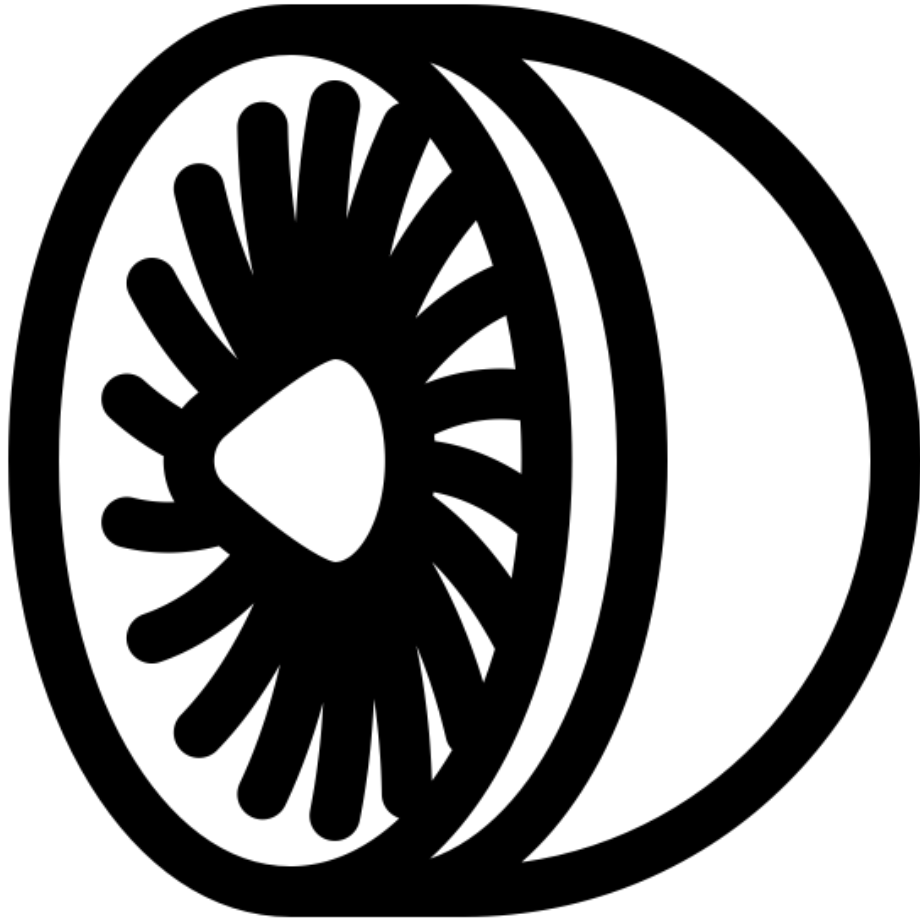


21 DE NOVIEMBRE DE 2025



DETECCIÓN ANOMALÍAS

NASA C-MAPSS

VÍCTOR VEGA SOBRAL
SANTIAGO SOUTO ORTEGA
LUIS SÁNCHEZ PATIÑO

Contenido

| | |
|--|---|
| Introducción | 2 |
| Contexto del Problema..... | 2 |
| Dataset NASA C-MAPSS..... | 2 |
| Metodología..... | 2 |
| Preprocesamiento de Datos..... | 2 |
| Métricas de Evaluación | 2 |
| Algoritmos Implementados | 3 |
| Resultados Experimentales | 3 |
| FD001: Escenario base | 3 |
| FD002: Complejidad Multimodal | 3 |
| FD003: Fallo Multimodal | 3 |
| FD004: Escenario Realista | 4 |
| Comparación Entre Algoritmos | 4 |
| Métodos Clásicos | 4 |
| Machine Learning No Supervisado | 4 |
| Deep Learning | 4 |
| Análisis de Variables Críticas | 5 |
| Sensores con Mayor Poder Discriminativo | 5 |
| Propuesta de Arquitectura: Gemelo Digital | 5 |
| Componentes del Sistema..... | 5 |
| Requisitos No Funcionales..... | 6 |
| Métricas Operativas | 6 |
| Conclusiones | 6 |
| Código | 6 |
| Apéndice: Configuración de Hiperparámetros | 7 |
| Autoencoder..... | 7 |
| LSTM-Autoencoder | 7 |
| TCN-VAE..... | 7 |
| Isolation Forest..... | 7 |
| PCA | 7 |

Introducción

Contexto del Problema

El mantenimiento predictivo en sistemas industriales críticos, como los motores de aeronaves, representa un desafío técnico y económico importante. La capacidad de detectar la degradación temprana permite reducir los costes de mantenimiento, aumentar la disponibilidad y prevenir fallos.

Dataset NASA C-MAPSS

El dataset C-MAPSS (Commercial Modular Aero-Propulsion System Simulation) fue desarrollado por la NASA con fines de investigación. Este contiene cuatro subdatasets que varían en complejidad.

FD001 incluye 100 unidades de entrenamiento y 100 de prueba, opera bajo una única condición operativa (nivel del mar) y presenta un solo modo de fallo (degradación HPC). FD002 aumenta la escala a 260 unidades de entrenamiento y 259 de prueba, introduce seis condiciones operativas diferentes, pero mantiene el modo de fallo único. FD003 vuelve a las 100 unidades, pero añade un segundo modo de fallo (HPC más ventilador) bajo condición única. Finalmente, FD004 combina la máxima complejidad con 248 unidades de entrenamiento, 249 de prueba, seis condiciones operativas y dos modos de fallo simultáneos.

Cada unidad temporal contiene tres configuraciones operativas (altitud, número de Mach y TRA), veintiún sensores que miden temperatura, presión, velocidad y relaciones de flujo, además del identificador de unidad y ciclo temporal.

Metodología

Preprocesamiento de Datos

Para todos los experimentos se aplicó análisis exploratorio de datos (EDA). Los datos se normalizaron mediante StandardScaler sobre las características de sensores. Se eliminaron las variables constantes, es decir, aquellos sensores sin variabilidad. Para los métodos temporales se crearon ventanas de 50 ciclos consecutivos. Las etiquetas de *ground truth* se definieron con base en el RUL (Remaining Useful Life).

Métricas de Evaluación

La comparación entre algoritmos se realizó utilizando varias métricas complementarias. La tasa de detección indica el porcentaje de unidades donde se identificaron anomalías. El RUL promedio en primera detección mide cuántos ciclos restaban hasta el fallo cuando se detectó la primera anomalía. El porcentaje del ciclo de vida indica el momento relativo de detección respecto a la vida útil total. Las detecciones tempranas cuentan las unidades identificadas antes del 50% de su vida útil. La tasa de falsos positivos mide el porcentaje de detecciones incorrectas durante la fase normal de operación.

Algoritmos Implementados

Los métodos clásicos incluyeron Z-score, que detecta anomalías basándose en la desviación estándar con la ventaja de ser computacionalmente eficiente, aunque limitado, y PCA, que realiza reducción dimensional seguida de reconstrucción para detectar anomalías.

Entre las técnicas de *machine learning* no supervisado se implementaron Isolation Forest.

Las arquitecturas de deep learning comprendieron el Autoencoder estándar con arquitectura de 21 entradas, capas ocultas de 64, 32 y 16 unidades en el encoder y su espejo en el decoder, utilizando activación ReLU y pérdida MSE para detectar anomalías mediante el error de reconstrucción. El LSTM-Autoencoder incorpora dos capas LSTM de 64 y 32 unidades en el encoder para capturar dependencias temporales en secuencias de 50 ciclos. El TCN-VAE combina convoluciones causales temporales en el encoder con un espacio latente gaussiano de dimensión 16 y optimiza la función de pérdida ELBO, ofreciendo un receptive field extenso junto con modelado probabilístico.

Resultados Experimentales

FD001: Escenario base

Este subdataset representa el caso más simple con condiciones operativas constantes y un único modo de degradación HPC.

TCN-VAE destacó con detección perfecta del 100%, identificando anomalías cuando restaban 148 ciclos de vida útil (27% del ciclo) y logrando que todas las unidades fueran detectadas tempranamente, aunque con 36% de falsos positivos. Z-score logró detección temprana con 172 ciclos restantes pero su alta sensibilidad podría generar falsos positivos. LSTM-Autoencoder mostró una tasa de falsos positivos notablemente baja de solo 3%, aunque detectó más tarde en el ciclo de vida. PCA fue el único método que no alcanzó cobertura completa, quedándose en 93% de detección.

FD002: Complejidad Multimodal

La introducción de seis condiciones operativas diferentes aumentó significativamente la complejidad del problema.

El resultado más notable fue el colapso total de Z-score, que solo logró detectar el 4.62% de las unidades, evidenciando su fragilidad ante la variabilidad de los datos complejos. Autoencoder consiguió un 100% de detección, buen balance en detección temprana con 96.83 ciclos de RUL restante y falsos positivos razonables del 18.46%. Isolation Forest logró detección muy temprana con 146 ciclos pero presentó una tasa de falsos positivos del 53.85%. LSTM-Autoencoder sufrió una caída significativa quedándose en 70% de detección.

FD003: Fallo Multimodal

Este dataset presenta condición operativa única pero dos modos de degradación simultáneos.

TCN-VAE ofreció el mejor balance con 100% de detección y solo 7% de falsos positivos. PCA y Autoencoder detectaron más temprano pero con tasas de falsos positivos entre 40% y 45%. Los métodos temporales como LSTM-AE y TCN-VAE mostraron mayor robustez ante múltiples patrones de degradación. Isolation Forest detectó muy tarde, cuando ya había transcurrido el 81% del ciclo de vida, con pocas detecciones tempranas.

FD004: Escenario Realista

El escenario más realista combina variabilidad operativa con múltiples modos de fallo.

TCN-VAE destacó nuevamente con la detección más temprana, identificando anomalías cuando restaban 176 ciclos de RUL (28% del ciclo) y logrando que el 99.2% de las unidades fueran detectadas tempranamente, aunque con 60% de falsos positivos. Autoencoder mantuvo 100% de detección con falsos positivos moderados del 29%. LSTM-Autoencoder mostró degradación significativa con solo 64% de detección en este escenario de máxima complejidad. Z-score nuevamente falló con apenas 4.82% de detección, confirmando su inviabilidad para escenarios multimodales. Isolation Forest balanceó bien la detección perfecta con falsos positivos del 22%.

Comparación Entre Algoritmos

Métodos Clásicos

Los métodos clásicos ofrecen ventajas en eficiencia computacional e interpretabilidad, con buen desempeño en escenarios controlados como FD001. Sin embargo, presentan debilidades importantes. Z-score colapsa totalmente en datasets multimodales, alcanzando apenas 5% de detección en FD002 y FD004. PCA presenta tasas de falsos positivos elevadas de hasta 45% en FD003. Ninguno de estos métodos captura dependencias temporales y ambos son frágiles ante heterogeneidad operativa. La recomendación es que solo son viables para FD001 y no se recomienda su uso en producción.

Machine Learning No Supervisado

Isolation Forest logra 100% de detección en todos los datasets con múltiples condiciones sin requerir modelado de dependencias temporales, mostrando robustez ante multimodalidad. Sus debilidades incluyen tasas de falsos positivos inaceptables en FD002 (53.85%), detección tardía en FD003 (81% del ciclo) y falta de interpretabilidad en las decisiones. Es viable para FD001 y FD004 pero requiere ajuste de hiperparámetros para reducir falsos positivos.

Deep Learning

TCN-VAE ofrece el mejor desempeño general, logrando en FD001 detección perfecta en el 27% del ciclo con 36% de falsos positivos, y en FD004 detección perfecta en el 28% del ciclo con 60% de falsos positivos. Autoencoder demuestra consistencia sobresaliente con 100% de detección en FD002, FD003 y FD004, manteniendo falsos positivos controlados entre 18% y 41%. Ambos capturan patrones complejos y temporales. LSTM-Autoencoder presenta degradación inesperada en FD002 (70%) y FD004 (64%). Estos métodos requieren más

recursos computacionales, mayor tiempo de entrenamiento y son sensibles a los hiperparámetros. Se recomienda TCN-VAE cuando la detección temprana es crítica, y Autoencoder para balance entre simplicidad y rendimiento.

Análisis de Variables Críticas

Sensores con Mayor Poder Discriminativo

Mediante análisis de importancia de reconstrucción en Autoencoders y contribución a componentes principales en PCA se identificaron los cinco sensores más críticos. El Sensor 11, que mide presión estática a la salida del HPC, es indicador directo de degradación HPC. El Sensor 4 mide temperatura de salida del LPC y refleja eficiencia del compresor de baja presión. El Sensor 7 registra temperatura total a la salida del HPC. El Sensor 15 mide presión total en el conducto bypass. El Sensor 21 monitorea el sangrado de refrigerante de HPT e indica eficiencia de la turbina de alta presión.

Propuesta de Arquitectura: Gemelo Digital

Componentes del Sistema

El sistema propuesto comienza con los 21 sensores del motor, cuyos datos se recolectan en un dispositivo de borde que también registra el identificador de unidad y el ciclo temporal. El preprocesamiento ocurre en este mismo dispositivo, aplicando normalización mediante StandardScaler, eliminando variables constantes y generando ventanas de 50 ciclos para los modelos temporales.

La ingesta y preprocesamiento ocurre en el propio dispositivo, aplicando normalización por régimen operativo mediante StandardScaler, eliminando variables constantes y generando ventanas de 50 ciclos para los modelos temporales.

El corazón del sistema es el Ensemble Detector, donde TCN-VAE actúa como detector primario realizando detección temprana sobre el espacio latente con umbral en el percentil 90, actualizándose cada 10 ciclos. Autoencoder funciona como sistema alternativo en caso de fallo del primario y confirmación

Un estimador de RUL basado en regresión temporal con LSTM estima la vida útil restante por unidad, generando una salida conjunta con el score de anomalía para priorización.

El modulo de inferencia combina el score de anomalía, RUL estimado, criticidad del sensor e historial para generar una puntuación de 0 a 100 la probabilidad de fallo de un componente del sistema.

Las salidas del sistema incluyen dashboard para visualizar scores de anomalía, RUL, tendencias por sensor y logs de detección y conexión a base de datos de mantenimiento para órdenes de trabajo automáticas e historial de eventos.

Requisitos No Funcionales

- La latencia objetivo es menor a 5 segundos por ciclo para inferencia.
- El reentrenamiento se programa trimestralmente en producción con capacidad de ajustes rápidos ante drift.
- El monitoreo de salud de modelos incluye seguimiento de tasa de falsos positivos, cobertura de detección y tiempo medio hasta alerta.
- Se exponen endpoints seguros para integración con sistemas de mantenimiento existentes.

Métricas Operativas

El sistema define latencia de detección menor a 5 segundos por ciclo, tasa de falsos positivos aceptable menor al 15% (equivalente a aproximadamente 3 alertas mensuales por motor), detección mínima antes del fallo de 30% (más de 100 ciclos de RUL restante), y actualización de modelos mediante reentrenamiento trimestral con datos nuevos.

Conclusiones

TCN-VAE es el algoritmo con mejor desempeño global, logrando detección temprana consistente entre 27% y 28% del ciclo en escenarios complejos, con cobertura perfecta en todos los datasets y robustez ante multimodalidad operativa y de fallos.

Autoencoder estándar ofrece el mejor balance costo-beneficio gracias a su arquitectura simple y entrenable, alcanzando 100% de detección en FD002, FD003 y FD004 con falsos positivos controlados entre 18% y 41%.

Los métodos clásicos no son viables para producción. Z-score colapsa en escenarios multimodales con menos del 5% de detección, mientras que PCA está limitado por falsos positivos elevados y falta de modelado temporal.

LSTM-Autoencoder muestra fragilidad inesperada con degradación severa del 70% en FD002 y 64% en FD004, posiblemente debido a sobreajuste a condiciones de entrenamiento específicas.

Isolation Forest es viable con ajustes pero requiere tuning agresivo del parámetro de contaminación para reducir falsos positivos, siendo útil como validador complementario en arquitectura ensemble.

Código

Poder consultar el código desarrollado, consulte:

https://github.com/Santisoutoo/Anomaly_detection.git

Apéndice: Configuración de Hiperparámetros

Autoencoder

La arquitectura utiliza capas de 21, 64, 32, 16, 32, 64 y 21 unidades con activación ReLU. Se optimiza con Adam a learning rate de 0.001, batch size de 256 durante 100 epochs máximo con early stopping de paciencia 10 monitoreando la pérdida de validación.

LSTM-Autoencoder

El encoder tiene capas de 64 y 32 unidades LSTM, el decoder las mismas en orden inverso. Utiliza secuencias de 50 pasos temporales con dropout de 0.2, optimizador Adam a learning rate 0.001, batch size de 128 durante 150 epochs.

TCN-VAE

El encoder usa filtros de 64 y 128 con kernel de tamaño 3 y dilataciones de 1, 2, 4 y 8. La dimensión latente es 16. El decoder tiene filtros de 128 y 64. El peso de KL divergence (beta) es 0.5. Se optimiza con Adam a learning rate 0.0005, batch size de 64 durante 200 epochs.

Isolation Forest

Utiliza 100 estimadores con contaminación de 0.1, max features de 1.0 sin bootstrap y semilla aleatoria 42.

PCA

Retiene componentes para explicar 95% de la varianza con whitening activado y umbral de anomalía en el percentil 95.