

MODELOS DE APRENDIZAJE AUTOMÁTICO PARA MANTENIMIENTO PREDICTIVO

ANDRÉS VICENTE MARTÍNEZ

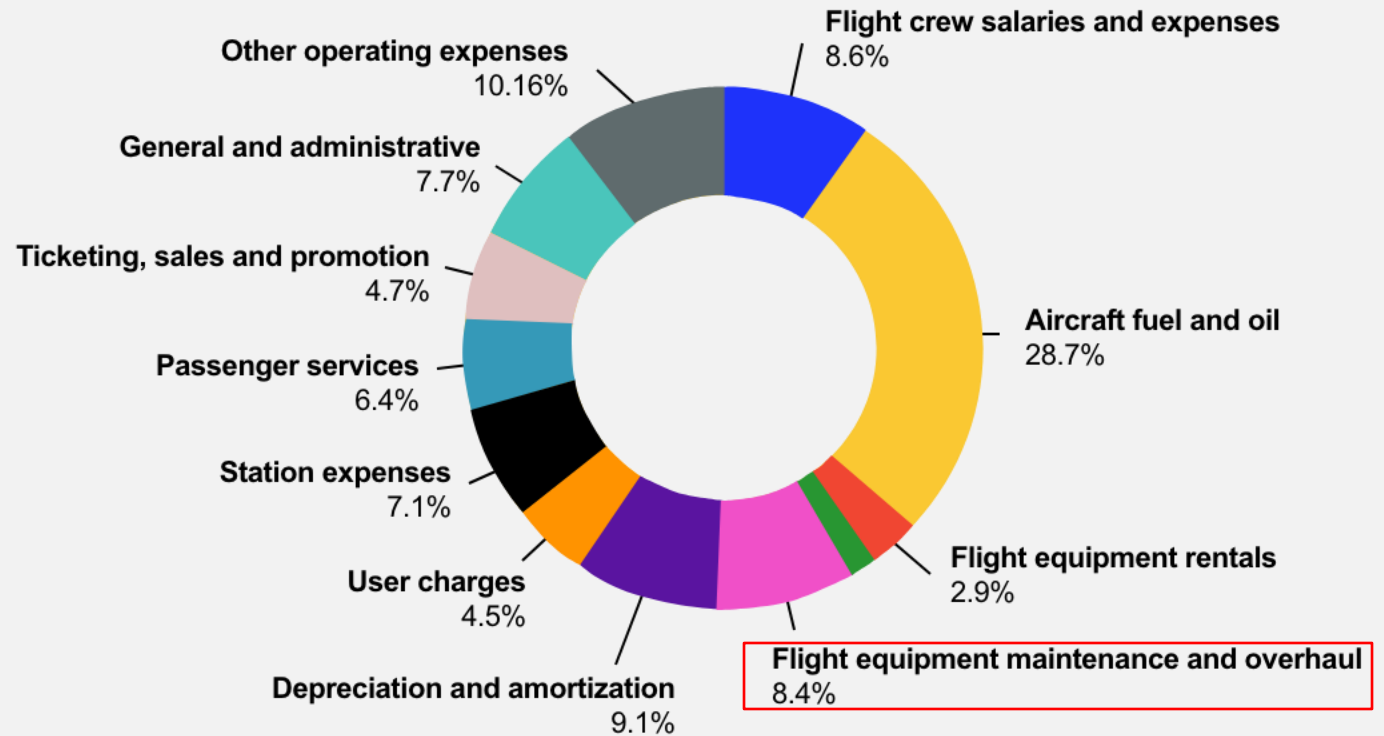
SEPTIEMBRE 2025

CONTENIDOS

1. Mantenimiento predictivo y Machine Learning
2. Caso 1: Aprendizaje Supervisado
3. Caso 2: Aprendizaje No Supervisado
4. Conclusiones y Futuros Trabajos

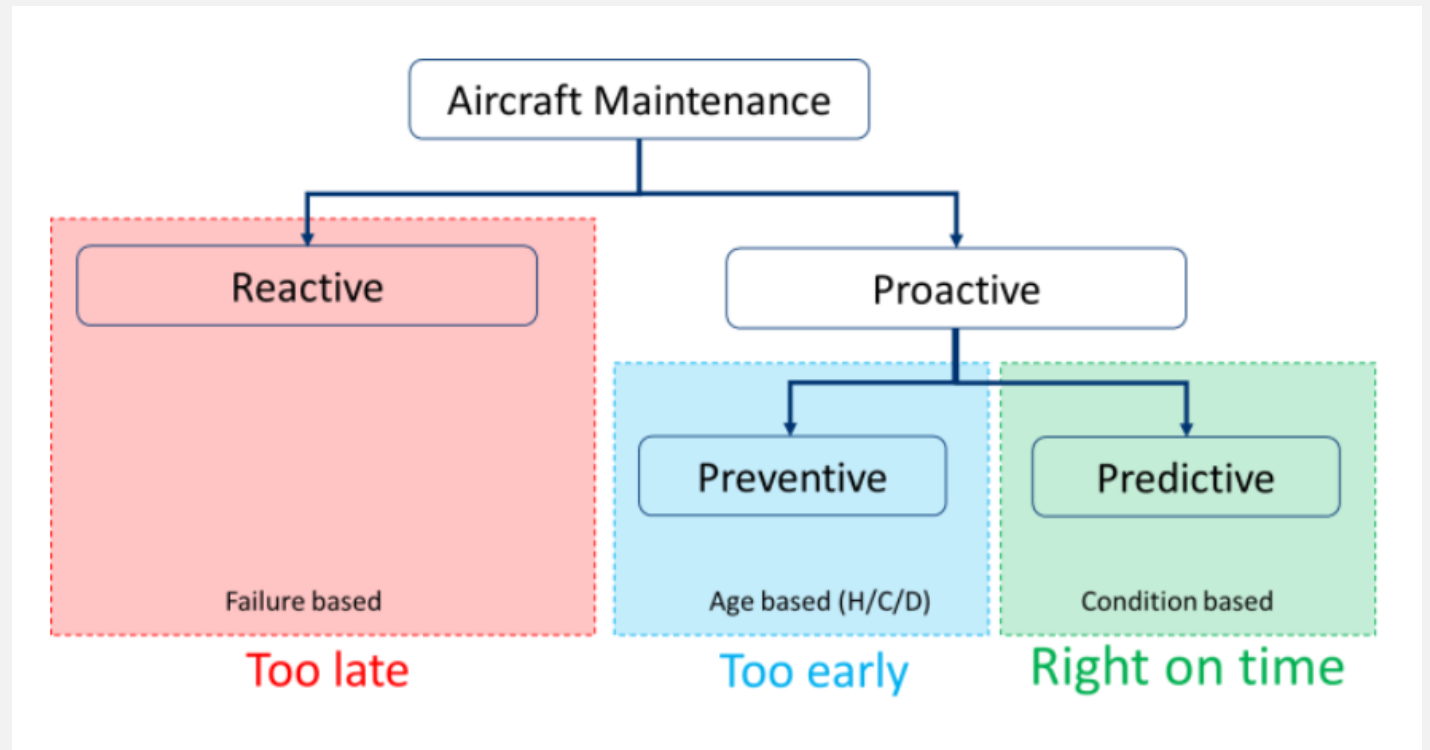
MANTENIMIENTO PREDICTIVO

- Fase crítica: mantenimiento, reparación y revisión mayor.
- Impacto en seguridad operacional.
- Representa el **8,4 % de los costes de las aerolíneas** (IATA, 2020).



¿QUÉ ES EL MANTENIMIENTO PREDICTIVO?

- Estrategias según momento de reparación
- Reactivo: tras fallo.
Solo en piezas no críticas.
- Preventivo: antes del fallo.
Riesgo de revisiones innecesarias
- Predictivo: en el momento óptimo.
Equilibrio entre costes y seguridad



APRENDIZAJE AUTOMÁTICO

- Machine Learning → procesar **grandes volúmenes de datos** de **sensores**.
- Permite diagnosticar fallos y degradación con fiabilidad.
- Dos enfoques principales:
 - Supervisado: → requiere datos etiquetados
→ clasificación o regresión
 - No supervisado: → identifica patrones sin etiquetas
→ clustering

Adquisición de Datos (sensores)

Preprocesamiento de Datos

Extracción de Características

Modelado (Machine Learning)

Prognosis

Planificación del Mantenimiento

CASO I: APRENDIZAJE SUPERVISADO

OBJETIVO Y DATOS

- **Dataset:** 1372 entradas de señales
- **Cuatro variables** estadísticas:
 - Varianza
 - Skewness
 - Curtosis
 - Entropía
- **Clase binaria:** normal (1) y fallo (0)
- Objetivo: predecir clase usando **decision trees, random forest y k-NN**.

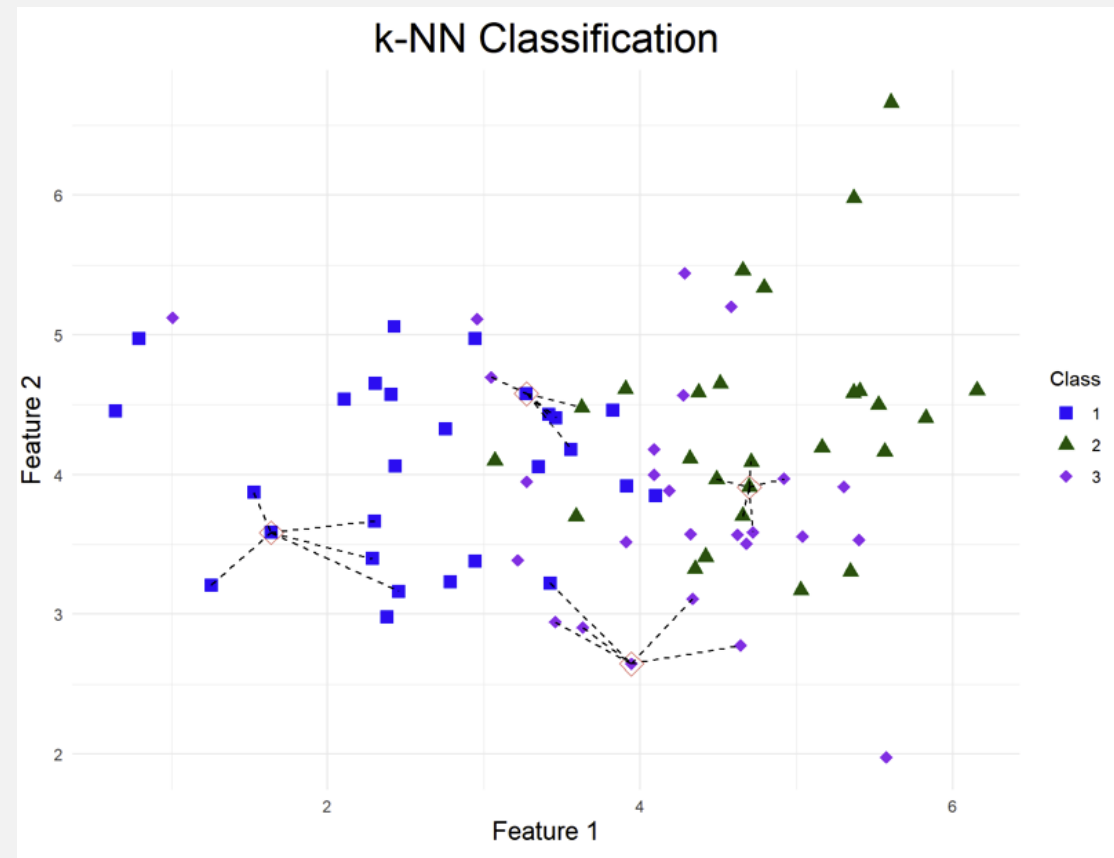
	Variance	Skewness	Curtosis	Entropy	Class
count	1372,000	1372,000	1372,000	1372,000	1372,000
mean	0,434	1,922	1,398	-1,192	0,445
std	2,843	5,869	4,310	2,101	0,497
min	-7,042	-13,773	-5,286	-8,548	0,000
25 %	-1,773	-1,708	-1,575	-2,413	0,000
50 %	0,496	2,320	0,617	-0,587	0,000
75 %	2,821	6,815	3,179	0,395	1,000
max	6,825	12,952	17,927	2,450	1,000

Variance	Skewness	Curtosis	Entropy	Class
3.6216	8.6661	-2.8073	-0.4470	0
4.5459	8.1674	-2.4586	-1.4621	0
3.8660	-2.6383	1.9242	0.1065	0
3.4566	9.5228	-4.0112	-3.5944	0
0.3292	-4.4552	4.5718	-0.9888	0

CASO I: APRENDIZAJE SUPERVISADO

ALGORITMOS: K – NEAREST NEIGHBORS

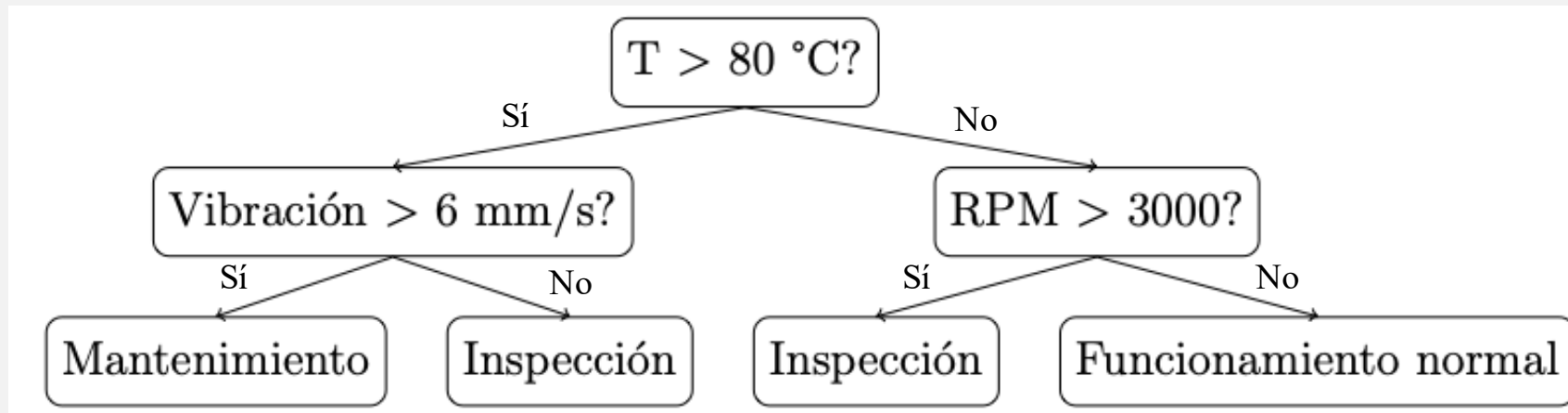
- Parámetro clave: **número de vecinos k**
- **Distancia euclídea** para clasificar un nuevo punto.
- Los resultados empeoran al aumentar la dimensionalidad



CASO I: APRENDIZAJE SUPERVISADO

ALGORITMOS: DECISION TREE

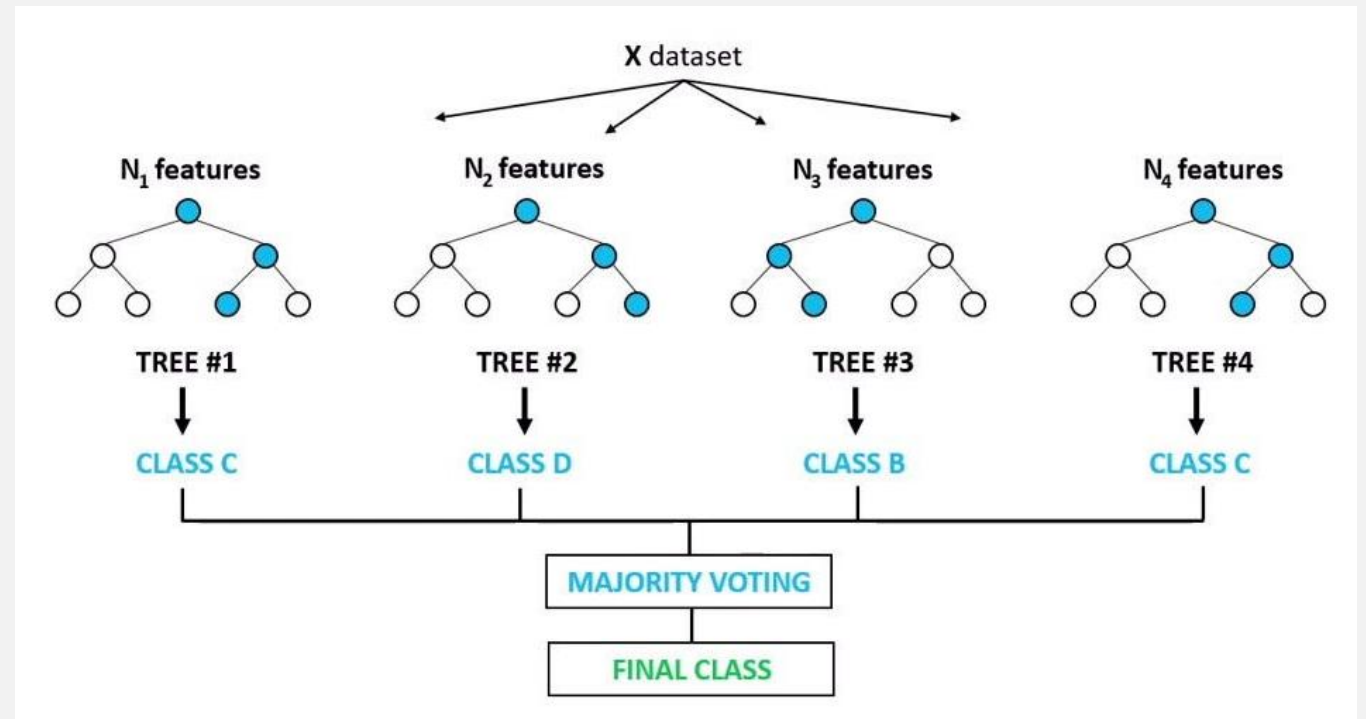
- Clasificación **mediante reglas secuenciales**
- Ventaja: **visualización gráfica**
- Riesgo: **overfitting** si demasiada profundidad.



CASO I: APRENDIZAJE SUPERVISADO

ALGORITMOS: RANDOM FOREST

- Serie de Decision Trees
- Selección del **número de árboles** ($N_{árboles} = 100$)
- Votación final



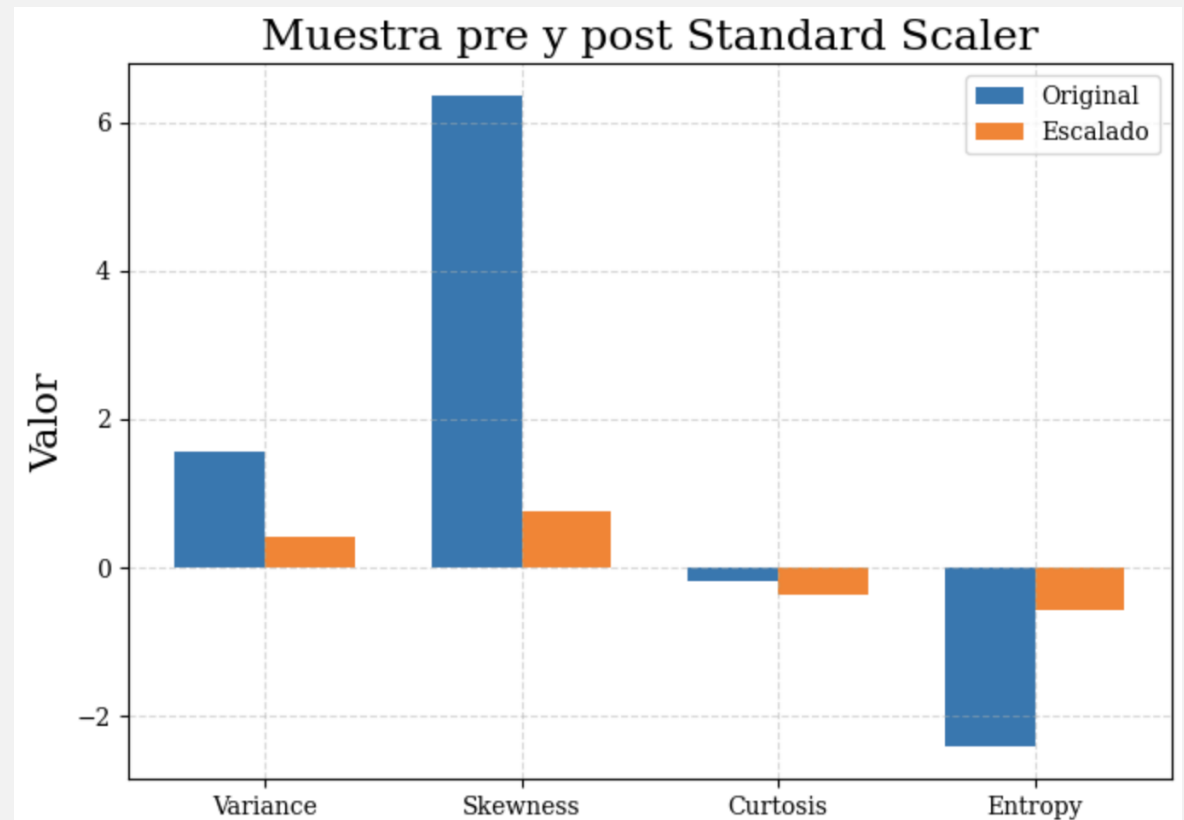
CASO I: APRENDIZAJE SUPERVISADO

PREPROCESADO Y NORMALIZACIÓN

- Normalización con **Standard Scaler**:

$$x_{scaled} = \frac{x - \bar{x}}{s}$$

- **División** del dataset
 - 70 % Entrenamiento
 - 30 % Test
- Primera **clasificación** única
- **Validación cruzada** (k-fold)



CASO I: APRENDIZAJE SUPERVISADO

RESULTADOS: CLASIFICACIÓN ÚNICA

- k-NN alcanza precisión del 100%
- Rendimiento general muy elevado
- Validación con k-Folds para comprobar generalización

Decision Tree

Real	Predicción	
	0	1
0	227	2
1	6	177

Random Forest

Real	Predicción	
	0	1
0	229	0
1	1	182

K- Nearest Neighbors

Real	Predicción	
	0	1
0	229	0
1	0	183

CASO I: APRENDIZAJE SUPERVISADO

RESULTADOS: VALIDACIÓN EN K-FOLDS Y COMPARACIÓN CON ESTUDIO ORIGINAL

- 5 particiones del dataset: 4 para entrenamiento y una para test
- Media entre resultados
- Accuracy, Precision, Recall y F1 > 97 %

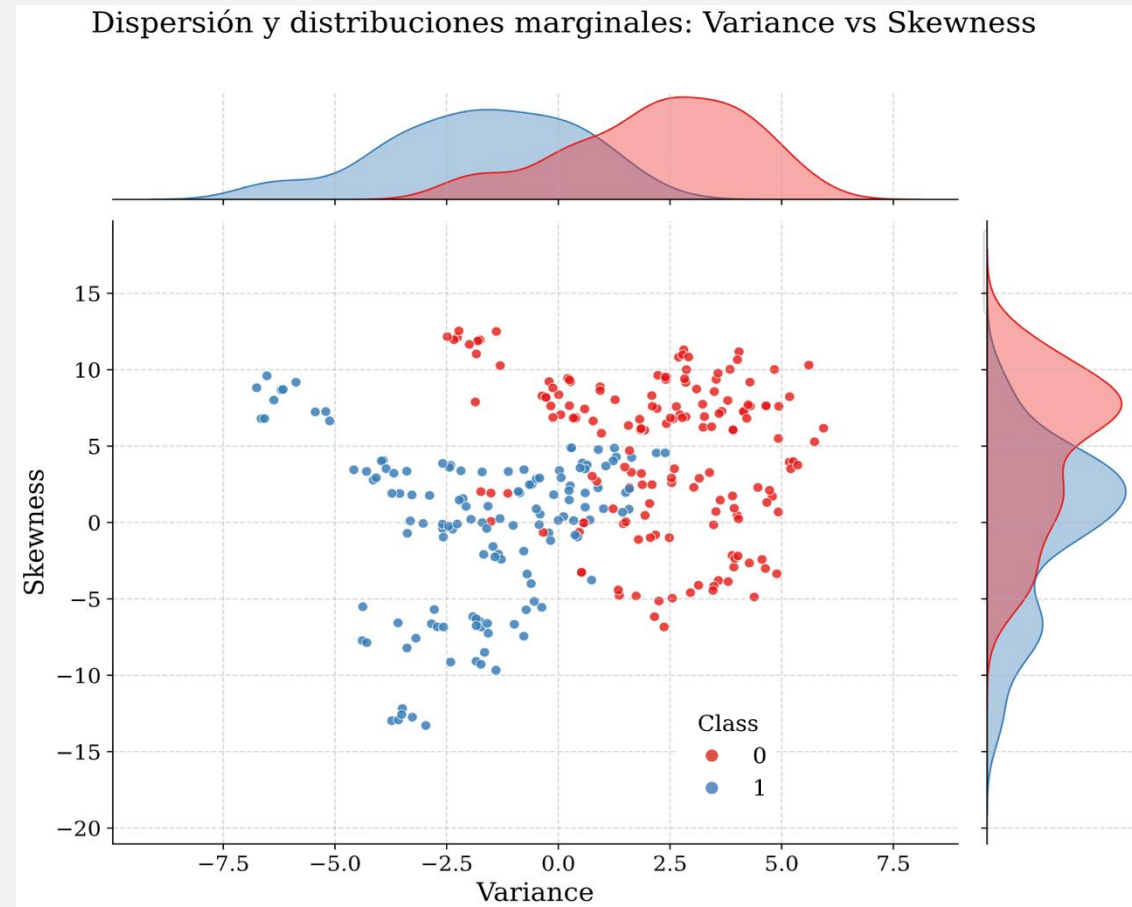
Modelo	Accuracy	Precisión	Recall	F1-score
Decision Tree	0.9811 \pm 0.0071	0.9756 \pm 0.0087	0.9820 \pm 0.0120	0.9787 \pm 0.0080
Random Forest	0.9956 \pm 0.0015	0.9903 \pm 0.0032	1.0000 \pm 0.0000	0.9951 \pm 0.0016
KNN	0.9985 \pm 0.0018	0.9967 \pm 0.0040	1.0000 \pm 0.0000	0.9984 \pm 0.0020

	Accuracy		Tiempo de Predicción (s)	
	Referencia	Experimental	Referencia	Experimental
Decision Tree	0.976	0.981	0.864	0.0002
Random Forest	0.984	0.996	4.245	0.0096
KNN	0.989	0.998	3.251	0.0035

CASO I: APRENDIZAJE SUPERVISADO

CONCLUSIONES

- Representación 2D entre variables
- Separación clara entre clases
- Dificultad reducida
- Métricas elevadas
- Limitación: dependencia de datos etiquetados



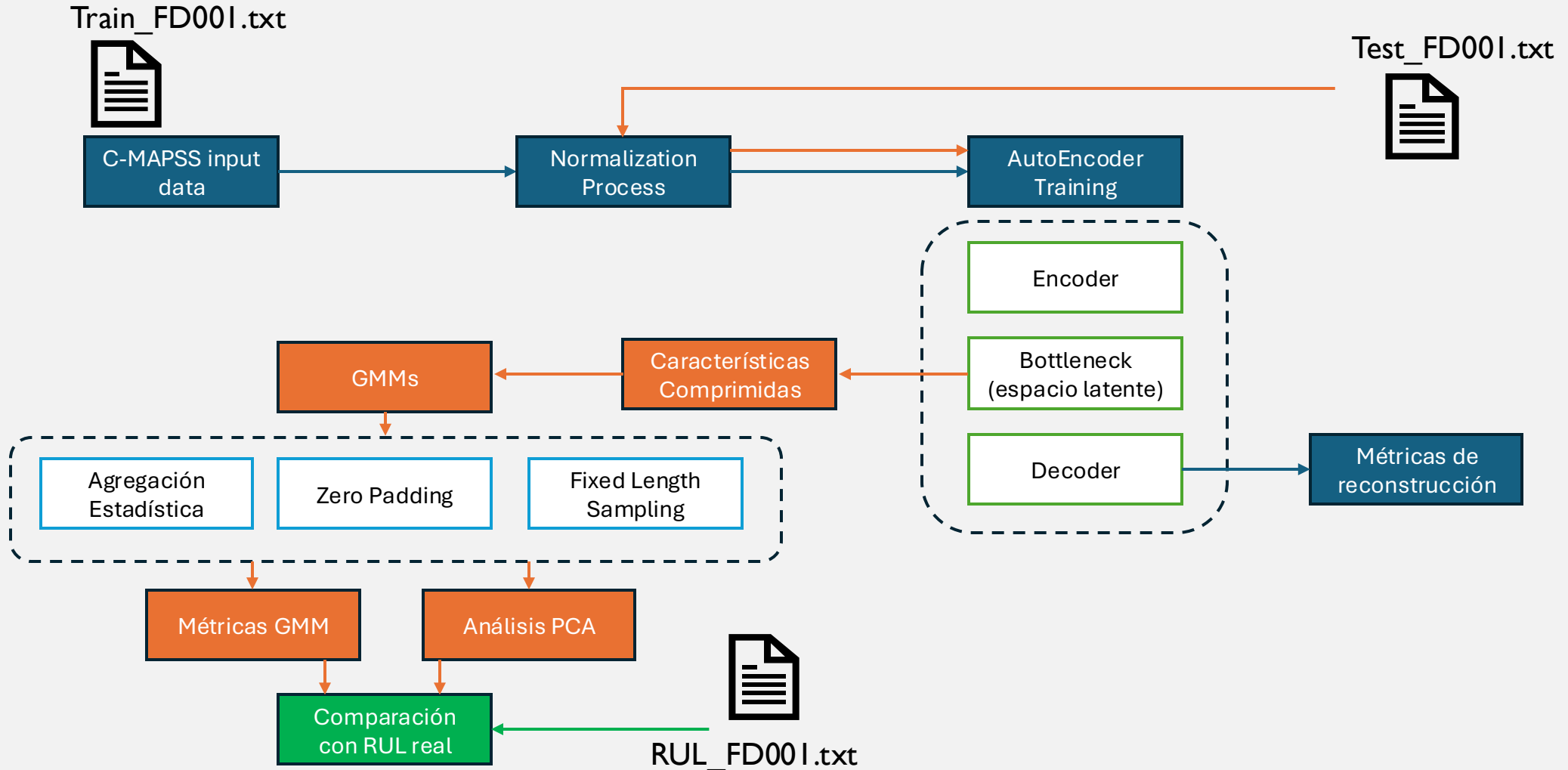
CASO 2: APRENDIZAJE NO SUPERVISADO

OBJETIVO Y DATOS

- Datos CMAPSS (NASA)
- Turbofán simulado
- Datasets disponibles:
 - Entrenamiento
 - Test: 100 motores
 - Vida útil remanente (RUL)
- Limitación: Motores con diferente número de ciclos
- Objetivo: clasificación de motores según estado de salud

Variable	Media	Desv. Típica	Mínimo	Máximo	Constante
op_setting_1	0,000	0,002	-0,0087	0,0087	No
op_setting_2	0,000	0,0003	-0,0006	0,0006	No
op_setting_3	100,000	0,000	100,000	100,000	Sí
sensor_1	518,670	0,000	518,670	518,670	Sí
sensor_2	642,681	0,500	641,210	644,530	No
sensor_3	1590,523	6,131	1571,040	1616,910	No
sensor_4	1408,934	9,001	1382,250	1441,490	No
sensor_5	14,620	0,000	14,620	14,620	Sí
sensor_6	21,610	0,000	21,610	21,610	Sí
sensor_7	555,277	1,272	550,500	561,000	No
sensor_8	2388,000	0,000	2388,000	2388,000	Sí
sensor_9	9046,430	22,705	8980,370	9165,500	No
sensor_10	1,300	0,000	1,300	1,300	Sí
sensor_11	47,590	0,014	47,530	47,650	No
sensor_12	521,413	0,738	518,690	523,380	No
sensor_13	2388,096	0,072	2387,880	2388,560	No
sensor_14	8143,753	19,076	8099,940	8293,720	No
sensor_15	8,442	0,038	8,3249	8,5848	No
sensor_16	0,030	0,000	0,030	0,030	Sí
sensor_17	393,211	1,549	388,000	400,000	No
sensor_18	2388,000	0,000	2388,000	2388,000	Sí
sensor_19	100,000	0,000	100,000	100,000	Sí
sensor_20	38,816	0,181	38,140	39,430	No
sensor_21	23,290	0,108	22,8942	23,6184	No

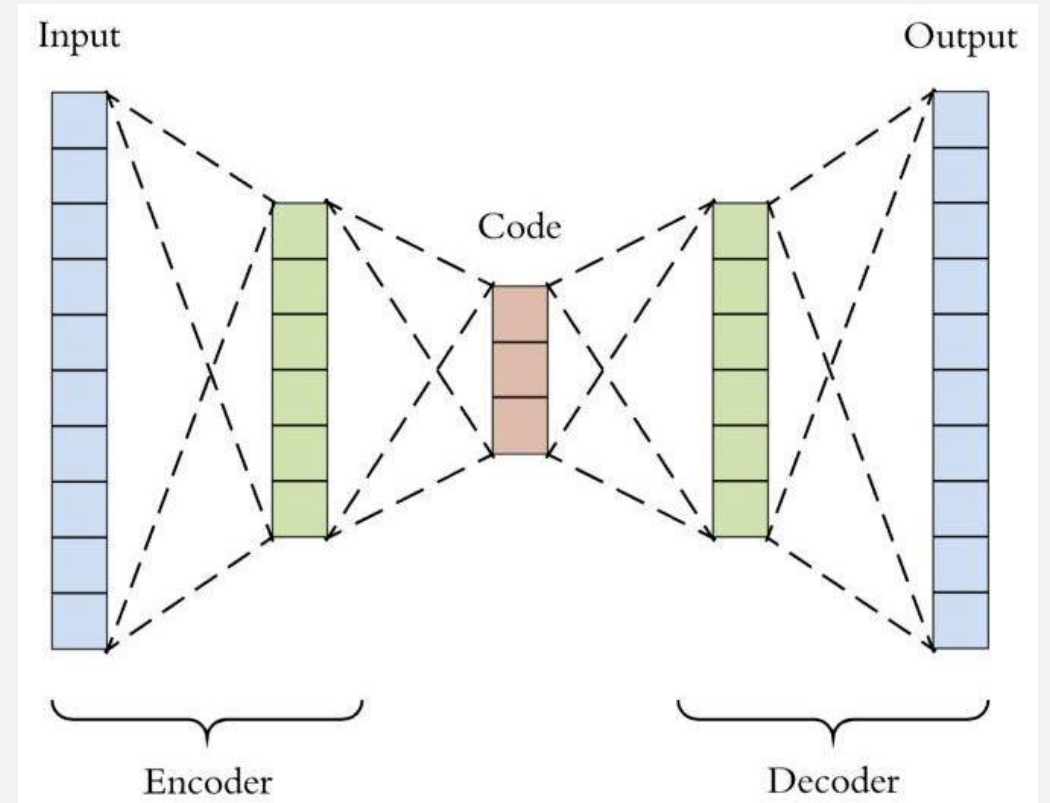
CASO 2: APRENDIZAJE NO SUPERVISADO METODOLOGÍA



CASO 2: APRENDIZAJE NO SUPERVISADO ENTRENAMIENTO DEL AUTOENCODER

- Proceso iterativo
- Necesidad de menos neuronas
- Artículo original: data set con datos reales y número más elevado de dimensiones

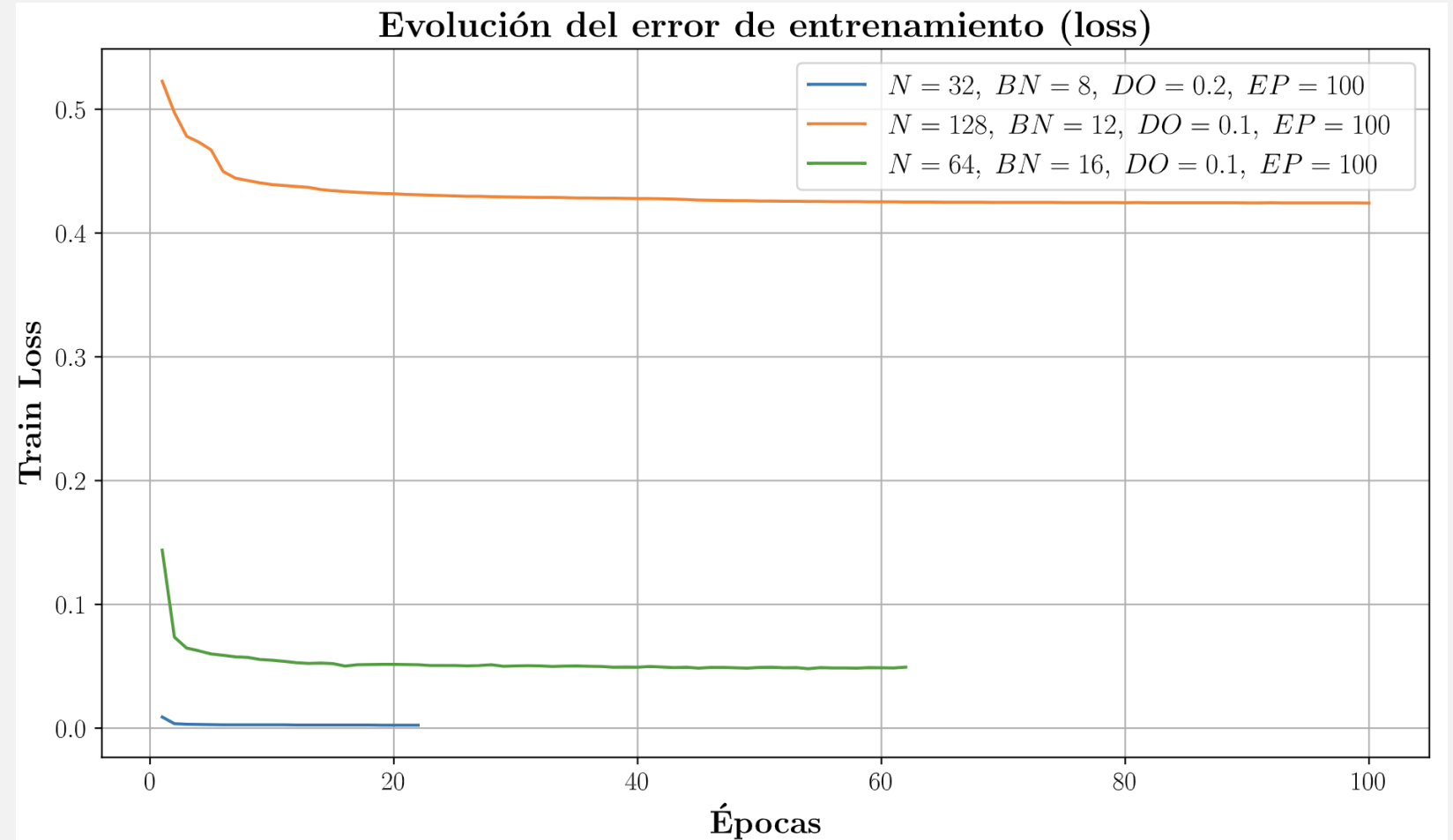
Capa	Tipo	# Neuronas (Real)	# Neuronas (CMAPSS)
1	Input	86	24
2	Encoder	(2x)128	32
3	Espacio Latente	32	8
4	Decoder	(2x)128	32
5	Output	86	24



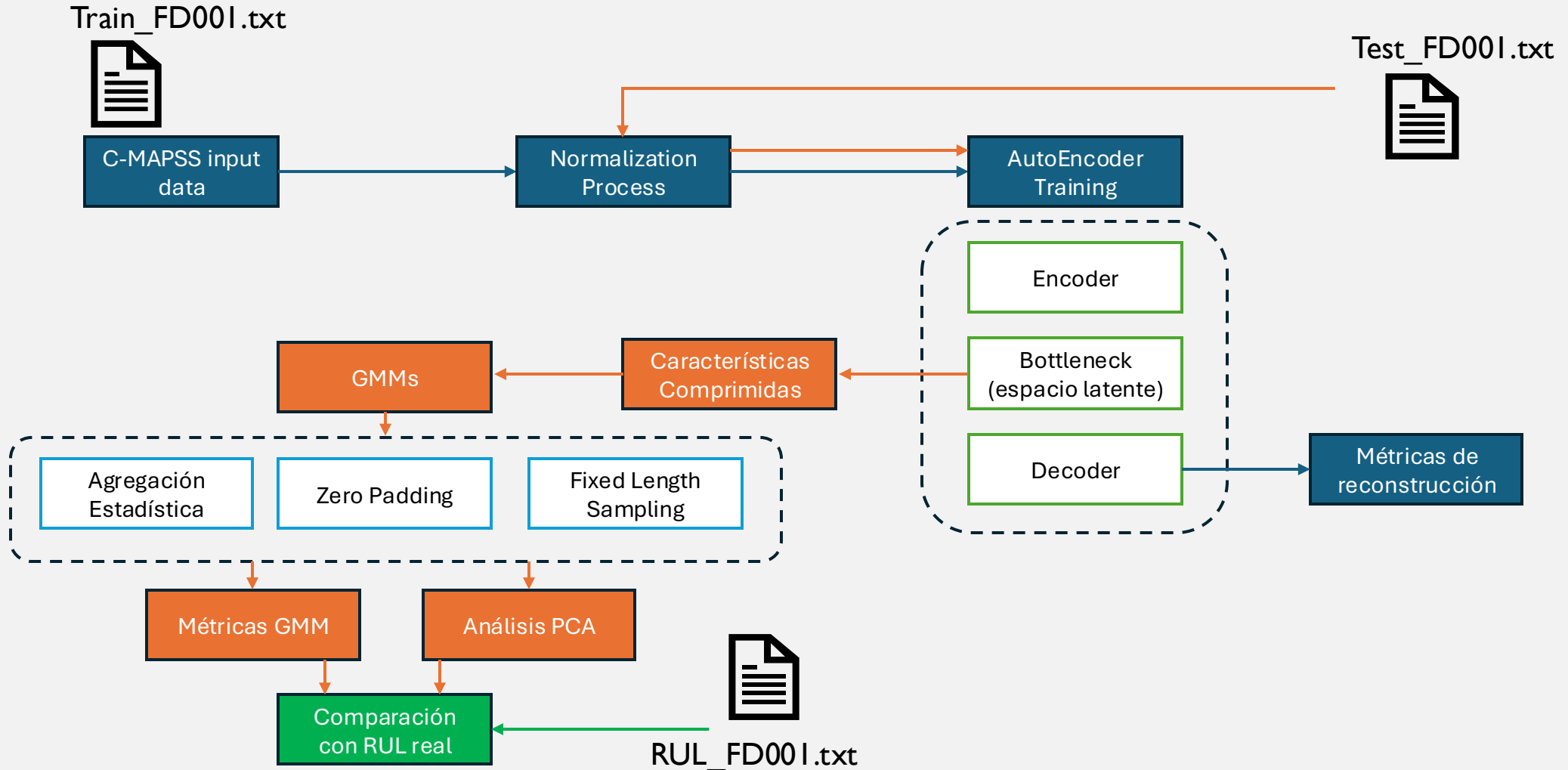
CASO 2: APRENDIZAJE NO SUPERVISADO ENTRENAMIENTO DEL AUTOENCODER

Parámetros de iteración:

- Neuronas capa latente (BN)
- Neuronas capas densas (N)
- Ratio de dropout (DO)
- Número de épocas (EP)



CASO 2: APRENDIZAJE NO SUPERVISADO METODOLOGÍA



CASO 2: APRENDIZAJE NO SUPERVISADO

CONSTRUCCIONES DE MATRIZ GMM

- **Espacio latente de 8 dimensiones**
- Cada motor tiene ciclos diferentes
- **Objetivo: crear un único vector por motor**
- Estrategias:
 - **Agregación Estadística**
 - **Zero Padding**
 - **Muestreo Fijo (75 ciclos)**

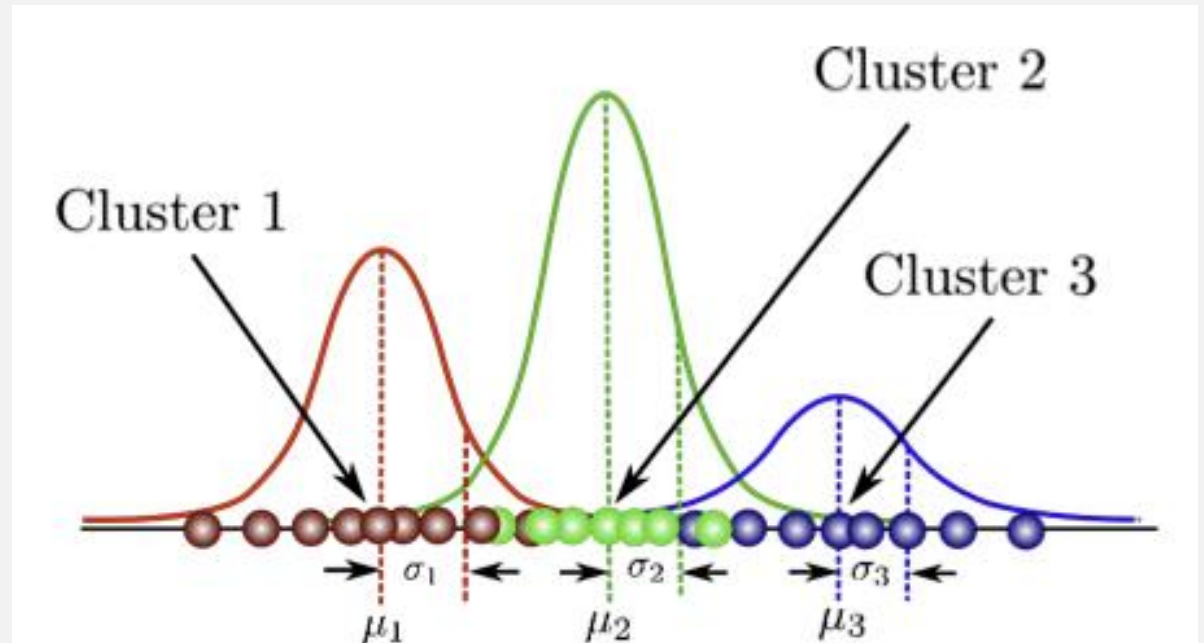
	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆	C ₇	C ₈	C ₉	C ₁₀
Motor 1	$M_1^{(1)}$	$M_1^{(2)}$	$M_1^{(3)}$	$M_1^{(4)}$	$M_1^{(5)}$	ZP	ZP	ZP	ZP	ZP
Motor 2	$M_2^{(1)}$	$M_2^{(2)}$	$M_2^{(3)}$	ZP	ZP	ZP	ZP	ZP	ZP	ZP
Motor 3	$M_3^{(1)}$	$M_3^{(2)}$	$M_3^{(3)}$	$M_3^{(4)}$	$M_3^{(5)}$	$M_3^{(6)}$	$M_3^{(7)}$	$M_3^{(8)}$	ZP	ZP
					⋮					
Motor 98	$M_{98}^{(1)}$	$M_{98}^{(2)}$	$M_{98}^{(3)}$	$M_{98}^{(4)}$	$M_{98}^{(5)}$	$M_{98}^{(6)}$	$M_{98}^{(7)}$	$M_{98}^{(8)}$	$M_{98}^{(9)}$	$M_{98}^{(10)}$
Motor 99	$M_{99}^{(1)}$	$M_{99}^{(2)}$	$M_{99}^{(3)}$	$M_{99}^{(4)}$	$M_{99}^{(5)}$	$M_{99}^{(6)}$	$M_{99}^{(7)}$	$M_{99}^{(8)}$	ZP	ZP
Motor 100	$M_{100}^{(1)}$	$M_{100}^{(2)}$	$M_{100}^{(3)}$	$M_{100}^{(4)}$	ZP	ZP	ZP	ZP	ZP	ZP

ID	Media			Desv. Típica			Valor Máximo		
1	$\bar{x}_1^{(1)}$...	$\bar{x}_1^{(8)}$	$\sigma_1^{(1)}$...	$\sigma_1^{(8)}$	$m_1^{(1)}$...	$m_1^{(8)}$
2	$\bar{x}_2^{(1)}$...	$\bar{x}_2^{(8)}$	$\sigma_2^{(1)}$...	$\sigma_2^{(8)}$	$m_2^{(1)}$...	$m_2^{(8)}$
⋮	⋮	⋱	⋮	⋮	⋱	⋮	⋮	⋱	⋮
100	$\bar{x}_{100}^{(1)}$...	$\bar{x}_{100}^{(8)}$	$\sigma_{100}^{(1)}$...	$\sigma_{100}^{(8)}$	$m_{100}^{(1)}$...	$m_{100}^{(8)}$

CASO 2: APRENDIZAJE NO SUPERVISADO

MODELOS DE MEZCLA GAUSSIANA

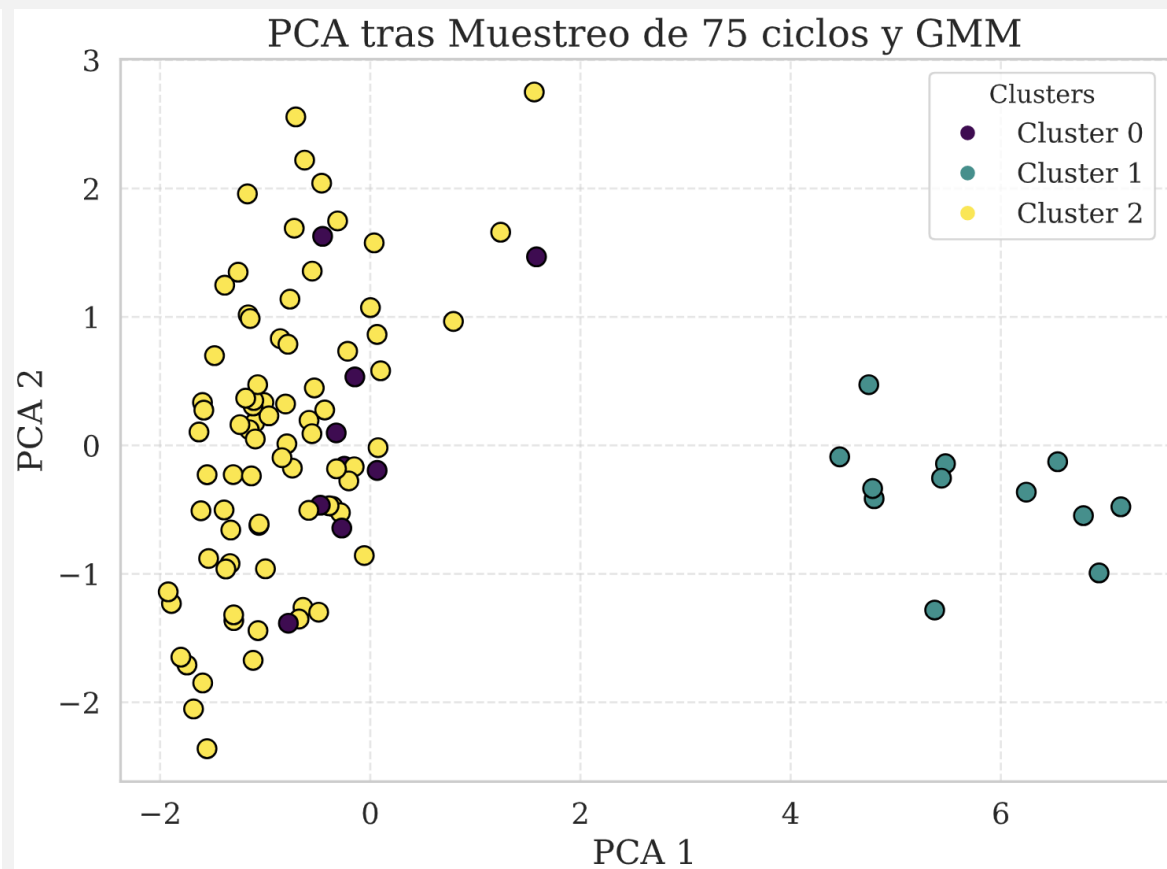
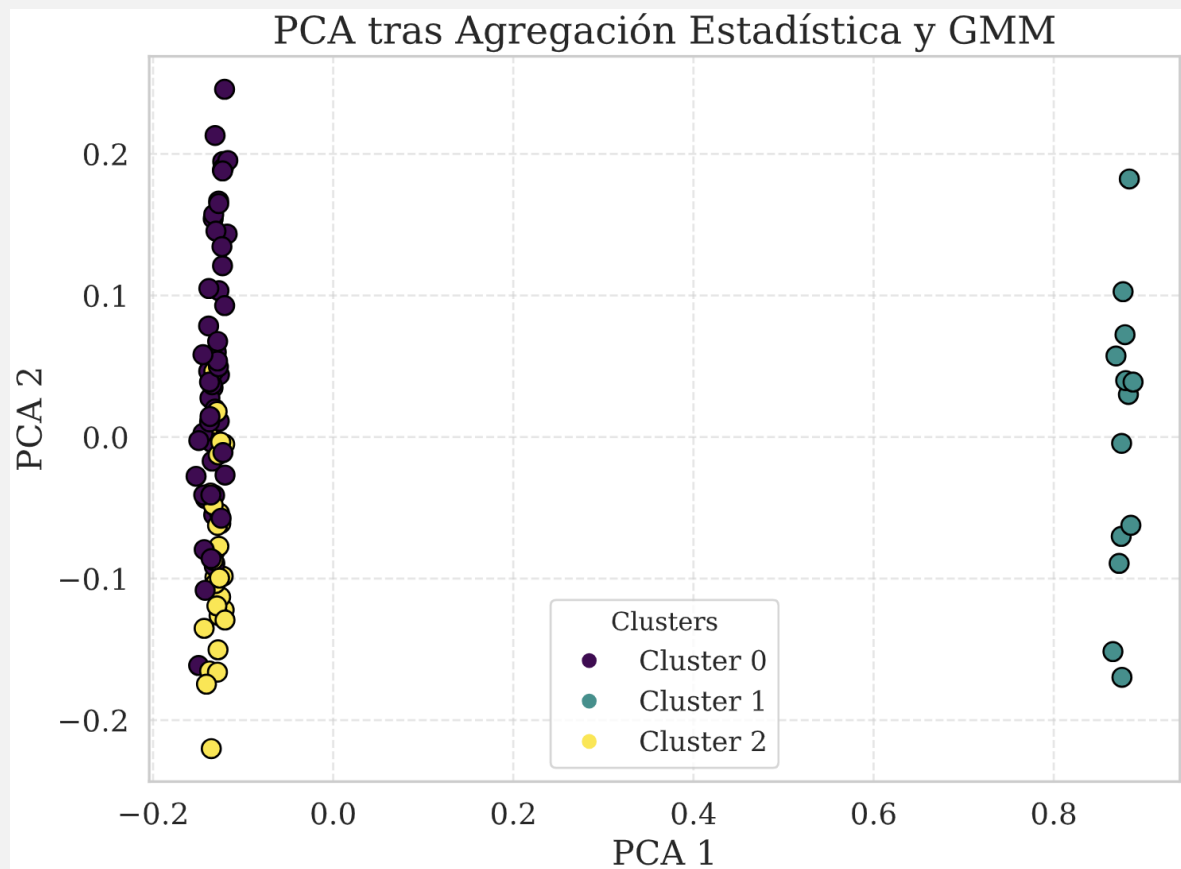
- Clasificación **probabilística**
- Se escoge el **número de clusters** deseados
- $\sum_{i=0}^2 p_i = 1 \rightarrow$ la suma de probabilidades de pertenencia para cada punto es uno
- Se realiza para cada estrategia



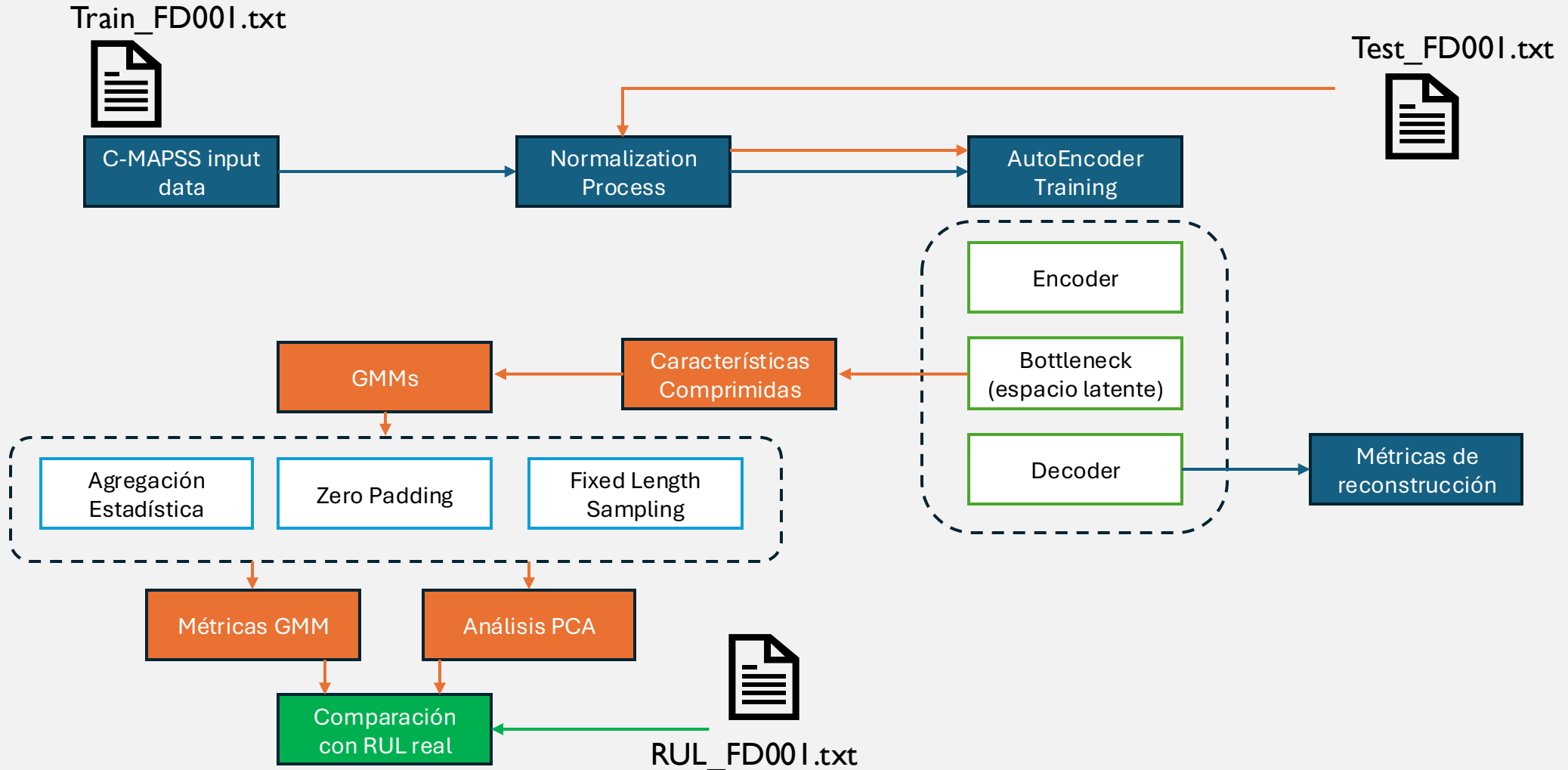
CASO 2: APRENDIZAJE NO SUPERVISADO

RESULTADOS: ANÁLISIS DE COMPONENTES PRINCIPALES

- Análisis de varianzas y representación 2D



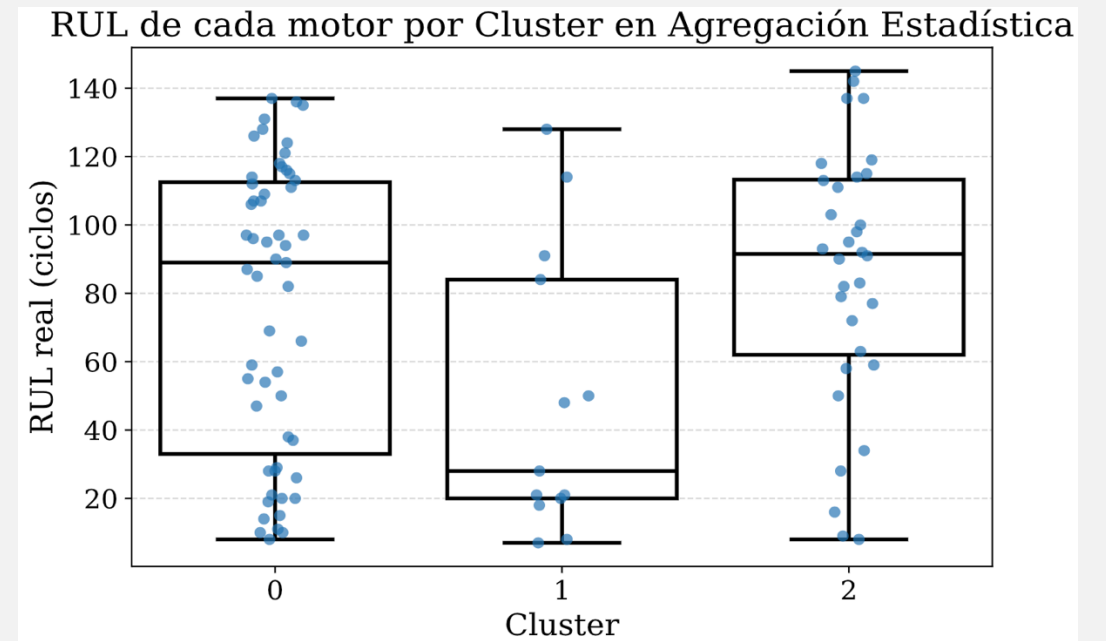
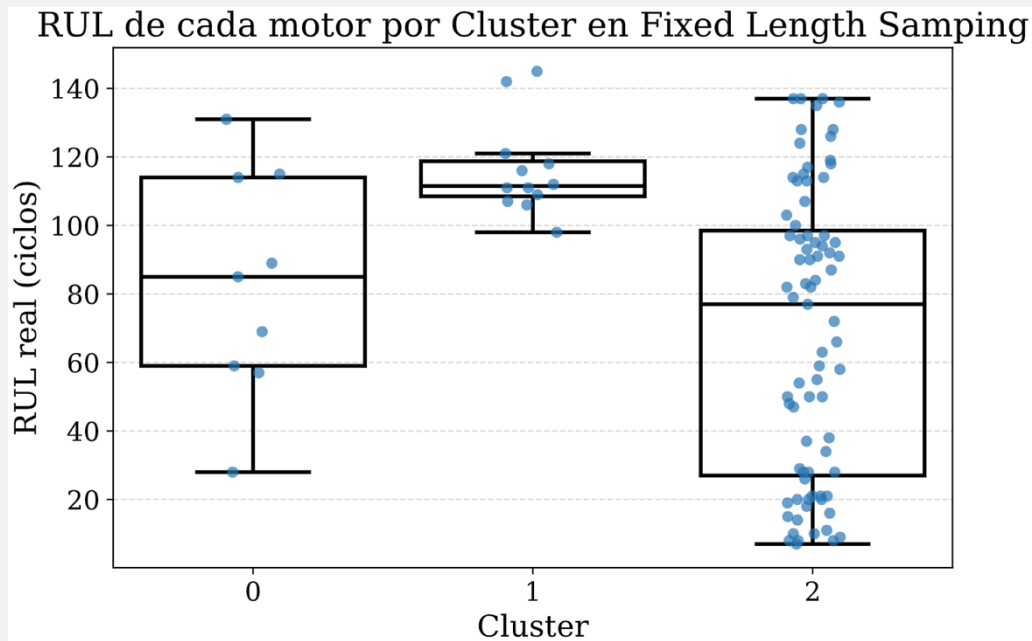
CASO 2: APRENDIZAJE NO SUPERVISADO METODOLOGÍA



CASO 2: APRENDIZAJE NO SUPERVISADO

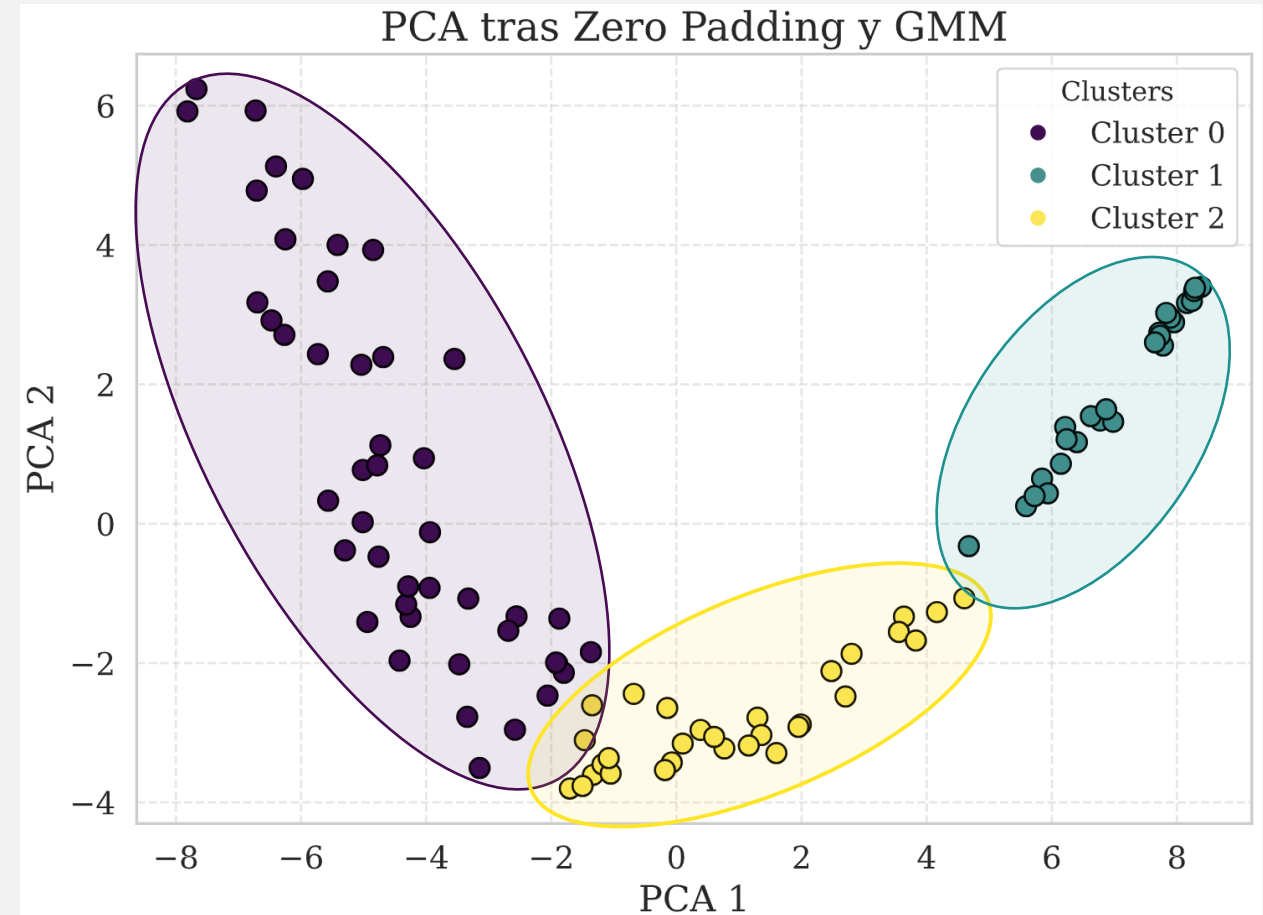
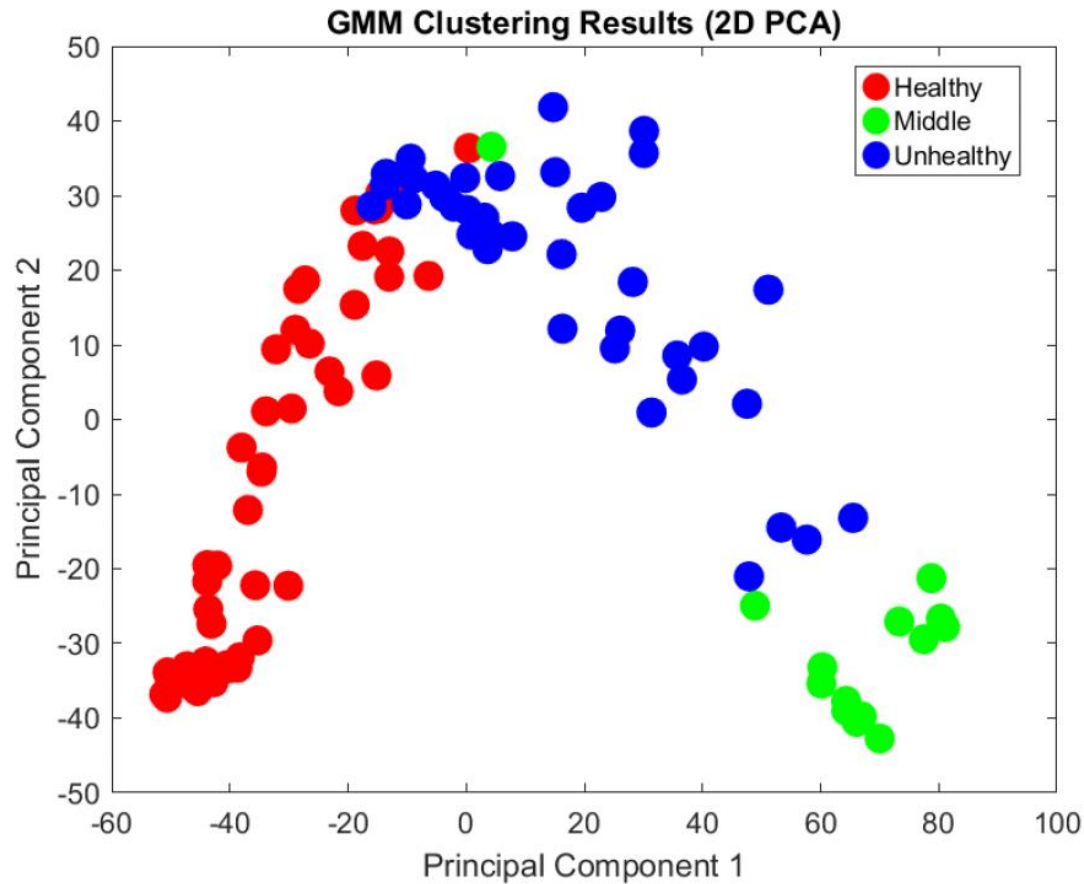
RESULTADOS: ANÁLISIS RUL

- Comparación con datos de RUL real
- Muestreo fijo y agregación estadística no muestran clusters muy diferenciados



CASO 2: APRENDIZAJE NO SUPERVISADO

RESULTADOS: ZERO PADDING

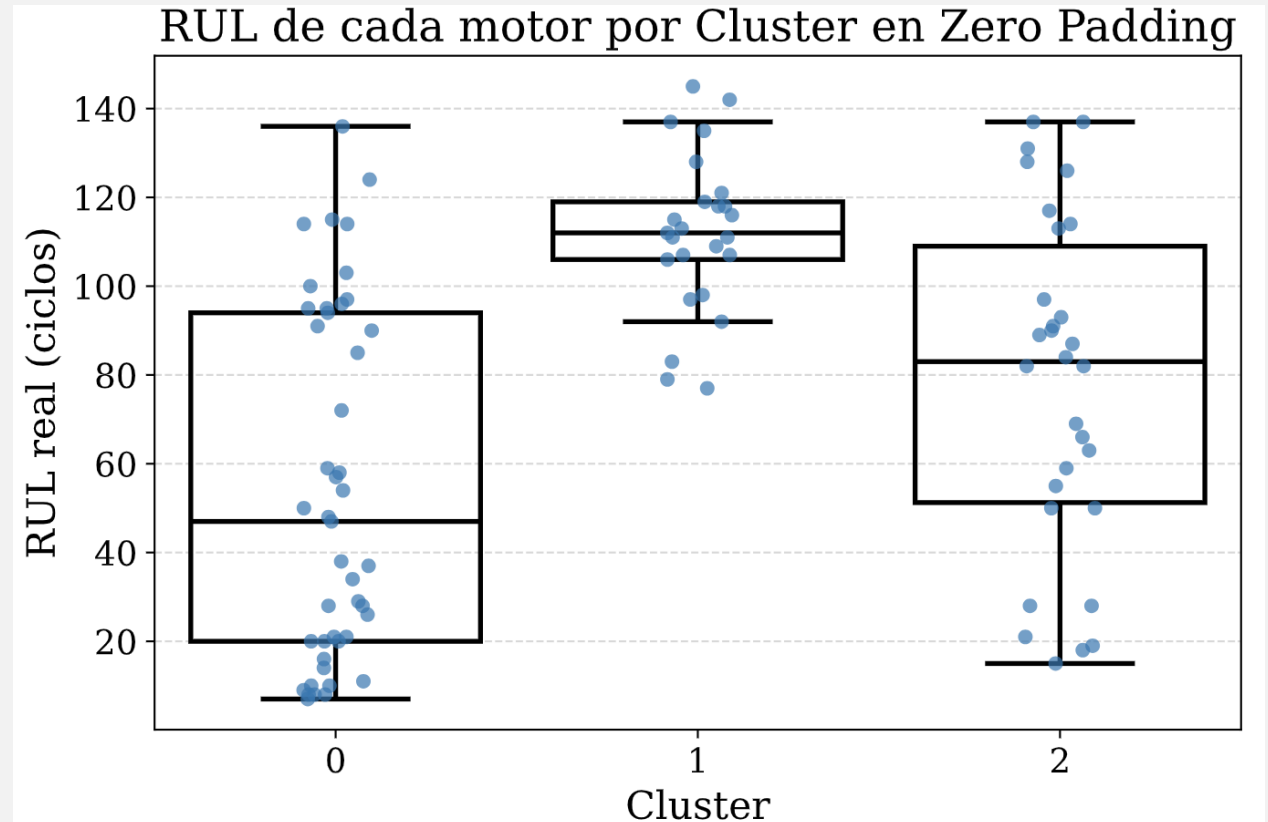


CASO 2: APRENDIZAJE NO SUPERVISADO

RESULTADOS: ZERO PADDING

- Comparación con RUL real
- Mejor resultado: Zero Padding
- Tres clusters bien diferenciados
- Presencia de datos anómalos

Cluster	Motores	Media	Mediana	Min	Max
0	45	53.71	47.0	7	136
1	25	111.84	112.0	77	145
2	30	77.97	83.0	15	137



CASO 2: APRENDIZAJE NO SUPERVISADO

CONCLUSIONES

- Zero padding obtuvo resultados similares a los del estudio original
- Grandes variaciones en cuanto a metodología
- Construcción de la matriz:
 - Adición de ruido (zero padding)
 - Pérdida de temporalidad (muestreo fijo y agregación estadística)

CONCLUSIONES Y FUTUROS TRABAJOS

- Aplicabilidad de los métodos a sistemas complejos
- Limitada por falta de datos reales
- Uso de banco de pruebas para obtener datos
- Mejoras:
 - Iteración de nuevos parámetros
 - Prueba de otros algoritmos
 - Aplicación a conjuntos de datos más complejos
 - Estimación del RUL mediante LSTM

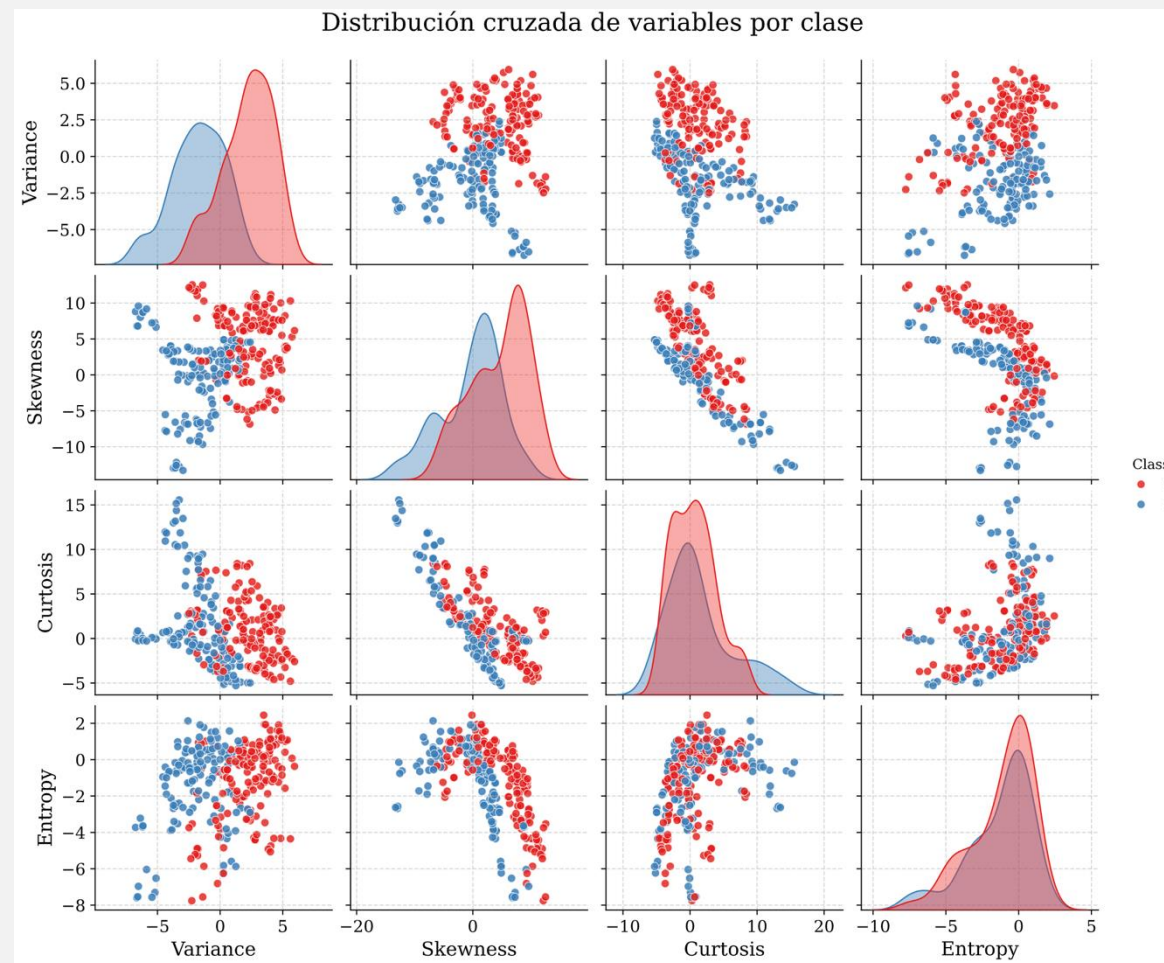
MUCHAS GRACIAS POR SU ATENCIÓN

DOI [10.5281/zenodo.16970727](https://doi.org/10.5281/zenodo.16970727)

BACKUP SLIDES

DISTRIBUCIONES CRUZADAS (CASO I)

Señales 0: mayor variación estadística, sobre todo en skewness y curtosis, lo que significa distribuciones asimétricas y colas más pesadas. Distribución asociadas al fallo.



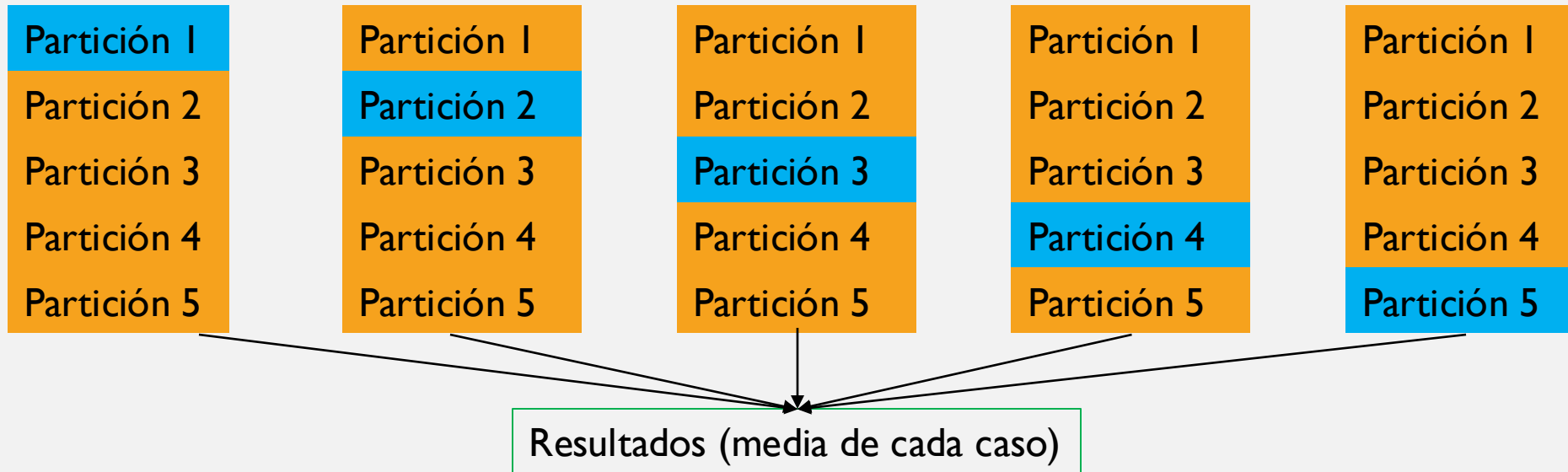
SUPPORT SLIDES → BACKUP PARA POSIBLES PREGUNTAS

- I.
- Adam:
- **Adam (Adaptive Moment Estimation)** es uno de los algoritmos de optimización más usados en *deep learning*.
Combina lo mejor de:
 - **Momentum** → acumula gradientes pasados para acelerar la convergencia y suavizar oscilaciones.
 - **RMSPProp** → ajusta dinámicamente la *learning rate* para cada parámetro, evitando que decaiga demasiado rápido.
- Resultado: Adam es **rápido, estable y robusto frente a datos ruidosos o gradientes dispersos**.
En nuestro caso práctico se utiliza porque entrenamos **redes neuronales con muchos parámetros**, y Adam ofrece la mejor combinación de velocidad y fiabilidad para optimizar este tipo de modelos.

BACKUP SLIDES

VALIDACIÓN EN K-FOLDS

- Ejemplo con 5 folds
- Azul → Test
- Naranja → Entrenamiento



TEORÍA A EXPLICAR PARA CADA MÉTODO

- Caso I:
 - Escalado: standard scaler → normaliza con la media y la desviación típica de cada variable, creando una distribución de media cero y varianza unidad.
 - Varianza es la media de los cuadrados de las diferencias respecto a la media
 - Desviación típica es la raíz de la varianza
 - kNN
 - Decision Trees y Random Forests
 - Matriz de confusión clara con ejemplo
 - Explicar con pairplot los resultados obtenidos, alta accuracy por la fácil separación de los datos por clase



- Caso 2
 - ANNs, retropropagación y optimización.
 - No interesa tanto el algoritmo usado (adam), tenerlo en las backup slides. Por qué se usa?
 - GMM: "Los GMMs son modelos probabilísticos que permiten identificar clusters cuando no hay fronteras claras entre estados. En mantenimiento predictivo, son muy útiles porque permiten detectar fases intermedias de degradación, dando una probabilidad de pertenencia en lugar de una clasificación tajante."

APRENDIZAJE SUPERVISADO Y NO SUPERVISADO

- Supervisado: datos etiquetados
- No Supervisado: sin etiquetas

Aprendizaje Supervisado

- k-NN
- Decision Trees
- Random Forests
- ...

Aprendizaje No Supervisado

- Autoencoders
- GMMs
- Reducción de dimensionalidad

Machine Learning Techniques

