss):** durante el entrenamiento, la red neuronal busca minimizar una función de pérdida, que mide el error entre la salida predicha y la salida verdadera. Esta función guía el proceso de aprendizaje al proporcionar una señal de retroalimentación que indica qué tan bien está funcionando el modelo.

Para problemas de regresión, una función común es el error cuadrático medio (MSE), que penaliza la distancia entre valores reales y predichos.

En problemas de clasificación con múltiples clases, como la identificación automática de la modulación en señales de comunicación, una función comúnmente utilizada es la **entropía cruzada categórica**. Esta función mide qué tan diferente es la distribución de clases que predice la red neuronal con respecto a la distribución real de clases.

Para obtener dicha distribución de clases predicha, se utiliza una capa *softmax* al final de la red. Esta capa toma las salidas crudas del modelo y las transforma en probabilidades normalizadas que suman uno. Así, cada salida representa la confianza del modelo en que una muestra pertenece a una clase determinada. Matemáticamente, la salida \hat{y}_i para la clase i se calcula como:

$$\hat{y}_i = \frac{e^{z_i}}{\sum_{j=1}^C e^{z_j}}$$

donde z_i es la salida cruda del modelo para la clase i , y C es el número total de clases.

Una vez convertidas las salidas a probabilidades, la entropía cruzada se utiliza para calcular la pérdida como [12]:

$$L_{CE} = -\sum_{i=1}^C y_i \cdot \log(\hat{y}_i),$$

donde:

- C : número total de clases
- y_i : valor real de la clase en formato *one-hot*,
- \hat{y}_i : probabilidad predicha para la clase i .

El formato one-hot es una manera de representar la clase verdadera como un vector binario de longitud C , en el que sólo la posición correspondiente a la clase correcta toma el valor 1, y todas las demás posiciones son 0. Por ejemplo, si hay cinco clases ($C = 5$) y la clase correcta es la número 3, entonces la codificación one-hot será: $[0, 0, 1, 0, 0]$.

Esta función penaliza con mayor severidad las predicciones incorrectas en las que el modelo muestra una alta confianza.

- **Regularización:** consiste en mecanismos que reducen el sobreajuste y mejoran la capacidad de generalización del modelo.
 - Dropout introduce *sparsity* estocástico al desactivar aleatoriamente un subconjunto de neuronas durante el entrenamiento, lo que fuerza a la red a no depender excesivamente de rutas específicas [23].
 - L2 weight decay agrega un término de penalización proporcional a la norma de los pesos, favoreciendo soluciones más simples y robustas frente a ruido en los datos.
 - Batch Normalization normaliza las activaciones de cada lote de entrenamiento. Esto reduce las variaciones en la distribución de las activaciones a lo largo del entrenamiento, lo que facilita la propagación de gradientes, mejora la estabilidad numérica y permite utilizar tasas de aprendizaje más altas sin que el entrenamiento diverja [24].

Estos hiperparámetros actúan como mecanismos de ajuste que se deben calibrar durante la experimentación. Su correcta configuración es clave para mejorar el rendimiento del modelo, ya que afectan tanto la velocidad como su capacidad de aprendizaje.

2.8. Resumen del capítulo

En este capítulo se presentaron los conceptos clave para entender la AMC en entornos inalámbricos. Se revisaron los fundamentos de las comunicaciones digitales, los esquemas de modulación desde BPSK a 256-QAM y la representación de señales en formato I-Q y polar.

También se explicaron los modelos de canal más comunes, como el canal AWGN, el desvanecimiento por trayectorias múltiples y la interferencia.

Finalmente, se compararon los métodos clásicos de clasificación, que usan extracción de características manuales, con los enfoques modernos basados en redes neuronales. Se describieron las arquitecturas más usadas: MLP, CNN, LSTM, Transformers, así como técnicas de regularización y los hiperparámetros clave para su entrenamiento. En el Capítulo 3 se revisará en profundidad el estado del arte de la clasificación automática de modulaciones.

3. Estado del Arte

3.1. Clasificación automática de señales: métodos tradicionales

3.1.1. Pruebas de verosimilitud

Estos métodos formulan la clasificación como hipótesis múltiples usando la FDP condicionada a cada esquema de modulación [2].

3.1.1.1. Average Likelihood Ratio Test (ALRT)

El ALRT integra la verosimilitud sobre la distribución a priori de parámetros $\Theta = \{\alpha, \sigma, \theta_0\}$ [25]:

$$L_{\text{ALRT}}(r) = \int_{\Theta} L(r | \Theta) f(\Theta) d\Theta = \int_{\Theta} \prod_{n=1}^N \sum_{m=1}^M \frac{1}{2\pi\sigma^2 M} e^{-\frac{|r[n] - \alpha e^{-j\theta_0} A_m|^2}{2\sigma^2}} f(\Theta) d\Theta.$$

- $r[n]$: muestra recibida en el instante discreto n ,
- N : número total de muestras observadas,
- M : número de símbolos de la constelación $\Lambda = \{A_m\}_{m=1}^M$,
- $\{A_m\}_{m=1}^M$: puntos de la constelación transmitida,
- α : ganancia (amplitud) del canal,
- θ_0 : desplazamiento de fase introducido por el canal,
- σ^2 : varianza del ruido aditivo Gaussiano,
- $\Theta = \{\alpha, \sigma, \theta_0\}$: conjunto de parámetros del canal a estimar,
- $f(\Theta)$: FDP a priori conjunta de los parámetros,
- $L(r | \Theta)$: función de verosimilitud conjunta de la secuencia recibida dadas las realizaciones de Θ , es decir

$$L(r | \Theta) = \prod_{n=1}^N \sum_{m=1}^M \frac{1}{2\pi\sigma^2 M} \exp \left\{ -\frac{|r[n] - \alpha e^{-j\theta_0} A_m|^2}{2\sigma^2} \right\}.$$

El método es óptimo en sentido bayesiano si la FDP supuesta coincide con la real; de lo contrario, su desempeño se degrada y su complejidad aumenta exponencialmente con M y N .

3.1.1.2. Generalized Likelihood Ratio Test (GLRT)

En lugar de integrar, el GLRT maximiza la verosimilitud respecto a Θ [25]:

$$L_{\text{GLRT}}(r) = \max_{\Theta} \prod_{n=1}^N \max_{A_m \in \Lambda} \frac{1}{2} e^{-\frac{|r[n]-ae^{-j\theta_0 A_m}|^2}{2\sigma^2}}.$$

Reduce la carga computacional frente al ALRT, pero introduce sesgos en SNR cuando es muy alto o muy bajo, confundiendo modulaciones anidadas ¹ [26].

3.1.1.3. Hybrid Likelihood Ratio Test (HLRT)

El HLRT combina ALRT y GLRT: promedia sobre la constelación y maximiza sobre parámetros de canal [27]. Su complejidad computacional es $O(M^K)$, donde M (como se mencionó previamente) es el número de símbolos por constelación y K el número de parámetros de canal estimados, ofreciendo un compromiso entre exactitud y costo computacional.

3.1.2. Métodos basados en momentos y cumulantes

Los cumulantes de orden superior (HOC) capturan propiedades estadísticas de la señal. Ali et al. [28] extraen cumulantes de orden 1 a 4 en ventanas de $N = 4096$ muestras, aplican un logaritmo para estabilizar varianza y clasifican M-QAM de órdenes hasta 1024.

Para cuantificar el desempeño, se define el porcentaje de clasificación correcta (PCC) como el cociente entre el número de decisiones de modulación correctas y el número total de pruebas:

$$\text{PCC} = \frac{\text{clasificaciones correctas}}{\text{total de pruebas}} \times 100\%.$$

Con esta métrica, [28] reporta:

- $P_{CC} > 99\%$ para $SNR \geq 5\text{dB}$.
- Rendimiento robusto en SNR bajos y canales AWGN, Rayleigh, Rician.

3.1.3. Pruebas estadísticas no paramétricas

El test de KS compara la función empírica de distribución acumulada (ECDF, *Empirical CDF*) de un estadístico de decisión ² con las funciones de distribución acumulada (CDF) teóricas de cada modulación [29]:

1. Calcular ECDF de magnitud o fase.
2. Medir la máxima distancia a las CDF candidatas.

¹Dos esquemas se dicen *anidados* cuando comparten subconjuntos de puntos en la misma constelación, p. ej., 16QAM dentro de 64QAM.

²Cualquier métrica calculada a partir de las muestras, p. ej., magnitud o fase.

3. Seleccionar la modulación con mínima divergencia.

Es eficaz con pocas muestras, robusto a desvanecimiento y ruido impulsivo (picos esporádicos de alta amplitud que distorsionan la señal), y aplicable a modulaciones de bajo y alto orden.

3.1.4. Visión condensada de métodos tradicionales

Como resumen de los métodos tradicionales, la Tabla 3.1 sintetiza las cuatro grandes familias de estos, con sus técnicas más representativas, ventajas y limitaciones principales.

Familia	Técnicas	Ventaja principal	Limitación
Verosimilitud	ALRT[25], GLRT [26], HLRT [27]	Óptimos bayesianos (ALRT) / menor complejidad (GLRT)	Alta carga computacional / sesgo en SNR extremo
Cumulantes de orden superior	HOC+log [28]	Robustez en modulaciones M-QAM altas	Depende de ventana de muestras y umbrales
Pruebas estadísticas	KS [29]	No paramétrico, menos muestras	Sensible a elección de estadístico

Tabla 3.1: Visión general de los métodos clásicos de AMC.

3.1.5. Comparación de métodos tradicionales

En la Tabla 3.2 se resume el desempeño de cada método a través del PCC obtenido para distintas modulaciones y rangos de SNR.

Método	Técnica	Modulaciones	Muestras	SNR (mín.–máx.)	P_{CC} min.	P_{CC} max.
ALRT	Integración bayesiana	BPSK, QPSK	100	-7-10dB	80 %	100 %
GLRT	Maximización	BPSK, . . . ,64-QAM	1000	0-20dB	55 %	100 %
HLRT	Híbrido	BPSK, . . . ,16-QAM	1000	-15-15dB	31 %	100 %
KS	No paramétrico	4-QAM, . . . ,64-QAM	50	0-20dB	40 %	100 %

Tabla 3.2: Comparación de métodos tradicionales de AMC.

Estos métodos, si bien sentaron las bases de la AMC, presentan limitaciones en complejidad computacional y robustez ante condiciones extremas de SNR, lo que motiva el paso a técnicas basadas en IA introducidas en la Sección 3.2.

3.2. Métodos de inteligencia artificial para AMC

3.2.1. Métodos estadísticos avanzados

Statistical Signal Characterization (SSC). Hossen et al. [30] extraen cuatro características estadísticas —amplitud promedio (M_a), periodo promedio (M_t), desviación de amplitud (D_a) y

desviación de periodo (D_i)— y las ingresan a una red neuronal artificial (ANN) de tres capas, logrando 100 % de eficiencia para $SNR \geq 7dB$ y $> 83\%$ para $SNR = 3dB$ [30].

La Tabla 3.3 presenta los parámetros y resultados obtenidos para el método SSC, destacando su PCC mínimo y máximo.

Resumen de desempeño: SSC [30]	
Conjunto de datos	Simulación Monte Carlo con modulaciones analógicas (Chan et al. [31]) y digitales ($M = 4$), $f_s = 40\text{ kHz}$, $f_c = 10\text{ kHz}$, $N = 2048$, AWGN a SNR 3–25 dB, 40 muestras/clase.
Modulaciones	AM-DSB-TC, AM-DSB-SC, SSB, FM, PM, ASK, PSK, FSK
Rango de SNR	3, 5, 7, 10, 15, 25 dB
PCC mínimo (%)	73.4 (a 3 dB)
PCC máximo (%)	100 (a ≥ 10 dB)
Muestras/señal	2048

Tabla 3.3: Parámetros y resultados para el método SSC.

- **Brecha específica (SSC)** A pesar de lograr PCC relativamente altos en SNR moderados, no se evalúa la robustez frente a modulaciones de orden superior tales como 16QAM o 64QAM.

Prueba de Kolmogorov-Smirnov (KS). Este método no paramétrico compara la función de distribución acumulada empírica (ECDF) de un estadístico de decisión con las función de distribución acumulada (CDF) teóricas de cada modulación. Ofrece robustez ante desvanecimiento y ruido impulsivo, requiriendo menos muestras que los métodos de cumulantes [29].

La Tabla 3.4 expone los parámetros y resultados obtenidos para el método basado en el Test de Kolmogorov-Smirnov.

Resumen de desempeño: Clasificador K-S	
Conjunto de datos	Señales simuladas en canales AWGN, desvanecimiento plano, OFDM y ruido no Gaussiano
Modulaciones	QAM (4, 16, 64-QAM) y PSK (4, 8, 16-PSK)
Rango de SNR	0-20 dB. Extendiéndose a 30 dB para PSK en desvanecimiento, y desde -10 dB para ruido no Gaussiano
PCC mínimo (%)	$\approx 25\%$ (observado en $SNR=0$ dB)
PCC máximo (%)	$\approx 100\%$ a alto SNR (≥ 15 dB).
Muestras/señal	100

Tabla 3.4: Parámetros y resultados para el método basado en el Test de Kolmogorov-Smirnov.

- **Brecha específica en el método KS** a pesar de su robustez, este no ha sido validado en señales reales ni en presencia de interferencias no gaussianas de banda adyacente. Además, no se detallan las herramientas ni parámetros empleados en la generación sintética de las señales. Esto limita su aplicabilidad práctica.

3.2.2. Redes neuronales en dominio polar

Polar+CNN (Peng et al. [32]). Basado en una CNN. Se transforma cada par I-Q a coordenadas polares (r, θ) y se renderiza como imagen JPEG de 227×227 píxeles, con esta imagen se alimenta a una CNN con base CaffeNet/AlexNet, alcanzando $>90\%$ de precisión en 8PSK y QPSK, aunque su desempeño decae en modulaciones de orden superior a bajo SNR [32].

Los parámetros experimentales y los resultados más relevantes de este enfoque se resumen en la Tabla 3.5, donde se detallan las condiciones de prueba y el rendimiento alcanzado en un rango de SNR de -4 a 14 dB.

Resumen de desempeño: Polar+CNN [32]	
Conjunto de datos	10000 train / 1000 test imágenes por modulación (1000 muestras/imagen). No se especifican número de muestras ni herramientas de simulación.
Modulaciones	QPSK, 8PSK, 16QAM, 64QAM
Rango SNR	-4 – 14 dB
PCC mínimo (%)	74.5% (a -2 dB)
PCC máximo (%)	100% (a ≥ 12 dB)
Imágenes/clase	10000 train / 1000 test

Tabla 3.5: Parámetros y resultados clave para el método Polar+CNN.

- **Brecha específica (Polar+CNN)** El uso de imágenes requiere un preprocesamiento previo, lo que limita el funcionamiento de esta sistema en tiempo real, afectando notoriamente en la práctica. Por otro lado, el trabajo carece de un detalle técnico de las características de las muestras I-Q de las señales.

CNN con parámetros compensatorios: CCN (Teng et al.) [33]. Presentan una arquitectura profunda basada en características polares que incorpora una red de compensación de canal CNN con el objetivo de mitigar el desvanecimiento en amplitud y fase. Los aspectos a recalcar son:

1. **Transformación polar / preprocesamiento:** cada $L = 1000$ muestras de la señal modulada se convierten de coordenadas I-Q a (r, θ) , con rango de radio $[0, 3]$, rango de fase $[-\pi, \pi]$ y resolución de imagen 36×36 .
2. **Arquitectura CNN:** cuatro capas convolucionales seguidas de tres capas densas.
3. **Red de compensación de canal (CCN):** cuatro capas densas que aprenden los parámetros inversos Δr (escala de radio) y $\Delta \theta$ (desplazamiento de fase), aplicados como

$$r' = r \times \Delta r, \quad \theta' = \theta + \Delta \theta.$$

La configuración específica de este modelo y su desempeño en un canal con desvanecimiento plano se consolidan en la Tabla 3.6.

Resumen de desempeño: CNN+CCN [33]	
Conjunto de datos	AWGN + desvanecimiento plano; 5 000 train / 1 000 test imágenes/modulación. No se mencionan parámetros de muestreo.
Modulaciones	QPSK, 8PSK, 16QAM, 64QAM
Resolución	36 × 36 px (polar)
Símbolos/imagen	1 000 muestras
CNN / CCN	4 Conv + 3 Dense / 4 Dense
Entrenamiento	100 000 it., lr=0.0005, decay=0.5, batch=64
Rango SNR	-4 – 12 dB
PCC mínimo (%)	25% (a -4dB)
PCC máximo (%)	100% (a 12dB)

Tabla 3.6: Parámetros y resultados clave para el método CNN con CCN.

- **Brecha específica (CNN+CCN)** La validación está restringida a constelaciones de hasta 64-QAM en un rango de SNR estrecho (-4 a 12 dB). No se ha probado en desvanecimiento Rayleigh o Rician (*aportando realismo*), ni en señales reales "over-the-air".

3.2.3. Modelos híbridos y atención

Jerárquico+algoritmo genético (GA). (Namjin Kim et al.) [34]. Se propone un clasificador jerárquico de señales de modulación, implementado en DSP (TMS320C6701), que combina redes neuronales MLP en cada nivel de la jerarquía con selección de características mediante un GA.

3.2.3.1. Metodología Jerárquico+GA

- Se extraen 31 características de dominio instantáneo y demodulación.
- Se define una jerarquía de 6 niveles, en cada uno de los cuales se clasifican subconjuntos de clases.
- En cada nivel, un GA selecciona el subconjunto óptimo de características para entrenar un MLP.
- Características retenidas por nivel: 4, 4, 7, 2, 8 y 6, respectivamente.

En la Tabla 3.7 se presentan los detalles de la configuración experimental y los resultados obtenidos por este clasificador, destacando su alta precisión en el rango de SNR evaluado.

Resumen de desempeño: Clasificador jerárquico + GA[34]	
Conjunto de datos	Ruido AWGN. 1 024 secuencias de entrenamiento por clase. Cada secuencia está formada por 30 segmentos de señal. No se mencionan frecuencias de muestreo.
Modulaciones	CW, AM, FM, SSB, FSK2, FSK4, PSK2, PSK4, OOK, QAM16, QAM32
Rango de SNR	7 – 12 dB
PCC mínimo (%)	93.32 (sin voto mayoritario)
PCC máximo (%)	99.62 (con voto mayoritario)

Tabla 3.7: Parámetros y resultados clave para el clasificador jerárquico con GA.

- **Brecha específica (Clasificador jerárquico + GA)** La validación está restringida a constelaciones de hasta 32-QAM además de no incluir escenarios dinámicos, se considera un único escenario de 7 a 12 dB. Ninguna herramienta ni proceso de generación de señales es detallado a profundidad.

3.2.4. Robustez y seguridad: clasificación en presencia de ACI

AlMakhlawaty et al.[35] se evalúan las CNN (AlexNet, VGG-16, VGG-19) en presencia de ACI y desvanecimiento, identificando a VGG-19 como la más robusta (> 95 % de precisión). En concreto, proponen un marco de clasificación basado en CNN que consta de:

1. Generación de diagramas de constelación de las señales recibidas con ACI, sobre canales AWGN, Rayleigh y Rician.
2. Renderizado de cada diagrama como imagen de 227×227 px (AlexNet) o 224×224 px (VGGNet).
3. Alimentación de las imágenes a tres arquitecturas CNN preentrenadas (AlexNet, VGG-16, VGG-19), con la capa final ajustada a 5 clases de modulación.
4. Ajuste fino de toda la red usando back-propagation para maximizar el PCC.

Los resultados de esta evaluación, que considera las tres arquitecturas CNN en diferentes canales, se consolidan en la Tabla 3.8, mostrando el notable rendimiento de VGG-19 incluso bajo interferencia.

Resumen de desempeño: ACI-CNN [35]	
Conjunto de datos	Diagramas con ACI en canales AWGN, Rayleigh y Rician; > 200,000 imágenes train, 1,000 test por modulación No se especifican frecuencias ni relaciones de muestreo en banda base.
Modulaciones	BPSK, QPSK, 8PSK, 16QAM, 64QAM
Rango de SNR	-10 a 15 dB (AWGN, Rayleigh); -10 a 13 dB (Rician)
PCC mínimo (%)	10 % (16QAM a 13 dB, canal Rician)
PCC máximo (%)	100 % (varias modulaciones a ≥ 5 dB)
Muestras/clase	1 000 imágenes de prueba por modulación

Tabla 3.8: Parámetros y resultados clave para la clasificación bajo ACI con CNN.

- **Brecha específica (ACI-CNN)** No se explora el desfase de portadora como efecto de canal en las señales procesadas. Por otro lado, requiere preprocesamiento para la generación de los diagramas aportando complejidad en el receptor.

3.2.5. Vulnerabilidades adversariales.

Sadeghi et al. [36] evidencian que las DNN son susceptibles a ataques adversariales (white-box y black-box), donde pequeñas perturbaciones inducen errores críticos de clasificación. En

el trabajo mencionado, este fenómeno es estudiado sobre el clasificador visual transformer CNN versión 2 (VT-CNN2) entrenado con el conjunto de datos GNU Radio ML RML2016.10a.

3.2.5.1. Metodología

- **Ataque white-box:** empleo de Fast Gradient Method (FGM) junto con búsqueda por bisección para generar perturbaciones específicas r_x que minimizan $\|r_x\|_2$ sujetas a clasificación errónea en VT-CNN2.
- **Perturbaciones universales (UAP):** emplea análisis de componentes principales (PCA) sobre las direcciones de gradiente normalizadas para crear un vector r capaz de engañar al modelo con alta probabilidad, independiente del input.
- **Métricas relacionadas:**
 - relación de perturbación a ruido (PNR)

$$\text{PNR(dB)} = \text{PSR(dB)} + \text{SNR(dB)}$$

- relación de perturbación a señal (PSR)

La Tabla 3.9 resume las condiciones del conjunto de datos RML2016.10a y el impacto de los ataques sobre el clasificador VT-CNN2, evidenciando cómo el rendimiento puede degradarse completamente bajo perturbaciones mínimas.

Resumen de desempeño: Adversarial Attacks [36]	
Conjunto de datos	GNU Radio ML RML2016.10a. 220 000 muestras,
Modulaciones	BPSK, QPSK, 8PSK, QAM16, QAM64, CPFSK, GFSK, PAM4, WBFM, AM-SSB, AM-DSB
Rango de SNR	-20 a 18 dB (paso 2 dB)
PCC mínimo (%)	0 % (PNR=0 dB bajo ataque)
PCC máximo (%)	100 % (sin ataque, SNR ≥ 10 dB)
Muestras/señal	256
Muestras/clase	1 000 por modulación y nivel de SNR

Tabla 3.9: Parámetros y resultados clave para los ataques adversariales en clasificación de modulación .

- **Brecha específica (Sadeghi et al. [36])** el estudio se restringe a perturbaciones sobre GNU Radio ML2016.10a en un entorno puramente simulado. En la página oficial de DeepSig [37], se indica que los conjuntos de datos *RadioML 2016.10a* y *RadioML 2018.01A* corresponden a investigaciones tempranas (2016–2017) y presentan algunas erratas conocidas. Se menciona además que actualmente no son utilizados en productos de aprendizaje profundo, y se sugiere a los investigadores generar sus propios conjuntos de datos mediante herramientas como MATLAB, GNU Radio o mediante capturas reales por aire. Esta observación no invalida su utilidad académica, pero sí sugiere precaución al emplearlos como referencia para desarrollos actuales.

3.2.6. Visión condensada de métodos de inteligencia artificial

La Tabla 3.10 ofrece una visión general de los trabajos analizados durante esta sección.

Familia	Modelo	Ventaja principal	Limitación
Estadísticos avanzados	SSC [30] KS [29]	Bajo costo computacional; pocas muestras requeridas	Desempeño decreciente a SNR muy bajo; sensible al estadístico elegido
Redes en dominio polar	Polar+CNN [32] CNN+CCN [33]	Robustez ante AWGN y desvanecimiento; compensación de amplitud/fase mejora constelaciones densas	Requiere gran volumen de datos e imágenes; decae en QAM de orden superior con bajo SNR
Híbridos y GA	Jerárquico+GA [34]	Clasificación jerárquica en tiempo real con alta PCC	Alta complejidad computacional; dependencia de los parámetros del GA
Robustez y seguridad	ACI-CNN [35] Adversarial Attacks [36]	Resistente a interferencia de canal adyacente y ataques adversariales	Incrementa complejidad y demanda defensas o entrenamiento adversarial

Tabla 3.10: Visión general de las familias de métodos de IA para clasificación automática de modulación (AMC).

3.2.7. Comparación de métodos de inteligencia Artificial

Finalmente, para facilitar una visión directa entre los distintos enfoques analizados, la Tabla 3.11 consolida los parámetros clave de cada método. Esta vista comparativa permite evaluar de manera rápida las fortalezas y debilidades de cada propuesta en función de su arquitectura, las modulaciones probadas y el rendimiento reportado.

Método	Arquitectura	Modulaciones	SNR (dB)	PCC mínimo (%)	PCC máximo (%)
SSC [30]	ANN de 3 capas	AM-DSB-TC, AM-DSB-SC, SSB, FM, PM, ASK, PSK, FSK	3, 5, 7, 10, 15, 25	73.4	100
Prueba [29]	KS Test no paramétrico KS	QAM (4, 16, 64), PSK (4, 8, 16)	0 a 20	≈ 25	≈ 100
Polar+CNN [32]	CNN (CaffeNet/AlexNet mod.)	QPSK, 8PSK, 16QAM, 64QAM	-4 a 14	74.5	100
CNN + CCN [33]	CNN + CCN (4 Conv + 3 Dense / 4 Dense)	QPSK, 8PSK, 16QAM, 64QAM	-4 a 12	25	100
Jerárquico + GA [34]	MLP jerárquico + GA	CW, AM, FM, SSB, FSK2, FSK4, PSK2, PSK4, OOK, 16QAM, 32QAM	7 a 12	93.32	99.62
ACI-CNN [35]	AlexNet/VGG-16/VGG-19 fine-tuned	BPSK, QPSK, 8PSK, 16QAM, 64QAM	-10 a 15	10	100
Adversarial Attacks [36]	VT-CNN2	BPSK, QPSK, 8PSK, QAM16, QAM64, CPFSK, GFSK, PAM4, WBFM, AM-SSB, AM-DSB	-20 a 18	0	100

Tabla 3.11: Comparación de métodos de IA para AMC.

3.2.8. Brechas y oportunidades en el contexto de la AMC

A partir de la revisión del estado del arte, se identifican las siguientes brechas y oportunidades para la clasificación automática de modulaciones:

- **Foco en modulaciones de bajo y medio orden.** SSC y KS se detienen en constelaciones simples (hasta 64-QAM); CNN+CCN y Polar+CNN tampoco superan 64-QAM. Ningún trabajo explora 128/256-QAM o modulaciones anidadas, dejando de lado la complejidad geométrica y la cercanía de símbolos en alto orden [2].
- **Ausencia de parámetros reproducibles.** Varios estudios (KS, Jerárquico+GA) omiten tasa de símbolo, roll-off, ancho de banda o herramientas de generación de señales. Esta falta de metadatos impide replicar resultados y comparar de forma adecuada y correcta diferentes arquitecturas.
- **Carencia de un generador abierto y estandarizado de conjuntos de datos.** Cada grupo/investigación sintetiza sus propias señales *a su manera*, lo que introduce sesgos y dificulta la comparación entre artículos. Es necesario un marco de trabajo de código abierto que confeccione colecciones I-Q con *metadatos completos*, parametrización de canal y soporte para modulaciones de alto orden; iniciativas recientes (p. ej. TorchSig [38]) apuntan en esa dirección, pero aún no se han adoptado como referencia común.

La literatura existente en métodos de IA se centra principalmente en entornos muy controlados (p. ej. AWGN puro o un solo tipo de desvanecimiento) y cubre modulaciones hasta 64-QAM. No

ofrece un enfoque que combine modulaciones de alto orden con validación en canales reales con varios efectos (desvanecimiento, interferencias, desfases). El presente trabajo propone cerrar estas brechas a través de:

- **Generación estandarizada de señales:** utilización de herramientas de código abierto para crear señales I-Q con metadatos detallados (tales como frecuencia de muestreo, roll-off, número de trayectorias, desfase). Garantizando así comparabilidad, trazabilidad y transparencia.
- **Escenarios realistas y variables:** definición de escenarios realistas que repliquen fielmente condiciones prácticas: rangos de variación de SNR habituales, multipropagación y desfases.
- **Clasificación de modulaciones de muy alto orden:** desarrollo de arquitecturas capaces de discriminar modulaciones de bajo orden (como *BPSK/QPSK*) así como aquellas de muy alto orden ($> 64\text{QAM}$) a la vez.

Con este enfoque, nuestro trabajo cierra las brechas clave identificadas en AMC: cubrimos modulaciones de muy alto orden, garantizamos reproducibilidad y estandarización de los datos y evaluamos un amplio espectro de condiciones de canal.

4. Metodología

A continuación se describen los recursos, el contexto y el flujo metodológico seguido en esta investigación.

4.1. Contexto y alcance

En un sistema de comunicación digital, el equipo que realiza la AMC se ubica en el receptor, como una etapa previa al demodulador. Su función es identificar el esquema de modulación empleado en la señal recibida, con el fin de que el demodulador pueda adaptarse dinámicamente y recuperar correctamente la información transmitida.

La señal que alimenta al clasificador corresponde a una versión preprocesada en banda base, afectada por fenómenos comunes del canal como AWGN, desvanecimiento por trayectorias múltiples y desfase de portadora. Este trabajo asume que la conversión a banda base y la sincronización en frecuencia ya han sido realizadas, por lo que el clasificador opera directamente sobre muestras I-Q complejas.

La Figura 4.1 muestra el esquema de recepción considerado, donde se observa la ubicación del módulo de clasificación, foco de este trabajo. El estudio se limita exclusivamente al problema de AMC a partir de señales en banda base, sin abordar el diseño del demodulador ni otras etapas de recuperación de información.

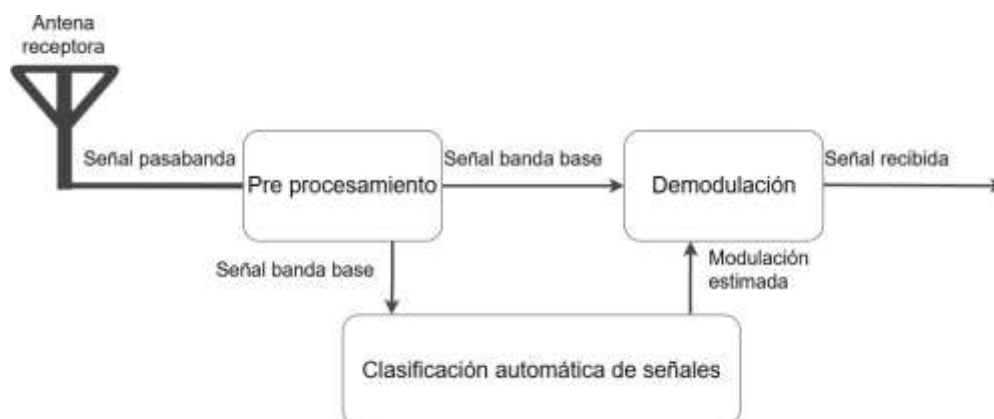


Figura 4.1: Arquitectura de recepción con clasificador automático de señales.

4.2. Recursos y herramientas

4.2.1. Entorno de desarrollo local

Para la generación, almacenamiento y procesamiento del conjunto de datos utilizado en esta investigación, se empleó el siguiente entorno computacional y conjunto de herramientas:

- **Biblioteca principal para la generación de señales.** Se utilizó TorchSig v1.1.0 [39] como herramienta principal para la generación de señales I-Q moduladas digitalmente y la aplicación de distorsiones típicas de canal (ruido, desvanecimiento, desfase). Esta biblioteca fue instalada y ejecutada localmente en Ubuntu 22.04.5 LTS.
- **Librerías complementarias en Python:**
 - `numpy` 2.2.5 – operaciones vectorizadas y manipulación de matrices.
 - `matplotlib` 3.10.1 – representación gráfica de señales, constelaciones.
 - `h5py` 3.13.0 – creación y manejo de archivos HDF5.
- **Formato HDF5:** contenedor binario autocontenible para múltiples archivos de datos.

4.2.2. Entorno de cómputo en la nube

El desarrollo y evaluación de los modelos se apoyaron en las siguientes plataformas y herramientas:

- **Google Colab** – plataforma de cómputo en la nube con acceso a GPU/TPU sin necesidad de configuración local [40]. En este, se empleó principalmente la tarjeta NVIDIA Tesla T4 – GPU con 16 GB de memoria GDDR6 y soporte CUDA.
- **Kaggle** – plataforma para almacenamiento, publicación y distribución de conjuntos de datos [41].
- **Librerías Python**
 - `pandas` 2.2.2 – manipulación y análisis de datos estructurados.
 - `scikit-learn` 1.6.1 – preprocesamiento, codificación de etiquetas, particionado y validación cruzada.
 - `TensorFlow` 2.18.0 – marco de trabajo para desarrollo de redes neuronales con API Keras.

4.2.3. Automatización y Gestión en la Nube

El flujo de trabajo de datos se orquestó en Kaggle mediante APIs, permitiendo:

- Descarga y versionado de datasets.

- Cacheo de archivos HDF5.
- Integración con cuadernos interactivos (formato `.ipynb`).

4.3. Generación y preparación del conjunto de datos

Previamente al uso de la biblioteca `TorchSig`, se evaluaron diversos datasets públicos como *RadioML2016.10a*, *RadioML2018.01A* y uno especializado en señales cicloestacionarias. Sin embargo, se optó por `TorchSig` debido a las limitaciones de estos conjuntos de datos en cuanto al control de las condiciones de los parámetros del canal.

Además, `TorchSig` es una herramienta de libre acceso y código abierto, desarrollada con el propósito de brindar mayor flexibilidad al trabajar con señales sintéticas. Permite definir distintos esquemas de modulación, niveles de ruido, condiciones de desvanecimiento e incluso desfases, ofreciendo un entorno controlado para la experimentación. Además, se encuentra en constante actualización por parte de su comunidad, incorporando progresivamente nuevos efectos de canal, lo que amplía las posibilidades de simulación. Esta capacidad de personalizar escenarios específicos resulta especialmente valiosa para investigaciones donde se requiere replicar y explorar el comportamiento de los clasificadores en condiciones particulares y reproducibles, algo que los conjuntos de datos públicos es difícil o imposible de conseguir.

El conjunto de datos final se generó mediante simulación utilizando `TorchSig`, aplicando modelos de canal en banda base compleja. Aunque en escenarios reales los efectos del canal operan sobre señales en pasabanda, esta aproximación simplifica el análisis al representar desvanecimiento, desfase y ruido directamente sobre la señal I-Q en banda base.

La Figura 4.2 muestra en resumen la lógica de generación del conjunto de datos.

4.3.1. Definición de esquemas de modulación

Se consideraron seis esquemas de modulación digital empleados en comunicaciones inalámbricas: *BPSK*, *QPSK*, *16-QAM*, *32-QAM Cruzada*, *64-QAM* y *128-QAM Cruzada*.

Además, para evaluar el impacto de una constelación de muy alta densidad, se generó un conjunto de datos adicional incluyendo la modulación *256-QAM*, formando así un total de siete modulaciones. Esta modulación se analizó exclusivamente en el escenario base el cual será introducido en la Sección 4.3.3.

Estos esquemas cubren desde modulaciones binarias hasta constelaciones de alta densidad.

4.3.2. Parámetros de generación

La longitud de cada señal simulada se fijó en **4096 muestras**. Dado que se utiliza un muestreo de **8 muestras por símbolo**, se tienen $\frac{4096}{8} = 512$ símbolos por señal. Estos valores se justifican:

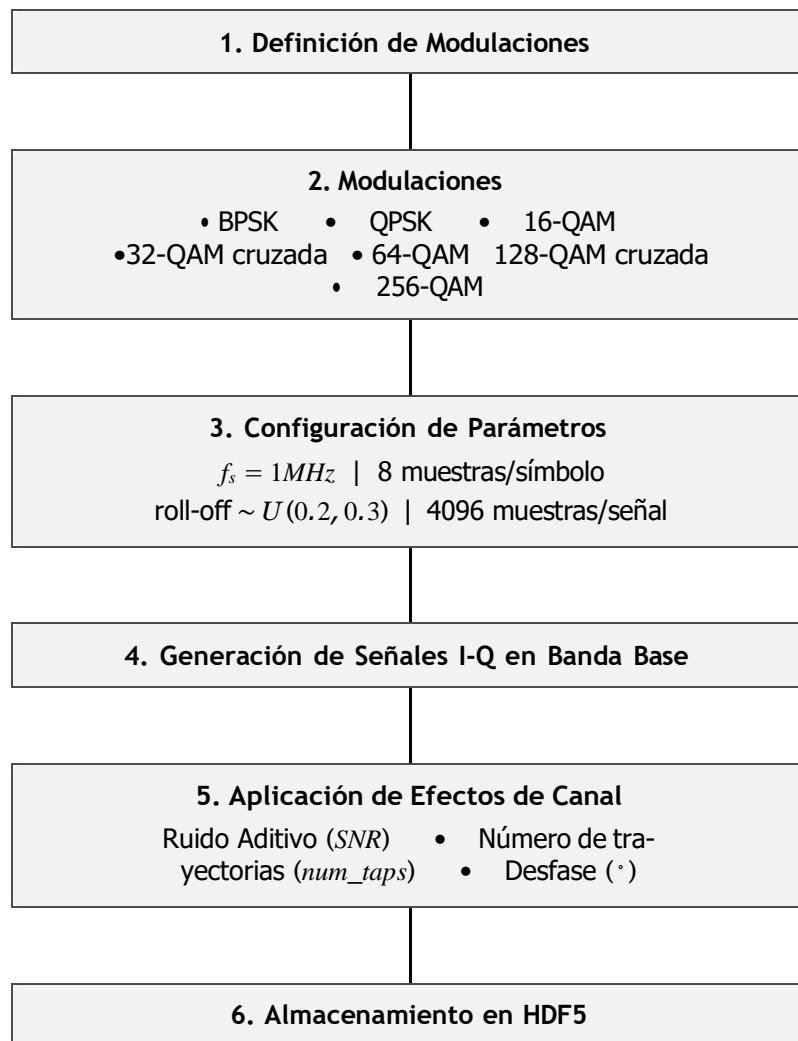


Figura 4.2: Metodología concisa para la generación, canalización y almacenamiento de señales moduladas.

- Cobertura de la constelación.** En modulaciones de alto orden, tales como 128-QAM o 256-QAM, una señal demasiado corta podría no contener ejemplos suficientes de cada símbolo de la constelación. Con 512 símbolos, la probabilidad de que cada símbolo posible aparezca al menos una vez es muy alta, garantizando que la red neuronal observe la geometría completa de la constelación durante el entrenamiento. Esta idea se puede demostrar desde el enfoque probabilístico. Supondremos que los símbolos son seleccionados de forma aleatoria, uniforme e independiente, y que existen M posibles símbolos en la constelación. Cada símbolo tiene entonces una probabilidad de aparición de:

$$p = \frac{1}{M}$$

Si tomamos una secuencia de n símbolos, la probabilidad de que un símbolo específico no aparezca en toda la secuencia es:

$$P(\text{símbolo no aparezca}) = 1 - \frac{1}{M}^n$$

Por lo tanto, la probabilidad de que ese símbolo aparezca al menos una vez es:

$$P(\text{símbolo aparece}) = 1 - \left(1 - \frac{1}{M}\right)^n$$

Como hay M símbolos posibles, la esperanza matemática del número de símbolos únicos que se observarán en una secuencia de longitud n es:

$$E[\text{símbolos únicos}] = M \cdot \left(1 - \left(1 - \frac{1}{M}\right)^n\right)$$

Tabla 4.1: Símbolos únicos esperados y cobertura de constelación según tamaño y longitud de la señal.

Modulación	128 símbolos		256 símbolos		512 símbolos	
	Únicos	Cobertura [%]	Únicos	Cobertura [%]	Únicos	Cobertura [%]
64-QAM	55.47	86.7	62.9	98.3	64.0	99.9
128-QAM cruzada	81.1	63.3	110.8	86.6	125.7	98.2
256-QAM	100.9	39.4	162.0	63.3	221.0	86.5

En la Tabla 4.1 se presenta el valor esperado del número de símbolos únicos observados, así como su cobertura porcentual respecto al total de símbolos posibles en la constelación, en función del tamaño de la señal y del esquema de modulación utilizado. Se observa que al utilizar señales de 4096 muestras, equivalentes a 512 símbolos, se alcanza una cobertura de hasta el 86.5 % incluso en modulaciones de alta densidad como 256-QAM, lo que representa una proporción elevada de símbolos. Consideramos que este nivel de cobertura es suficiente para que las redes neuronales puedan aprender de constelaciones complejas de forma adecuada.

En longitudes menores, como 1024 o 2048 muestras por señal, ciertos símbolos de la constelación podrían omitirse, disminuyendo la capacidad de generalización del clasificador. En la Figura 4.3 se muestra que, dependiendo de la longitud de la señal, existen símbolos que no aparecen en modulaciones de alto orden.

- **Precisión temporal.** Un factor de muestreo de 8 muestras/símbolo permite representar con mayor resolución la forma temporal de las señales, capturando con mayor resolución la forma de onda modulada.

El parámetro de **roll-off** α del filtro de forma de pulso se tomó aleatoriamente de una distribución uniforme

$$\alpha \sim U(0.2, 0.3),$$

valor que coincide con los rangos habitualmente empleados en sistemas de comunicación reales. Al variar aleatoriamente α en cada secuencia, se introduce diversidad en la densidad espectral de las señales, reforzando la robustez del clasificador y minimizando el riesgo de sobreajuste a un único filtro.

La Tabla 4.2 resume todos los parámetros no aleatorios utilizados en la generación de señales moduladas.

Tabla 4.2: Parámetros de configuración de señales moduladas

Parámetro	Valor
Frecuencia de muestreo	1 MHz
Muestras por símbolo	8
Longitud por señal	4096 muestras (512 símbolos)

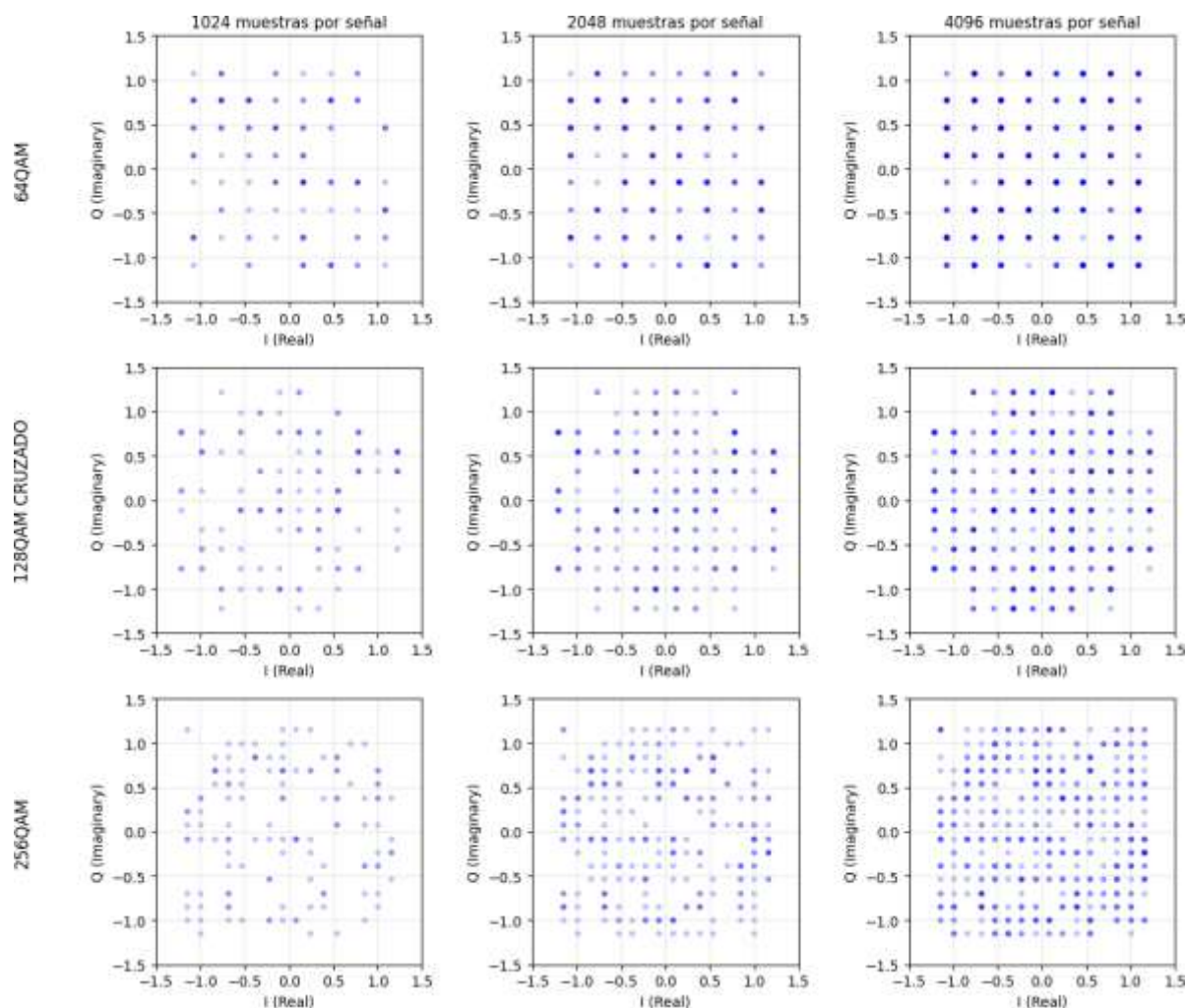


Figura 4.3: Impacto de la longitud de señal en la ocurrencia de símbolos para modulaciones de alto orden.

4.3.3. Escenarios de canal

Se definieron 11 escenarios univariados agrupados según el parámetro variado: *SNR*, Número de trayectos múltiples y desfase. La Tabla 4.3 muestra los escenarios considerados divididos en 3 grupos principales. En cada grupo se especifica el parámetro que varía y aquellos que se mantienen fijos.

Tabla 4.3: Escenarios univariados: se varía un solo parámetro mientras los otros se mantienen fijos.

Escenario	Parámetro	Rango	Descripción
Variación de SNR		<i>(Número de trayectos = 2, Desfase = 0°-45°)</i>	
1	SNR	0-4 dB	Ruido extremo
2	SNR	4-8 dB	Conectividad baja
3	SNR	8-12 dB	Referencia media — <i>Escenario base</i>
4	SNR	12-16 dB	Buena conectividad
5	SNR	16-20 dB	Alta calidad
Variación de número de trayectos		<i>(SNR = 8-12 dB, Desfase = 0°-45°)</i>	
6	Número de trayectos	1	Canal ideal sin reflexiones
7	Número de trayectos	4	Multipropagación moderada
8	Número de trayectos	6	Multipropagación severa
Variación de desfase		<i>(SNR = 8-12 dB, Número de trayectos = 2)</i>	
9	Desfase	45°-90°	Parcial
10	Desfase	90°-135°	Intermedio
11	Desfase	135°-180°	Antifase

La Figura 4.4 presenta de forma comparativa las constelaciones I-Q en banda base, mostrando el caso ideal frente al escenario con distorsiones introducidas por el canal (SNR = 10 dB, Número de trayectos = 2 y desfase = 45°).

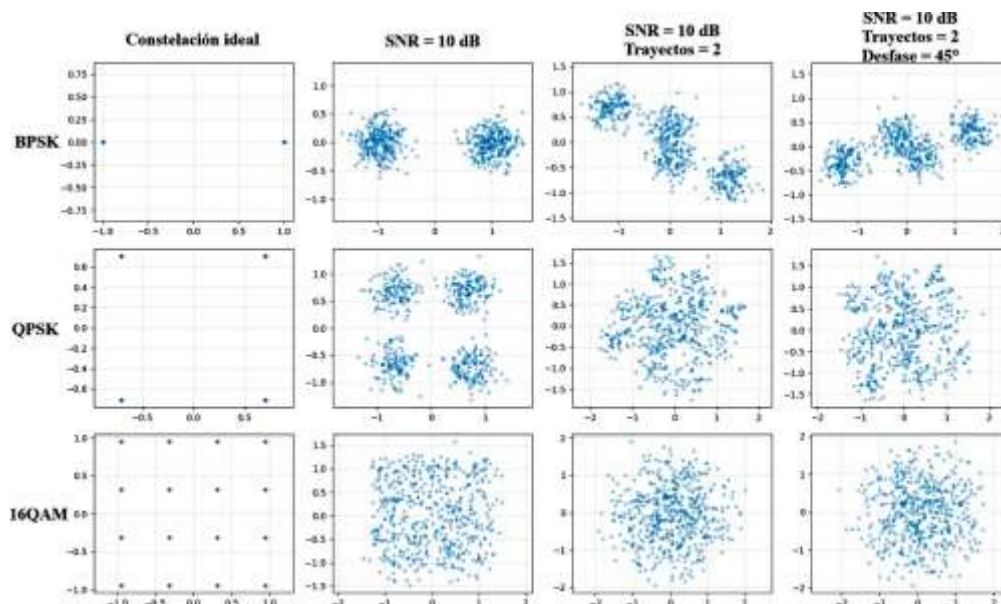


Figura 4.4: Constelación de BPSK, QPSK y 16-QAM (filas) en cuatro condiciones de canal (columnas): (1) señal ideal; (2) con ruido (SNR = 10dB); (3) con ruido y desvanecimiento (2 trayectos); (4) con ruido, desvanecimiento y desfase (45°).

Una de las ventajas de TorchSig es su capacidad para definir directamente el perfil de potencia del canal mediante un vector de retardo (`power_delay_profile`). El perfil de potencia de

trayectos empleado sigue un decaimiento exponencial de 3 dB por cada trayecto (tap), conforme a la recomendación ITU-R P.1238-7 de comunicaciones inalámbricas. En escala lineal, cada tap sucesivo tiene la mitad de la potencia del anterior, lo que da los valores normalizados [1, 0.5, 0.25, 0.125, 0.06, 0.03]. La generación de señales con este perfil produce una respuesta en magnitud que refleja la dispersión temporal típica de un canal selectivo en frecuencia.

Es importante recalcar que el parámetro roll-off (α) es aplicado por igual a todas las señales en detrimento del escenario.

La Figura 4.5 muestra la magnitud promedio de los taps obtenida tras generar un total de 4096 señales I-Q. Esta visualización permite verificar la correcta aplicación del perfil de potencia y el comportamiento estadístico del canal simulado.

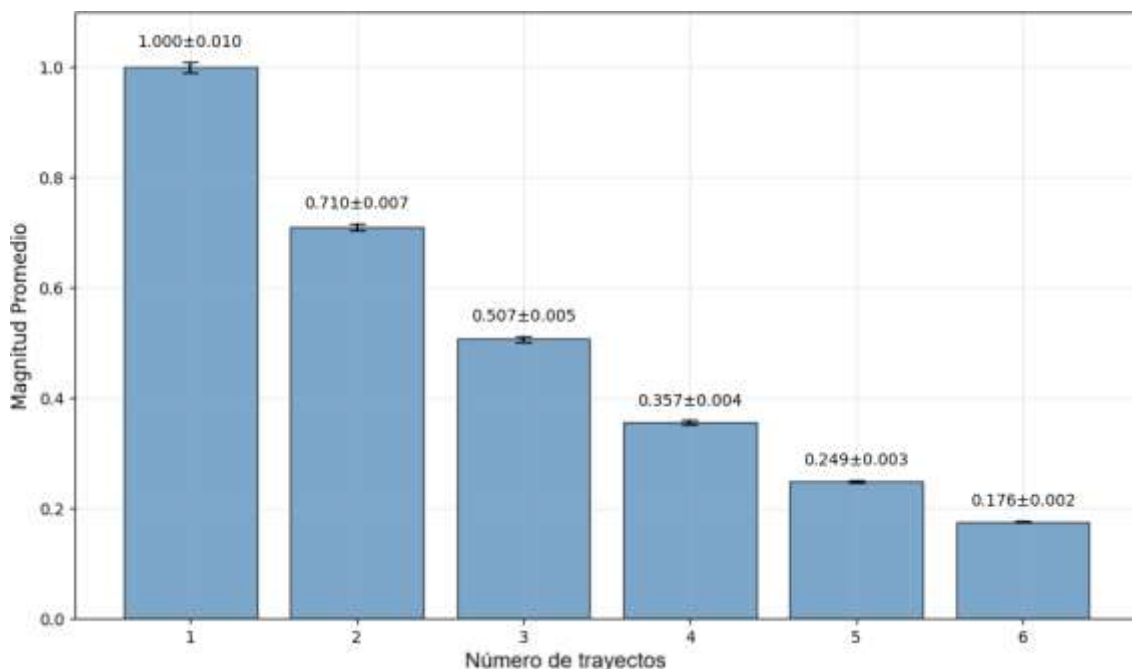


Figura 4.5: Magnitud promedio de 6 taps en un canal multipropagación con perfil de potencia [1, 0.5, 0.25, 0.125, 0.06, 0.03], calculada con TorchSig sobre 4096 señales I-Q.

4.3.4. Conjuntos de Datos

En este proyecto de investigación se evalúa el desempeño de los clasificadores mediante el ePCC, equivalente a la métrica "accuracy" en la terminología de aprendizaje automático. Este se define como:

$$\text{PCC} = \frac{\text{\#clasificaciones correctas}}{\text{\#total de señales}} \times 100\%$$

Para optimizar la validación, el entrenamiento y la evaluación de las arquitecturas, se definieron tres **conjuntos de datos** independientes que varían en sus tamaños:

1. Conjunto 1: Validación temprana.

- **Tamaño:** $N = 4096$ señales por modulación. Teniendo en total 24576 señales para cada escenario.
- **División:** `StratifiedKFold` ($k = 5$, 4 repeticiones): 90 % entrenamiento/validación, 10 % test.
- **Propósito:** brindar una primera valoración del desempeño de las arquitecturas con un subconjunto reducido pero suficientemente representativo para guiar las siguientes etapas sin agotar recursos.

2. Conjunto 2: Entrenamiento exhaustivo.

- **Tamaño:** $N = 8192$ señales por modulación. Teniendo en total 49152 señales para cada escenario.
- **Propósito:** uso exclusivo para ajuste de pesos y validación interna; sin fase de test.
- **División:** 80% para entrenamiento y 20% para validación interna.

3. Conjunto 3: Prueba final.

- **Tamaño:** $N = 1024$ señales por modulación. Teniendo en total 6144 señales para cada escenario.
- **Propósito:** Realizar una evaluación comparativa del desempeño de los modelos, en términos de PCC y $loss$, utilizando el *Conjunto 2*. Este conjunto está destinado únicamente a evaluación.

Para el experimento comparativo con siete modulaciones, se emplearon las mismas estructuras de Conjunto 2 y Conjunto 3 ampliadas a siete clases (seis originales + 256-QAM):

- **Escenario de evaluación:** Se limita al escenario base (SNR 8–12 dB, Número de trayectos=2, desfase 0–45°).
- **Entrenamiento (variante 7 clases):** $N = 8192$ señales por cada una de las 7 modulaciones, conserva el protocolo del Conjunto 2.
- **Prueba (variante 7 clases):** $N = 1024$ señales por cada una de las 7 modulaciones, conserva el protocolo del Conjunto 3.
- **Propósito:** Comparar de forma directa el PCC y $loss$ al clasificar 6 vs. 7 modulaciones bajo idénticas condiciones de canal.

4.3.5. Estructura de Almacenamiento y Etiquetado

Cada conjunto se almacenó en un archivo HDF5 que permite almacenar distintos datasets en un solo archivo. Teniendo así los siguientes datasets para cada escenario:

- **X:** Arreglo de señales I-Q con forma $(K \cdot N, 4096, 2)$, donde el número K corresponde a la cantidad de clases distintas disponibles, y cada muestra contiene una secuencia compleja representada por sus componentes en cuadratura y fase.

- **Y:** Etiquetas de modulación codificadas en formato one-hot, arreglo de forma $(K \cdot N, K)$. Cada fila representa una de las siguientes K modulaciones digitales, en nuestro estudio principal ($K = 6$):

Modulación	Vector one-hot
BPSK	[1, 0, 0, 0, 0, 0]
QPSK	[0, 1, 0, 0, 0, 0]
16QAM	[0, 0, 1, 0, 0, 0]
32QAM cruzada	[0, 0, 0, 1, 0, 0]
64QAM	[0, 0, 0, 0, 1, 0]
128QAM cruzada	[0, 0, 0, 0, 0, 1]

- **Z:** Valor promedio de SNR (en dB), representado como un arreglo unidimensional $(K \cdot N,)$. Este valor permite identificar el escenario de canal correspondiente. Por ejemplo, para el escenario 3 con SNR en el rango [8, 12] dB, la etiqueta Z toma el valor 10.
- **Dataset Effects:** Conjunto de parámetros de canal aplicados a cada muestra, incluyendo: SNR exacto, número de trayectorias (Número de trayectos), desfase exacto y valor de roll-off.

4.4. Validación Temprana de Arquitecturas

Con el *Conjunto 1*, se evaluaron múltiples arquitecturas de red, registrando las métricas de *PCC* y *loss*, y aplicando análisis de intervalos de confianza del 95 % para una evaluación estadística más robusta. Durante esta etapa exploratoria, se experimentó con diversas arquitecturas y combinaciones de hiperparámetros, afinando progresivamente las redes a partir de sus resultados preliminares. Finalmente, se seleccionaron para entrenamiento exhaustivo aquellas arquitecturas que cumplían el criterio de aceptación (Sección 4.4.1).

4.4.1. Criterio de aceptación

Una arquitectura avanza a la fase exhaustiva si, en el escenario base (SNR = 8 – 12dB, num_taps = 2, desfase = 0 – 45°), alcanza un PCC ≥ 92 % y un *loss* ≤ 0.30 .

- El umbral de PCC (92 %) se fijó tras revisar la evolución de los enfoques más recientes *deep learning* para AMC: estas superan regularmente el 92 % en condiciones intermedias de canal (canal AWGN con SNR=10 dB).
- El umbral de *loss* menor que 0.30 se definió como condición de desempeño mínimo, validando la capacidad del modelo para generalizar sin caer en sobreajuste.

4.5. Diseño de Arquitecturas Neuronales

Durante la etapa de validación temprana, se exploró un amplio conjunto de arquitecturas neuronales, abarcando múltiples combinaciones de bloques convolucionales, residuales, recurrentes y de atención. A partir de los resultados, se identificaron aquellas configuraciones con mejor desempeño y mayor potencial de generalización.

En esta sección se presentan las tres arquitecturas seleccionadas para entrenamiento exhaustivo y análisis comparativo. El código completo del proyecto se encuentra disponible en el repositorio [42].

4.5.1. Red CNN-BiLSTM-Attention

La Figura 4.6 muestra el esquema general de esta arquitectura, que combina bloques convolucionales, una capa Bi-LSTM y un mecanismo de atención personalizado para clasificación de señales I-Q. A continuación, se detallan sus componentes:

- **Entrada I-Q:** Tensor de forma (4096, 2) correspondiente a 4096 muestras de señales complejas I-Q (canales I y Q).
- **Bloques convolucionales:** Tres capas `Conv1D` configuradas con filtros [64, 128, 128], kernels [7, 5, 3] y `strides` [2, 1, 1], cada una seguida de `BatchNormalization`. Después de la segunda convolución se aplica `MaxPooling1D` con tamaño de ventana 2.
- **Capa Bi-LSTM:** Una capa bidireccional `LSTM` con 256 unidades, que procesa la secuencia extraída por el front-end convolucional.
- **Mecanismo de atención:** Se emplea una capa personalizada `AttentionPool` con 128 unidades, que calcula una representación global de la secuencia ponderando cada paso temporal según su relevancia. Esto se realiza mediante una proyección no lineal (`Dense + tanh`), seguida de una puntuación escalar (`Dense(1)`), normalización con `softmax` en el tiempo y una suma ponderada de los vectores de entrada. Este mecanismo de atención tiene la estructura mostrada en la Figura 4.7
- **Capa densa intermedia:** Capa `Dense` con 128 unidades y activación `ReLU`, seguida de `Dropout` con tasa 0.35 para regularización.
- **Capa de salida:** Capa `Dense` con 6 unidades y activación `softmax` para clasificación multiclase.

4.5.2. Red Inception-Res-SE-Attn

La Figura 4.8 muestra el esquema general de esta arquitectura, que combina bloques inception, conexiones residuales y un bloque Multi-Head Attention para clasificación de señales I-Q. A continuación, se detallan sus componentes:

- **Entrada I-Q:** Entrada de forma (4096, 2) que representa señales I-Q en dos canales.

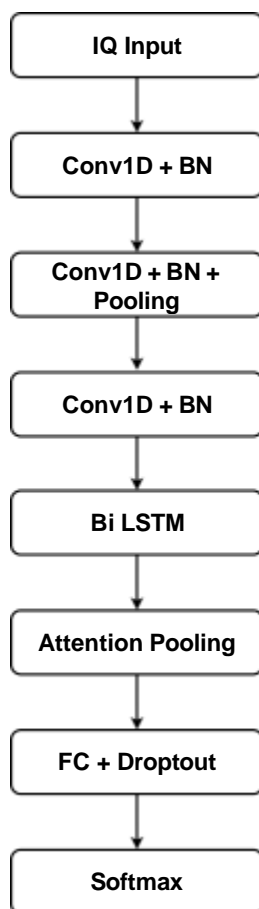


Figura 4.6: Esquema general de la arquitectura CNN-BiLSTM-Attention.

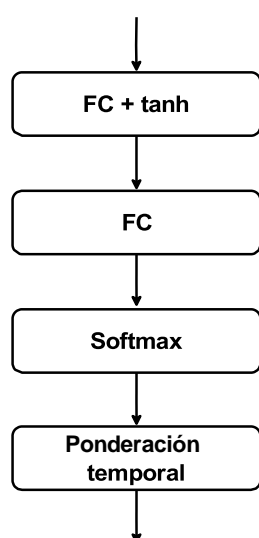


Figura 4.7: Esquema de la arquitectura del bloque Attention Pooling.

- **Bloque inicial (Conv1D + BN + Pooling):** Capa Conv1D con 32 filtros, kernel de tamaño 8 y *stride* de 2, seguida de una normalización por lotes (BatchNormalization) y MaxPooling1D con tamaño de ventana 2.
- **Bloques Inception + ResNet + SE (×2):** Cada bloque aplica tres convoluciones paralelas Conv1D con 32 filtros y kernels de tamaño 4, 6 y 8, regularizadas con $L2=2e-4$. Las salidas se concatenan y normalizan, seguidas de un bloque *Squeeze-and-Excitation*. Se incorpora una conexión residual, seguida de activación ReLU. Este bloque se repite dos veces y su estructura se encuentra en la Figura 4.9.
- **Bloque de Atención:** Capa MultiHeadAttention con 2 cabezas y $key_dim=16$, precedida de normalización (LayerNormalization), seguida de Dropout con tasa 0.1 y conexión residual.
- **Global Average Pooling:** Reducción temporal de dimensionalidad mediante una capa de tipo GlobalAveragePooling1D que extrae un vector por canal.
- **Capa densa intermedia:** Capa Dense con 14 unidades y activación ReLU, regularizada con $L2=2e-4$, seguida de LayerNormalization y Dropout con tasa 0.3.
- **Salida Softmax:** Capa Dense con 6 unidades y activación softmax para clasificación multiclase.

4.5.3. Red Res-TCN-SE

La Figura 4.10 muestra el esquema general de esta arquitectura, que combina conexiones residuales, convoluciones dilatadas y un mecanismo de atención personalizado para clasificación de señales I-Q. A continuación, se detallan sus componentes:

- **I-Q Input:** Tensor de forma (4096, 2) que representa la señal compleja en formato I-Q, con 4096 muestras y dos canales (I y Q).
- **Inception + BN + Pooling:** Bloque Inception-1D compuesto por tres convoluciones paralelas con kernels de tamaños 3, 5 y 7, cada una con 64 filtros y activación ReLU. Las salidas se concatenan y se aplica BatchNormalization, seguido de MaxPooling1D con tamaño 4.
- **Bloque ResDilated ×2:** Dos bloques residuales con convoluciones dilatadas:
 - Primer bloque: 128 filtros, dilatación 1.
 - Segundo bloque: 256 filtros, dilatación 2.

Cada bloque incluye dos Conv1D, normalización, conexión residual (con proyección si es necesario) y activación ReLU. Al final de cada bloque se aplica MaxPooling1D.

- **Transformer Encoder:** Este bloque, presentado en la Figura 4.11, implementa la codificación basada en atención propia. Recibe una secuencia y primero aplica una capa de MultiHeadAttention con 2 cabezas y dimensión de clave 16, seguida de Dropout(0.1).

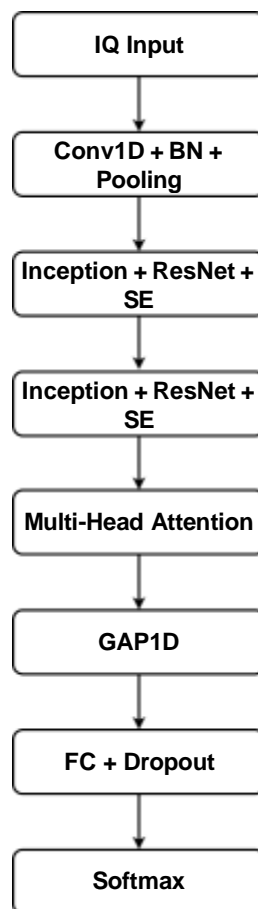


Figura 4.8: Esquema general de la arquitectura Inception-Res-SE-Attn.

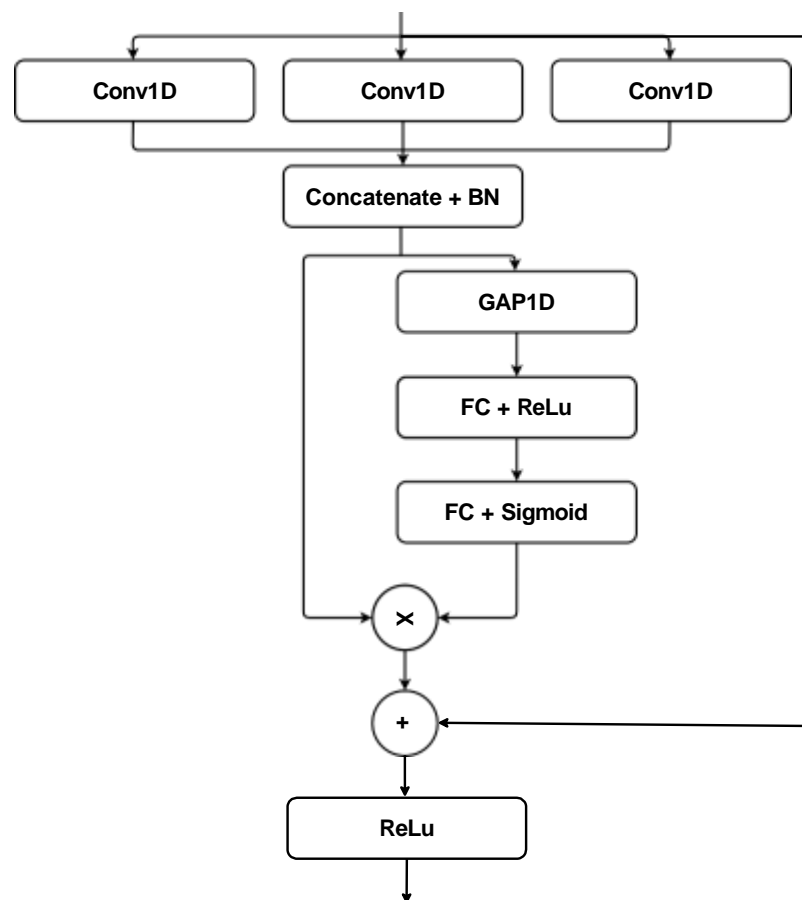


Figura 4.9: Esquema de la arquitectura del bloque Inception + ResNet + SE.

Luego, la salida se combina con la entrada original mediante una suma residual, y se normaliza con `LayerNormalization`. A continuación, se aplica una red feedforward (`Dense(256, ReLU) → Dense(units)`), cuya salida también se suma residualmente a su entrada y se normaliza. Este bloque permite modelar relaciones de largo alcance en la secuencia de forma eficiente.

- **GAP1D:** Reducción de la dimensión temporal mediante `GlobalAveragePooling1D`, que produce un vector resumen representativo de toda la secuencia procesada.
- **FC + ReLU + Dropout:** Capa totalmente conectada con 256 unidades y activación `ReLU`, seguida de `Dropout` con probabilidad de 0.3 para regularización.
- **Softmax:** Capa densa final con 6 unidades y activación `softmax`, que entrega la distribución de probabilidad sobre las clases.

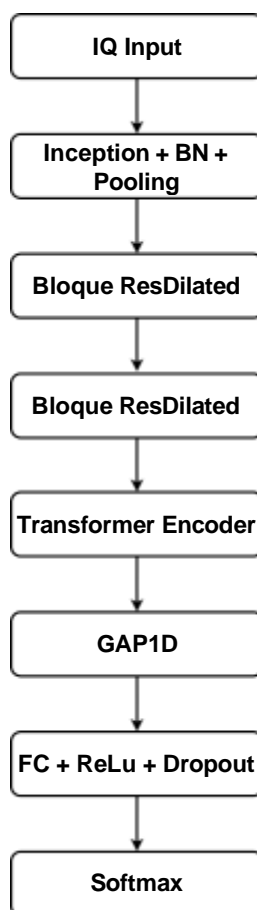


Figura 4.10: Esquema general de la arquitectura Res-TCN-SE.

4.6. Entrenamiento y Validación Exhaustiva

A partir del proceso de validación temprana, se definieron los hiperparámetros de entrenamiento que se aplicaron de forma uniforme a todas las arquitecturas seleccionadas. En parti-

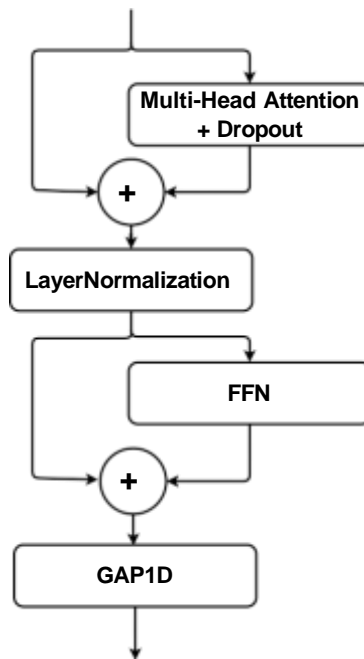


Figura 4.11: Esquema de la arquitectura del bloque Transformer Encoder.

cular, se utilizó el optimizador *Adam* con una tasa de aprendizaje inicial de 0.001 y un tamaño de lote (*batch size*) de 32.

Para mejorar la estabilidad del entrenamiento y mitigar el sobreajuste, se implementaron las siguientes estrategias de control:

- **EarlyStopping:** con un criterio basado en `val_accuracy` y una paciencia de 10 épocas.
- **ReduceLRonPlateau:** monitorizando `val_loss`, con un factor de reducción de 0.2, paciencia de 6 épocas, `cooldown` de 2, `min_delta` de 1×10^{-4} , y una tasa mínima de aprendizaje de 2×10^{-5} .

Este esquema de entrenamiento fue aplicado utilizando el *Conjunto 2*, el cual contiene una mayor cantidad de muestras, con el objetivo de realizar un reentrenamiento más robusto. Para cada arquitectura, se ejecutaron 10 entrenamientos independientes, cada uno con una semilla distinta, a fin de evaluar la estabilidad y consistencia del rendimiento del modelo bajo distintas condiciones iniciales.

4.7. Selección del Modelo Final

Como se mencionó previamente en la Sección 4.6, para cada arquitectura se realizaron 10 repeticiones de entrenamiento independientes, variando la semilla de inicialización. Todos los modelos fueron entrenados con el *Conjunto 2* y evaluados posteriormente con el *Conjunto 3*.

En lugar de seleccionar un único modelo por arquitectura, el entregable final está compuesto por los k mejores modelos de cada arquitectura, seleccionados en función de la métrica *PCC*

obtenida en la evaluación. Estos modelos conforman un comité que opera mediante un sistema de votación por mayoría, donde la modulación seleccionada se determina a partir de las predicciones emitidas por los $3k$ modelos seleccionados.

El valor de k se determina de forma experimental, buscando el punto óptimo en el que la combinación de múltiples modelos mejora la precisión y robustez del sistema.

Para la votación del comité, cada modelo seleccionado aporta un voto con igual peso, independientemente de su arquitectura. Aunque sería posible implementar un esquema de ponderación basado en el PCC promedio de cada arquitectura, en este trabajo optamos por la votación uniforme para preservar la diversidad de inicializaciones y simplificar el procedimiento.

En futuros trabajos, se podría explorar un sistema de voto ponderado—por ejemplo, asignando a cada modelo un peso proporcional a su PCC medio—con el fin de potenciar la contribución de los mejores modelos y, potencialmente, elevar aún más el rendimiento global del comité.

La Figura 4.12 muestra un esquema del sistema propuesto de votación entre modelos.

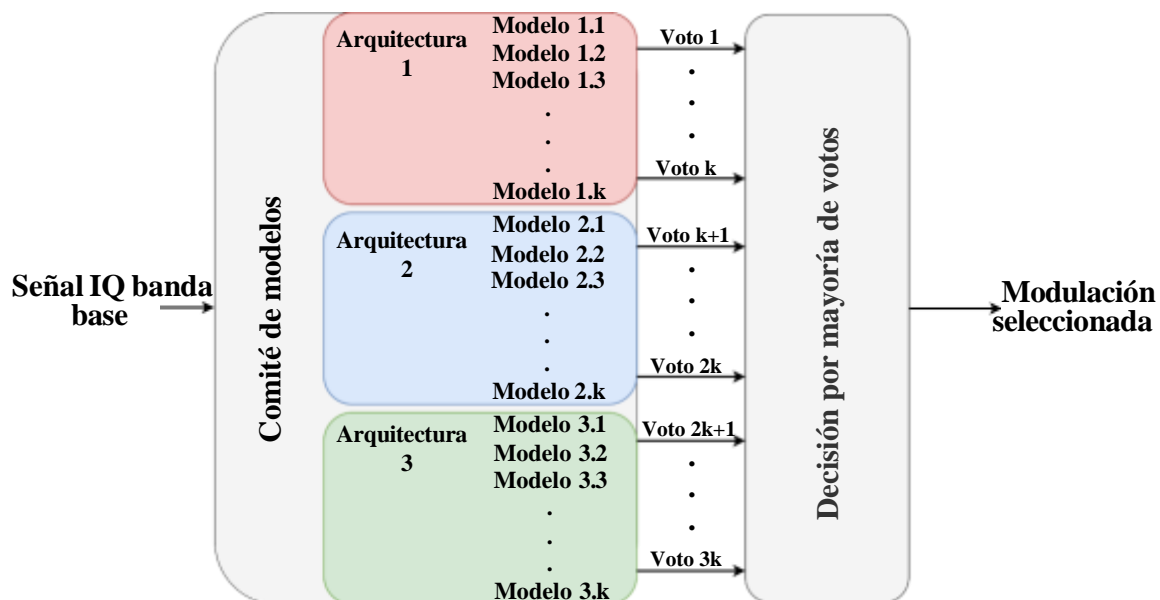


Figura 4.12: Esquema del sistema de votación con múltiples modelos por arquitectura.

5. Resultados y Discusión

En este capítulo se presentan los resultados obtenidos en las cuatro fases definidas en la Metodología: (i) validación temprana, (ii) entrenamiento y validación exhaustiva, (iii) experimento con siete modulaciones, y (iv) sistema de votación por comité.

5.1. Validación temprana de arquitecturas

En esta sección realizamos una validación temprana de tres arquitecturas frente a los once escenarios univariados definidos en la Metodología (Tabla 4.3.3).

Se evalúa el impacto aislado de cada parámetro —SNR, número de trayectos y desfase— sobre el desempeño de los modelos, midiendo PCC y pérdida promedio. El criterio de aceptación se aplica a las arquitecturas antes de la fase de entrenamiento y validación exhaustiva.

5.1.1. Desempeño global frente a parámetros de canal

Para evaluar el impacto de los principales parámetros de canal, se realizaron 20 ejecuciones independientes por cada condición univariable. La Tabla 5.1 presenta la exactitud (%) y la pérdida media (loss) con su desviación estándar para cada arquitectura y cada parámetro de canal.

Tabla 5.1: Resultados de validación temprana: exactitud (%) y pérdida (loss) media \pm desviación estándar, 20 corridas por condición univariable.

Condición	CNN-BiLSTM-Attn		Inception-Res-SE-Attn		Res-TCN-SE	
	Acc (%)	Loss	Acc (%)	Loss	Acc (%)	Loss
1. Variación de SNR ($\text{num_taps}=2$, $\text{phase_offset}=0-45^\circ$)						
0-4 dB	79.50 \pm 1.05	0.456 \pm 0.044	60.07 \pm 0.92	0.868 \pm 0.022	69.80 \pm 5.62	1.019 \pm 0.377
4-8 dB	88.98 \pm 0.69	0.333 \pm 0.110	86.28 \pm 3.24	0.457 \pm 0.140	86.03 \pm 1.09	0.526 \pm 0.164
8-12 dB	96.10 \pm 1.09	0.148 \pm 0.048	94.43 \pm 1.42	0.286 \pm 0.072	94.25 \pm 0.53	0.218 \pm 0.038
12-16 dB	98.10 \pm 0.49	0.066 \pm 0.020	97.74 \pm 0.85	0.130 \pm 0.039	95.95 \pm 1.44	0.144 \pm 0.053
16-20 dB	99.09 \pm 0.22	0.032 \pm 0.009	98.75 \pm 0.49	0.095 \pm 0.029	97.42 \pm 0.66	0.088 \pm 0.027
2. Variación de multitrayectorias (SNR=8-12 dB, $\text{phase_offset}=0-45^\circ$)						
1 tap	98.15 \pm 0.39	0.069 \pm 0.019	96.86 \pm 1.26	0.156 \pm 0.036	96.22 \pm 0.85	0.154 \pm 0.030
2 taps	96.10 \pm 1.09	0.148 \pm 0.048	94.43 \pm 1.42	0.286 \pm 0.072	94.25 \pm 0.53	0.218 \pm 0.038
4 taps	86.94 \pm 1.64	0.381 \pm 0.103	79.85 \pm 8.53	0.724 \pm 0.341	85.07 \pm 1.49	0.423 \pm 0.104
6 taps	80.58 \pm 1.64	0.499 \pm 0.172	67.44 \pm 6.62	0.824 \pm 0.175	72.90 \pm 7.24	0.972 \pm 0.477
3. Variación de desfase (SNR=8-12 dB, $\text{num_taps}=2$)						
0-45°	96.10 \pm 1.09	0.148 \pm 0.048	94.43 \pm 1.42	0.286 \pm 0.072	94.25 \pm 0.53	0.218 \pm 0.038
45-90°	95.59 \pm 1.46	0.173 \pm 0.068	94.48 \pm 1.00	0.256 \pm 0.055	94.44 \pm 0.62	0.221 \pm 0.037
90-135°	95.65 \pm 1.36	0.180 \pm 0.066	94.41 \pm 1.64	0.251 \pm 0.063	93.42 \pm 2.07	0.233 \pm 0.086
135-180°	95.89 \pm 1.20	0.172 \pm 0.053	94.63 \pm 0.96	0.240 \pm 0.039	93.58 \pm 1.34	0.252 \pm 0.063

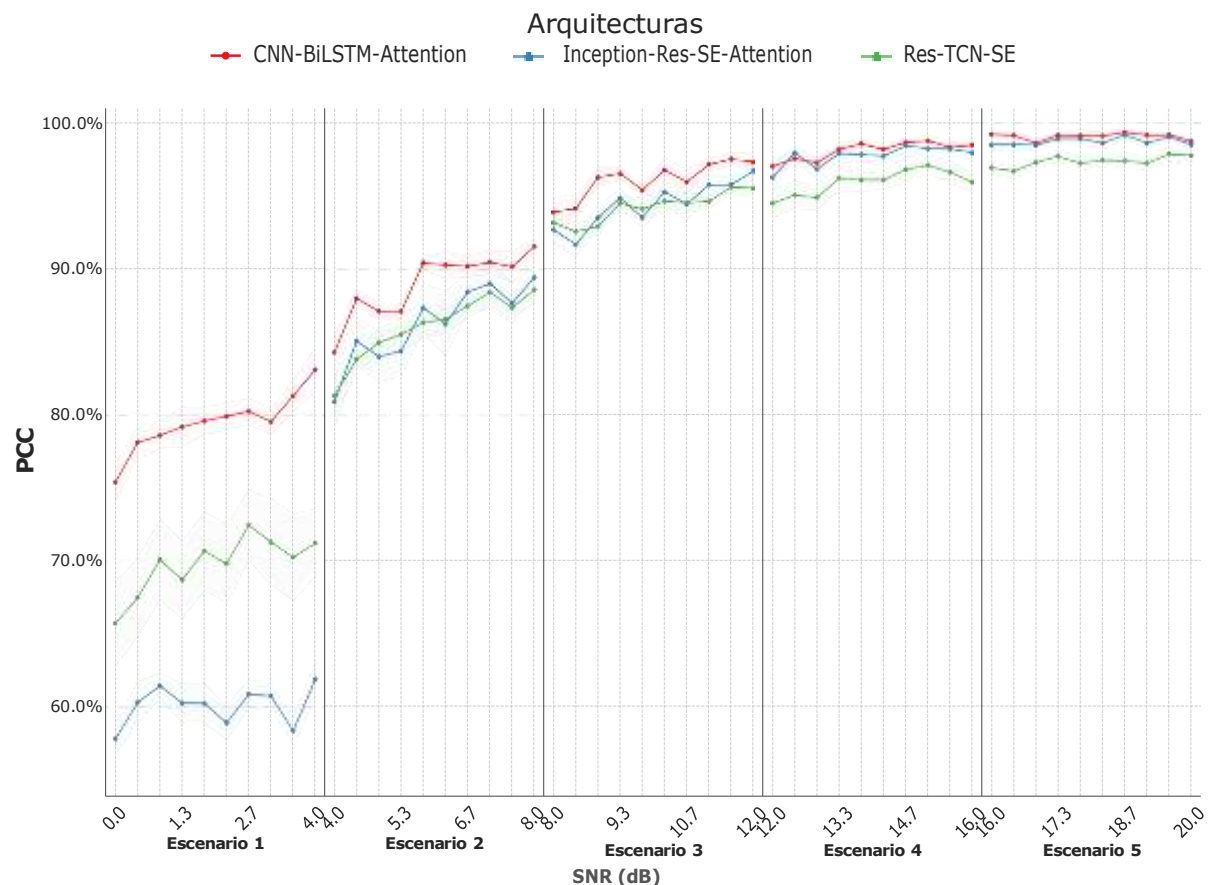


Figura 5.1: PCC medio con IC 95 % vs. SNR (escenarios 1–5).

En la validación para los escenarios univariados del SNR (*referirse a Figura 5.1*), se observa que todas las redes muestran un aprendizaje consistente a lo largo del aumento del SNR. Sin embargo, se constata que la arquitectura Inception-Res-SE-Attention presenta sensibilidad a los parámetros de inicialización, esto genera mayor variabilidad, que se evidencia en un rango más amplio para el IC 95 %, en su desempeño en los escenarios de más bajo SNR (*0-4dB y 4-8dB, escenario 1 y 2, respectivamente*).

Debido a esto, fue necesario ajustar los parámetros de entrenamiento para esta red en estos 2 escenarios. Los cambios puntuales fueron: incremento en el parámetro paciencia parada temprana a 20 y modificación el umbral del *ReduceLROnPlateau* de 0.2 a 0.5. Los resultados al modificar estos parámetros se evalúan en la Sección 5.2.

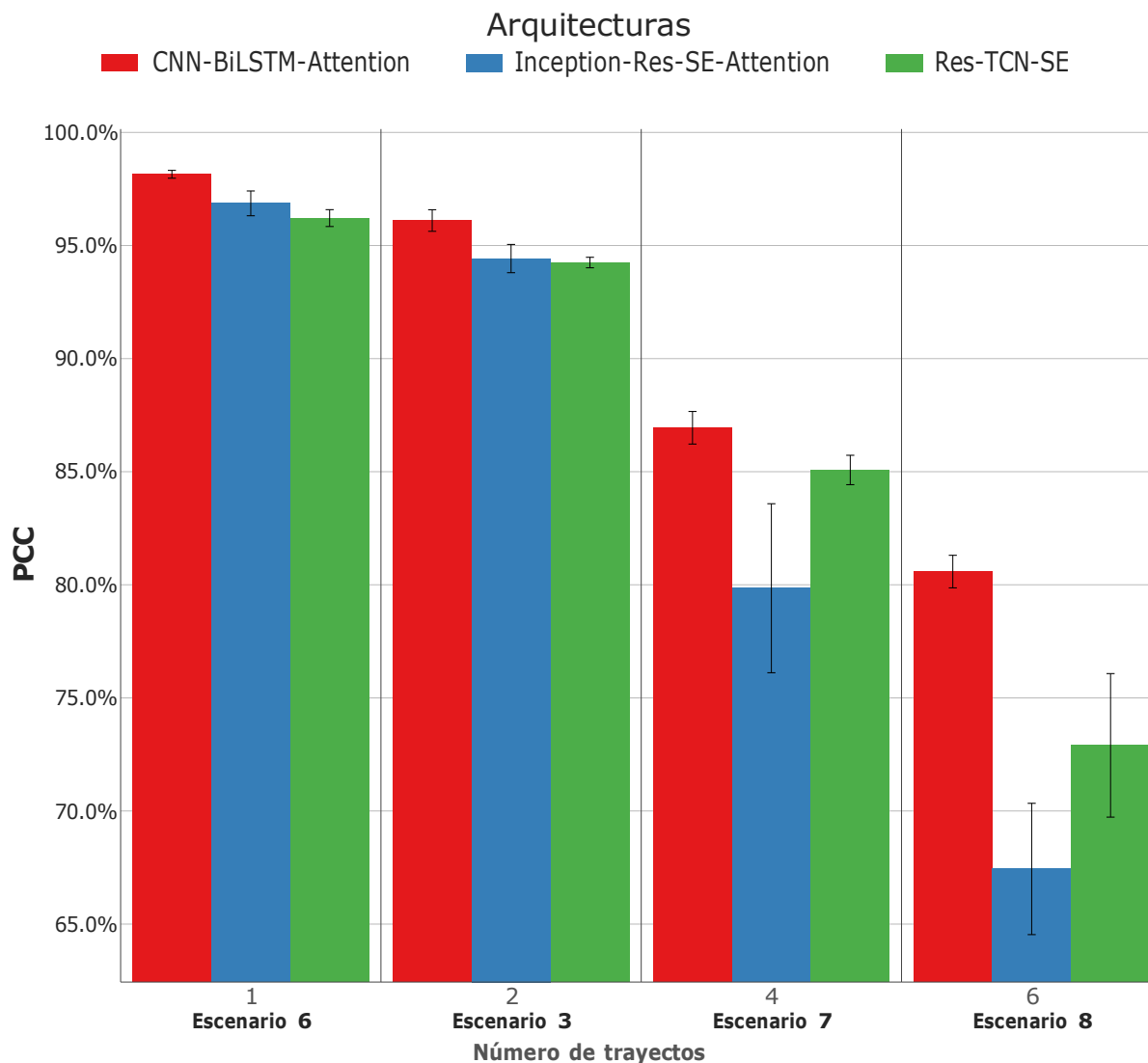


Figura 5.2: PCC medio con IC 95% vs. número de trayectos.

La Figura 5.2 muestra el desempeño en los escenarios de variación del número de trayectos. Se observa claramente una tendencia decreciente en el PCC medio conforme aumenta la complejidad del escenario, esto evidencia la relevancia del parámetro número de trayectos en un AMC.

En los escenarios 6 y 3 (*1 y 2 trayectos, respectivamente*), los modelos alcanzan un PCC superior al 93 %, pero conforme aumenta el número de trayectos, esta precisión se reduce progresivamente, llegando en el caso más extremo a valores cercanos al 65 % en el caso de la arquitectura *Inception-Res-SE-Attention*.

Adicionalmente, se observa que la arquitectura *CNN-BiLSTM-Attention* presenta un desempeño más robusto, manteniéndose como la mejor, por su PCC elevado y variabilidad limitada de IC, en todos los escenarios evaluados. Por el contrario, la arquitectura *Inception-Res-SE-Attention* muestra una caída más pronunciada, especialmente en los escenarios con 4 y 6 trayectos, tanto en PCC como en el aumento de la variabilidad reflejada en el rango del IC.

La arquitectura *Res-TCN-SE* mantiene un rendimiento intermedio en la mayoría de los casos, con una estabilidad mayor que la de *Inception-Res-SE-Attention*.

Finalmente, se puede notar que las barras del intervalo de confianza aumentan conforme se incrementa el número de trayectos, lo cual refleja una mayor incertidumbre en los rendimientos del modelo.

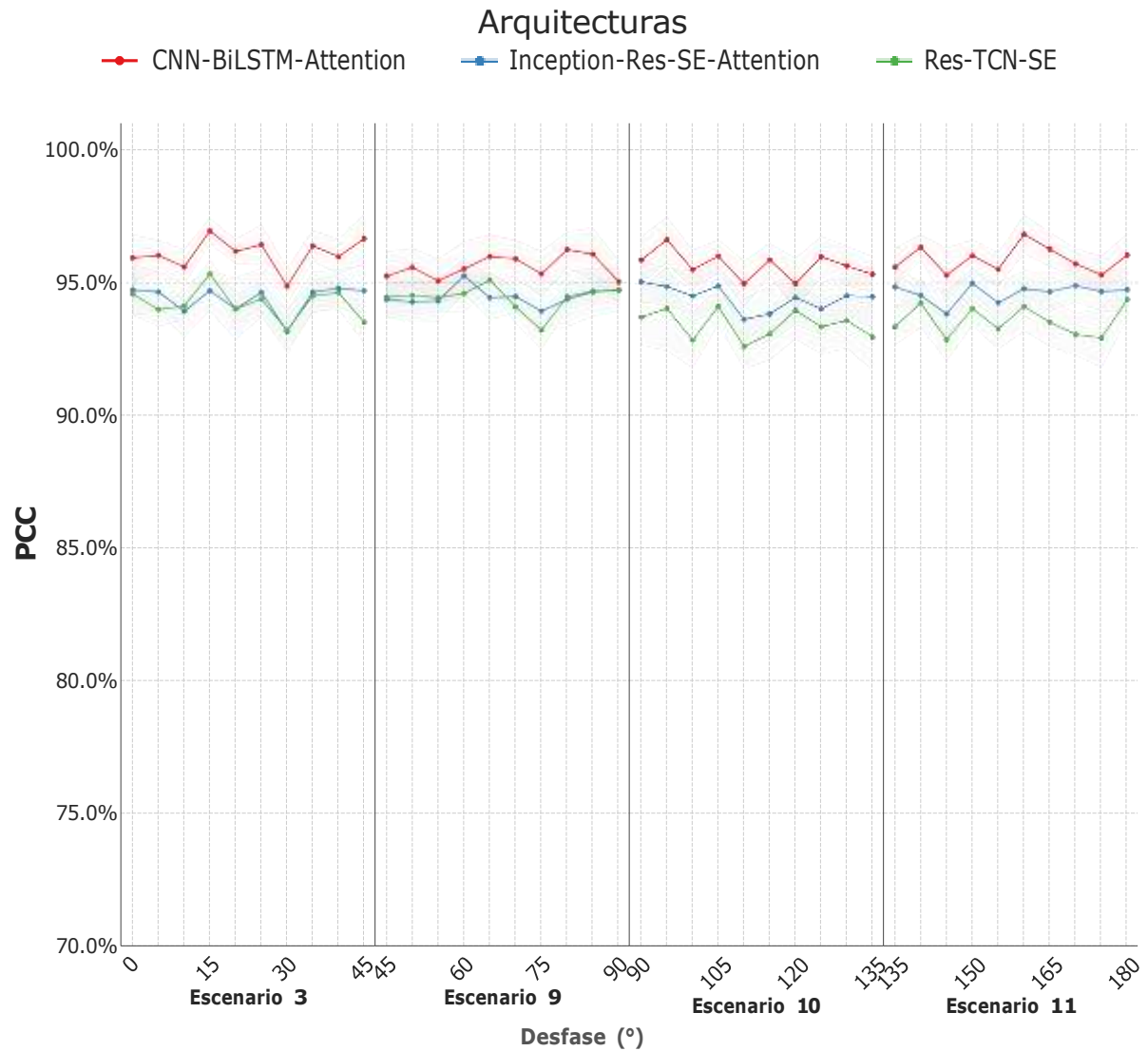


Figura 5.3: PCC medio con IC 95 % vs. desfase.

La Figura 5.3 presenta el comportamiento del PCC en los escenarios de variabilidad del desfase angular (3, 9, 10 y 11).

Se observa que el desajuste de fase tiene un impacto limitado sobre el desempeño de las redes, ya que los valores de PCC se mantienen altos (en torno al 94–96 %) a lo largo de toda la variación angular (de 0° a 180°). Esta tendencia sugiere que las arquitecturas evaluadas presentan una alta robustez frente a variaciones de fase.

Síntesis. Las tres arquitecturas mantienen $PCC > 90\%$ mientras $SNR \geq 8$ dB; los resultados

en SNR son consistentes con la teoría. El aumento de trayectos genera la mayor penalización (hasta ≈ 25 punto porcentual (pp) en Inception-Res-SE-Attention), mientras que el desfase produce una caída prácticamente irrelevante.

5.1.2. Análisis detallado por clase

5.1.2.1. Mapa de calor F1 por escenario y clase

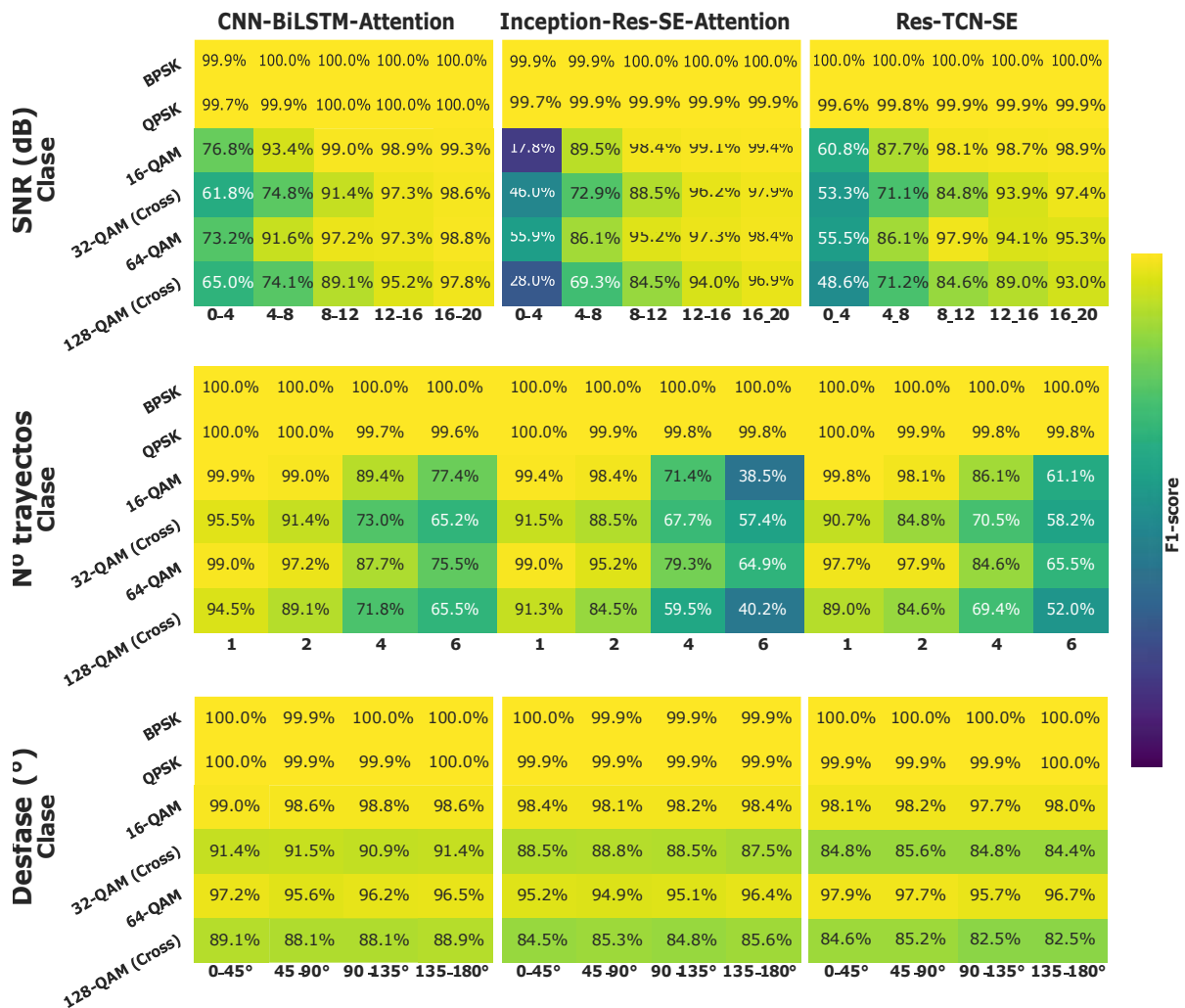


Figura 5.4: Mapa de calor de F1 por clase y escenario univariable.

Comentario. En la Figura 5.4 se observa un gradiente claro conforme disminuye la SNR y aumentan los trayectos. La caída más brusca ocurre en $SNR < 4$ dB para las modulaciones QAM, donde se puede observar que la métrica F1 en conjunto bajan a un promedio de 63.55%. Por otro lado, es importante destacar que BPSK y QPSK tiene un F1 casi perfecto ($> 99.6\%$) en todos los escenarios.

5.1.2.2. Contribución al error global

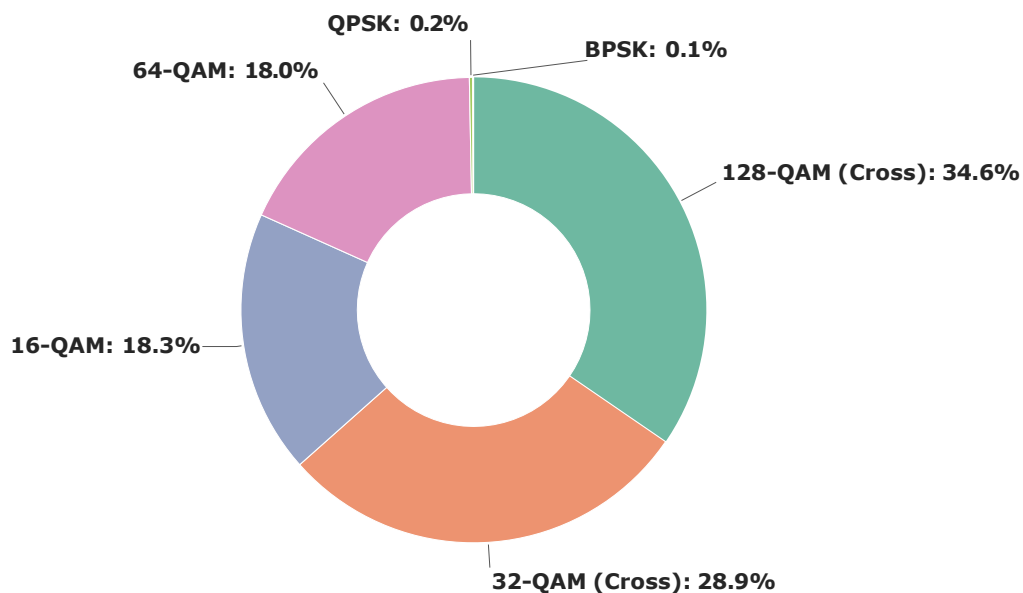


Figura 5.5: Distribución de la contribución al error (1– Recall) normalizada por clase. Todos los escenarios son considerados.

Análisis. La Figura 5.5 muestra que las modulaciones de orden elevado dominan el error global (*como se había previsto*):

- 128-QAM cruzada contribuye con un 34,6 %, seguida de la 32-QAM cruzada con un 28,9 %. Esto sugiere que, debido a que ambas modulaciones comparten una distribución cruzada y presentan símbolos con niveles de amplitud y fase similares en determinadas regiones de la constelación, ambas modulaciones presentan una fuerte correlación en el origen del error.
- Las modulaciones 16-QAM y 64-QAM aportan un 18,3 % y un 18,0 %, respectivamente. Esta similitud en el error puede atribuirse al hecho de que ambas modulaciones comparten una estructura geométrica de constelación similar y se diferencian principalmente en la densidad de símbolos, de forma análoga a las modulaciones 32-QAM y 128-QAM cruzadas.
- Las modulaciones BPSK y QPSK resultan prácticamente irrelevantes, con menos de un 0,3 % del error total.

La Figura 5.5 expone que, incluso con un sobremuestreo de $\times 8$ y señales de 512 símbolos, la complejidad geométrica de las constelaciones cuadradas y cruzadas de alto orden de modulación siguen siendo el principal desafío para las tres arquitecturas.

5.2. Entrenamiento y validación exhaustiva

5.2.1. Desempeño global frente a parámetros de canal

La Tabla 5.2 detalla los resultados obtenidos luego del entrenamiento exhaustivo. En breves rasgos, en comparación con la fase de validación temprana, la variabilidad —medida por la anchura de los intervalos de confianza— se acorta en un 60 % de media, y el PCC se incrementa por encima del 2.5 % en diversos escenarios.

Tabla 5.2: Resultados de entrenamiento y validación exhaustiva: exactitud (%) y pérdida (loss) media \pm desviación estándar, 10 corridas por condición univariable.

Condición	CNN-BiLSTM-Attn		Inception-Res-SE-Attn		Res-TCN-SE	
	Acc (%)	Loss	Acc (%)	Loss	Acc (%)	Loss
1. Variación de SNR ($\text{num_taps}=2$, $\text{phase_offset}=0-45^\circ$)						
0-4 dB	81.39 \pm 0.70	0.419 \pm 0.060	80.41 \pm 0.58	0.458 \pm 0.020	79.76 \pm 1.00	0.441 \pm 0.032
4-8 dB	91.29 \pm 0.34	0.284 \pm 0.087	90.99 \pm 0.56	0.303 \pm 0.055	89.62 \pm 0.34	0.330 \pm 0.137
8-12 dB	97.64 \pm 0.19	0.108 \pm 0.011	97.42 \pm 0.47	0.140 \pm 0.023	96.19 \pm 0.29	0.155 \pm 0.020
12-16 dB	98.89 \pm 0.24	0.044 \pm 0.010	98.90 \pm 0.21	0.076 \pm 0.011	97.76 \pm 0.50	0.088 \pm 0.019
16-20 dB	99.38 \pm 0.11	0.022 \pm 0.005	99.40 \pm 0.14	0.047 \pm 0.007	98.74 \pm 0.20	0.046 \pm 0.013
2. Variación de multitrayectorias (SNR=8-12 dB, $\text{phase_offset}=0-45^\circ$)						
1 tap	98.89 \pm 0.20	0.048 \pm 0.009	98.95 \pm 0.21	0.069 \pm 0.009	98.42 \pm 0.25	0.070 \pm 0.007
2 taps	97.64 \pm 0.19	0.108 \pm 0.011	97.42 \pm 0.47	0.140 \pm 0.023	96.19 \pm 0.29	0.155 \pm 0.020
4 taps	90.42 \pm 0.39	0.283 \pm 0.042	90.52 \pm 1.14	0.313 \pm 0.033	88.52 \pm 0.24	0.284 \pm 0.040
6 taps	85.12 \pm 0.74	0.398 \pm 0.087	85.53 \pm 4.24	0.449 \pm 0.176	83.82 \pm 0.81	0.370 \pm 0.019
3. Variación de desfase (SNR=8-12 dB, $\text{num_taps}=2$)						
0-45°	97.64 \pm 0.19	0.108 \pm 0.011	97.42 \pm 0.47	0.140 \pm 0.023	96.19 \pm 0.29	0.155 \pm 0.020
45-90°	97.94 \pm 0.21	0.084 \pm 0.013	97.34 \pm 0.61	0.133 \pm 0.019	96.16 \pm 0.28	0.142 \pm 0.024
90-135°	97.72 \pm 0.24	0.084 \pm 0.012	97.30 \pm 0.57	0.141 \pm 0.026	96.36 \pm 0.29	0.137 \pm 0.014
135-180°	97.69 \pm 0.52	0.101 \pm 0.034	97.23 \pm 0.34	0.166 \pm 0.027	96.33 \pm 0.37	0.149 \pm 0.019

5.2.2. Curvas PCC por parámetro

Las Figuras 5.6, 5.7 y 5.8 muestran de manera gráfica el PCC para los escenarios de variación de SNR, número de trayectorias y desfase, respectivamente.

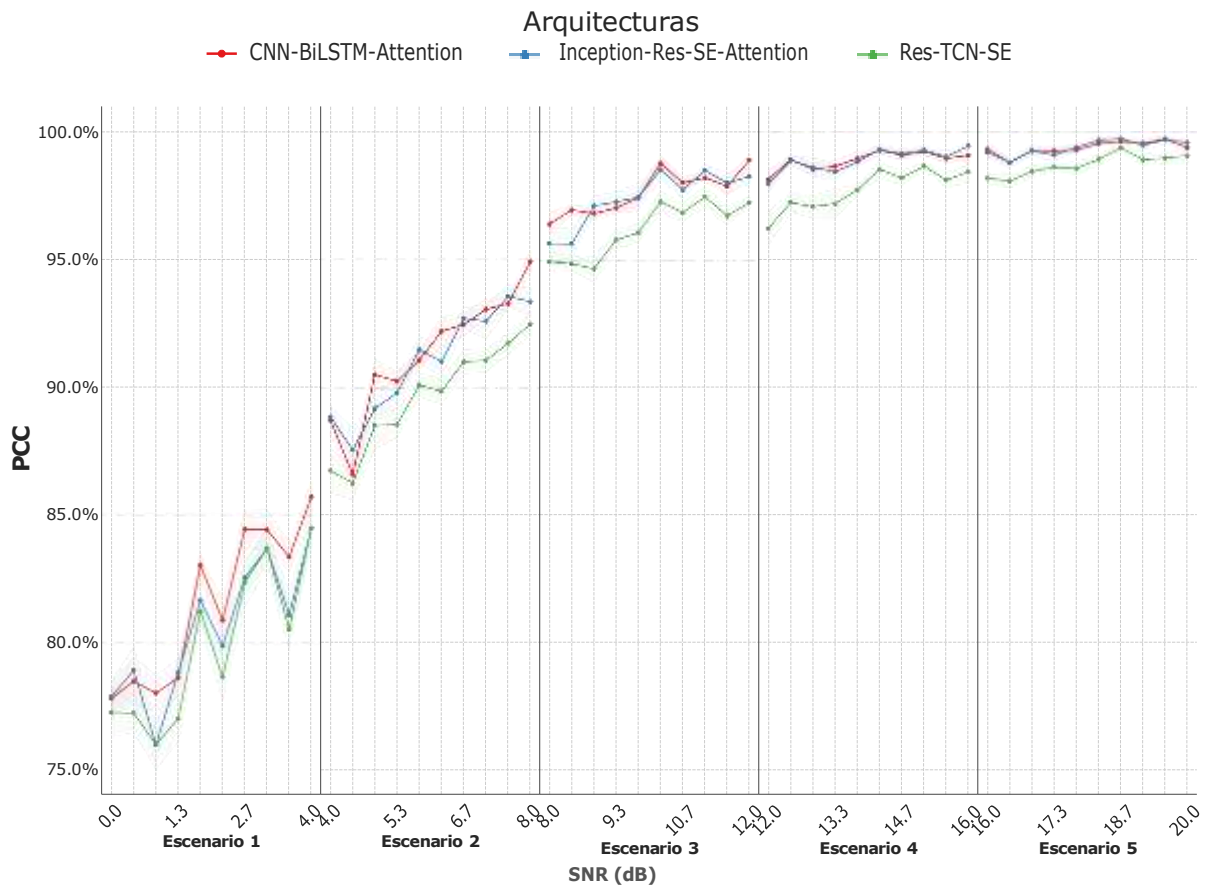


Figura 5.6: PCC medio con IC 95% vs. SNR. Entrenamiento exhaustivo

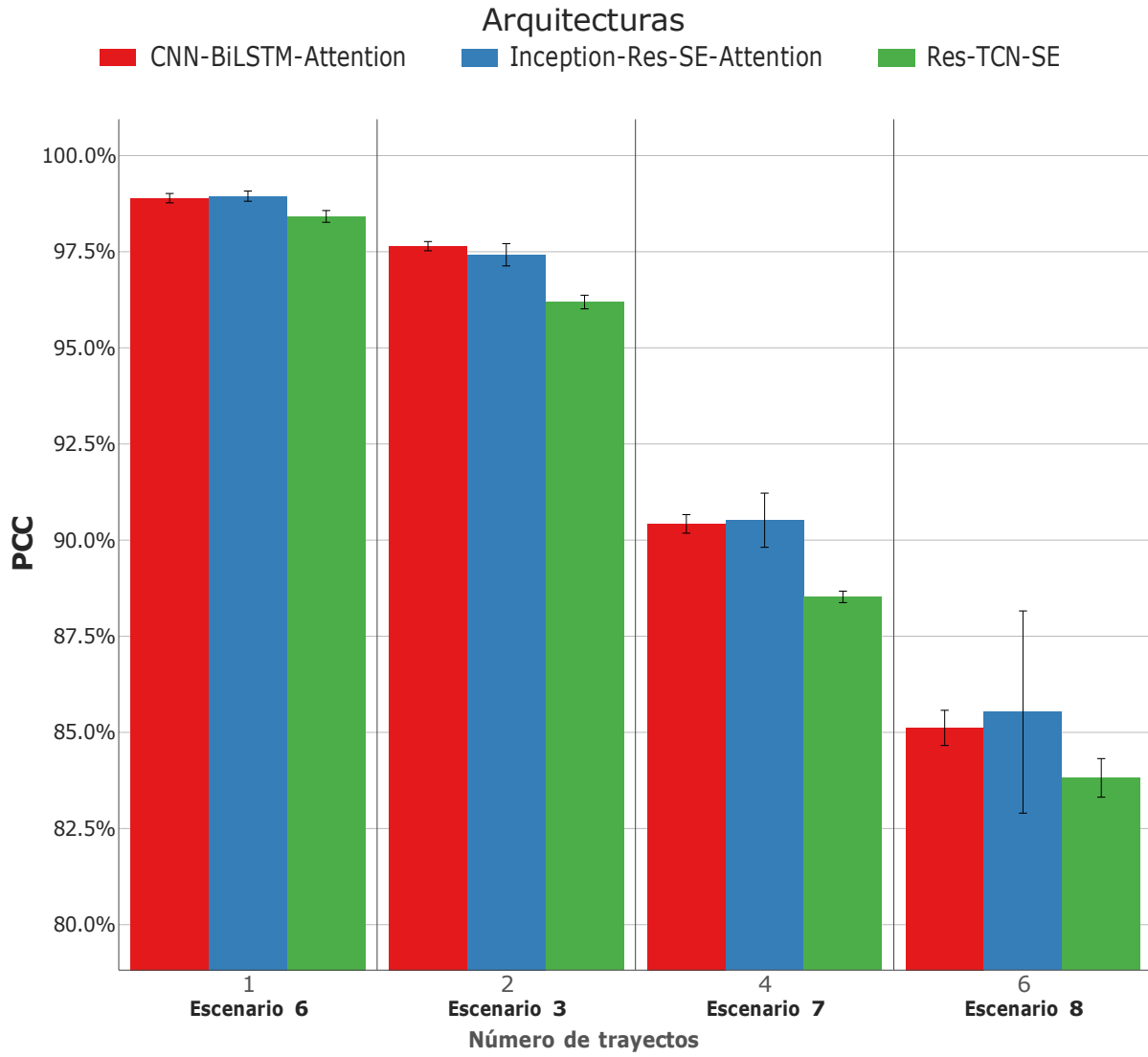


Figura 5.7: PCC medio con IC 95% vs. número de trayectos. Entrenamiento exhaustivo

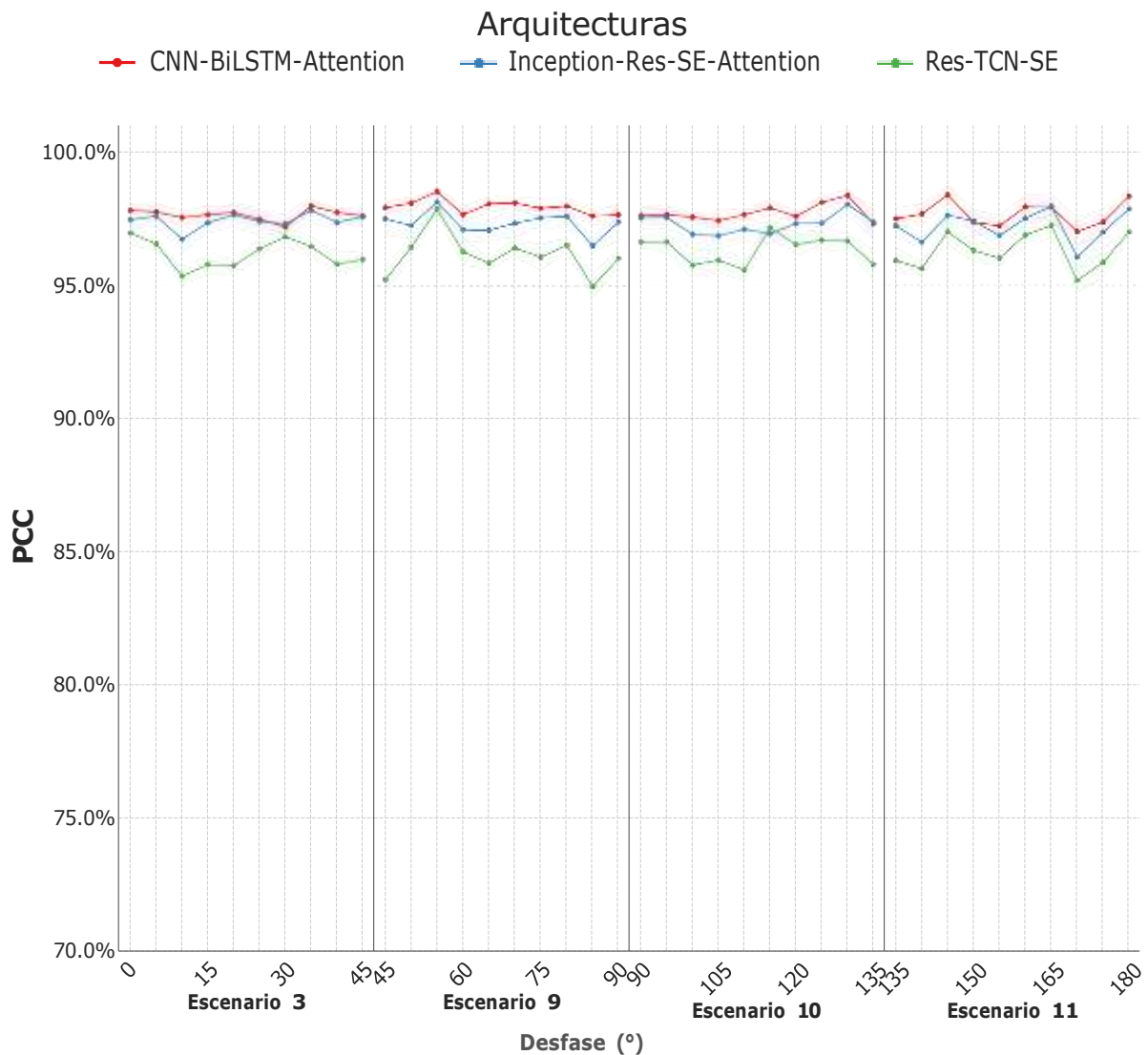


Figura 5.8: PCC medio con IC 95% vs. desfase. Entrenamiento exhaustivo

Comentario. El margen entre arquitecturas se reduce a < 1 pp. De forma concreta evidenciamos una disminución significativa de la variabilidad (*rango del IC más pequeño*) en comparación con los resultados de la fase de validación. De igual forma, constatamos un aumento en PCC en las 3 arquitecturas, siendo los escenarios de bajo SNR los más beneficiados.

5.2.3. Mapa de calor F1 por escenario y clase

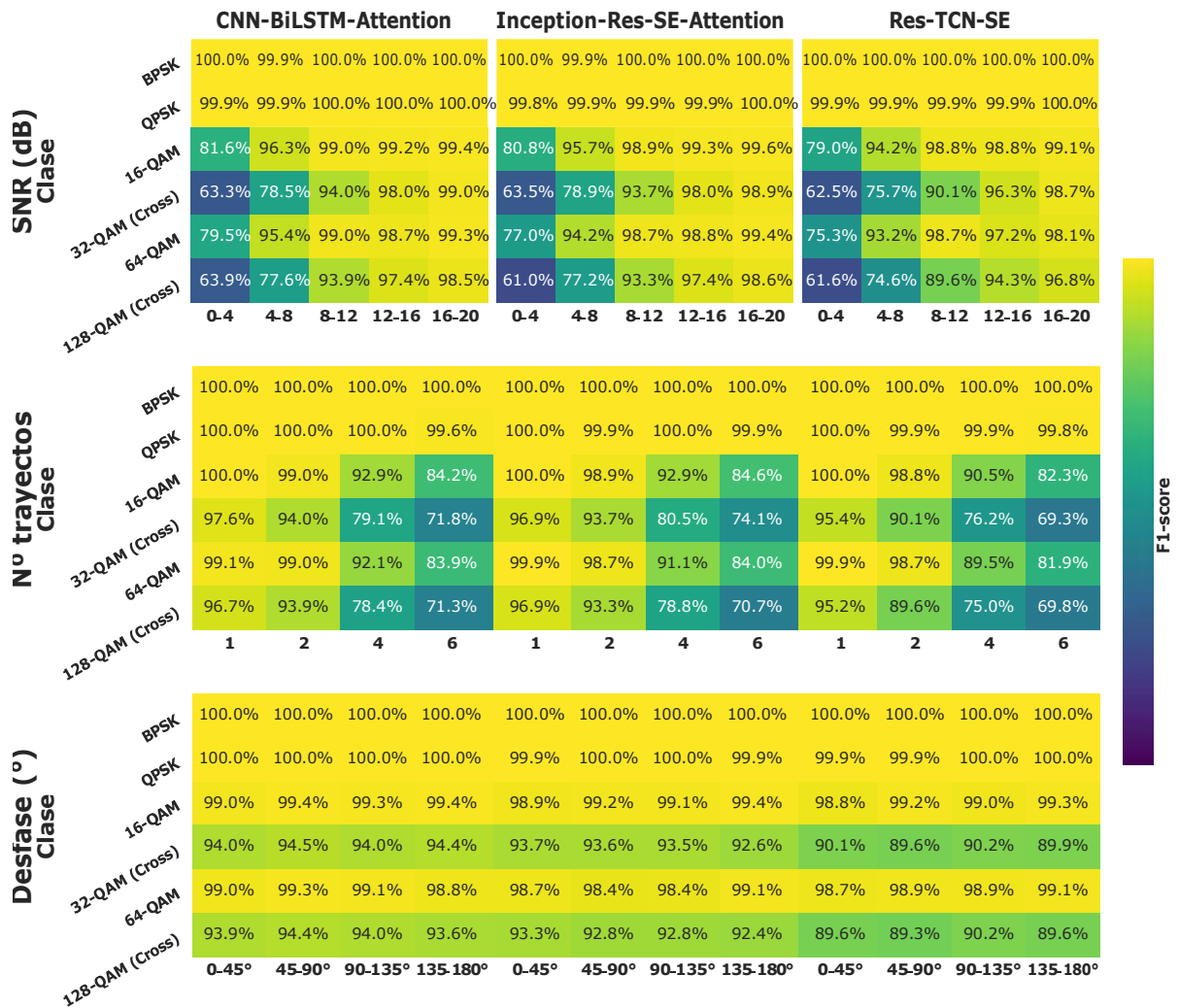


Figura 5.9: Mapa de calor de F1 por clase y escenario tras entrenamiento exhaustivo.

Como se observa en la Figura 5.9, la mejora en F1 es homogénea. Cabe recalcar que, existe un aumento sustancial de la métrica F1 en los escenarios de bajo SNR (0-4 dB y 4-8dB) y de alto número de trayectos (4 y 6 trayectos). Este resultado muestra la efectividad de entrenar los modelos con un conjunto amplio de datos.

5.2.4. Contribución al error global

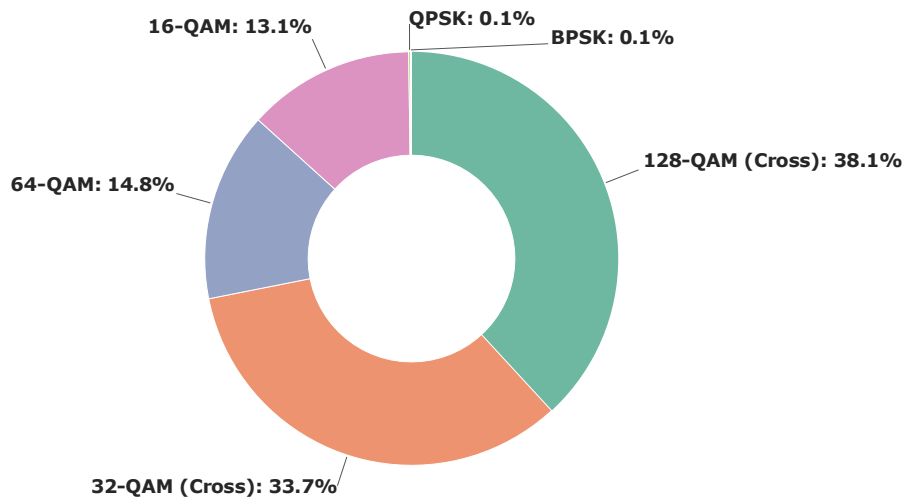


Figura 5.10: Contribución al error después del reentrenamiento exhaustivo.

Comparación con fase previa. Al comparar los resultados presentados en la Figura 5.5 con los nuevos datos de la Figura 5.10, se observa una reducción en el aporte al error por parte de las clases BPSK, QPSK, 16-QAM y 64-QAM, lo cual indica que el modelo se benefició del mayor volumen de datos en relación al volumen de datos empleado en la fase de validación temprana. Sin embargo, se presenta un aumento en el error asociado a las clases 32-QAM cruzada y 128-QAM cruzadas, las cuales en conjunto representan más de la mitad del error total.

En otras palabras, el incremento en el volumen de muestras resultó insuficiente para solventar el problema de discernimiento de modulaciones con diagramas de constelación anidados.

5.2.5. Análisis de matrices de confusión

Como se mencionó en la Sección 4.6, para cada combinación de arquitectura y escenario se realizaron diez ejecuciones independientes, lo que produjo 330 matrices de confusión brutas. Tras normalizarlas y promediarlas, obtuvimos 33 matrices finales (3 arquitecturas \times 11 escenarios). Los hallazgos principales son:

- **Bajo SNR (0-4 dB y 4-8 dB).** BPSK y QPSK mantienen tasas de acierto del 100 %, pero las QAM intermedias y de alta densidad exhiben un fuerte ensanchamiento off-diagonal, como se observa en la Figura 5.11. En el escenario más adverso (0-4 dB), se registran hasta un 34 % de asignaciones erróneas entre 32-QAM (cruzada) y 128-QAM (cruzada), y confusiones del 16%-19% entre 16-QAM y 64-QAM.
- **Multipath (4 y 6 taps).** Al incrementar el número de trayectos (Figura 5.12), las QAM cruzadas ven caer su diagonal: con 4 trayectos, las tasas de acierto de 32-QAM (Cross)

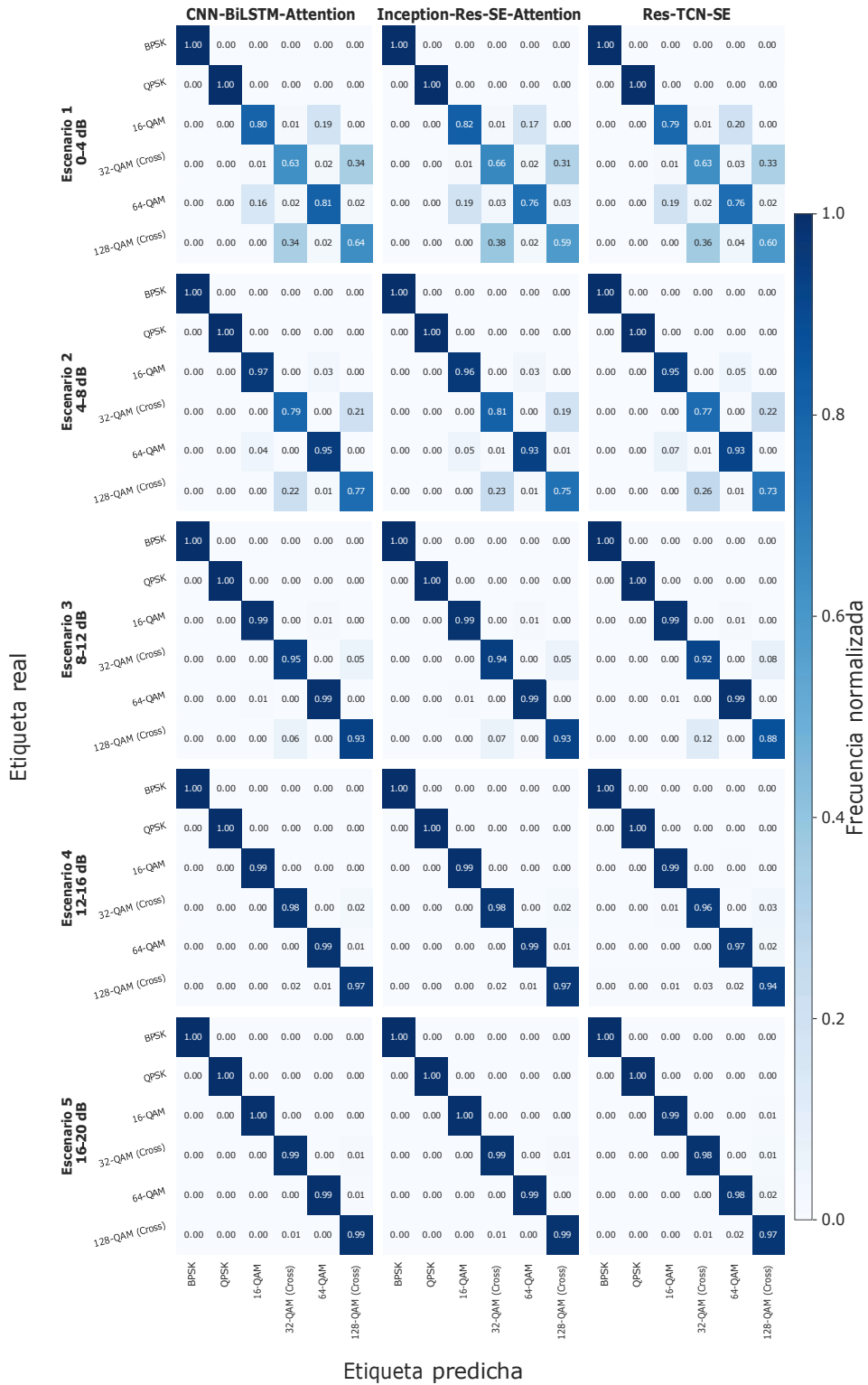


Figura 5.11: Matriz de confusión promedio para escenarios con variación de SNR.

y 128-QAM (Cross) descienden al 74%–80%, y con 6 taps la precisión desciende aún más, al 68 %-74 %.

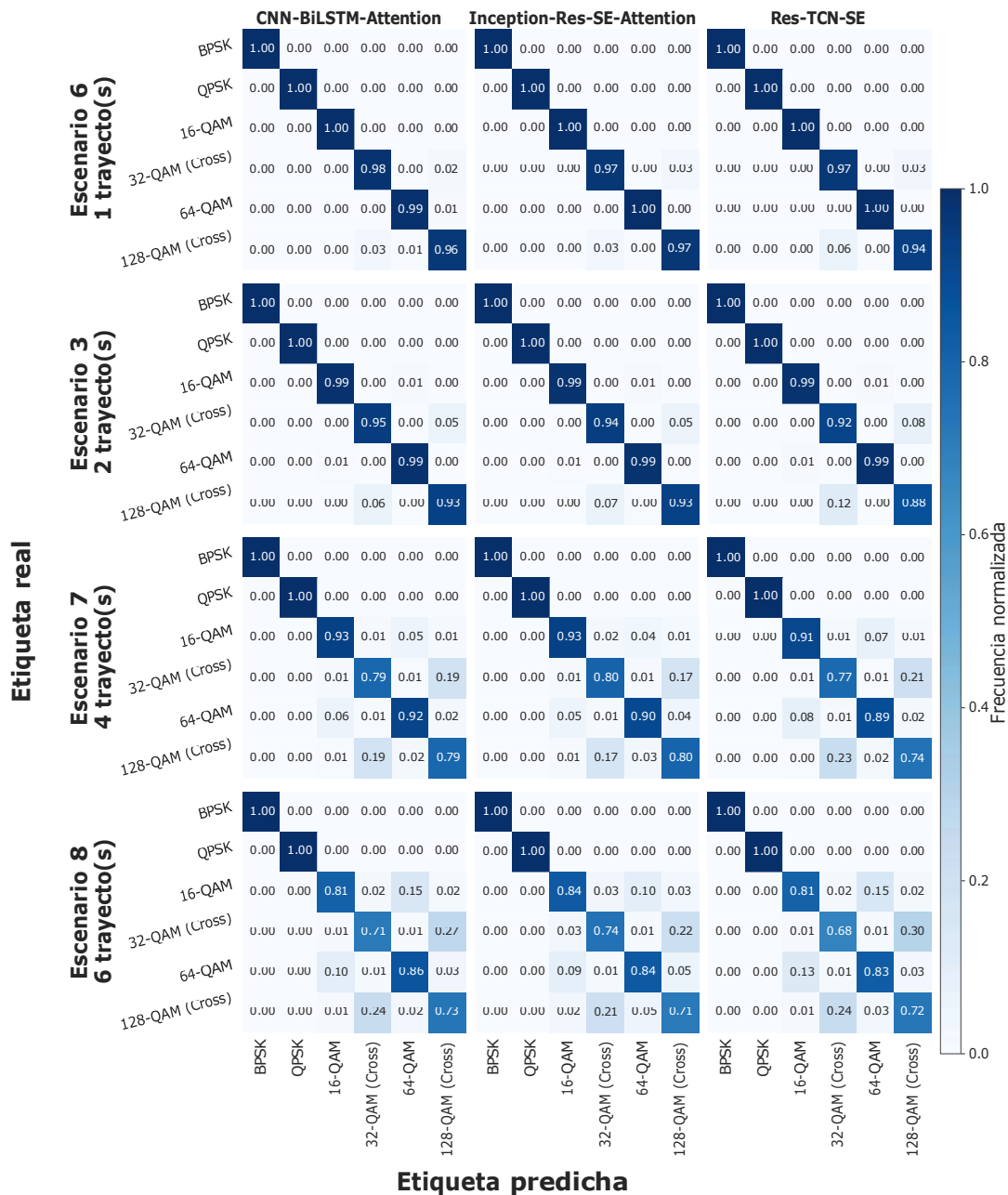


Figura 5.12: Matriz de confusión promedio para escenarios con variación en número de trayectos.

- Desfase angular (0° -180°).** Como muestra la Figura 5.13, las variaciones de fase apenas alteran la pureza diagonal de las matrices: todas las clases conservan tasas promedio de acierto superiores al 95 %, confirmando la alta inmunidad de las arquitecturas frente a este parámetro.

Los resultados ponen de manifiesto que la degradación por SNR y el número de trayectos son los factores más críticos en AMC, mientras que el desfase queda en un segundo plano.

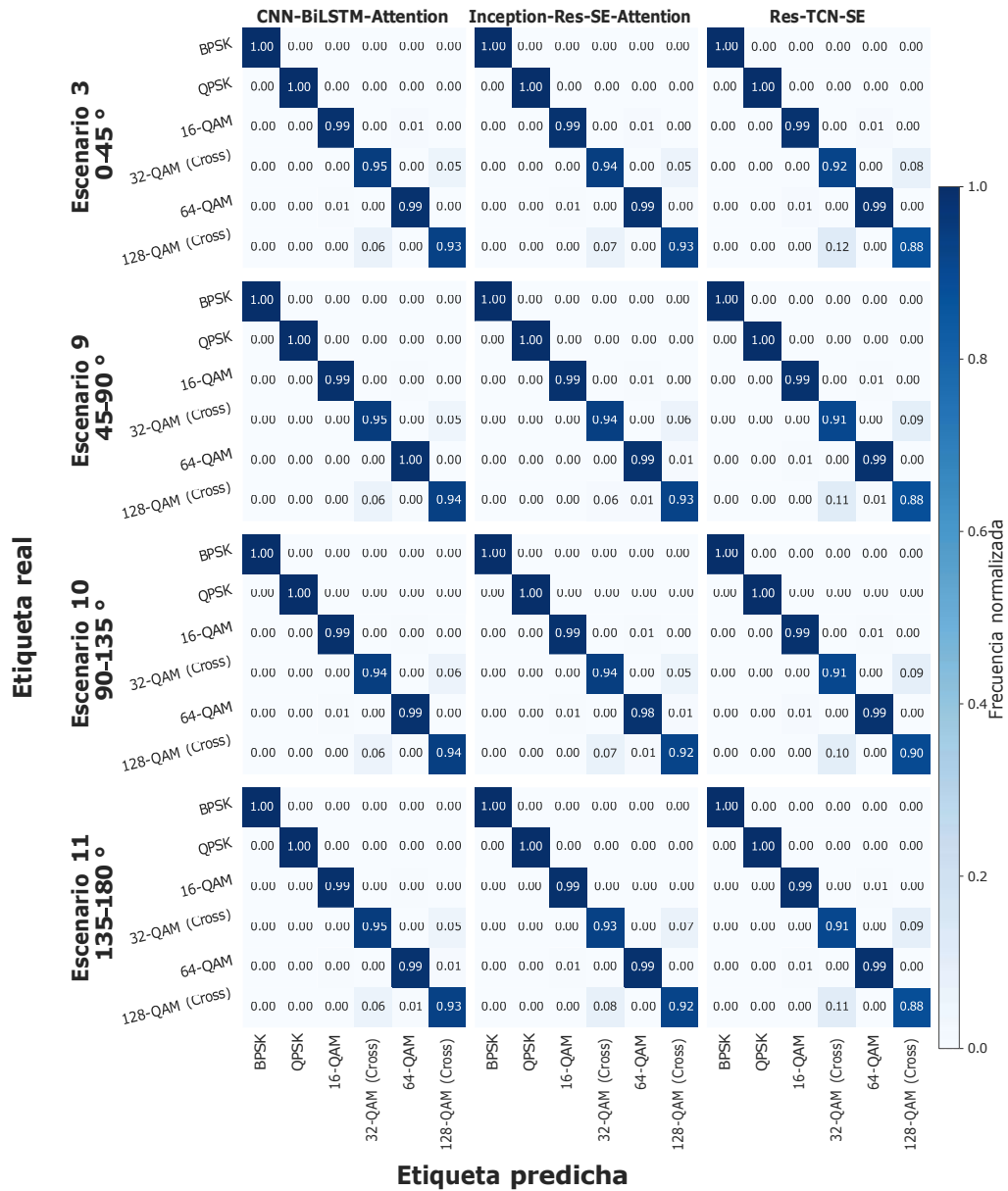


Figura 5.13: Matriz de confusión promedio para escenarios con variación en desfase.

5.3. Experimento con siete modulaciones (inclusión de 256-QAM)

5.3.1. PCC medio con IC 95% en el escenario base

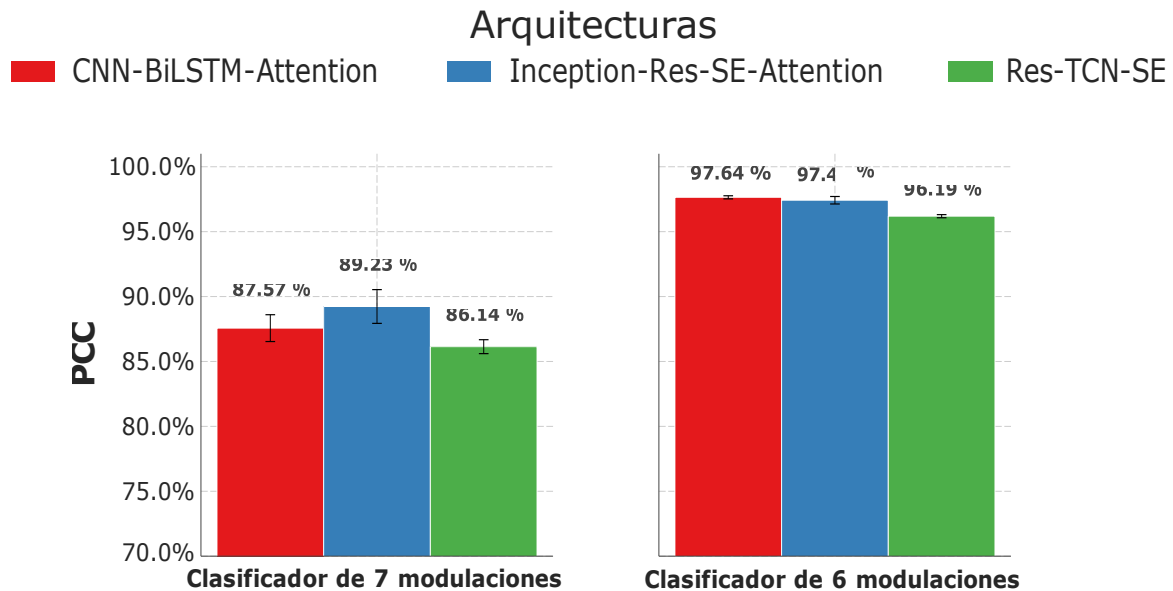


Figura 5.14: Comparación de PCC con IC 95% (escenario 3) con seis vs. siete modulaciones.

Hallazgo clave. La introducción de 256-QAM provoca una reducción media de 9.43 pp en las 3 arquitecturas; no obstante la arquitectura Inception-Res-SE-Attention amortigua mejor el impacto (-8.19 pp).

De acuerdo a lo que se evidenció en la Sección 5.2, la arquitectura CNN-BiLSTM-Attention ofrecía el mejor desempeño en todos los escenarios evaluados. Sin embargo, al ampliar el conjunto de clases a siete, observamos que Inception-Res-SE-Attention se impone en el escenario base. Esto se debe a la mayor capacidad discriminativa de su estructura:

- Bloques residuales heredados de Inception ResNet potencian el flujo de información.
- Módulo Squeeze-and-Excitation refina dinámicamente la representación de canales.
- Atención multi-cabeza profundiza en los matices más sutiles de la señal IQ.

Estos hallazgos, en cierto grado contrarios a lo observado en secciones previas, ponen en evidencia que la elección del modelo debe considerar no solo la arquitectura, sino también la complejidad y el número de clases del problema; ya que una red con mayor capacidad y especialización puede ser necesaria cuando se incrementa la dificultad de clasificación.

5.3.2. Mapa de calor F1 por clase

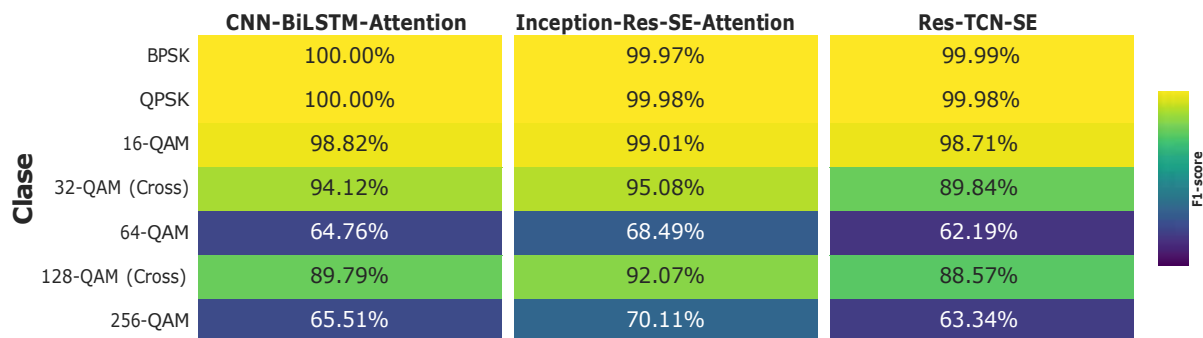


Figura 5.15: Mapa de calor F1 por clase con siete modulaciones (escenario 3).

5.3.2.1. Análisis.

Para analizar el rendimiento individual de cada arquitectura en el escenario de siete modulaciones, la Figura 5.15 presenta un mapa de calor del valor F1 obtenido por clase. Esta representación visual facilita la identificación rápida de las modulaciones más problemáticas y permite comparar directamente el desempeño de los tres modelos.

- **Desempeño en modulaciones de orden bajo y medio.** 16-QAM, 32-QAM (cruzada) y 128-QAM (cruzada) mantienen un F1 superior al 89 %, reflejando que la complejidad añadida por 256-QAM no penaliza la clasificación de estas clases intermedias.
- **Dificultad en modulaciones de orden alto.** Los mayores retos se concentran en 64-QAM y 256-QAM, con valores de F1 que oscilan entre 62 % y 70 % según la arquitectura.
 - *CNN-BiLSTM-Attention*: 64.76 % (64-QAM) y 65.51 % (256-QAM).
 - *Inception-Res-SE-Attention*: 68.49 % (64-QAM) y 70.11 % (256-QAM).
 - *Res-TCN-SE*: 62.19 % (64-QAM) y 63.34 % (256-QAM).
- **Mejora de modulaciones cruzadas.** 32-QAM y 128-QAM cruzada pasan de valores cercanos al 63 % (en el escenario de seis clases) a superar el 90 %, lo que indica que la presencia de 256-QAM favorece fronteras de decisión más definidas para estas clases.

5.3.3. Contribución al error

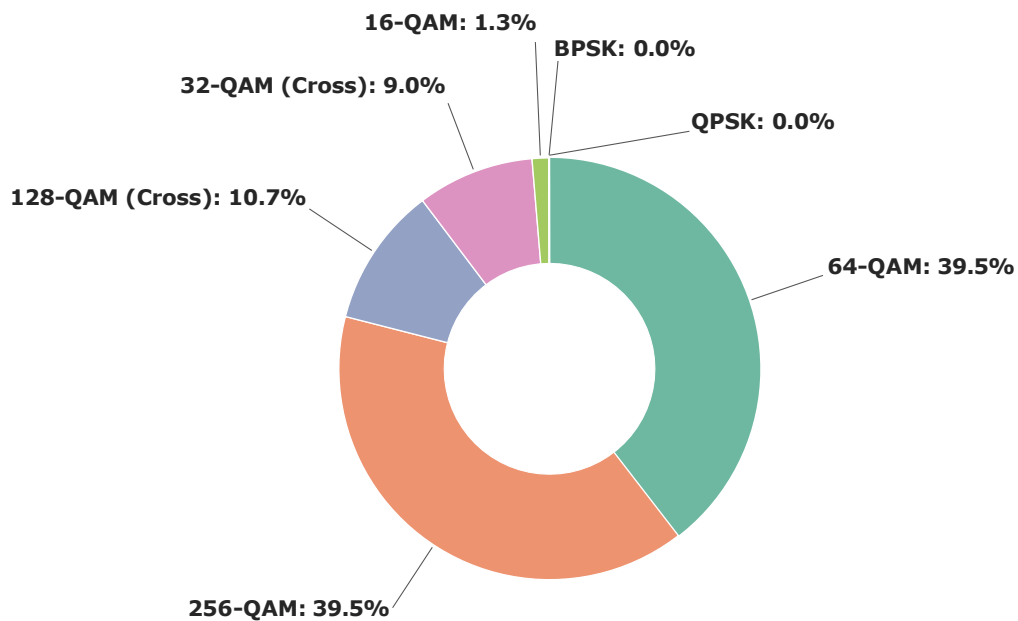


Figura 5.16: Contribución al error con siete modulaciones.

5.3.3.1. Análisis

En referencia a la Figura 5.16, 256-QAM y 64-QAM:

- **Alta contribución al error:** 64-QAM y 256-QAM representan cada una el 39.5 % del error, sumando un 79 % en conjunto. Esto reafirma que la mayor dificultad radica en distinguir entre constelaciones de alta densidad.
- **Contribución media:** 128-QAM (cruzada) y 32-QAM (cruzada) aportan 10.7 % y 9.0 %, respectivamente. Como se mencionó en la Sección 5.3.2, la introducción de 256QAM aporta una mejor discriminación de estas 2 clases.
- **Baja y nula contribución:** 16-QAM representa solo el 1.3 % del error, mientras que BPSK y QPSK no aportan error detectable, indicando que las arquitecturas las discrimina con alta confianza sin sensibilidad a la inclusión de 256QAM.

5.4. Sistema de votación por comité

5.4.1. Selección óptima del parámetro k

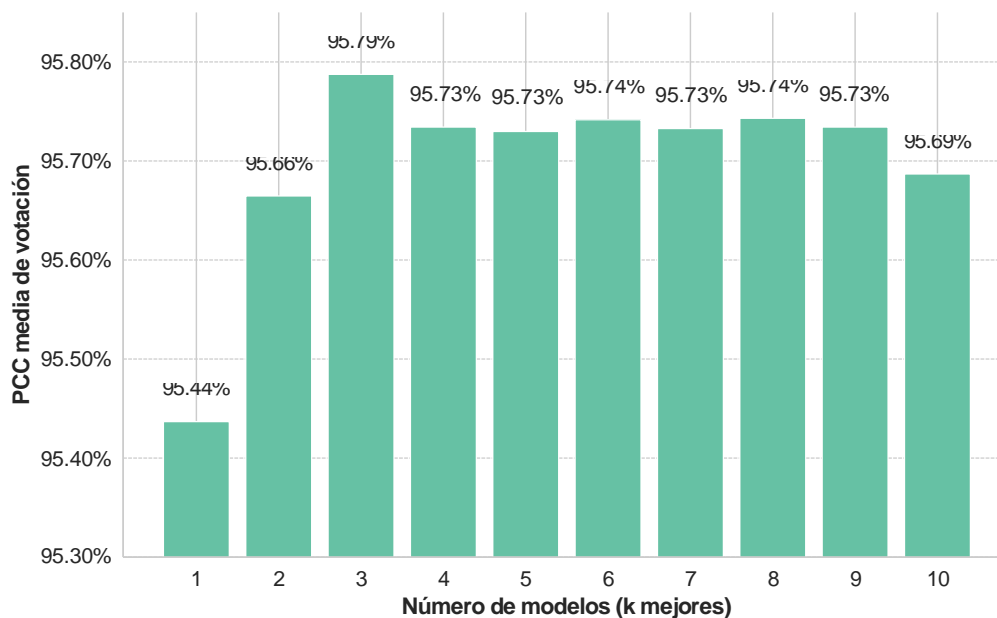


Figura 5.17: PCC promedio global vs. tamaño del subconjunto k .

Importante: el PCC medio de votación representado en la Figura 5.17 corresponde al promedio del PCC de las tres arquitecturas evaluadas en cada uno de los once escenarios.

Análisis de la selección de k . La Figura 5.17 muestra cómo varía el PCC medio al modificar el tamaño del subconjunto k en el comité de votación por arquitectura.

- **Óptimo en $k = 3$.** El PCC medio alcanza su máximo de 95.79 % considerando los 3 mejores modelos por arquitectura.
- **Compromiso precisión-complejidad.** Dado que la mejora absoluta entre $k = 3$ y $k = 10$ es inferior a 0.1 %, elegir $k = 3$ nos ofrece el mejor equilibrio entre exactitud y carga computacional.

5.4.2. Comparación por parámetro univariable

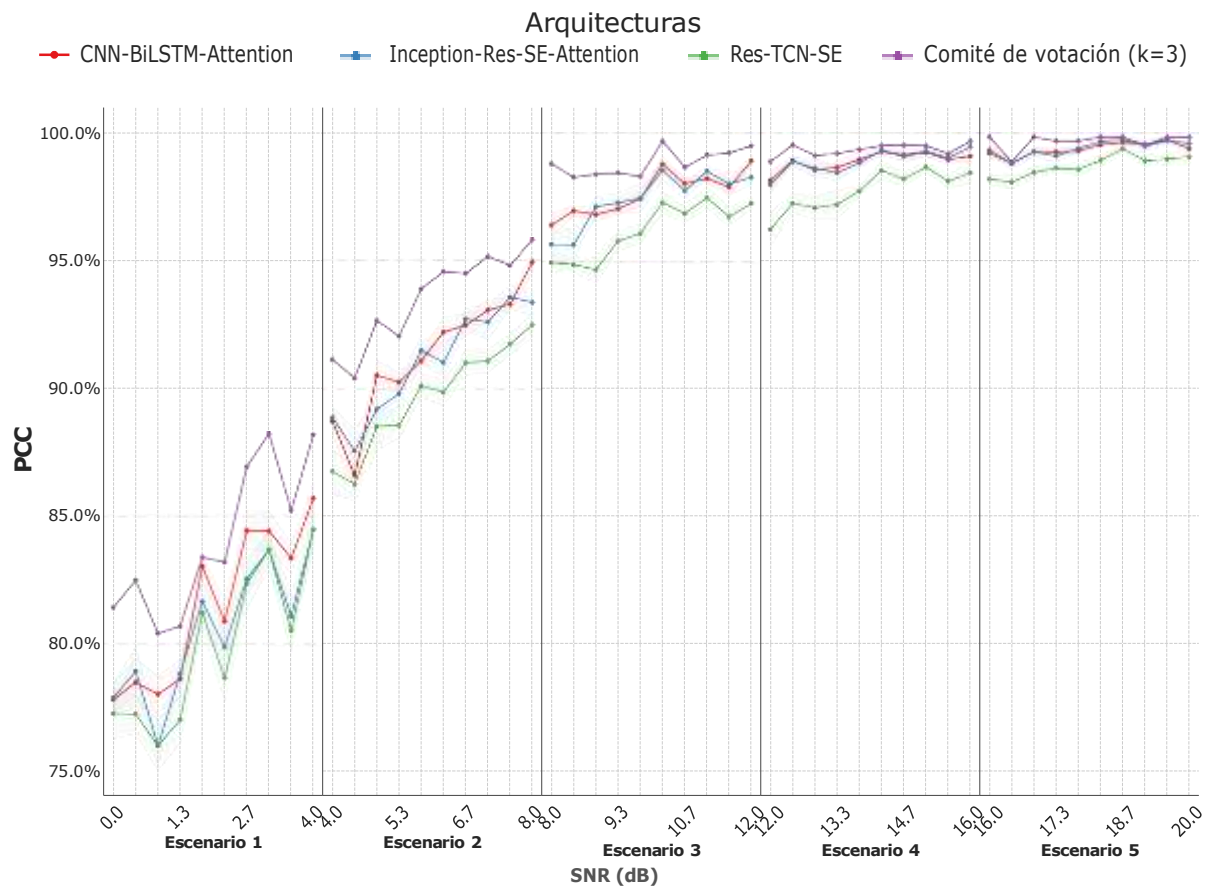


Figura 5.18: PCC vs. SNR: arquitecturas individuales vs. comité ($k = 3$).

La Figura 5.18 presenta la evolución del PCC en función del SNR para tres arquitecturas individuales y un sistema de votación basado en comité con $k = 3$. En este esquema, en lugar de emplear una única instancia por arquitectura, se generan tres modelos independientes por cada una de las tres arquitecturas, cada uno inicializado con diferentes pesos aleatorios. La decisión final se toma mediante votación mayoritaria entre las salidas de estas nueve redes.

Claramente este enfoque basado en comité mejora el desempeño en todos los rangos de SNR, particularmente en los escenarios de bajo y medio SNR. El comité mejora la robustez del sistema alcanzando valores de PCC superiores a los obtenidos de forma individual por arquitectura.

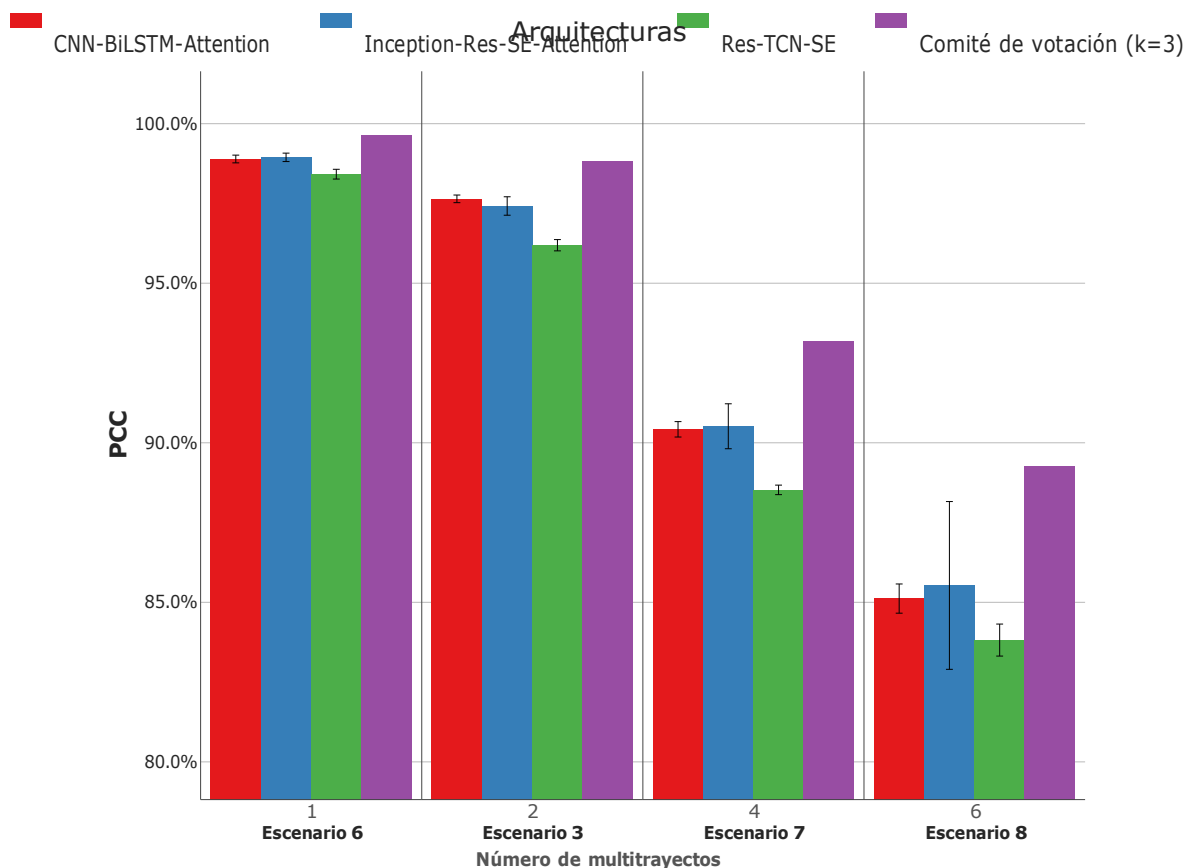


Figura 5.19: PCC vs. número de trayectos: arquitecturas individuales vs. comité ($k = 3$).

La Figura 5.19 muestra la variación del PCC en función del número de trayectos presentes en la señal.

Los resultados reflejan que el sistema de votación logra superar en todos los escenarios a las arquitecturas individuales. La ventaja es más evidente en los escenarios complejos de 4 y 6 trayectos, donde la precisión cae significativamente para los modelos individuales, pero el comité obtiene una precisión cercana al 90 %.

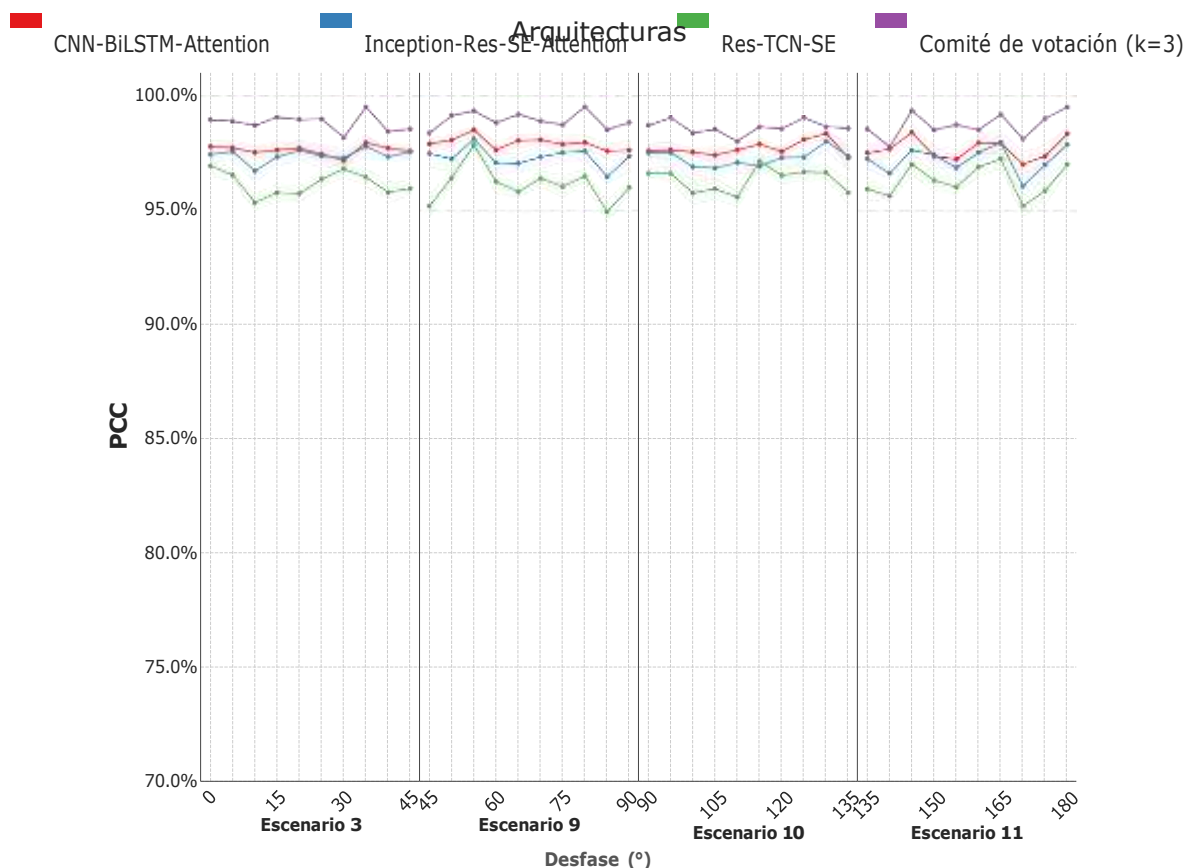


Figura 5.20: PCC vs. desfase: arquitecturas individuales vs. comité ($k = 3$).

La Figura 5.20 presenta la evolución del PCC frente a variaciones en el desfase en grados para los escenarios de variación de desfase.

Como ya se revisó y constató previamente, el desfase no es un parámetro que afecte significativamente la capacidad de las arquitecturas para distinguir modulaciones. Sin embargo, se observa que la tendencia del sistema de votación sigue siendo de provecho, ya que mejora levemente la capacidad de reconocimiento en este tipo de distorsión.

Discusión.

- En $SNR \geq 12$ dB la ganancia del comité es marginal (< 1 pp), pero para $SNR < 8$ dB asciende a 3 pp.
- Con 6 taps el aumento es de 4–5 pp, confirmando la mayor robustez frente a dispersión temporal.
- En conjunto, los resultados obtenidos a lo largo de los tres escenarios univariados evidencian que el sistema de votación basado en $k = 3$ instancias por arquitectura proporciona una mejora robusta frente a las fuentes de degradación impuestas.

5.5. Síntesis general

- El entrenamiento exhaustivo mejora el F1 promedio en 5 pp respecto a la validación temprana, con especial impacto en modulaciones de orden alto.
- La adición de 256-QAM confirma que la densidad de constelación representa el principal factor de dificultad: $\Delta PCC = -4.6$ pp en el escenario base.
- El voto mayoritario con $k = 3$ modelos por arquitectura ofrece una ganancia neta de 3 pp en promedio global y hasta 8 pp en escenarios de fuerte multitraectoria.

Líneas futuras. Trabajos posteriores pueden explorar técnicas de data augmentation espectral y pérdida focalizada para reducir la brecha de F1 en 256-QAM, así como estrategias de ponderación de votos basadas en la fiabilidad por clase.

5.6. Resumen del capítulo

Los resultados demuestran que, con un $SNR \geq 8dB$, las tres arquitecturas alcanzan un $PCC > 90\%$ y mantienen un F1 alto, validando la consistencia del aprendizaje temprano y exhaustivo. El número de trayectos se etiqueta como la fuente de degradación más crítica (hasta -25 pp en Inception-Res-SE-Attention), mientras que el desfase muestra un impacto apenas perceptible ($\Delta PCC < 2$ pp).

El experimento con siete modulaciones revela que las constelaciones de alta densidad (64-QAM y 256-QAM) concentran la mayor parte del error ($\approx 79\%$), a la vez que modulaciones cruzadas intermedias mejoran su F1 al superar el 90% .

Finalmente, el sistema de votación por comité con $k = 3$ modelos por arquitectura eleva el PCC hasta en un $+5$ pp en escenarios adversos y estabiliza la varianza.

Perspectivas de trabajo futuro:

- Integrar un *curriculum learning* progresivo de QPSK a 64-QAM con una *focal loss* adaptativa (γ) para centrar el entrenamiento en las regiones de mayor densidad y confusión.
- Generación de datos sintéticos y técnicas de *augmentation* focalizadas en constelaciones complejas.
- Votación ponderada y esquemas adaptativos de k según complejidad o varianza del escenario.
- Diseño de arquitecturas híbridas con módulos de atención espacial que refinen la separación de símbolos cercanos.

6. Conclusiones, recomendaciones y trabajos futuros

6.1. Conclusiones

Conclusiones generales: Las tres arquitecturas evaluadas alcanzan un PCC superior al 95 % para $\text{SNR} \geq 8 \text{ dB}$, destacando CNN-BiLSTM-Attn por su mejor rendimiento ante canales con múltiples trayectos. La inclusión de la modulación 256-QAM incrementa la complejidad del problema y provoca una reducción promedio de 9.43 pp en la precisión, donde Inception-Res-SE-Attn supera a las otras arquitecturas gracias a su capacidad para extraer características en distintas escalas. A pesar del aumento en el número de clases, la dificultad está más relacionada con la similitud geométrica entre constelaciones que con la cantidad total de modulaciones.

El reentrenamiento con un mayor volumen de datos y ajuste de hiperparámetros mejora la estabilidad y ofrece una convergencia adecuada. Las modulaciones de alto orden siguen siendo las más complejas de distinguir, destacando 64-QAM y 256-QAM (en el experimento con 7 modulaciones). Se detecta que el desfase angular tiene poco impacto en el PCC. Por otro lado, el sistema de votación por mayoría de votos con los $k = 3$ mejores modelos por arquitectura eleva el PCC promedio global hasta un 95.79 %; mejorando notablemente la precisión y estabilidad en escenarios con bajo SNR y canales con 4 y 6 trayectos. Este sistema de votación actúa como un mecanismo efectivo de regularización independiente del tipo de red.

Cumplimiento de objetivos: El presente trabajo ha cumplido satisfactoriamente el objetivo general planteado: desarrollar y evaluar un sistema robusto de clasificación automática de señales para comunicaciones inalámbricas, utilizando técnicas avanzadas de inteligencia artificial. A continuación detallamos las conclusiones de los objetivos específicos:

- **Objetivo 1:** Se realizó una exhaustiva revisión del estado del arte, revisando los métodos tradicionales así como los métodos basados en IA más relevantes para la clasificación automática de modulaciones (AMC). Este proceso fue clave para comprender los parámetros importantes para experimentar con arquitecturas neuronales, permitiendo también identificar las métricas adecuadas (*PCC*, *loss*, *F1*) para la comparación y seleccionar las modulaciones candidatas para el estudio.
- **Objetivo 2:** Se definieron 11 escenarios usando la biblioteca TorchSig, considerando modulaciones como BPSK, QPSK, 16-QAM, 32-QAM cruzada, 64-QAM, 128-QAM cruzada y 256-QAM, bajo condiciones controladas de AWGN, multitrayectorias y desfase de portadora. Se estudió en detalle la aplicabilidad de los sistemas AMC en diversos escenarios realistas del canal inalámbrico. Es así que, definimos 3 efectos de canal (ruido, multitrayectoria y desfase angular) que, en conjunto, permitieron simular 11 escenarios

diferentes, desde condiciones ideales (10dB con LoS sin multitrayectos) hasta entornos difíciles (6 trayectos acompañado de desfase de 0-45 grados).

- **Objetivo 3:** Se cumplió con el desarrollo e implementación exitosa de 3 arquitecturas de redes neuronales. Estas redes fueron expuestas a una fase de validación temprana la cual mostró resultados satisfactorios en términos de capacidad de clasificación en los 11 escenarios. Posteriormente, fueron entrenadas exhaustivamente con un conjunto de datos más amplio, es importante destacar la importancia del ajuste adecuado de hiperparámetros para asegurar que las redes converjan correctamente y generalicen bien en los distintos esquemas de modulación evaluados.
- **Objetivo 4:** se cumplió en varias etapas. Se comparó el desempeño de las tres arquitecturas utilizando el PCC en escenarios univariados. Los resultados confirmaron lo reportado en la literatura: un aumento de la relación señal-ruido (SNR) incrementa el PCC, mientras que un aumento en el número de trayectorias provoca una reducción significativa en esta métrica. En promedio, las redes lograron valores superiores al 95 % cuando el SNR fue ≥ 8 dB. Se evaluó también el efecto de añadir la modulación 256-QAM, observando una reducción promedio del 9 % en el PCC, lo que refleja la complejidad adicional introducida por modulaciones de alto orden. Como etapa final, se implementó un sistema de votación por mayoría, que mostró mejoras sustanciales de hasta 5 puntos porcentuales en los escenarios más difíciles.

6.2. Recomendaciones

Durante el desarrollo de este trabajo se constató que la etapa de validación de las redes neuronales debe abordarse con especial cuidado. Los parámetros ajustados en esta fase son clave para determinar la capacidad de aprendizaje y el desempeño final de cada arquitectura.

Asimismo, es fundamental seleccionar un tamaño adecuado para el conjunto de datos, de modo que se incluya una cantidad representativa y variada de señales.

6.3. Trabajos futuros

A partir de los resultados obtenidos, se identifican varias líneas de trabajo para mejorar los sistemas de AMC:

- **Ampliar el tamaño del conjunto de datos:** Incrementar la cantidad de ejemplos disponibles para entrenamiento podría ayudar a las redes neuronales a aprender de una mayor cantidad de patrones. Esto podría ser especialmente útil para modulaciones de alto orden.
- **Reducir la cantidad de símbolos:** Desarrollar técnicas que mantengan un buen nivel de clasificación con menos símbolos por señal es un reto importante para aplicaciones con restricciones de latencia o ancho de banda.

- **Preprocesamiento de señales:** Incorporar etapas de ecualización, a ciegas o con información del canal, para mitigar efectos de desvanecimiento y mejorar la calidad de la señal de entrada.
- **Arquitecturas en cascada:** Implementar sistemas jerárquicos o en cascada para facilitar la clasificación entre modulaciones similares.
- **Exploración de nuevas arquitecturas:** Evaluar modelos alternativos como Transformers, redes convolucionales profundas 1D o autoencoders para mejorar el reconocimiento.
- **Evaluación con señales reales:** Validar los modelos con datos reales capturados mediante un SDR para medir la robustez en escenarios prácticos.
- **Incorporación de nuevas degradaciones:** Considerar efectos adicionales como el desplazamiento Doppler, interferencia de canales adyacentes, desviación de frecuencia del oscilador local, entre otros, para simular escenarios más realistas y complejos.

Referencias

- [1] "Explore the ericsson mobility report november 2024." [En línea]. Disponible: <https://www.ericsson.com/en/reports-and-papers/mobility-report/reports/november-2024>
- [2] O. A. Dobre, A. Abdi, Y. Bar-Ness, y W. Su, "Survey of automatic modulation classification techniques: Classical approaches and new trends," *IET Communications*, vol. 1, num. 2, pp. 137–156, 2007.
- [3] J. Proakis y M. Salehi, *Digital Communications*, 5ésima ed. New York, NY: McGraw-Hill Professional, Nov. 2007.
- [4] B. Sklar, *Digital Communications: Fundamentals and Applications*. Old Tappan, NJ: Prentice Hall, 2001.
- [5] C. Pintos, "Diseño, simulación e implementación de códigos de canal en sistemas ofdm," 01 2014.
- [6] A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.
- [7] D. Tse y P. Viswanath, "Fundamentals of wireless communication," Online draft version, University of California, Berkeley and University of Illinois, Urbana-Champaign, Berkeley, CA and Urbana-Champaign, IL, Tech. Rep., 2005, draft, to be published by Cambridge University Press. [En línea]. Disponible: <https://web.stanford.edu/~dntse/papers/book121004.pdf>
- [8] J. G. Proakis y M. Salehi, *Digital Communications*, 4ésima ed. New York, NY: McGraw-Hill, 2001.
- [9] E. Azzouz y A. K. Nandi, *Automatic Modulation Recognition of Communication Signals*. Springer Science & Business Media, 1996.
- [10] T. J. O'Shea y J. Corgan, "Convolutional radio modulation recognition networks," *arXiv preprint arXiv:1602.04105*, 2016.
- [11] N. E. West, T. J. O'Shea, y T. C. Clancy, "Deep architectures for modulation recognition," in *2017 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2017.
- [12] I. Goodfellow, Y. Bengio, y A. Courville, *Deep Learning*. MIT Press, 2016.
- [13] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, num. 6, 1958.

- [14] Y. LeCun, L. Bottou, Y. Bengio, y P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, num. 11, 1998.
- [15] A. Krizhevsky, I. Sutskever, y G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, F. Pereira, C. Burges, L. Bottou, y K. Weinberger, Eds., vol. 25. Curran Associates, Inc., 2012. [En línea]. Disponible: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf
- [16] K. Simonyan y A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014. [En línea]. Disponible: <https://arxiv.org/abs/1409.1556>
- [17] K. He, X. Zhang, S. Ren, y J. Sun, "Deep residual learning for image recognition," 2015. [En línea]. Disponible: <https://arxiv.org/abs/1512.03385>
- [18] S. Hochreiter y J. Schmidhuber, "Long short-term memory," *Neural Computation*, 1997.
- [19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, y I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017.
- [20] J. Hu, L. Shen, S. Albanie, G. Sun, y E. Wu, "Squeeze-and-Excitation networks," 2017.
- [21] J. Kiefer y J. Wolfowitz, "Stochastic estimation of the maximum of a regression function," *The Annals of Mathematical Statistics*, vol. 23, num. 3, pp. 462–466, 1952.
- [22] D. P. Kingma y J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations (ICLR)*, 2015. [En línea]. Disponible: <https://arxiv.org/abs/1412.6980>
- [23] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, y R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, num. 56, pp. 1929–1958, 2014. [En línea]. Disponible: <http://jmlr.org/papers/v15/srivastava14a.html>
- [24] S. Ioffe y C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *International Conference on Machine Learning*, 2015.
- [25] P. Panagiotou, A. Anastasopoulos, y A. Polydoros, "Likelihood ratio tests for modulation classification," in *MILCOM 2000 Proceedings. 21st Century Military Communications. Architectures and Technologies for Information Superiority*, vol. 2, 2000, pp. 670–674.
- [26] Z. Zhu, "Automatic classification of digital communication signal modulations," Ph.D. dissertation, Brunel University London, UK, 2014.
- [27] F. Hameed, O. A. Dobre, y D. C. Popescu, "On the likelihood-based approach to modulation classification," *IEEE Transactions on Wireless Communications*, vol. 8, num. 12, pp. 5884–5892, 2009.
- [28] A. K. Ali y E. Erçelebi, "An m-qam signal modulation recognition algorithm in awgn channel," *Scientific Programming*, vol. 2019, pp. 1–17, May 2019.

- [29] F. Wang y X. Wang, "Fast and robust modulation classification via kolmogorov-smirnov test," *IEEE Transactions on Communications*, vol. 58, num. 8, pp. 2324–2332, 2010.
- [30] A. Hossen, F. Al-Wadahi, y J. A. Jervase, "Classification of modulation signals using statistical signal characterization and artificial neural networks," *Engineering Applications of Artificial Intelligence*, vol. 20, num. 4, p. 463–472, Jun. 2007. [En línea]. Disponible: <http://dx.doi.org/10.1016/j.engappai.2006.08.004>
- [31] Y. T. Chan, L. G. Gadbois, y P. Yansouni, "Identification of the modulation type of a signal," in *Proceedings of the 1985 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, 1985, pp. 838–841.
- [32] S. Peng, H. Jiang, H. Wang, H. S. Alwageed, y Y.-d. Yao, "Modulation classification using convolutional neural network based deep learning model," *2017 26th Wireless and Optical Communication Conference (WOCC)*, pp. 1–5, 2017.
- [33] C.-F. Teng, C.-C. Liao, C.-H. Chen, y A.-Y. A. Wu, "Polar feature based deep architectures for automatic modulation classification considering channel fading," in *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2018, pp. 554–558.
- [34] N. Kim, N. Kehtarnavaz, M. Yearly, y S. Thornton, "Dsp-based hierarchical neural network modulation signal classification," *IEEE Transactions on Neural Networks*, vol. 14, num. 5, p. 1065–1071, Sep. 2003. [En línea]. Disponible: <http://dx.doi.org/10.1109/TNN.2003.816037>
- [35] R. M. Al-Makhlaway, A. A. Hefnawy, M. M. Abd Elnaby, y F. E. Abd El-Samie, "Modulation classification in the presence of adjacent channel interference using convolutional neural networks," *International Journal of Communication Systems*, vol. 33, num. 13, p. e4295, Feb. 2020. [En línea]. Disponible: <http://dx.doi.org/10.1002/dac.4295>
- [36] M. Sadeghi y E. G. Larsson, "Adversarial attacks on deep-learning based radio signal classification," *IEEE Wireless Communications Letters*, vol. 8, num. 1, pp. 213–216, Feb. 2019.
- [37] DeepSig, "Datasets," <https://www.deepsig.ai/datasets/>, 2024, accedido en junio de 2025.
- [38] T. Contributors, "Torchsig: A pytorch signal processing and machine learning toolkit," GitHub repository, <https://github.com/TorchDSP/torchsig>, 2023, accessed: 2025-05-10.
- [39] L. Boegner, M. Gulati, G. Vanhoy, P. Vallance, B. Comar, S. Kokalj-Filipovic, C. Lennon, y R. D. Miller, "Large scale radio frequency signal classification," 2022, arXiv:2207.09918. [En línea]. Disponible: <https://arxiv.org/abs/2207.09918>
- [40] Google Research, "Google colab,," <https://colab.research.google.com/>, 2018, accedido: 2025-05-10.
- [41] Kaggle Inc., "Kaggle: Your machine learning and data science community," <https://www.kaggle.com/>, 2010, accedido: 2025-05-10.

- [42] J. P. Pérez, "AMC: Flujos de trabajo y pipelines para entrenamiento, evaluación y comparación de redes neuronales de clasificación." https://github.com/juanpablo-perez/AMC_ArQ, 2025, Último acceso: 29 de junio de 2025.