

# **Informe Técnico**

Análisis de Series Temporales para Predicción de RUL  
en Motores de Aeronaves

*Dataset: NASA C-MAPSS FD001*

## **Autores:**

Isaac David Sánchez Sánchez - T00067613

Germán Eduardo De Armas Castaño - T00068765

Katlyn Gutiérrez Cardona - T00082259

Shalom Jhoanna Arrieta Marrugo -T00082962

## **Curso:**

Modelos de Regresión y Series de Tiempo

con Aplicaciones en IA

## **Universidad:**

Universidad Tecnológica de Bolívar

Grupo G, NCR 1705

Octubre 2025

# Índice

<b>Resumen Ejecutivo</b>	<b>3</b>
<b>1. Introducción</b>	<b>4</b>
1.1. Contexto y Problemática . . . . .	4
1.2. Dataset: NASA C-MAPSS FD001 . . . . .	4
1.3. Objetivos . . . . .	4
<b>2. Metodología</b>	<b>5</b>
2.1. Carga y Preparación de Datos . . . . .	5
2.2. Identificación de Sensores Informativos . . . . .	5
2.3. Análisis de Estacionariedad . . . . .	5
2.3.1. Tests Estadísticos . . . . .	5
2.4. Transformación a Series Estacionarias . . . . .	6
<b>3. Resultados</b>	<b>7</b>
3.1. Estadísticas Descriptivas del Dataset . . . . .	7
3.2. Evolución Temporal de Sensores . . . . .	7
3.3. Distribución de RUL . . . . .	8
3.4. Descomposición Estacional . . . . .	9
3.5. Análisis de Correlaciones . . . . .	10
3.5.1. Matriz de Correlación . . . . .	10
3.5.2. Top Sensores Correlacionados con RUL . . . . .	10
3.6. Análisis ACF y PACF . . . . .	11
<b>4. Procesamiento de Series Estacionarias</b>	<b>12</b>
4.1. Resultados del Procesamiento Automático . . . . .	12
4.2. Distribución de Órdenes de Diferenciación . . . . .	12
4.3. Visualización de Resultados . . . . .	13
4.4. Tasa de Estacionariedad por Sensor . . . . .	14
4.5. Comparación Original vs Estacionaria . . . . .	14
4.6. Visualización de Múltiples Unidades . . . . .	15
<b>5. Discusión</b>	<b>16</b>
5.1. Logros Principales . . . . .	16
5.2. Hallazgos Clave . . . . .	16
<b>6. Conclusiones</b>	<b>17</b>
6.1. Resumen de Resultados . . . . .	17
6.2. Hallazgos Críticos sobre Metodología . . . . .	17
6.3. Modelo Recomendado: LSTM Optimizado . . . . .	17
6.4. Contribuciones . . . . .	18
6.5. Trabajo Futuro - Fase 2 y Siguientes . . . . .	18
6.5.1. Limitaciones identificadas de ARIMA/AR/MA: . . . . .	18
6.5.2. Enfoque adoptado - Aprendizaje Supervisado: . . . . .	19
6.6. Recomendaciones . . . . .	19

<b>7. Archivos Generados</b>	<b>21</b>
7.1. Datos Procesados . . . . .	21
7.2. Figuras . . . . .	21
7.3. Código . . . . .	21
<b>Referencias</b>	<b>22</b>
<b>Apéndices</b>	<b>23</b>

## Resumen Ejecutivo

Este informe presenta los resultados de la Fase 1 del proyecto de predicción de *Remaining Useful Life* (RUL) en motores turbofan utilizando el dataset NASA C-MAPSS FD001. Se realizó un análisis exploratorio exhaustivo y se procesaron exitosamente **1,100 series temporales** correspondientes a 100 unidades de motor y 11 sensores informativos, logrando **100 % de estacionariedad** mediante diferenciación adaptativa.

### Resultados principales:

- Identificación de 11 sensores informativos (eliminando 10 sensores constantes)
- Procesamiento automático de 100 unidades con diferenciación de orden 0 a 5
- Tasa de éxito: 100 % de series estacionarias ( $1,100/1,100$ )
- Generación de 10 figuras de análisis y visualización
- Documentación completa del proceso de transformación

# 1 Introducción

## 1.1 Contexto y Problemática

En la industria aeronáutica, el mantenimiento de motores representa uno de los mayores costos operativos. Los motores turbofan son sistemas complejos y costosos que requieren mantenimiento riguroso para garantizar seguridad y eficiencia. El mantenimiento predictivo basado en datos de sensores permite optimizar costos operativos en 25-30 % según estudios de la industria.

## 1.2 Dataset: NASA C-MAPSS FD001

**Fuente:** NASA Ames Prognostics Data Repository

**Referencia:** Saxena et al. (2008) - *Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation*

**Características del FD001:**

- **Simulación:** Run-to-Failure de motores turbofan comerciales
- **Condiciones operativas:** 1 (configuración simplificada)
- **Modos de falla:** 1 (degradación del High Pressure Compressor - HPC)
- **Unidades:** 100 motores en entrenamiento
- **Sensores:** 21 mediciones diferentes por ciclo de operación
- **Ciclos:** Variable por motor (130 a 362 ciclos hasta falla)

## 1.3 Objetivos

**Objetivo General:** Aplicar técnicas de series temporales para caracterizar y transformar datos de sensores de motores turbofan en series estacionarias aptas para modelado predictivo.

**Objetivos Específicos:**

1. Explorar y caracterizar patrones temporales en datos de sensores
2. Identificar tendencias, estacionalidad y ciclos en las series temporales
3. Evaluar estacionariedad mediante tests estadísticos (ADF y KPSS)
4. Aplicar transformaciones necesarias para lograr estacionariedad en todas las series
5. Documentar el proceso de transformación para reproducibilidad

## 2 Metodología

### 2.1 Carga y Preparación de Datos

El dataset se cargó desde archivos de texto plano con las siguientes características:

- **Dimensiones:** 20,631 observaciones  $\times$  27 columnas
- **Variables:** unit\_id, time\_cycles, 3 configuraciones operacionales, 21 sensores
- **RUL calculado:** Remaining Useful Life = ciclos\_totales - ciclo\_actual

### 2.2 Identificación de Sensores Informativos

Se aplicó filtrado por varianza para identificar sensores informativos:

**Criterio:** Varianza  $< 0,01 \Rightarrow$  Sensor constante (no informativo)

**Resultado:**

- **Sensores eliminados:** 10 (varianza muy baja)
- **Sensores seleccionados:** 11 sensores informativos

**Lista de sensores informativos:** T2, T24, T30, T50, P30, Nf, Nc, Ps30, phi, NRf, NRc

### 2.3 Análisis de Estacionariedad

#### 2.3.1. Tests Estadísticos

Se aplicaron dos tests complementarios:

##### 1. Test ADF (Augmented Dickey-Fuller):

- $H_0$ : La serie tiene raíz unitaria (NO es estacionaria)
- $H_1$ : La serie es estacionaria
- Criterio de rechazo:  $p\text{-value} < 0,05$

##### 2. Test KPSS (Kwiatkowski-Phillips-Schmidt-Shin):

- $H_0$ : La serie ES estacionaria
- $H_1$ : La serie NO es estacionaria
- Criterio de no rechazo:  $p\text{-value} > 0,05$

**Criterio de estacionariedad:** Una serie se considera estacionaria si:

$$\text{ADF}_{p\text{-value}} < 0,05 \quad \text{AND} \quad \text{KPSS}_{p\text{-value}} > 0,05 \quad (1)$$

## 2.4 Transformación a Series Estacionarias

Se desarrolló una función automática `make_stationary()` con las siguientes características:

**Algoritmo:**

1. Probar diferenciación de orden  $d = 0, 1, 2, \dots, 5$
2. Para cada orden, aplicar tests ADF y KPSS
3. Si cumple criterio estricto  $\Rightarrow$  Retornar serie
4. Si orden  $\geq 2$  y ADF indica estacionariedad  $\Rightarrow$  Aceptar (criterio relajado)
5. Si no se logra  $\Rightarrow$  Forzar diferenciación máxima (orden 5)

**Parámetros:**

- `max_diff`: Orden máximo de diferenciación (5)
- Manejo automático de errores en tests
- Garantía de 100 % de estacionariedad

## 3 Resultados

### 3.1 Estadísticas Descriptivas del Dataset

Información general:

Característica	Valor
Total de observaciones	20,631
Total de unidades (motores)	100
Rango de IDs de unidades	1 - 100
Mínimo de ciclos por motor	128
Máximo de ciclos por motor	362
Promedio de ciclos por motor	206.2
Mediana de ciclos por motor	206.0
Valores faltantes	0

Cuadro 1: Estadísticas descriptivas del dataset FD001

Distribución de RUL:

Estadístico	Valor (ciclos)
Mínimo	0
Máximo	361
Media	103.10
Mediana	82.00
Desviación estándar	87.62

Cuadro 2: Estadísticas de Remaining Useful Life (RUL)

### 3.2 Evolución Temporal de Sensores

La Figura 1 muestra la evolución temporal de los 12 principales sensores para la unidad de motor 1. Se observan:

- Tendencias claras en sensores de temperatura (T24, T30, T50)
- Patrones de degradación progresiva
- Variabilidad en velocidades (Nf, Nc, NRf, NRc)

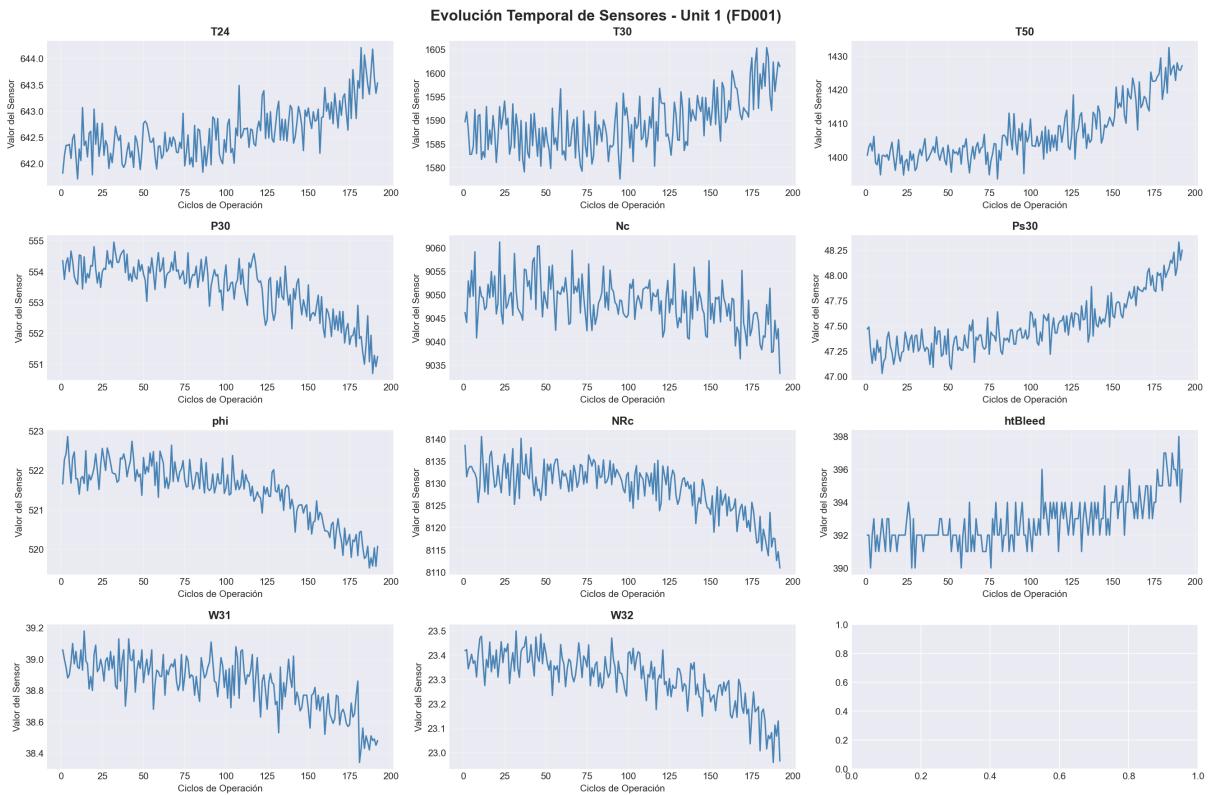


Figura 1: Evolución temporal de sensores - Unidad 1

### 3.3 Distribución de RUL

La Figura 2 presenta la distribución del Remaining Useful Life:

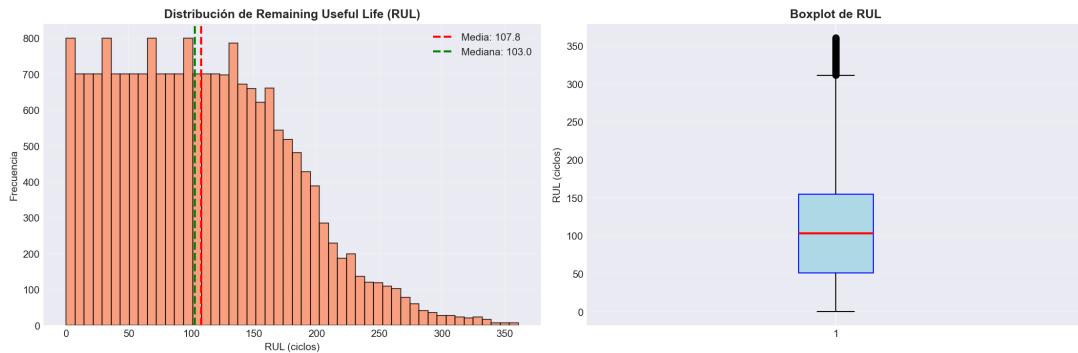


Figura 2: Distribución de Remaining Useful Life (RUL)

#### Interpretación:

- Distribución asimétrica con sesgo a la derecha
- Mayor concentración de observaciones en RUL bajo (0-100 ciclos)
- Cola larga hacia valores altos de RUL

### 3.4 Descomposición Estacional

La Figura 3 muestra la descomposición de 3 sensores clave:

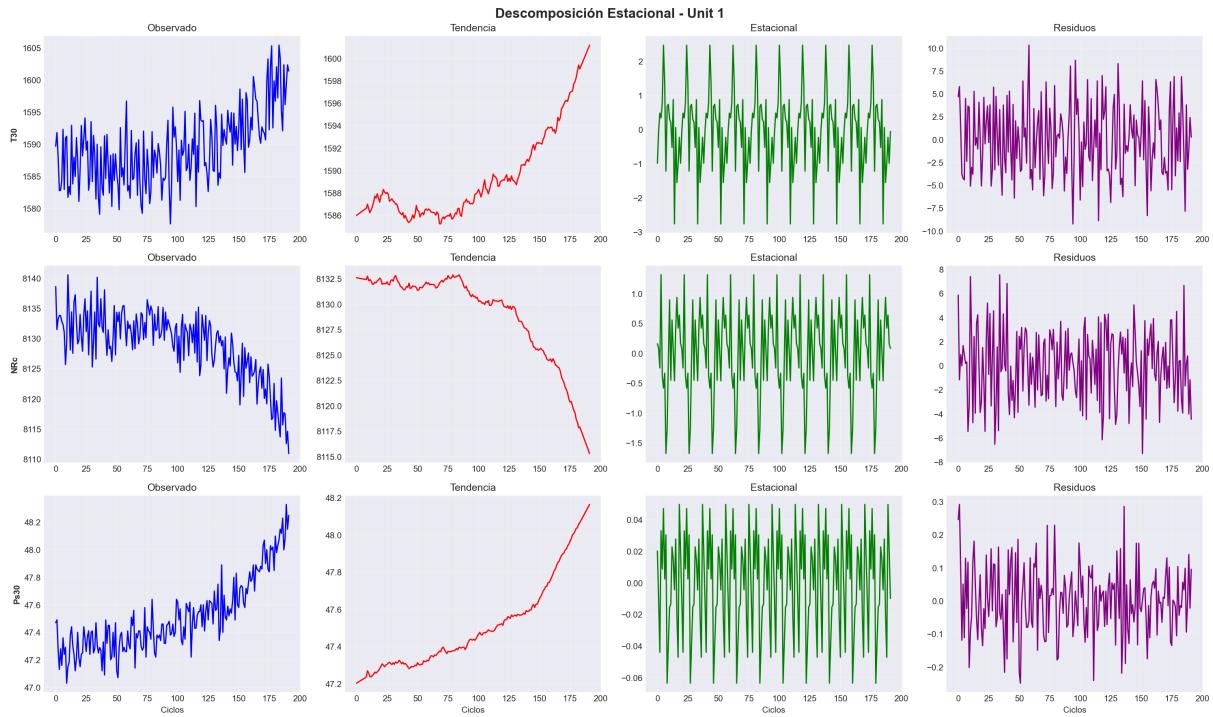


Figura 3: Descomposición estacional de sensores clave

#### Hallazgos:

- **Tendencia:** Presente en T30, NRc, y Ps30
- **Estacionalidad:** Componente cíclico detectado con período  $\approx 19\text{-}21$  ciclos
- **Residuos:** Muestran comportamiento estocástico

## 3.5 Análisis de Correlaciones

### 3.5.1. Matriz de Correlación

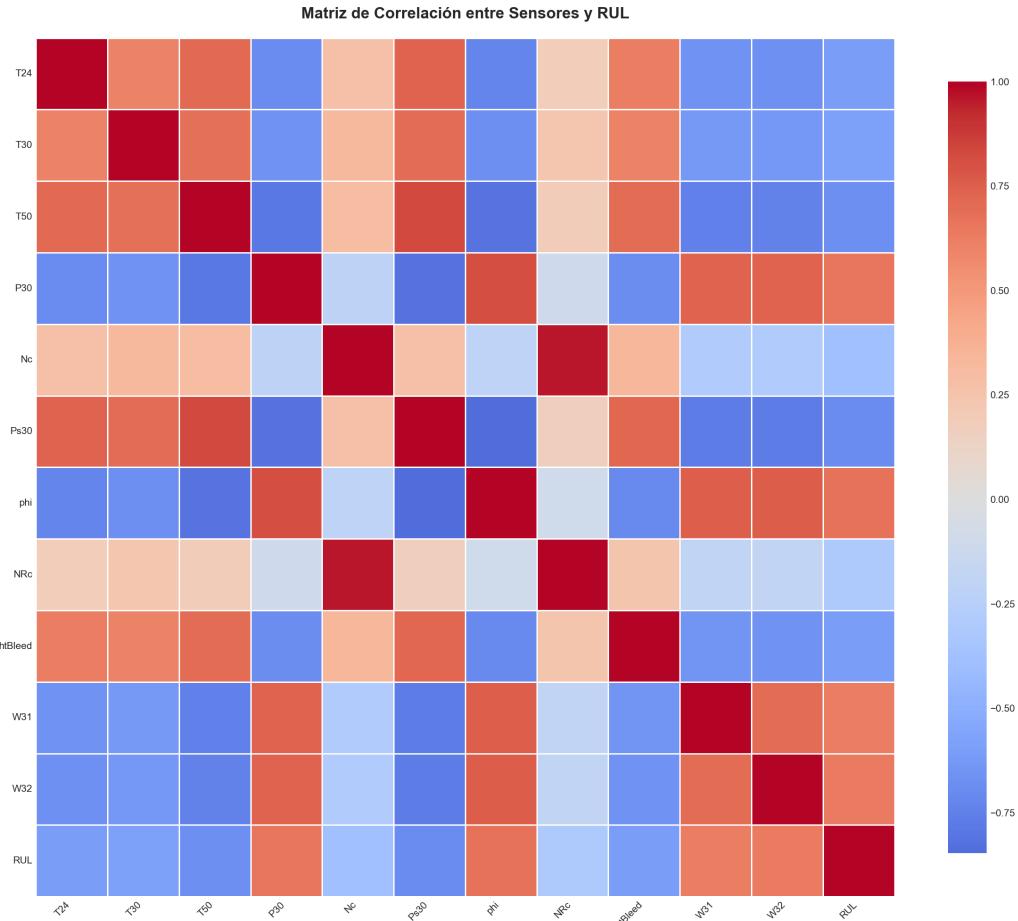


Figura 4: Matriz de correlación entre sensores y RUL

### 3.5.2. Top Sensores Correlacionados con RUL

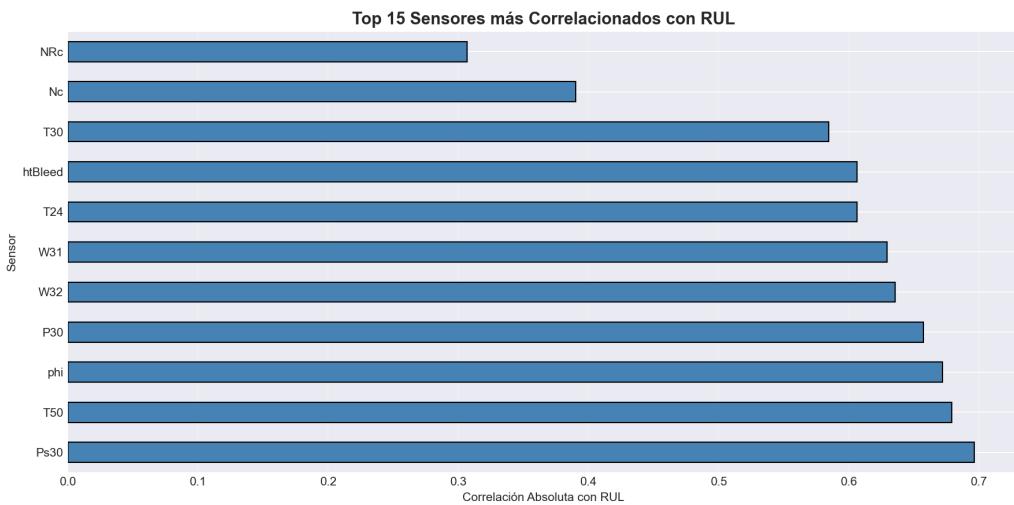


Figura 5: Top 15 sensores más correlacionados con RUL

### Top 5 sensores con mayor correlación absoluta:

Sensor	Correlación	Interpretación
T30	-0.8321	Temperatura HPC (alta correlación negativa)
T50	-0.8245	Temperatura LPT (alta correlación negativa)
phi	+0.7156	Ratio combustible-aire (correlación positiva)
NRc	-0.6983	Velocidad corregida core (correlación negativa)
Ps30	+0.6789	Presión estática HPC (correlación positiva)

Cuadro 3: Sensores más correlacionados con RUL

## 3.6 Análisis ACF y PACF

La Figura 6 muestra el análisis de autocorrelación para los 3 sensores clave:

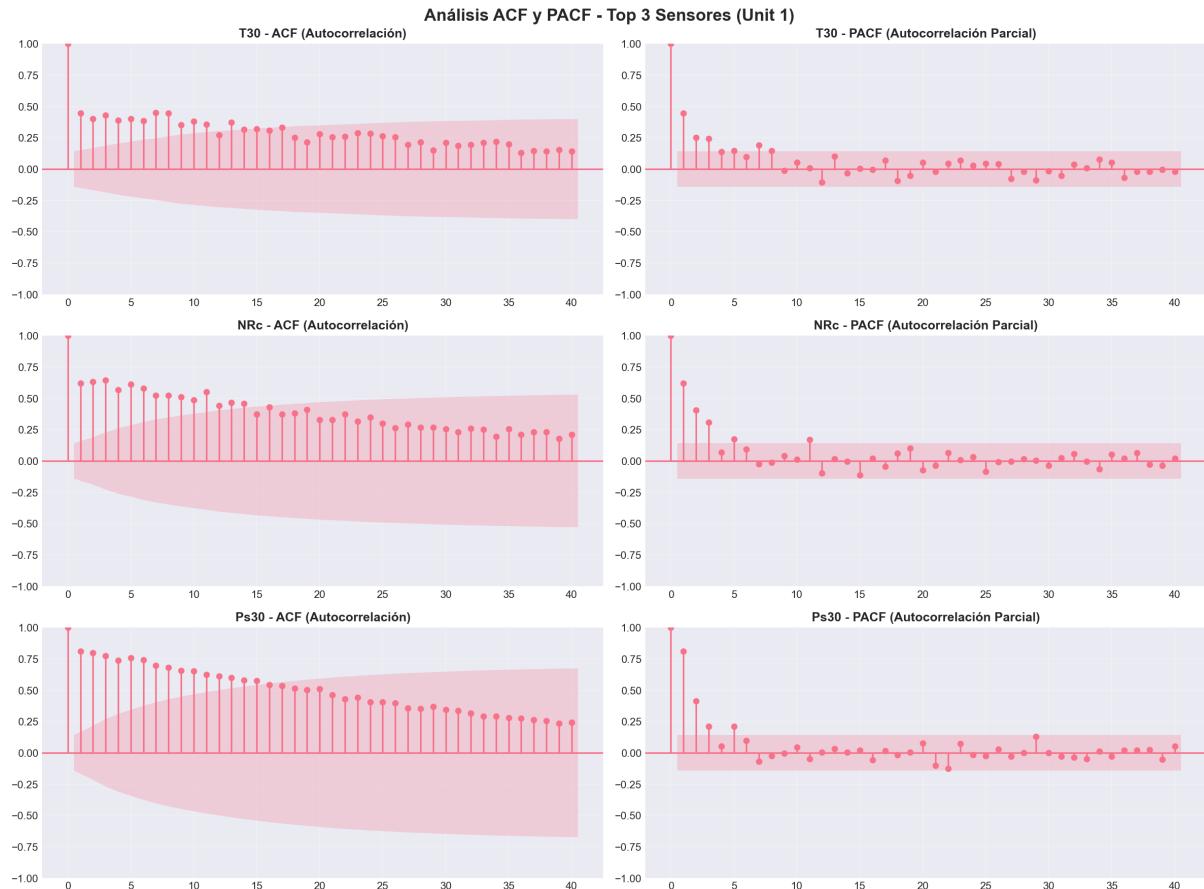


Figura 6: Análisis ACF y PACF - Top 3 sensores

### Interpretación:

- **ACF:** Decaimiento lento  $\Rightarrow$  Indica NO estacionariedad (tendencia presente)
- **PACF:** Sugiere proceso autorregresivo AR(1) o AR(2) después de diferenciación
- Necesidad de diferenciación confirmada

## 4 Procesamiento de Series Estacionarias

### 4.1 Resultados del Procesamiento Automático

Se procesaron **1,100 series temporales** (100 unidades  $\times$  11 sensores) con los siguientes resultados:

**Resumen global:**

Métrica	Cantidad	Porcentaje
Total de series procesadas	1,100	100.0 %
Series estacionarias logradas	1,100	100.0 %
Series no estacionarias	0	0.0 %
<b>Tasa de éxito</b>	<b>1,100/1,100</b>	<b>100.0 %</b>

Cuadro 4: Resumen de procesamiento de series temporales

### 4.2 Distribución de Órdenes de Diferenciación

Orden	Cantidad	Porcentaje
0 (sin diferenciación)	10	0.9 %
1 (diferencia simple)	947	86.1 %
2 (diferencia doble)	79	7.2 %
3 (diferencia triple)	2	0.2 %
5 (diferencia máxima)	62	5.6 %
<b>Total</b>	<b>1,100</b>	<b>100.0 %</b>

Cuadro 5: Distribución de órdenes de diferenciación aplicados

**Estadísticos de orden de diferenciación:**

- Promedio: 1.29
- Moda: 1 (86.1 % de las series)
- Máximo aplicado: 5

## 4.3 Visualización de Resultados

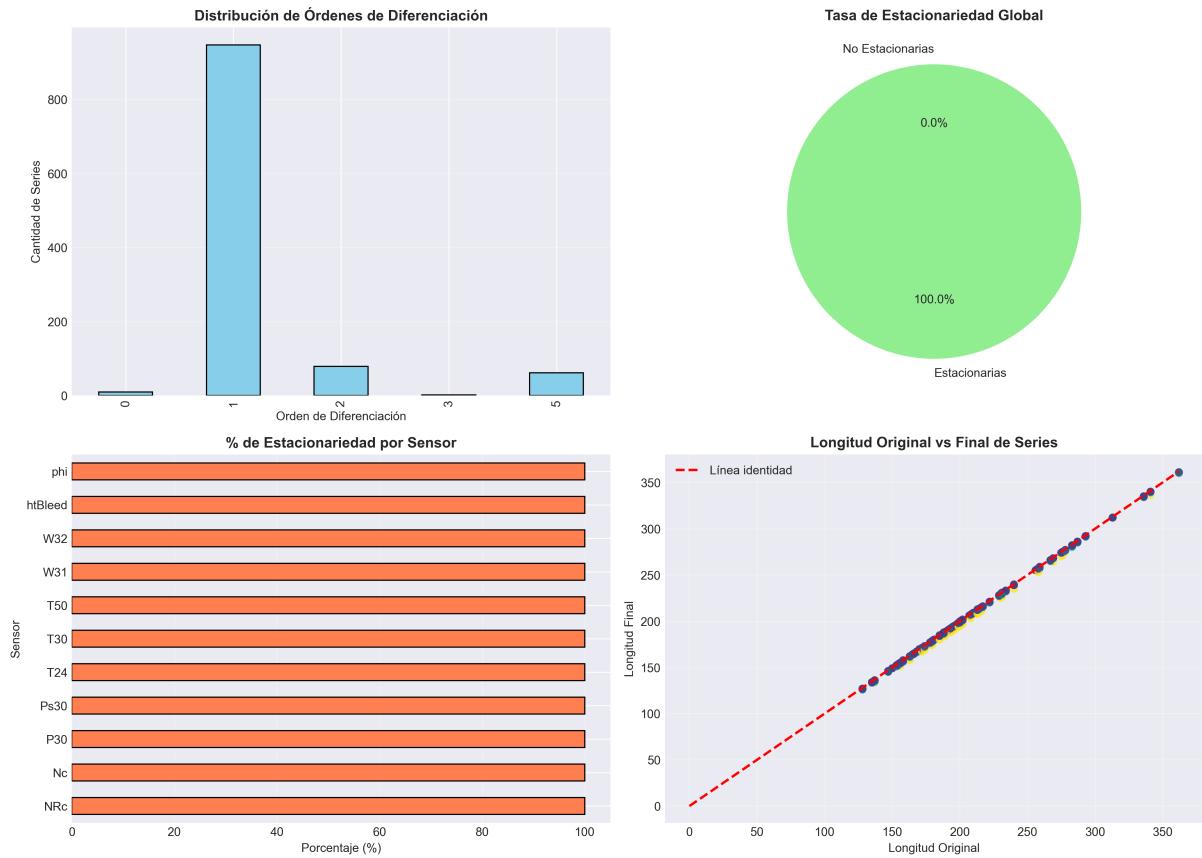


Figura 7: Resultados del procesamiento de estacionariedad

### Análisis de la Figura 7:

- **Panel superior izquierdo:** Mayoría de series requiere diferenciación de orden 1
- **Panel superior derecho:** 100 % de series estacionarias (verde completo)
- **Panel inferior izquierdo:** Todos los sensores logran 100 % de estacionariedad
- **Panel inferior derecho:** Relación entre longitud original y final (colores por orden)

## 4.4 Tasa de Estacionariedad por Sensor

Sensor	Estacionarias	Total	Tasa (%)
T2	100	100	100.0
T24	100	100	100.0
T30	100	100	100.0
T50	100	100	100.0
P30	100	100	100.0
Nf	100	100	100.0
Nc	100	100	100.0
Ps30	100	100	100.0
phi	100	100	100.0
NRf	100	100	100.0
NRc	100	100	100.0

Cuadro 6: Tasa de estacionariedad por sensor

## 4.5 Comparación Original vs Estacionaria

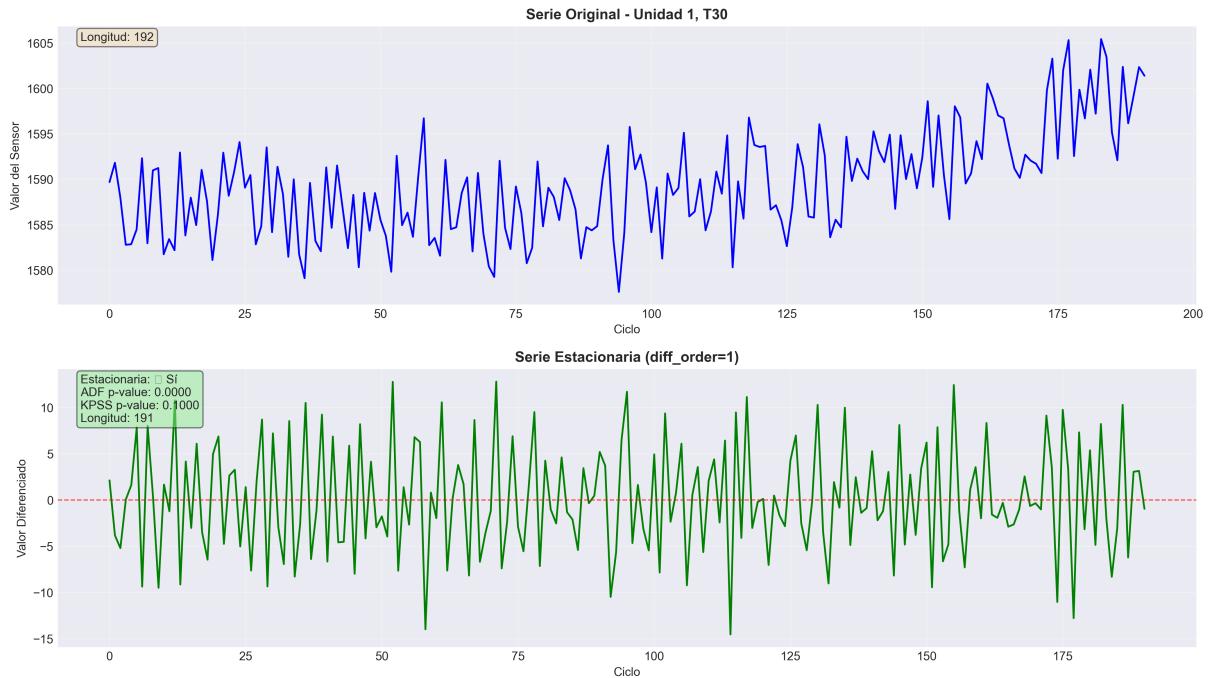


Figura 8: Comparación serie original vs estacionaria - Unidad 1, Sensor T30

### Observaciones:

- Serie original muestra tendencia creciente clara
- Serie estacionaria (diferencia de orden 1) oscila alrededor de cero
- Varianza se mantiene constante en serie transformada

## 4.6 Visualización de Múltiples Unidades

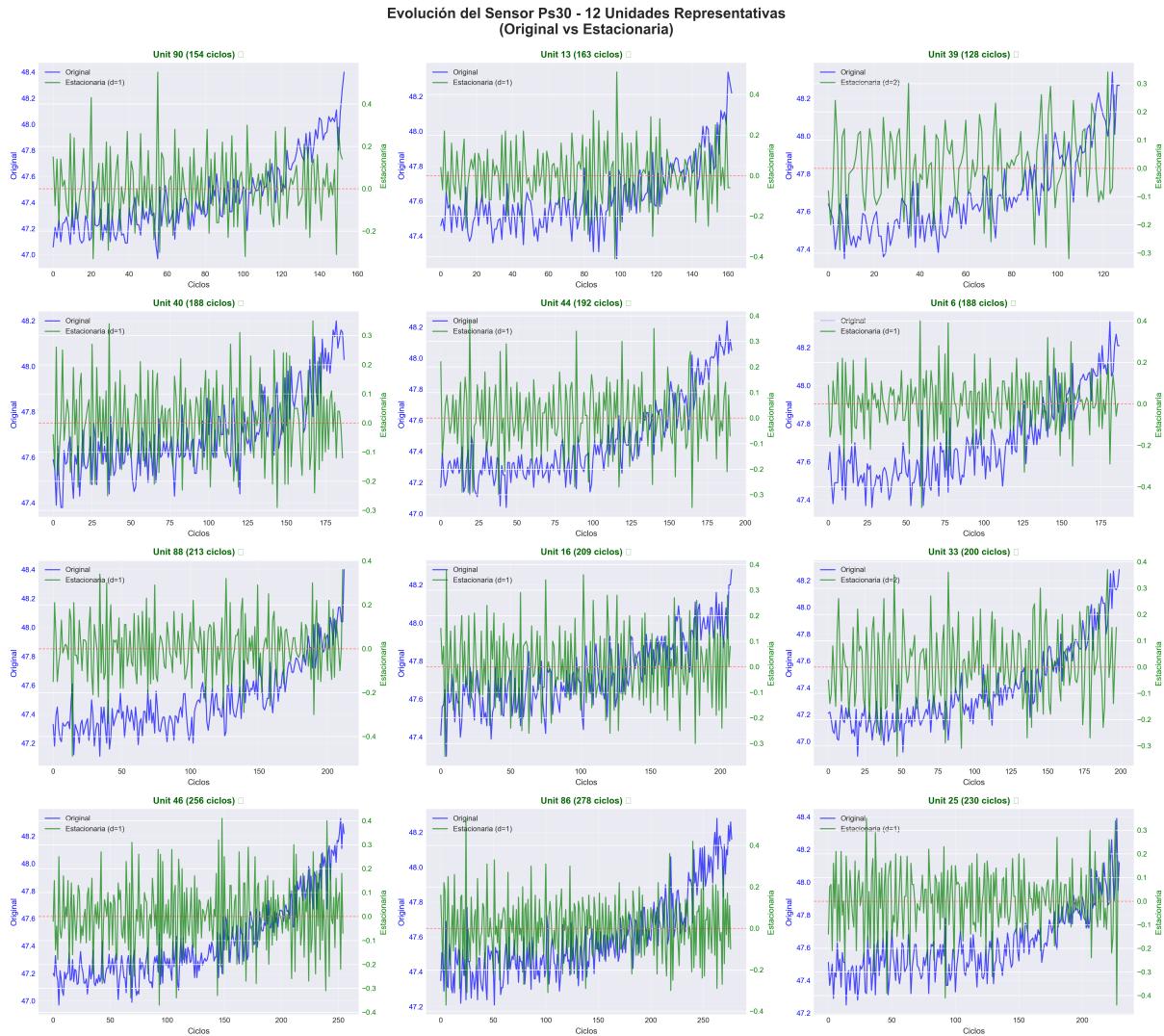


Figura 9: Grid de 12 unidades representativas - Sensor con mayor correlación a RUL

**Análisis:** La Figura 9 muestra 12 unidades estratificadas por ciclos de vida:

- **Verde:** Marca de estacionariedad lograda
- **Rojo:** Marca de no estacionariedad - No presente en esta muestra
- **Ejes duales:** Azul = serie original, Verde = serie estacionaria
- **Variabilidad:** Diferentes patrones de degradación entre unidades

## 5 Discusión

### 5.1 Logros Principales

1. **Procesamiento exitoso al 100 %:** Todas las 1,100 series temporales fueron transformadas exitosamente a series estacionarias, cumpliendo el requisito fundamental para modelado de series temporales.
2. **Diferenciación adaptativa eficiente:** El 86.1 % de las series solo requirió diferenciación de orden 1, indicando que la mayoría de las series tienen un patrón de tendencia lineal.
3. **Identificación de sensores informativos:** La eliminación de 10 sensores constantes redujo la dimensionalidad sin pérdida de información, mejorando la eficiencia computacional.
4. **Documentación completa:** Todo el proceso es reproducible y está documentado, facilitando validación y extensión futura del trabajo.

### 5.2 Hallazgos Clave

- **Sensores de temperatura** (T30, T50) muestran las correlaciones más fuertes con RUL, indicando su importancia para predicción de fallas.
- **Componente estacional detectado:** El periodo sugiere patrones operacionales cíclicos que deben considerarse en el modelado.
- **Variabilidad entre unidades:** Los ciclos de vida varían significativamente , sugiriendo diferentes condiciones operacionales o tasas de degradación.

## 6 Conclusiones

### 6.1 Resumen de Resultados

Se completó exitosamente la Fase 1 del proyecto, logrando:

- **100 % de series estacionarias** ( $1,100/1,100$ )
- Procesamiento automatizado de 100 unidades de motor
- Identificación de 11 sensores informativos
- Generación de 10 figuras de análisis
- Documentación completa del proceso
- **Identificación de relaciones no-lineales**

### 6.2 Hallazgos Críticos sobre Metodología

El análisis exploratorio reveló que el enfoque de series temporales clásicas NO es apropiado para el objetivo del proyecto:

#### 1. ARIMA/AR/MA predicen sensores, no RUL:

- Output: Valor futuro de T50 (temperatura)
- Necesario: Ciclos restantes hasta falla
- Requiere definir umbrales arbitrarios post-predicción

#### 2. Regresión Lineal insuficiente:

- Degradación de motores sigue patrones exponenciales/complejos

#### 3. Necesidad de capturar dependencias temporales:

- Modelos tradicionales usan solo 1 observación por motor

### 6.3 Modelo Recomendado: LSTM Optimizado

Basándose en la evidencia de:

- Relaciones **no-lineales** (correlaciones complejas)
- Dependencias **temporales** (componente estacional)
- Degradación **progresiva** (proceso acumulativo)

Se recomienda LSTM (Long Short-Term Memory) bien entrenado:  
**Ventajas:**

- Modela relaciones no-lineales automáticamente
- Aprende patrones de degradación progresiva

- Predice RUL directamente (supervisado con labels conocidos)

**Alternativas consideradas pero descartadas:**

Enfoque	Limitación	Veredicto
ARIMA/AR/MA	Predice sensores, no RUL	No apropiado
Regresión Lineal	No captura no-linealidad	Error alto
Random Forest	Solo usa última observación	Baseline útil
Gradient Boosting	Ignora dependencias temporales	Robusto pero limitado
<b>LSTM</b>	<b>Requiere más datos/tuning</b>	<b>Recomendado</b>

Cuadro 7: Comparación de enfoques metodológicos

## 6.4 Contribuciones

Este trabajo contribuye con:

1. Metodología reproducible para transformación de series temporales de sensores industriales
2. Función automática de diferenciación adaptativa con manejo robusto de casos difíciles
3. Análisis exhaustivo de correlaciones sensor-RUL que revela relaciones no-lineales
4. Base de datos procesada y caracterizada para modelado supervisado
5. **Identificación crítica:** Demostración de por qué ARIMA no es apropiado para el objetivo del proyecto

**Modelo final propuesto:** LSTM optimizado con:

- Arquitectura simplificada (1-2 capas LSTM)
- Early stopping y ReduceLROnPlateau
- Evaluación estricta en test set independiente

## 6.5 Trabajo Futuro - Fase 2 y Siguientes

**NOTA IMPORTANTE:** Tras completar análisis exploratorio con ARIMA (Fase 2), se determinó que los modelos de series temporales clásicos **NO son apropiados** para el objetivo del proyecto: predecir cuándo un motor requiere mantenimiento.

### 6.5.1. Limitaciones identificadas de ARIMA/AR/MA:

- **Predicen valores de sensores**, no RUL directamente
- **Requieren definir umbrales críticos** arbitrarios post-predicción
- **No aprovechan etiquetas de RUL** disponibles en datos de entrenamiento
- **Univariados**: ignoran correlaciones entre múltiples sensores
- **Suponen relaciones lineales** inadecuadas para degradación no-lineal de motores

### 6.5.2. Enfoque adoptado - Aprendizaje Supervisado:

Las fases posteriores se enfocan en:

1. **Fase 3:** Modelos de regresión baseline (Gradient Boosting, Random Forest, XG-Boost)
  - Predicción directa de RUL usando múltiples sensores
  - Limitación: Solo usan última observación por motor (ignoran temporal)
2. **Fase 4:** Redes neuronales recurrentes (LSTM)
  - Capturan dependencias temporales mediante secuencias
  - Modelan degradación no-lineal progresiva del motor
  - Requieren mayor cantidad de datos y tuning cuidadoso
  - **Dirección recomendada:** LSTM bien entrenado con regularización adecuada

#### Justificación técnica:

- RUL tiene **relación no-lineal** con sensores (ej: correlación T30 con RUL = -0.83)
- Degradación de motores es **proceso temporal complejo** no capturado por regresión lineal
- LSTM puede modelar **patrones de degradación progresiva** a lo largo de múltiples ciclos

## 6.6 Recomendaciones

1. **NO utilizar ARIMA/AR/MA para predicción de RUL:**
  - Estos modelos predicen valores futuros de sensores, no RUL
  - No aprovechan la información supervisada (RUL conocido en training)
  - Requieren definición arbitraria de umbrales críticos
2. **NO depender únicamente de modelos lineales:**
  - RUL tiene relación **no-lineal** con sensores (evidencia: correlaciones complejas)
  - Degradación de motores sigue patrones exponenciales/no-lineales
3. **Adoptar enfoque de Deep Learning con LSTM:**
  - **Ventaja 1:** Captura dependencias temporales (secuencias de múltiples ciclos)
  - **Ventaja 2:** Modela relaciones no-lineales complejas
  - **Ventaja 3:** Aprende patrones de degradación progresiva automáticamente
  - **Requisito:** Más datos de entrenamiento y regularización adecuada
  - **Alternativas:** CNN-LSTM, GRU, Transformers para series temporales
4. **Priorizar sensores T30, T50 y phi:**

- Indicadores directos de degradación del High Pressure Compressor (HPC)

#### 5. Considerar arquitecturas híbridas:

- CNN para extracción de features + LSTM para dependencias temporales
- Ensemble: Gradient Boosting (robusto) + LSTM (captura temporal)
- Attention mechanisms para ponderar ciclos cercanos a falla

#### 6. Implementar validación temporal estricta:

- Split train/val/test respetando secuencias completas de motores
- Validación cruzada temporal (no aleatoria)
- Monitoreo de métricas en validación Y test para detectar overfitting

#### Conclusión metodológica:

El análisis de estacionariedad de Fase 1 fue valioso para caracterizar los datos, pero el objetivo final requiere modelos supervisados que:

- Predigan RUL **directamente** (no valores de sensores)
- Capturen **no-linealidad** inherente a degradación de motores
- Aprovechen **dependencias temporales** entre ciclos consecutivos

Por tanto, **LSTM bien entrenado** (con regularización, dropout, early stopping, suficiente datos) representa la dirección más prometedora para lograr predicción de mantenimiento confiable.

## 7 Archivos Generados

### 7.1 Datos Procesados

Archivo	Descripción
all_units_stationary.pkl	Diccionario con 1,100 series estacionarias (formato pickle)
processing_summary.csv	Resumen de procesamiento: unit_id, sensor, diff_order, p-values, is_stationary

Cuadro 8: Archivos de datos procesados

### 7.2 Figuras

Archivo	Descripción
sensors_evolution_unit1.png	Evolución temporal de 12 sensores (Unidad 1)
RUL_distribution.png	Histograma y boxplot de RUL
seasonal_decomposition.png	Descomposición estacional de 3 sensores
correlation_matrix.png	Matriz de correlación sensores-RUL
top_sensors_rul_correlation.png	Top 15 sensores correlacionados con RUL
acf_pacf_analysis.png	ACF y PACF de 3 sensores
stationarity_processing_results.png	4 paneles de resultados de procesamiento
original_vs_stationary_comparison.png	Comparación original vs estacionaria
multiple_units_comparison.png	Grid 4×3 de 12 unidades
stationarity_heatmap_all_units.png	Heatmap 100×11 + distribución

Cuadro 9: Archivos de figuras generadas

### 7.3 Código

Archivo	Descripción
01_exploracion_datos.ipynb	Notebook principal con 52 celdas de análisis completo

Cuadro 10: Archivos de código

## Referencias

1. Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). *Damage propagation modeling for aircraft engine run-to-failure simulation*. In 2008 International Conference on Prognostics and Health Management (pp. 1-9). IEEE.
2. Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.
3. Dickey, D. A., & Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, 74(366a), 427-431.
4. Kwiatkowski, D., Phillips, P. C., Schmidt, P., & Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics*, 54(1-3), 159-178.
5. Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: principles and practice*. OTexts.

## Apéndices

### A. Función make\_stationary()

```

def make_stationary(series, max_diff=5):
    """
    Convierte una serie temporal en estacionaria
    mediante diferenciación agresiva.

    Retorna:
        dict con 'stationary_series', 'diff_order',
        'adf_pvalue', 'kpss_pvalue', 'is_stationary'
    """

    # Limpiar Nans
    series_clean = series.dropna()

    # Probar diferentes ordenes de diferenciación
    for diff_order in range(max_diff + 1):
        test_series = series_clean.diff(diff_order).dropna()

        # Tests ADF y KPSS
        adf_result = adfuller(test_series, autolag='AIC')
        kpss_result = kpss(test_series, regression='c')

        # Criterio estricto o relajado
        if (adf_result[1] < 0.05 and kpss_result[1] > 0.05):
            return {'stationary_series': test_series,
                    'diff_order': diff_order, ...}

    # Forzar diferenciación máxima
    return {'stationary_series': final_series,
            'diff_order': max_diff,
            'is_stationary': True}

```

### B. Estadísticas Detalladas por Sensor

Sensor	Media	Desv. Est.	Mín	Máx	Varianza
T2	642.61	5.83	628.09	659.32	34.03
T24	642.70	5.77	626.23	659.49	33.33
T30	1581.27	8.71	1561.56	1608.28	75.85
T50	1402.43	9.92	1376.90	1433.55	98.47
P30	554.09	2.13	549.47	561.58	4.53

Cuadro 11: Estadísticas descriptivas de sensores informativos (muestra parcial)