

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

**FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA,
INFORMÁTICA Y MECÁNICA**

ESCUELA PROFESIONAL DE INGENIERÍA MECÁNICA



TESIS

**IDENTIFICACIÓN DEL DESALINEAMIENTO
DE EJES EN UN BANCO DE PRUEBAS
MEDIANTE ALGORITMOS DE INTELIGENCIA ARTIFICIAL**

PRESENTADO POR:

Br. WILFREDO OMAR ESTRADA ABARCA

Br. HERBERT AUGUSTO CAMPANA SOTELO

**PARA OPTAR AL TÍTULO PROFESIONAL
DE INGENIERO MECÁNICO**

ASESOR:

Mg. ARTURO MACEDO SILVA

**CUSCO – PERÚ
2026**



Universidad Nacional de San Antonio Abad del Cusco

INFORME DE SIMILITUD

(Aprobado por Resolución Nro. CU-321-2025-UNSAAC)

El que suscribe, el Asesor Mgt. ARTURO MACEDO SILVA
quien aplica el software de detección de similitud al trabajo de investigación/tesis titulada:
IDENTIFICACION DEL DESALINEAMIENTO DE EJES EN UN BANCO DE PRUEBAS MEDIANTE ALGORITMOS DE INTELIGENCIA ARTIFICIAL

Presentado por:
ESTRADA ABARCA, Wilfredo Omar DNI N° 72746132

Presentado por:
CAMPANA SOTELO, Herbert Augusto DNI N° 47654400

Para optar el título profesional/Grado Académico de INGENIERO MECANICO

Informo que el trabajo de investigación ha sido sometido a revisión por 2 veces, mediante el Software Antiplagio, conforme al Art. 6° del **Reglamento para Uso de Sistema Antiplagio de la UNSAAC** y de la evaluación de originalidad se tiene un porcentaje de 7%.

Evaluación y acciones del reporte de coincidencia para trabajos de investigación conducentes a grado académico o título profesional, tesis

Porcentaje	Evaluación y Acciones	Marque con una (X)
Del 1 al 10%	No se considera plagio.	X
Del 11 al 30 %	Devolver al usuario para las correcciones.	
Mayor a 31%	El responsable de la revisión del documento emite un informe al inmediato jerárquico, quien a su vez eleva el informe a la autoridad académica para que tome las acciones correspondientes. Sin perjuicio de las sanciones administrativas que correspondan de acuerdo a Ley.	

Por tanto, en mi condición de asesor, firmo el presente informe en señal de conformidad y adjunto las primeras páginas del reporte del Sistema de Detección de Similitud

Cusco, 05 de Junio del 2026

Firma

Post firma Arturo Macedo Silva

Nro. de DNI 23821894

ORCID del Asesor 0000-0002-2794-949X

Se adjunta:

1. Reporte generado por el Sistema Antiplagio.
2. Enlace del Reporte Generado por el Sistema Antiplagio: trn: oid: 27259:597620858
3. <https://unsaac.turnitin.com/viewer/submissions/oid:27259:597620858?locale=es-MX>

Wilfredo Omar y Herbert Augusto ESTRADA ABARC...

IDENTIFICACIÓN DEL DESALINEAMIENTO DE EJES EN UN BANCO DE PRUEBAS MEDIANTE ALGORITMOS DE INTELIGE...

 Universidad Nacional San Antonio Abad del Cusco

Detalles del documento

Identificador de la entrega

trn:oid:::27259:597620858

Fecha de entrega

5 jun 2026, 10:36 a.m. GMT-5

Fecha de descarga

5 jun 2026, 10:55 a.m. GMT-5

Nombre del archivo

tesis_desalineamiento_ejes.pdf

Tamaño del archivo

36.1 MB



234 páginas

40.985 palabras

223.670 caracteres

7% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...




Filtrado desde el informe

- ▶ Bibliografía
- ▶ Texto citado
- ▶ Texto mencionado
- ▶ Coincidencias menores (menos de 9 palabras)
- ▶ Trabajos entregados
- ▶ Base de datos de Crossref
- ▶ Base de datos de contenido publicado de Crossref

Exclusiones


- ▶ N.º de coincidencias excluidas

Fuentes principales

- 7%  Fuentes de Internet
- 2%  Publicaciones
- 0%  Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alerta de integridad para revisión

-  **Caracteres reemplazados**
51 caracteres sospechosos en N.º de páginas
Las letras son intercambiadas por caracteres similares de otro alfabeto.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

Dedicatoria

A mi familia, por estar siempre a mi lado, por apoyarme en los momentos difíciles y por darme la fuerza para seguir adelante. Gracias por su cariño y paciencia, este logro también es de ustedes.

Wilfredo Estrada

Dedico este trabajo de investigación a mi familia, por su apoyo incondicional, motivación y palabras de aliento durante el desarrollo de esta etapa académica. Su respaldo fue importante para seguir adelante y culminar este proyecto con esfuerzo y excelencia.

Herbert Campana

Presentación

El diagnóstico de fallas en maquinaria rotativa constituye una parte esencial del mantenimiento predictivo, ya que permite detectar condiciones anómalas en etapas tempranas, antes de que generen daños severos, paradas no programadas o pérdidas económicas. En este tipo de equipos, fallas como el desalineamiento de ejes pueden alterar el comportamiento dinámico del sistema, incrementar los niveles de vibración y producir esfuerzos adicionales en rodamientos, acoplamientos y soportes, reduciendo progresivamente la vida útil de los componentes.

Aunque el análisis vibracional y espectral es una técnica ampliamente utilizada para la detección de estas fallas, su interpretación puede depender de la experiencia del especialista y verse afectada por factores como el ruido de la señal, las variaciones de velocidad o las condiciones propias del montaje. Esto puede dificultar la identificación precisa del nivel de severidad de la falla, especialmente cuando se requiere un diagnóstico repetible y automatizado.

En este contexto, la presente investigación propone una metodología experimental y computacional orientada a la identificación automática del nivel de desalineamiento angular en ejes rotativos. Para ello, se emplea un módulo de pruebas diseñado para reproducir distintos niveles de desalineamiento bajo condiciones controladas. Durante los ensayos, las señales generadas por el sistema son adquiridas mediante un sensor CKP, posteriormente procesadas y utilizadas para entrenar algoritmos de aprendizaje automático.

De esta manera, el enfoque propuesto busca mejorar la precisión y repetibilidad del diagnóstico, reduciendo la dependencia de la interpretación manual. Además, la base de datos experimental obtenida permite evaluar el comportamiento del sistema frente a diferentes niveles de falla y validar modelos predictivos aplicables al monitoreo de maquinaria rotativa.

Índice general

Introducción	III
Índice de cuadros	XI
Lista de Tablas	XII
Índice de figuras	XIII
Lista de Figuras	XVII
1. Problema de Investigación	5
1.1. Problema General	7
1.2. Problemas Específicos	7
1.3. Justificación	8
2. Objetivos	10
2.1. Objetivo General	10
2.2. Objetivos Específicos	10
3. Hipótesis	12
4. Variables	13
4.1. Variables de la investigación	13
4.1.1. Variable independiente	13

4.1.2. Variable dependiente	14
4.1.3. Variables de control	15
4.1.4. Matriz de operacionalización de variables	16
5. ALCANCES Y LIMITACIONES	17
5.1. Alcances	17
5.2. Limitaciones	18
6. Metodología	19
6.1. Tipo de Investigación	21
6.2. Métodos de Validación Propuestas	21
7. Estado del Arte	23
7.1. Desalineamiento de ejes	23
7.2. Tipos de desalineamiento presentes en ejes	24
7.2.1. Desalineamiento paralelo	24
7.2.2. Desalineamiento angular	25
7.3. Detección del desalineamiento en ejes	26
7.4. Análisis espectral y de forma de onda	29
7.4.1. Señales continuas y discretas	30
7.4.2. Señal determinística	32
7.4.3. Señal aleatoria	33
7.4.4. Teoría del muestreo	35
7.4.5. Muestreo ideal	36
7.5. Sensores de posición de cigueñal CKP (Crankshaft Position Sensor).	37
7.6. Aprendizaje de máquina (Machine Learning)	38
7.7. Clasificación de algoritmos de Machine Learning	39
7.8. Perceptrón	42

7.9. Redes neuronales	45
7.9.1. Redes neuronales completamente conectadas	47
7.10. Transformada de Fourier	48
8. Antecedentes	50
8.1. Estudios relacionados con análisis vibracional en maquinaria rotativa .	51
8.1.1. Determinación de fallas mediante análisis vibracional en equipos críticos	51
8.1.2. Plan de mantenimiento predictivo basado en análisis vibracional en sistemas de bombeo	52
8.2. Estudios relacionados con bancos experimentales	53
8.2.1. Diseño e implementación de un banco didáctico para alineación de elementos rotativos	53
8.3. Estudios sobre el fenómeno de desalineamiento en ejes	54
8.3.1. Desalineamiento como falla en maquinaria rotativa	54
8.3.2. Análisis vibracional para detección de desalineamiento	55
8.4. Modelamiento dinámico y procesamiento de señales	56
8.4.1. Modelamiento dinámico del desalineamiento	56
8.4.2. Técnicas de procesamiento de señales para diagnóstico de desalineamiento	57
8.5. Enfoques multisensoriales y aprendizaje automático	58
8.5.1. Enfoques multisensoriales para detección de desalineamiento . .	58
8.5.2. Aprendizaje automático para diagnóstico de fallas en maquinaria rotativa	59
8.6. Aporte global de los antecedentes a la presente investigación	60
9. DISEÑO Y CONSTRUCCIÓN DE MÓDULO DE PRUEBAS	61
9.1. DISEÑO CONCEPTUAL	61
9.1.1. Metodología de diseño	61

ÍNDICE GENERAL

9.1.2.	Lista de exigencias	63
9.1.3.	Caja negra	64
9.1.4.	Estructura de funciones	65
9.1.5.	Matriz morfológica	66
9.1.6.	Valoración técnica económica	70
9.2.	Diseño modular	72
9.2.1.	Sistema motriz	73
9.2.2.	Placa base de soporte	74
9.2.3.	Volante de inercia	82
9.2.4.	Acople universal	88
9.2.5.	Simulación computacional del acople	89
9.2.6.	Análisis de esfuerzos	90
9.2.7.	Sensor de campo magnético	91
9.2.8.	Sistema de adquisición de datos	94
9.2.9.	Rodamiento de bolas	95
9.3.	Análisis por Elementos Finitos del módulo de pruebas	102
9.3.1.	Propiedades de la geometría	102
9.3.2.	Modelamiento tridimensional	102
9.3.3.	Mallado	105
9.3.4.	Análisis modal	106
9.4.	Componentes del equipo	112
10.	CAPTURA DE SEÑALES DE VIBRACIÓN	115
10.1.	Parámetros de operación generales	116
10.2.	Selección de frecuencia de muestreo según teorema de Nyquist	117
10.3.	Diseño y desarrollo experimental	120
10.3.1.	Planteamiento experimental	122

10.3.2. Definición de los niveles de desalineamiento	123
10.3.3. Procedimiento general de ensayo	124
10.3.4. Estructuración del conjunto de datos experimental	125
10.4. Captura de señales de vibración	126
11. PROCESAMIENTO DE SEÑALES	130
11.1. Análisis de señales en el dominio del tiempo	130
11.2. Procesamiento de señales mediante la Transformada Rápida de Fourier FFT	131
11.2.1. Extracción de características de la base de datos	134
12. DISEÑO E IMPLEMENTACIÓN DEL ALGORITMO DE APRENDIZAJE AUTOMÁTICO	138
12.1. Diseño del espacio de posibles soluciones	140
12.1.1. Modelamiento matemático del fenómeno	141
12.2. Selección del Algoritmo de Aprendizaje Automático	145
12.2.1. Random Forest Regressor	146
12.2.2. XGBoost Regressor	149
12.2.3. Perceptrón Multicapa: MLPRegressor	151
12.2.4. Criterio de selección de los modelos	153
12.3. Ejecución del algoritmo	155
12.4. Validación de resultados experimentales	158
12.5. Validación del modelo con conjunto de datos externo	160
12.5.1. Referencia del conjunto de datos empleado	160
12.5.2. Correspondencia de clases para el preentrenamiento y la validación	161
12.5.3. Procedimiento de comprobación con el conjunto de datos real .	162
12.5.4. Compatibilización de señales y extracción de características . . .	164
12.5.5. Metodología de preentrenamiento y ajuste fino con conjunto de datos externo	164

12.5.6. Resultados de la validación externa	165
13. Presupuesto	170
14. Conclusiones	171
15. Recomendaciones	175
Bibliografía	176
Bibliografía	180
A. Anexo A: CÓDIGO FUENTE RANDOM FOREST	181
B. Anexo B: CÓDIGO FUENTE XG BOOST	184
C. Anexo C: CÓDIGO FUENTE RED NEURONAL	187
D. Anexo D: CÓDIGO FUENTE DE EXTRACCIÓN DE CARACTERÍSTICAS	190
E. Anexo E: CÓDIGO FUENTE DEL ACONDICIONAMIENTO DE SEÑAL	193
F. Anexo F: CÓDIGO FUENTE DEL PREENTRENAMIENTO DEL MODELO	196
G. Anexo G: CÓDIGO FUENTE DEL AJUSTE FINO DEL MODELO	199
H. Anexo H: CÓDIGO FUENTE DEL ENTRENAMIENTO CON EL CONJUNTO DE REFERENCIA	203
I. Anexo I: CÓDIGO FUENTE DE LA PREPARACIÓN INICIAL DE LOS DATOS	207
J. Anexo J: Especificaciones técnicas de la interfaz de captura de audio PRESOUNUS STUDIO 24C	210
K. Anexo K: Especificaciones técnicas del rodamiento de bolas con chumacera KFL08	212

Anexo L: Planos

214

Índice de cuadros

4.1. Operacionalización de la variable independiente	13
4.2. Operacionalización de la variable dependiente	14
4.3. Variables de control consideradas en la investigación	15
4.4. Matriz de operacionalización de variables	16
9.1. Evaluación de criterios económicos	70
9.2. Evaluación de criterios técnicos.	71
9.3. Parámetros geométricos y físicos del modelo	74
9.4. Resultados (según DCL y diagramas obtenidos)	76
9.5. Refinamiento de malla.	106
9.6. Frecuencias naturales.	110
9.7. Lista de materiales del módulo de pruebas	114
11.1. Resumen de fórmulas estadísticas utilizadas en el análisis de señales . .	135
12.1. Funcionamiento general de Random Forest Regressor.	147
12.2. Funcionamiento general de XGBoost Regressor.	149
12.3. Funcionamiento general de MLPRegressor.	152
12.4. Criterio de selección de los algoritmos de regresión evaluados.	154
12.5. Parámetros utilizados para Random Forest Regressor	156
12.6. Parámetros utilizados para XGBoost Regressor	157

12.7. Parámetros utilizados para la red neuronal MLPRegressor 158

12.8. Resultados obtenidos con los modelos de regresión evaluados 159

12.9. Agrupación de clases empleada en el preentrenamiento con datos del módulo experimental. 161

12.10Correspondencia de clases en el *dataset* real de desalineamiento angular horizontal. 162

12.11Comparación del desempeño en la validación externa. 166

12.12Mejora obtenida mediante la técnica de preentrenamiento. 167

12.13Comparación de tiempos de entrenamiento en la validación externa. . . 169

13.1. Recursos para el desarrollo del proyecto de investigación. 170

Índice de figuras

7.1. Fuerzas presentes debido al desalineamiento paralelo. Fuente: White (2010).	25
7.2. Fuerzas presentes debido al desalineamiento angular White (2010).	26
7.3. Procedimiento estandarizado para la detección de fallas mediante análisis vibracional, adaptado de Navarro-Navarro et al. (2025).	27
7.4. Análisis FFT de señales de vibración en un módulo de pruebas bajo régimen estacionario para diferentes niveles de desalineamiento de ejes, adaptado de Navarro-Navarro et al. (2025).	27
7.5. Sistema experimental para la medición láser del desalineamiento en ejes, adaptado de Simm et al. (2016).	28
7.6. Sistema de alineación láser Easy-Laser Extreme utilizado como referencia de equipamiento industrial para la medición y corrección del desalineamiento de ejes, adaptado de Damalini AB (2014).	29
7.7. Señal continua en el tiempo ESPOLBlogSenalesContinuas2017.	31
7.8. Señal discreta en el tiempo ESPOLBlogSenalesContinuas2017.	32
7.9. Señal determinística Aviles (2003).	33
7.10. Señal aleatoria. Aviles (2003).	34
7.11. Señal continua en el tiempo reconstruida de manera digital mediante un muestreo con ventanas de tiempo uniforme Nyquist (1928).	35
7.12. Diagrama de bloques de la teoría del muestreo, en el que se representa el proceso de muestreo de una señal continua para su conversión al dominio discreto.	36
7.13. Principio de funcionamiento de sensor CKP Castellanos Martínez, José Francisco (2015).	38

7.14. Clasificación de los algoritmos de Aprendizaje Automático (Machine Learning).	39
7.15. Esquema general de un árbol de decisión, donde cada nodo interno representa una regla de decisión, cada rama una condición asociada y cada nodo hoja una salida o predicción. Adaptado de scikit-learn scikit-learn developers (2026).	41
7.16. Representación esquemática del perceptrón, donde las entradas son ponderadas mediante pesos, sumadas junto con un sesgo y evaluadas mediante una función de activación para obtener la salida del modelo Rosenblatt (1958).	44
7.17. Arquitectura de una red neuronal completamente conectada, compuesta por una capa de entrada, capas ocultas y una capa de salida. Adaptado de Basogain Olabe Olabe (s f).	48
7.18. Transformada de Fourier de diferentes espectros de vibración.	49
9.1. Procedimiento de diseño Adaptado: VDI 2221.	62
9.2. Lista de exigencias.	63
9.3. Caja negra.	64
9.4. Estructura de funciones del módulo de pruebas desarrollada según la metodología VDI 2221. Se consideran como funciones principales el control de velocidad, la rotación del eje, la transmisión de potencia, la medición de la señal y la captura de la señal.	66
9.5. Matriz morfológica elaborada para la selección conceptual del módulo de pruebas.	67
9.6. Solución 1	68
9.7. Solución 2	69
9.8. Solución 3	69
9.9. Gráfico de evaluación técnico-económica de las soluciones propuestas. El eje horizontal representa el puntaje técnico porcentual y el eje vertical representa el puntaje económico porcentual. La línea diagonal indica una condición de equilibrio entre ambos criterios.	72
9.10. Modelo de motor seleccionado	73
9.11. Modelo conceptual estático del módulo de pruebas.	74
9.12. Diagrama de cuerpo libre estático del módulo de pruebas.	76

9.13. Diagrama de fuerzas cortantes.	77
9.14. Diagrama de momentos flectores.	77
9.15. Representación matemática de un volante de inercia. Richard G. Budynas (2012)	82
9.16. Volante de inercia seleccionado	86
9.17. Acople universal tipo cardán de 8 mm de diámetro de eje.	88
9.18. Propiedades del aluminio. Fuente: ANSYS Materials.	89
9.19. Propiedades de calidad del mallado inicial del acople flexible.	90
9.20. Esfuerzos equivalentes en acople con malla refinada de 2.5mm.	90
9.21. Esfuerzos equivalentes en pines internos	91
9.22. Sensor de posición de sigueñal CKP.	92
9.23. Modo de funcionamiento de un sensor CKP.	93
9.24. Interfaz de audio PreSonus Studio 24c Pre (2019)	94
9.25. Entorno de trabajo de software de captura de audio (Digital Audio Workstation) Studio One.	95
9.26. Rodamiento de bolas con chumacera tipo KFL08 utilizado en el módulo de pruebas.	102
9.27. Modelo tridimensional del módulo de pruebas.	104
9.28. Modelo mallado.	105
9.29. Calidad inicial de malla.	105
9.30. Convergencia de malla.	107
9.31. Primer modo de vibración.	108
9.32. Segundo modo de vibración.	108
9.33. Tercer modo de vibración.	109
9.34. Cuarto modo de vibración.	109
9.35. Quinto modo de vibración.	109
9.36. Sexto modo de vibración.	110

9.37. Vista de explosión del módulo de pruebas con identificación de componentes.	113
10.1. Medición de frecuencia de operación con tacómetro digital.	116
10.2. Principio de Aliasing.	118
10.3. Frecuencia de muestreo de interfaz de audio PreSonus Studio 24cPre (2019)	118
10.4. Arreglo general de las diferentes pruebas realizadas.	119
10.5. Diagrama de flujo de la metodología experimental y computacional empleadas.	121
10.6. Resumen visual de la adquisición experimental, la generación de ventanas de 1 segundo y la extracción de características a partir de las señales de vibración medidas para cada nivel de desalineamiento.	123
10.7. Resumen visual de la construcción del conjunto de datos, la partición en subconjuntos, el entrenamiento de los algoritmos y la evaluación del modelo final para la identificación del nivel de desalineamiento.	126
10.8. Montaje de sensor con referencia a la rueda fónica Castellanos Martínez, José Francisco (2015).	127
10.9. Montaje del sensor.	127
10.10 Ejemplo de la señal capturada durante el proceso de adquisición de datos en el módulo de pruebas.	128
10.11 Niveles de desalineamiento en ejes.	129
11.1. Señal en el tiempo capturada de distintos niveles de desalineamiento angular.	131
11.2. Comparación espectral mediante FFT de las señales obtenidas con rueda fónica de 4 y 10 pines, considerando condiciones de eje alineado y desalineamientos angulares de 5°, 10° y 15°.	133
11.3. Base de datos obtenida de la extracción de parámetros.	136
11.4. Dispersión de pruebas a distintos niveles de desalineamiento analizado por un análisis de componentes principales PCA.	137
12.1. Flujograma para la implementación de un sistema inteligente.	139

12.2. Variación temporal de la velocidad angular de salida para diferentes ángulos de desalineamiento, considerando una velocidad de entrada constante de 940 rpm.	145
12.3. Esquema general del funcionamiento de Random Forest Regressor para la estimación del ángulo de desalineamiento.	148
12.4. Diagrama simplificado del funcionamiento de Random Forest, donde se observa el uso de múltiples árboles de decisión para generar una predicción más robusta que la obtenida con un solo árbol. Adaptado de GeeksforGeeks (2026).	148
12.5. Esquema general del funcionamiento de XGBoost Regressor para la estimación del ángulo de desalineamiento.	150
12.6. Diagrama simplificado del funcionamiento de XGBoost, donde se muestra la construcción secuencial de árboles y la suma de sus contribuciones para obtener el resultado final. Adaptado de GeeksforGeeks (2025). . .	151
12.7. Esquema general del funcionamiento de MLPRegressor para la estimación del ángulo de desalineamiento.	153
12.8. Diagrama esquemático del desalineamiento angular empleado como referencia conceptual del procedimiento experimental. Fuente: Kechik et al. (2021).	161
12.9. Resumen visual del procedimiento de comprobación con el conjunto de datos real, incluyendo la selección del subconjunto de desalineamiento angular horizontal, la segmentación en ventanas de 1 segundo, la extracción de características y la comparación entre entrenamiento desde cero y preentrenamiento con ajuste fino.	163
12.10 Procedimiento general de aprendizaje por transferencia aplicado al modelo de red neuronal, considerando el preentrenamiento con datos del módulo experimental, el ajuste fino con el conjunto externo y la comparación frente al entrenamiento directo.	165
12.11 Comparación de la precisión obtenida en la validación externa. La estrategia basada en preentrenamiento y ajuste fino presentó una mejora relativa de 8.47 % respecto al entrenamiento directo.	168

Resumen

El desalineamiento angular en ejes rotativos representa una condición de falla relevante debido a que modifica la transmisión del movimiento, altera la respuesta dinámica del sistema y genera variaciones medibles en las señales asociadas a su operación. Su detección resulta importante dentro del mantenimiento predictivo, ya que permite reconocer cambios en el comportamiento del equipo antes de que se produzcan daños mayores en rodamientos, acoplamientos o elementos de soporte.

En esta investigación se desarrolla una metodología experimental y computacional para la identificación automática de niveles de desalineamiento angular en ejes rotativos. Para ello, se diseñó y construyó un módulo de pruebas capaz de reproducir condiciones controladas de falla, considerando incrementos angulares de 1° en 1° . Durante los ensayos, las señales fueron adquiridas mediante un sensor electromagnético de posición de cigüeñal CKP, obteniéndose una base de datos experimental representativa del comportamiento del sistema.

Posteriormente, las señales registradas fueron segmentadas, procesadas y organizadas para extraer características relevantes asociadas al fenómeno estudiado. Con esta información se implementaron algoritmos de aprendizaje automático orientados al reconocimiento de patrones.

Finalmente, los modelos fueron evaluados con datos experimentales y datos reales de referencia, evidenciando el potencial de la metodología propuesta para el diagnóstico automático de desalineamiento.

Palabras clave: Desalineamiento Angular, Procesamiento de Señales, Aprendizaje Automático, Inteligencia Artificial, Módulo de Pruebas.

Abstract

Angular misalignment in rotating shafts represents a relevant fault condition because it modifies motion transmission, alters the dynamic response of the system, and generates measurable variations in the signals associated with its operation. Its detection is important within predictive maintenance, as it allows changes in equipment behavior to be identified before major damage occurs in bearings, couplings, or support elements.

This research develops an experimental and computational methodology for the automatic identification of angular misalignment levels in rotating shafts. For this purpose, a test module was designed and built to reproduce controlled fault conditions, considering angular increments of 1° in 1° . During the tests, the signals were acquired using a CKP electromagnetic crankshaft position sensor, obtaining an experimental database representative of the system behavior.

Subsequently, the recorded signals were segmented, processed, and organized to extract relevant features associated with the phenomenon under study. With this information, machine learning algorithms aimed at pattern recognition were implemented.

Finally, the models were evaluated using experimental data and real reference data, demonstrating the potential of the proposed methodology for the automatic diagnosis of misalignment.

Keywords: Angular Misalignment, Signal Processing, Machine Learning, Artificial Intelligence, Test Module.

1 | Problema de Investigación

Actualmente, la detección del desalineamiento de ejes en máquinas rotativas se realiza principalmente mediante monitoreo de vibraciones, siendo esta una de las técnicas más utilizadas en la industria. También existen otros métodos, como el uso de cámaras termográficas, pero el análisis de vibraciones sigue siendo el más empleado porque permite identificar variaciones en el comportamiento dinámico del sistema.

Generalmente, el diagnóstico del nivel de desalineamiento se obtiene a partir del análisis de señales realizado por personal con experiencia en procesamiento de datos vibracionales. Para ello, se observan características y patrones presentes en la señal, los cuales permiten relacionar el comportamiento medido con posibles fallas mecánicas. Sin embargo, este procedimiento depende en gran parte de la experiencia del analista, por lo que puede presentar errores, especialmente cuando la falla recién empieza a desarrollarse o cuando las diferencias entre un nivel de desalineamiento y otro no son tan evidentes. Además, este tipo de evaluación requiere tiempo y personal especializado, lo que incrementa el costo del diagnóstico.

Frente a esta situación, en el presente trabajo se propone desarrollar una metodología que permita identificar de manera automática el nivel de desalineamiento angular en ejes. Para ello, la investigación se organiza en cuatro etapas principales, las cuales comprenden el diseño y construcción de un módulo de pruebas, la captura de datos experimentales, el procesamiento de la información obtenida y el entrenamiento

y validación de un algoritmo de aprendizaje automático.

1.1. Problema General

¿Cómo identificar de manera automática el nivel de desalineamiento angular en ejes a partir del diseño de un módulo de pruebas, la captura y procesamiento de datos experimentales, y el entrenamiento y validación de un algoritmo de aprendizaje automático?

1.2. Problemas Específicos

- ¿Cuáles son los parámetros técnicos necesarios para el diseño, dimensionamiento y selección de los componentes del módulo de pruebas?
- ¿Cómo se realizará la captura y organización de los datos experimentales correspondientes a los distintos niveles de desalineamiento angular?
- ¿Qué técnicas de procesamiento de datos permitirán extraer características útiles para diferenciar los distintos estados de desalineamiento?
- ¿Cómo entrenar y validar un algoritmo de aprendizaje automático que permita identificar automáticamente los niveles de desalineamiento angular, incluyendo su evaluación con datos de referencia obtenidos de una fuente de datos externa?

1.3. Justificación

El desalineamiento de ejes es una de las fallas más comunes en máquinas rotativas y puede generar efectos negativos en el funcionamiento del sistema, como incremento de vibraciones, desgaste prematuro de componentes, reducción de la vida útil de los elementos mecánicos y mayores costos de mantenimiento. Por ello, la detección oportuna de esta falla es importante para evitar daños mayores y mejorar la confiabilidad de los equipos.

En la práctica, el diagnóstico del desalineamiento suele depender del análisis de señales realizado por personal con experiencia, lo cual puede hacer que el proceso tome tiempo y esté sujeto a errores de interpretación, especialmente cuando las variaciones entre los niveles de desalineamiento no son tan evidentes. Frente a ello, surge la necesidad de desarrollar metodologías que permitan apoyar o automatizar este proceso, haciendo uso de herramientas de procesamiento de datos y aprendizaje automático.

La presente investigación se justifica porque plantea el desarrollo de una metodología completa que abarca desde el diseño de un módulo de pruebas hasta la validación de un modelo de identificación automática de desalineamiento angular. En primer lugar, el diseño y construcción del módulo permitirá contar con una plataforma experimental propia para reproducir condiciones controladas de desalineamiento y generar datos representativos del fenómeno estudiado. En segundo lugar, la captura y organización de los datos permitirá construir una base de datos útil para el análisis y entrenamiento del modelo. Posteriormente, el procesamiento de la información permitirá extraer características relevantes de las señales medidas, y finalmente, el entrenamiento y validación del algoritmo permitirán evaluar la posibilidad de automatizar la identificación del nivel de desalineamiento.

Además, la investigación tiene relevancia porque no se limita únicamente al análisis de los datos obtenidos en el módulo desarrollado, sino que también incorpora una etapa de validación empleando un conjunto de datos de referencia proveniente de una configuración experimental más cercana a condiciones reales de operación. Esto permite analizar la capacidad de generalización del modelo y fortalecer la propuesta planteada en la tesis.

Por lo tanto, este trabajo se justifica en el aporte que brinda al diagnóstico de fallas en maquinaria rotativa, proponiendo una metodología experimental y computacional que puede servir como base para futuras aplicaciones en monitoreo de condición y mantenimiento predictivo.

2 | Objetivos

2.1. Objetivo General

Implementar algoritmos de inteligencia artificial para identificar automáticamente el nivel de desalineamiento entre ejes mediante señales obtenidas en un módulo de pruebas.

2.2. Objetivos Específicos

- Diseñar y construir un módulo de pruebas para reproducir distintos estados de desalineamiento en ejes, permitiendo la adquisición controlada de señales de vibración.
- Capturar y organizar señales de vibración en diferentes condiciones de desalineamiento, para generar una base de datos representativa del comportamiento dinámico del sistema.
- Procesar la base de datos de señales vibratorias mediante técnicas de extracción y selección de características estadísticas para la identificación del desalineamiento.
- Implementar un algoritmo de aprendizaje automático que procese la base de datos

obtenida para automatizar la identificación de niveles de desalineamiento.

3 | Hipótesis

La aplicación de algoritmos de aprendizaje automático a las características estadísticas extraídas de las señales adquiridas mediante un sensor inductivo en un módulo experimental permite estimar con precisión el grado de desalineamiento angular en ejes rotativos.

4 | Variables

4.1. Variables de la investigación

En la presente investigación, las variables se definen en función del proceso experimental y computacional desarrollado para la identificación automática del nivel de desalineamiento angular en ejes mediante el análisis de señales adquiridas con un sensor CKP y el uso de algoritmos de inteligencia artificial.

4.1.1. Variable independiente

La variable independiente corresponde al nivel de desalineamiento angular del eje. Esta variable representa la condición mecánica impuesta en el módulo de pruebas y constituye el fenómeno principal que se desea identificar.

Cuadro 4.1: Operacionalización de la variable independiente

Variable	Definición operacional	Indicador	Escala
Nivel de desalineamiento angular del eje	Se establece mecánicamente en el módulo experimental mediante la variación controlada del ángulo entre los ejes acoplados. En esta investigación se consideran niveles de desalineamiento desde 0° hasta 15° , con incrementos de 1° .	Ángulo de desalineamiento configurado en el módulo de pruebas.	Grados sexagesimales ($^\circ$)

4.1.2. Variable dependiente

La variable dependiente corresponde a la identificación automática del nivel de desalineamiento angular. Esta variable representa la respuesta obtenida por los modelos de inteligencia artificial a partir de las características extraídas de las señales adquiridas experimentalmente.

Cuadro 4.2: Operacionalización de la variable dependiente

Variable	Definición operacional	Indicador	Escala
Identificación automática del nivel de desalineamiento angular	Se obtiene mediante el entrenamiento y evaluación de modelos de inteligencia artificial, tales como Random Forest, XGBoost y redes neuronales, utilizando características extraídas de las señales CKP adquiridas en el módulo experimental.	Ángulo estimado por el modelo, error de predicción, coeficiente de determinación y desempeño del algoritmo.	Grados ($^{\circ}$) y porcentaje

4.1.3. Variables de control

Las variables de control corresponden a aquellas condiciones experimentales que deben mantenerse constantes o dentro de un rango definido, con la finalidad de reducir su influencia sobre la señal adquirida y asegurar que las variaciones observadas estén asociadas principalmente al nivel de desalineamiento angular.

Cuadro 4.3: Variables de control consideradas en la investigación

Variable de control	Descripción	Forma de control
Velocidad de rotación	Influye directamente en la frecuencia fundamental de la señal y en la aparición de componentes armónicas.	Se mantiene una velocidad nominal de operación aproximadamente constante durante los ensayos.
Frecuencia de muestreo	Define la cantidad de muestras adquiridas por segundo y condiciona la calidad del registro de la señal.	Se emplea una frecuencia de muestreo fija de 44,1 kHz.
Tipo de sensor	El sensor utilizado determina la naturaleza y forma de la señal adquirida.	Se emplea el mismo sensor CKP durante todos los ensayos experimentales.
Configuración de la rueda fónica	El número de pines metálicos influye en la resolución temporal y en la forma de onda de la señal capturada.	Se utiliza una configuración definida de rueda fónica durante la adquisición de datos.
Tiempo de adquisición	Determina la cantidad de información disponible para el procesamiento y entrenamiento de los modelos.	Se mantiene un tiempo de captura similar para cada condición de desalineamiento.
Condiciones de montaje	La rigidez del soporte, la distancia entre sensor y rueda fónica, y la estabilidad del módulo pueden afectar la señal.	Se mantiene fija la geometría del montaje, el soporte del sensor y la disposición de los elementos mecánicos.

4.1.4. Matriz de operacionalización de variables

La matriz de operacionalización permite relacionar cada variable con su dimensión, indicadores, técnica de medición e instrumento utilizado durante el desarrollo de la investigación.

Cuadro 4.4: Matriz de operacionalización de variables

Tipo de variable	Variable	Dimensión	Indicador	Instrumento
Independiente	Nivel de desalineamiento angular	Condición mecánica del sistema rotativo	Ángulo impuesto entre los ejes	Módulo experimental de desalineamiento
Dependiente	Identificación automática del desalineamiento	Predicción del nivel de desalineamiento	Ángulo estimado por el modelo	Algoritmos de inteligencia artificial
Dependiente	Error de predicción	Precisión del modelo	MAE y RMSE	Evaluación computacional en Python
Dependiente	Calidad del ajuste	Capacidad predictiva del modelo	Coefficiente de determinación R^2	Evaluación computacional
Control	Velocidad de rotación	Régimen de operación del eje	Velocidad angular del sistema	Señal CKP / medición experimental
Control	Frecuencia de muestreo	Adquisición de señal	Número de muestras por segundo	Interfaz de audio
Control	Configuración de la rueda fónica	Generación de pulsos de referencia	Número de pines metálicos	Rueda fónica impresa en 3D

5 | ALCANCES Y LIMITACIONES

5.1. Alcances

El presente trabajo de investigación comprende los siguientes alcances:

- Se desarrollará un módulo de pruebas experimental capaz de reproducir distintos niveles de desalineamiento angular en ejes, con el propósito de obtener señales representativas para el estudio.
- Se adquirirán señales experimentales bajo diferentes condiciones de desalineamiento angular, estableciendo una base de datos propia para el análisis y evaluación del fenómeno.
- El procesamiento de las señales adquiridas estará orientado a la extracción de características relevantes en el dominio de la vibración, con el fin de identificar patrones asociados a cada nivel de desalineamiento.
- Se diseñará e implementará un modelo de aprendizaje automático entrenado con la base de datos experimental generada, para automatizar la identificación de los niveles de desalineamiento evaluados.
- La investigación se enfocará en la detección de desalineamiento angular a partir de datos experimentales obtenidos en un entorno controlado, sin abarcar otras fallas mecánicas como desbalanceo, holguras o fallas en rodamientos.

- La validación del modelo se realizará con los datos obtenidos en el módulo de pruebas y con datos experimentales de referencia, con el fin de evaluar el desempeño del algoritmo propuesto.
- El trabajo incluirá el análisis del comportamiento dinámico del módulo de pruebas como soporte para la interpretación de los resultados experimentales y computacionales obtenidos.

5.2. Limitaciones

Las limitaciones del presente estudio se resumen en los siguientes puntos:

- La investigación estará enfocada únicamente en el análisis del desalineamiento angular, por lo que no se considerará el estudio de desalineamiento paralelo.
- La adquisición de datos se realizará únicamente mediante el sensor implementado en el módulo de pruebas, por lo que no se considerará el uso de otros sensores o variables adicionales, como temperatura u otras magnitudes físicas complementarias.
- El comportamiento dinámico del módulo de pruebas no será evaluado mediante técnicas experimentales avanzadas como OMA, EMA u ODS, limitándose el estudio a los ensayos y análisis planteados dentro de la metodología desarrollada.
- La propuesta se centrará en la identificación automática del nivel de desalineamiento a partir del procesamiento de datos y el entrenamiento del modelo, por lo que no se desarrollará una interfaz gráfica de usuario para su visualización o uso final.
- La metodología propuesta no estará orientada a la detección en tiempo real, sino al análisis y clasificación de datos previamente adquiridos y procesados.

6 | Metodología

El presente trabajo de investigación se desarrolla con base en una metodología experimental, la cual inicia con el diseño y construcción de un módulo de pruebas destinado a reproducir condiciones controladas de desalineamiento angular en ejes. En dicho módulo se inducirán distintos niveles de desalineamiento, permitiendo la adquisición de señales experimentales mediante un sensor de posición de cigüeñal.

Posteriormente, las señales obtenidas serán sometidas a una etapa de procesamiento orientada a la extracción de características estadísticas relevantes, con el fin de conformar una base de datos representativa de los diferentes casos de estudio. Esta base de datos será empleada para el entrenamiento de algoritmos de aprendizaje automático enfocados en la identificación de patrones asociados a los niveles de desalineamiento evaluados.

Para el procesamiento de datos y la implementación de los modelos se utilizó el lenguaje de programación Python, debido a su facilidad de uso y a la disponibilidad de herramientas orientadas al análisis de señales, manejo de datos y aprendizaje automático. Asimismo, se emplearon librerías especializadas que permitieron organizar la información, extraer características, entrenar los modelos y evaluar su desempeño de manera ordenada.

Concluida la etapa de entrenamiento, se analizará el desempeño del modelo desarrollado en función de su precisión para identificar el nivel de desalineamiento presente en los ejes. Finalmente, los resultados obtenidos serán comparados con un caso real o con datos experimentales de referencia, con el propósito de evaluar la capacidad de generalización del modelo y validar la propuesta metodológica en un contexto de aplicación más cercano a condiciones reales de operación.

6.1. Tipo de Investigación

La presente investigación corresponde a un estudio de tipo experimental, debido a que contempla el diseño y construcción de un módulo de pruebas para la obtención de datos bajo condiciones controladas de desalineamiento angular en ejes. Mediante este enfoque, se busca observar, medir y analizar el comportamiento del sistema a partir de señales adquiridas experimentalmente, lo cual permite establecer una relación entre las condiciones de desalineamiento inducidas y las características presentes en los datos obtenidos.

Los datos experimentales recolectados constituyen la base para el procesamiento de señales, la extracción de características y el entrenamiento de modelos de aprendizaje automático orientados a la identificación de niveles de desalineamiento. Finalmente, la naturaleza experimental de la investigación también permite contrastar los resultados obtenidos con datos de referencia o con un caso real, con el propósito de evaluar la validez de la metodología propuesta y su potencial de aplicación en escenarios distintos al entorno controlado del módulo de pruebas.

6.2. Métodos de Validación Propuestas

La validación de la metodología propuesta se llevará a cabo de manera progresiva en las diferentes etapas de la investigación. En una primera etapa, se evaluará el correcto funcionamiento del módulo de pruebas y la calidad de las señales adquiridas, verificando que los datos experimentales obtenidos representen de manera consistente los distintos niveles de desalineamiento angular inducidos.

En una segunda etapa, se validará el desempeño del modelo de aprendizaje automático a partir de métricas que permitan cuantificar su capacidad de identificación. Entre los criterios de evaluación se considerarán la precisión alcanzada por el modelo, la evolución de la precisión durante el entrenamiento y el comportamiento de la función de pérdida en relación con el número de épocas, con el fin de analizar su proceso de aprendizaje y su estabilidad.

Finalmente, se efectuará una validación externa mediante la comparación de los resultados obtenidos con datos experimentales de referencia o con un caso real, lo que permitirá evaluar la capacidad de generalización del modelo y la pertinencia de la metodología desarrollada para su aplicación en escenarios diferentes al entorno controlado del módulo de pruebas.

7 | Estado del Arte

7.1. Desalineamiento de ejes

El desalineamiento de ejes en máquinas es un problema mecánico frecuente que afecta a equipos rotativos y sistemas acoplados, como motores, bombas y compresores. Se produce cuando los ejes de dos componentes que deberían estar perfectamente alineados no coinciden en su posición o en su orientación geométrica. Este problema puede parecer menor, pero tiene un impacto significativo en el rendimiento, la vida útil y los costos operativos del sistema, generando vibraciones, desgaste acelerado y fallos prematuros si no se corrige a tiempo.

Entre unas de las causas que producen el desalineamiento de ejes se encuentran:

- Fallas en el montaje.
- Alteraciones en las posiciones relativas de los ejes posteriores al montaje.
- Distorción debido a fuerzas externas.
- Expansiones y contracciones en el sistema debido a cambios de temperatura.
- Falta de rigidez en la cimentación de la máquina.

7.2. Tipos de desalineamiento presentes en ejes

El desalineamiento en ejes acoplados es un fenómeno común en sistemas mecánicos donde dos o más ejes deben trabajar en conjunto para transmitir potencia o movimiento. Este desalineamiento puede surgir, como ya se mencionó, debido a imperfecciones en la instalación, desgaste de los componentes o deformaciones provocadas por fuerzas externas o variaciones térmicas. Existen principalmente dos tipos de desalineamiento, el desalineamiento de tipo angular, en el que los ejes no están en el mismo plano pero convergen en un ángulo, y el desalineamiento en paralelo, donde los ejes son paralelos pero se encuentran desplazados de forma lateral. Estas condiciones generan vibraciones, sobrecargas y un desgaste acelerado de los componentes, comprometiendo la eficiencia y la vida útil del sistema.

7.2.1. Desalineamiento paralelo

El desalineamiento paralelo en ejes ocurre cuando los ejes de dos componentes mecánicos conectados mediante acoples, como los ejes de un motor y un dispositivo impulsado (por ejemplo, una bomba o un compresor), no están correctamente alineados en el espacio. Aunque los ejes son paralelos entre sí, sus centros no coinciden. Este tipo de desalineamiento se distingue de otros como el angular o el mixto. La condición de desalineamiento paralelo genera fuerzas de cizallamiento. En la Figura 7.1 se muestra este fenómeno.

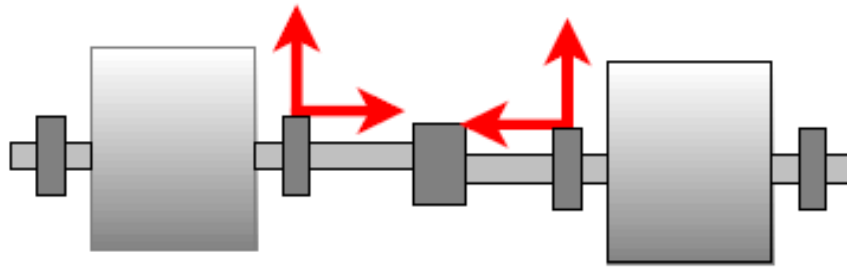


Figura 7.1: Fuerzas presentes debido al desalineamiento paralelo. Fuente: White (2010).

La desalineación paralela produce una fuerza de cizallamiento y un momento de flexión en la extremidad acoplada de cada flecha, niveles de vibración altos en $2x$ y en $1x$. Se producen en las direcciones radiales o tangenciales en los rodamientos en cada lado del acoplamiento, y son de fase opuesta. En la mayoría de los casos, los componentes $2x$ estarán más altos que los $1x$. Los niveles axiales $1x$ y $2x$ estarán bajos solamente en desalineación paralela. y su fase estará opuesta.

7.2.2. Desalineamiento angular

El desalineamiento angular en ejes ocurre cuando los ejes de dos componentes mecánicos acoplados, como el eje de un motor y el eje de un equipo impulsado, no mantienen la misma orientación angular en el espacio. En esta condición, los centros de los ejes pueden coincidir en un punto, pero los ejes forman un ángulo entre sí, ya sea en el plano vertical, horizontal o en una combinación de ambos. Este tipo de desalineamiento se diferencia del desalineamiento paralelo y del desalineamiento mixto por la presencia de una discrepancia angular entre los ejes.

La condición de desalineamiento angular genera momentos flectores y cargas alternantes en los acoplamientos y rodamientos, lo que incrementa los niveles de vibración y acelera el desgaste de los componentes mecánicos. En la Figura 7.2 se ilustra este fenómeno, mostrando la diferencia angular existente entre los ejes acoplados.

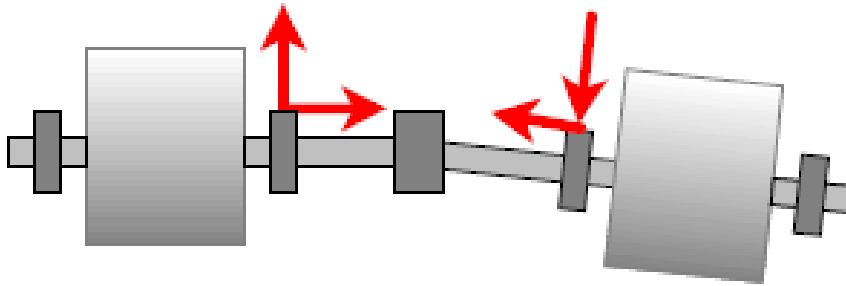


Figura 7.2: Fuerzas presentes debido al desalineamiento angular White (2010).

7.3. Detección del desalineamiento en ejes

La detección del desalineamiento puede realizarse, de forma general, mediante dos enfoques básicos. El primero corresponde al análisis de señales durante la operación del equipo, principalmente mediante acelerómetros u otros sensores capaces de medir la respuesta dinámica del sistema. El segundo corresponde a la medición geométrica mediante equipamiento láser, práctica común en la industria para verificar la posición relativa entre ejes acoplados, normalmente con el equipo detenido.

En el enfoque basado en vibraciones, el desalineamiento se identifica a partir de las variaciones que aparecen en la señal medida por el acelerómetro. Estas señales pueden analizarse en el dominio del tiempo y, principalmente, en el dominio de la frecuencia mediante la Transformada Rápida de Fourier (FFT). En el análisis espectral, el desalineamiento suele reflejarse mediante incrementos en la frecuencia de giro del eje y en sus armónicos, especialmente en componentes asociadas a 1X y 2X de la velocidad de rotación, como indican Guan et al. (2017); Palomino et al. (2023).

El procedimiento general para la detección de fallas mediante análisis vibracional consiste en medir la señal, acondicionarla, obtener su representación espectral e identificar componentes frecuenciales asociadas al estado mecánico del sistema, como indica Navarro-Navarro et al. (2025). Este procedimiento se muestra en la Figura 7.3.

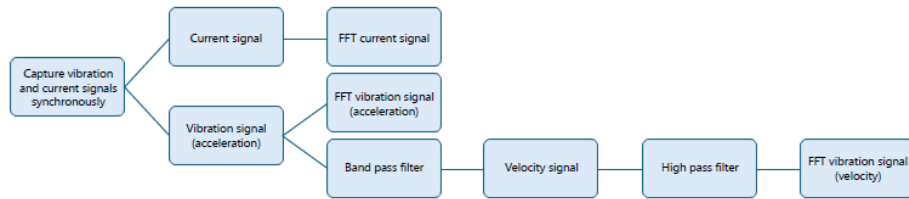


Figura 7.3: Procedimiento estandarizado para la detección de fallas mediante análisis vibracional, adaptado de Navarro-Navarro et al. (2025).

Para el caso del presente trabajo, el análisis espectral permite comparar la respuesta del sistema en condición alineada y bajo distintos niveles de desalineamiento. De esta manera, es posible observar cambios en la amplitud de las componentes frecuenciales asociadas a la rotación del eje y a sus armónicos. Un ejemplo de este tipo de procesamiento se presenta en la Figura 7.4, donde se muestran resultados FFT obtenidos para diferentes condiciones de desalineamiento.

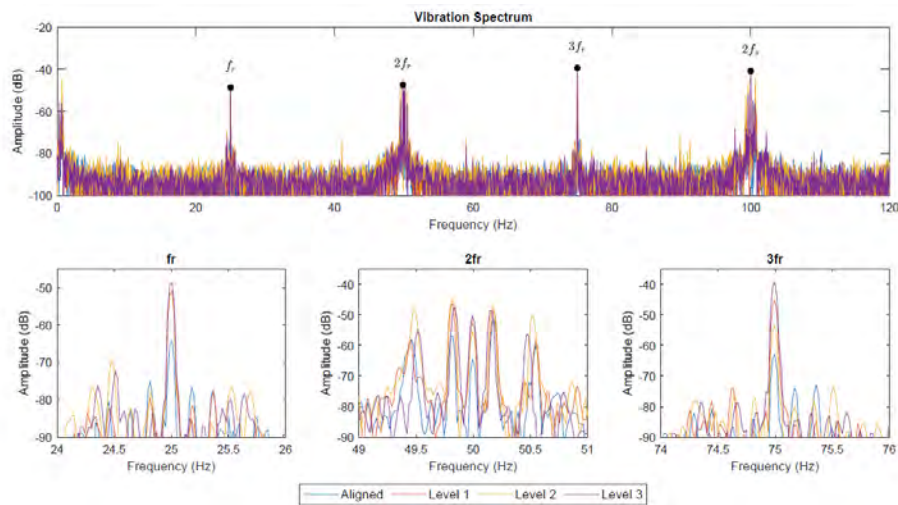


Figura 7.4: Análisis FFT de señales de vibración en un módulo de pruebas bajo régimen estacionario para diferentes niveles de desalineamiento de ejes, adaptado de Navarro-Navarro et al. (2025).

Por otro lado, la detección mediante láser se basa en medir directamente la posición relativa entre los ejes. Este método es usado de forma común en labores de montaje, mantenimiento y corrección de alineamiento, ya que permite verificar desviaciones geométricas sin depender únicamente de la respuesta vibracional del sistema.

A diferencia del análisis espectral, la medición láser permite evaluar directamente el error de posición entre los ejes, incluyendo condiciones de desalineamiento paralelo y angular, como indica Simm et al. (2016).

La Figura 7.5 muestra un esquema de medición láser aplicado al monitoreo del desalineamiento en ejes, como indica Simm et al. (2016). Este tipo de configuración representa el principio general usado en equipos de alineamiento láser, donde se compara la posición relativa de los ejes mediante un emisor, un receptor y un sistema de referencia geométrica.

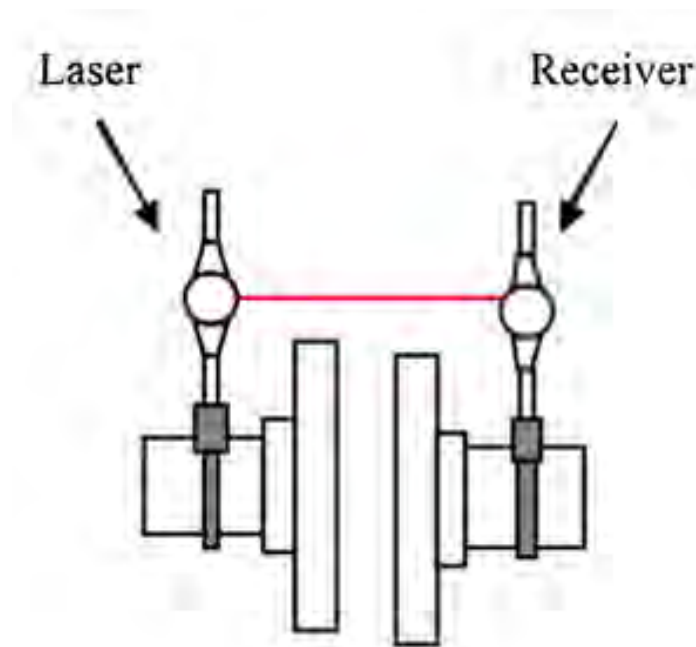


Figura 7.5: Sistema experimental para la medición láser del desalineamiento en ejes, adaptado de Simm et al. (2016).

Como referencia del equipo usado para la alineación láser, el manual del sistema Easy-Laser Extreme muestra que este equipo cuenta con una unidad de visualización, dos unidades de medición identificadas como S y M, y soportes con cadena para fijar los sensores sobre los ejes. Este sistema permite medir la posición relativa entre dos ejes y verificar si existe desalineamiento, como indica Damalini AB (2014).

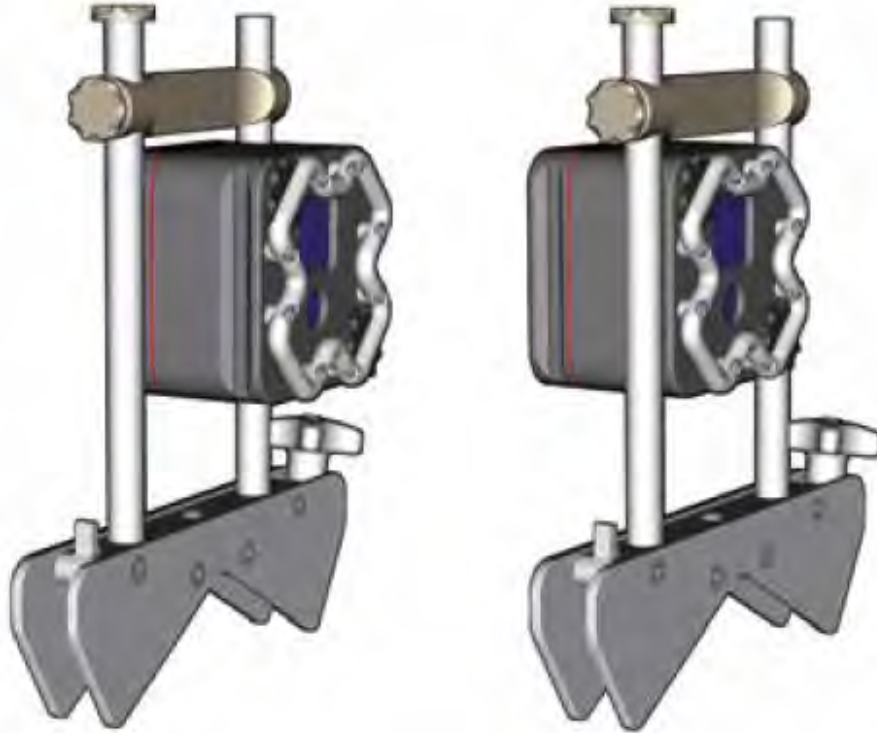


Figura 7.6: Sistema de alineación láser Easy-Laser Extreme utilizado como referencia de equipamiento industrial para la medición y corrección del desalineamiento de ejes, adaptado de Damalini AB (2014).

7.4. Análisis espectral y de forma de onda

Un sistema motriz el cual presenta indicios de desalineamiento angular entre el eje motriz y el eje conducido, no podrá ser diagnosticado mediante una inspección visual o métodos de inspección no intrusivos, deberá analizarse mediante un análisis vibracional como método de diagnóstico. Este procedimiento se realizará mediante acelerómetros pegados a las chumaceras que albergan a dicho sistema y se analizará la

data capturada en equipos analizadores de espectros. De la data capturada se observará que se producirán amplitudes de vibración a frecuencias que no serán múltiplos de la velocidad de giro y serán componentes no armónicos de la frecuencia principal de giro. La presencia de espectros de vibración no armónicos será la primera señal de advertencia para detectar que se puede estar desarrollando una falla por desalineamiento; esto se da principalmente debido al desfase provocado por los defectos en los elementos internos que componen el sistema. Por esto, las fuentes de vibración armónica deberán ser tomadas como parámetro de calibración para alimentar al algoritmo de inteligencia artificial.

7.4.1. Señales continuas y discretas

Una señal continua en el tiempo es aquella que se encuentra definida para todo instante de tiempo real, es decir, su variable independiente puede tomar cualquier valor dentro de un intervalo continuo. Este tipo de señales es comúnmente generado por fenómenos físicos naturales, tales como vibraciones mecánicas, señales acústicas, variaciones de temperatura o señales eléctricas analógicas. Matemáticamente, una señal continua se representa como una función $x(t)$, donde $t \in \mathbb{R}$ Oppenheim et al. (1997).

Las señales continuas presentan una variación suave y no cuantizada tanto en el dominio del tiempo como en el dominio de amplitud. Su análisis puede realizarse mediante herramientas matemáticas como la transformada de Fourier continua, que permite describir el contenido espectral de la señal. En sistemas físicos reales, este tipo de señales constituye la representación más fiel del comportamiento del fenómeno observado y su procesamiento directo requiere sistemas analógicos o técnicas de digitalización para su tratamiento computacional.

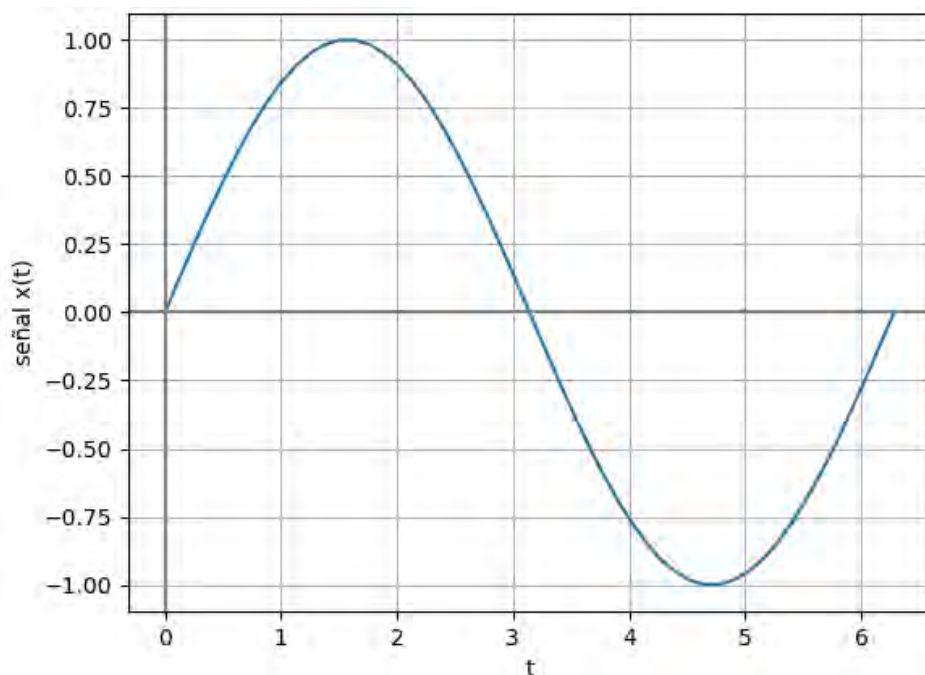


Figura 7.7: Señal continua en el tiempo ESPOLBlogSenalesContinuas2017.

Una señal continua en el tiempo es aquella que se encuentra definida para todo instante de tiempo real, es decir, su variable independiente puede tomar cualquier valor dentro de un intervalo continuo. Este tipo de señales es comúnmente generado por fenómenos físicos naturales, tales como vibraciones mecánicas, señales acústicas, variaciones de temperatura o señales eléctricas analógicas. Matemáticamente, una señal continua se representa como una función $x(t)$, donde $t \in \mathbb{R}$ Oppenheim et al. (1997).

La principal diferencia entre las señales continuas y discretas radica en el dominio temporal en el cual están definidas. Mientras que las señales continuas existen para todo instante de tiempo, las señales discretas solo están definidas en puntos específicos. Esta distinción tiene implicancias directas en las técnicas de análisis y procesamiento empleadas, ya que las señales discretas pueden ser manipuladas mediante algoritmos digitales, mientras que las señales continuas requieren herramientas analógicas o transformaciones previas para su tratamiento computacional Mitra (2011).

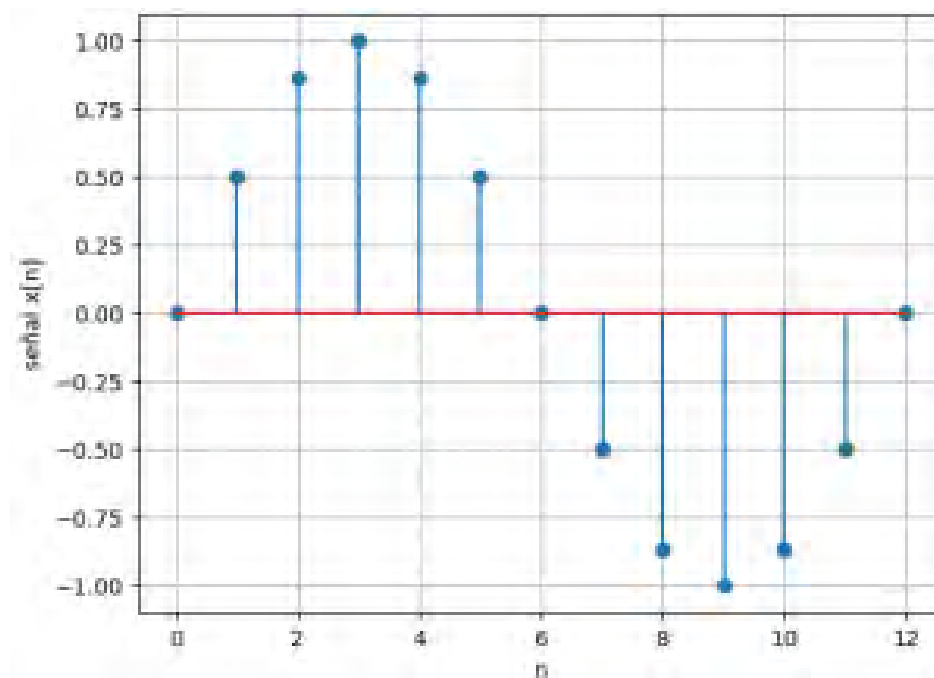


Figura 7.8: Señal discreta en el tiempo ESPOLBlogSenalesContinuas2017.

7.4.2. Señal determinística

Una señal determinística es una señal cuyo comportamiento y características pueden describirse completamente mediante una función matemática o un conjunto de parámetros bien definidos. En otras palabras, su forma, amplitud, frecuencia y fase están completamente determinadas y son predecibles en cualquier punto del tiempo. No hay aleatoriedad en la evolución de la señal, y su patrón puede ser modelado con precisión.

Las señales determinísticas son opuestas a las señales estocásticas, que exhiben algún grado de aleatoriedad o imprevisibilidad en su comportamiento. Ejemplos de señales determinísticas incluyen funciones matemáticas como senos, cosenos, ondas cuadradas, impulsos, entre otras. Estas señales son fundamentales en el análisis y diseño de sistemas de comunicación, procesamiento de señales, control y otras áreas de la ingeniería.

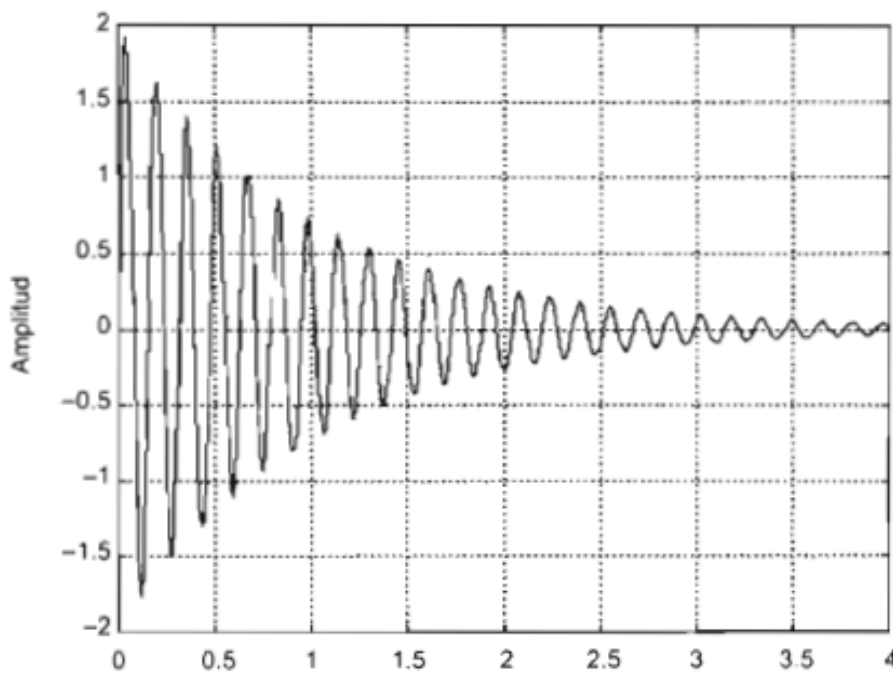


Figura 7.9: Señal determinística Aviles (2003).

7.4.3. Señal aleatoria

Una señal aleatoria o estocástica es una señal cuyo comportamiento no puede describirse completamente mediante una función matemática determinística, ya que presenta un grado inherente de aleatoriedad o variabilidad. A diferencia de las señales determinísticas, las señales aleatorias no siguen un patrón exacto y predecible, por lo que su evolución temporal está sujeta a incertidumbre Papoulis and Pillai (2002); Proakis and Manolakis (2007).

Las señales aleatorias se emplean comúnmente para modelar fenómenos reales en los que intervienen factores impredecibles o fluctuaciones de origen aleatorio. Entre los ejemplos más representativos se encuentran el ruido en sistemas de telecomunicaciones, las variaciones en condiciones atmosféricas y la respuesta de ciertos sistemas dinámicos sometidos a excitaciones no controladas. Debido a esta naturaleza, las señales aleatorias constituyen una representación más cercana al comportamiento observado en aplicaciones reales de ingeniería Bendat and Piersol (2010).

Desde el punto de vista matemático, las señales aleatorias se describen mediante herramientas estadísticas, tales como funciones de distribución de probabilidad y funciones de densidad espectral de potencia (PSD, del inglés *Power Spectral Density*). Estas funciones permiten caracterizar la probabilidad de ocurrencia de los valores de la señal y la distribución de su contenido energético en el dominio de la frecuencia, proporcionando una base formal para su análisis y modelado Oppenheim et al. (1997).

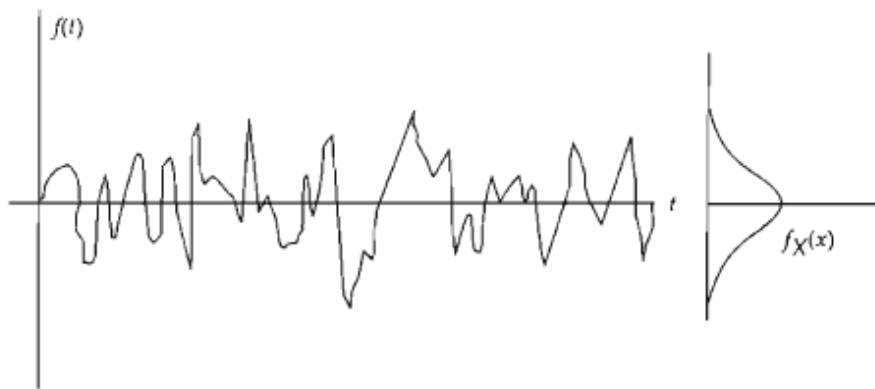


Figura 7.10: Señal aleatoria. Aviles (2003).

7.4.4. Teoría del muestreo

La teoría del muestreo constituye un fundamento esencial en el proceso de adquisición de señales, ya que permite obtener una representación discreta a partir de una señal continua en el tiempo $x(t)$ generada por el sistema motriz del módulo experimental. Esta representación discreta posibilita el almacenamiento, procesamiento y análisis digital de grandes volúmenes de datos, lo cual resulta especialmente ventajoso en aplicaciones de ingeniería y análisis de señales Shannon (1949); Unser (2000).

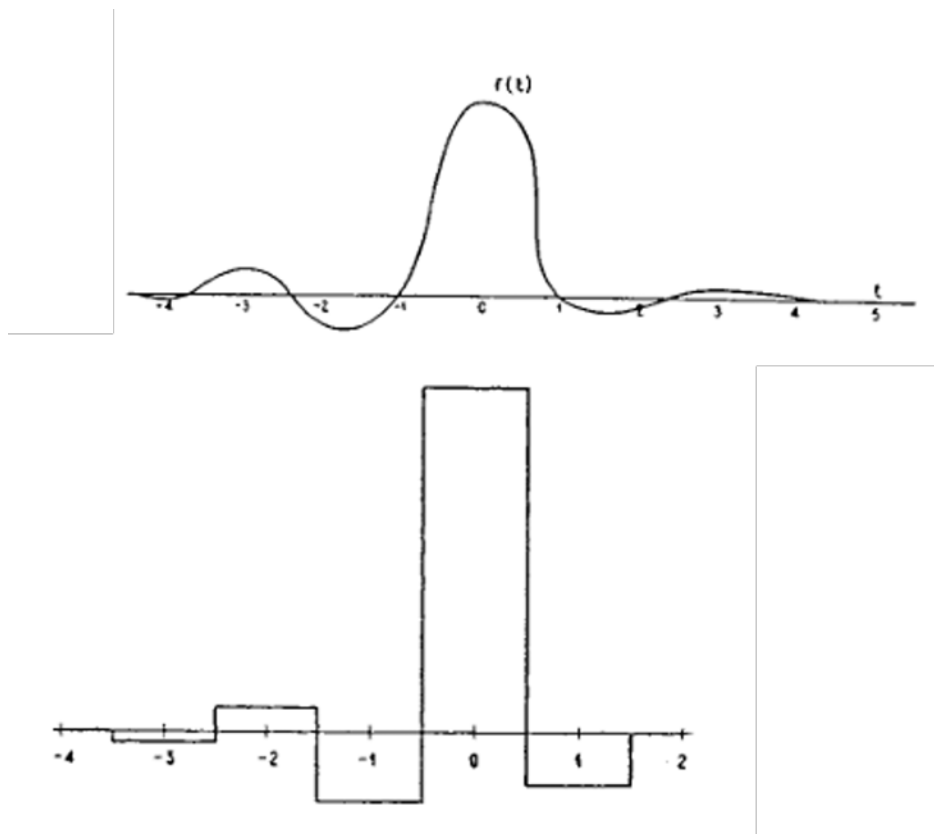


Figura 7.11: Señal continua en el tiempo reconstruida de manera digital mediante un muestreo con ventanas de tiempo uniforme Nyquist (1928).

No obstante, la discretización de una señal continua introduce restricciones inherentes al proceso de muestreo. En particular, para evitar distorsiones en la señal digitalizada, es necesario que la señal original sea de banda limitada y que la frecuencia de muestreo sea suficientemente alta en relación con su contenido espectral. De lo contra-

rio, pueden producirse fenómenos como el aliasing, los cuales comprometen la fidelidad de la señal reconstruida Nyquist (1928); Marvasti (2001). Estas consideraciones justifican el estudio detallado de la teoría del muestreo dentro del presente trabajo de investigación.

7.4.5. Muestreo ideal

El proceso de muestreo de manera conceptual consiste en multiplicar, en el dominio del tiempo, una señal continua $x(t)$ por una función de tren de impulsos, y así obtener valores instantáneos a la cadencia determinada por el tren de impulsos, si lo vemos como una estructura de control el proceso se sintetiza de la siguiente manera:

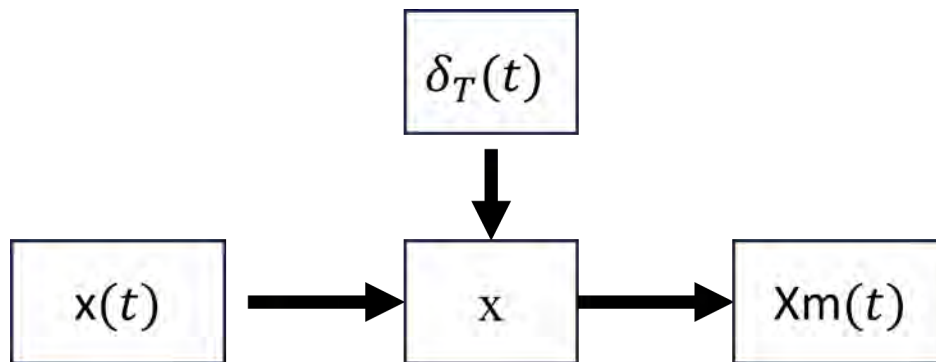


Figura 7.12: Diagrama de bloques de la teoría del muestreo, en el que se representa el proceso de muestreo de una señal continua para su conversión al dominio discreto.

Donde:

- $x(t)$ es la señal en el tiempo.
- es el tren de impulsos unitarios con un periodo T .
- $X_m(t)$ es la señal del tiempo muestreada en periodos de tiempo T .

7.5. Sensores de posición de cigüeñal CKP (Cranckshaft Position Sensor).

Los sensores de posición del cigüeñal (Cranckshaft Position Sensor, CKP), magnéticos, son dispositivos electromagnéticos pasivos que operan bajo el principio de la inducción electromagnética. Están compuestos por una bobina enrollada sobre un núcleo de material ferromagnético permanente. Frente a este sensor se encuentra una rueda dentada o reluctor acoplada al cigüeñal.

El principio de funcionamiento de este tipo de sensores se basa en la Ley de Faraday de la Inducción Electromagnética, la cual establece que un voltaje inducido aparece en una espira cuando el flujo magnético a través de ella cambia con el tiempo.

Ley de Faraday:

$$\mathcal{E} = -N \frac{d\Phi_B}{dt} \quad (7.1)$$

Donde:

- \mathcal{E} es la fuerza electromotriz (fem) inducida [V],
- N es el número de espiras,
- Φ_B es el flujo magnético [Wb],
- $\frac{d\Phi_B}{dt}$ es la tasa de cambio del flujo magnético.

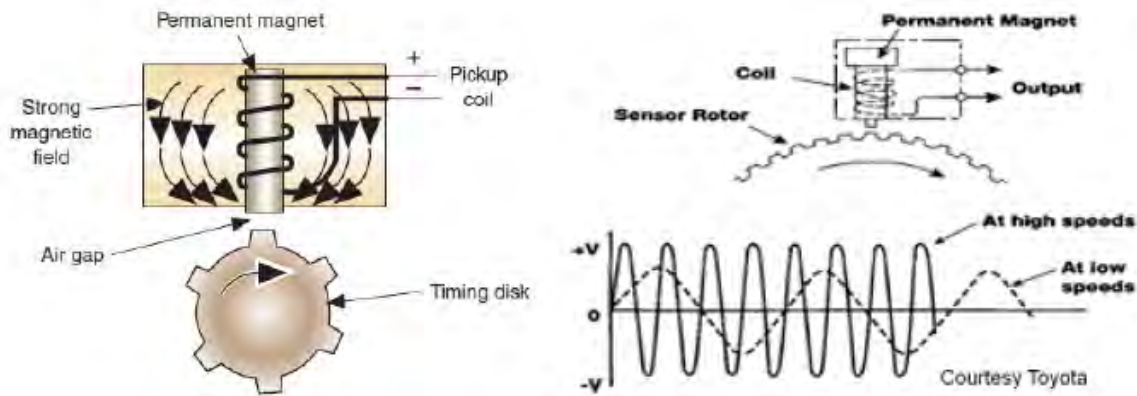


Figura 7.13: Principio de funcionamiento de sensor CKP Castellanos Martínez, José Francisco (2015).

7.6. Aprendizaje de máquina (Machine Learning)

El aprendizaje de máquinas (del inglés, *machine learning*) es una rama de la inteligencia artificial cuyo objetivo es desarrollar métodos que permitan a los sistemas computacionales aprender patrones a partir de datos y tomar decisiones sin una programación explícita para cada tarea Mitchell (1997). Este aprendizaje se logra mediante una etapa de entrenamiento, en la cual el modelo es expuesto a conjuntos de datos representativos con el fin de ajustar sus parámetros internos. En este contexto, el aprendizaje de máquinas puede describirse como un proceso en el cual un computador observa datos, construye un modelo a partir de ellos y utiliza dicho modelo para generar predicciones o resolver problemas específicos Goodfellow et al. (2016).

En los últimos años, una subdisciplina del aprendizaje de máquinas conocida como aprendizaje profundo (del inglés, *deep learning*) ha sido aplicada con éxito en diversos sectores industriales. Este avance ha sido posible gracias al incremento en la capacidad de cómputo y a la disponibilidad de grandes volúmenes de datos experimentales, tales como espectros de vibraciones, señales en el dominio del tiempo y registros termográficos, entre otros LeCun et al. (2015). El aprendizaje profundo se basa en re-

des neuronales artificiales con múltiples capas, las cuales permiten modelar relaciones complejas y no lineales en los datos, y constituyen la base de los modelos presentados en las siguientes secciones.

7.7. Clasificación de algoritmos de Machine Learning

Los algoritmos de aprendizaje automático se organizan según la información disponible en los datos y el tipo de tarea que se busca resolver. Pueden agruparse en aprendizaje supervisado, aprendizaje no supervisado y aprendizaje por refuerzo. Cada grupo responde a una lógica de modelado distinta y se selecciona en función de la existencia de clases, del tipo de salida esperada y de las características del problema. La Figura 7.14 muestra una clasificación general de los algoritmos de *Machine Learning*.

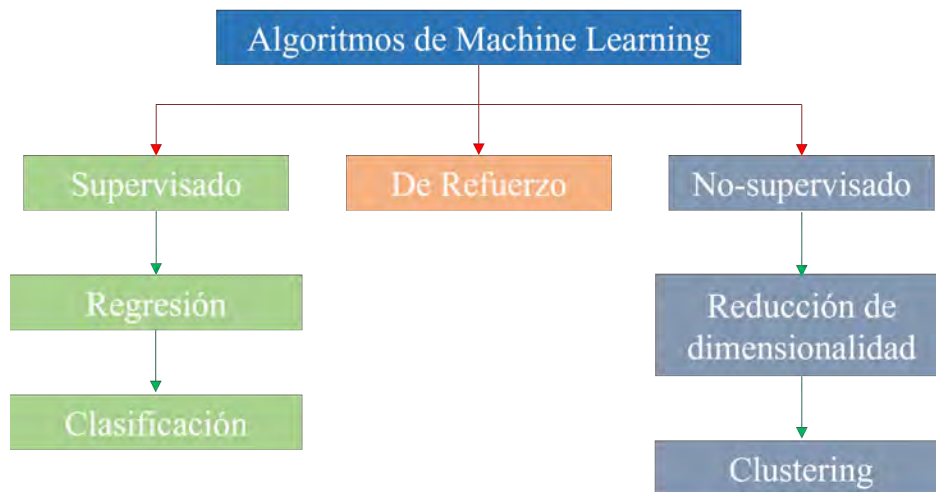


Figura 7.14: Clasificación de los algoritmos de Aprendizaje Automático (Machine Learning).

Dentro del aprendizaje supervisado se emplean conjuntos de datos etiquetados, en los que cada vector de entrada está asociado a una salida conocida. A partir de esta relación, el modelo ajusta sus parámetros para aproximar una función que permita predecir nuevas salidas a partir de nuevas entradas. Este grupo incluye problemas orientados tanto a la estimación de variables continuas como a la asignación de categorías discretas.

Entre las técnicas más conocidas de aprendizaje supervisado se encuentran:

- Regresión Lineal
- Regresión Logística
- k-Nearest Neighbors (k-NN)
- Árboles de Decisión
- Support Vector Machines (SVM)
- Naive Bayes
- Redes Neuronales (Perceptrón Multicapa / MLP)
- Bagging (p. ej. Random Forest)
- Boosting (p. ej. AdaBoost, Gradient Boosting, XGBoost, LightGBM, CatBoost)

Entre estos métodos, los árboles de decisión son una de las técnicas más usadas dentro del aprendizaje supervisado. Su funcionamiento se basa en dividir los datos mediante preguntas o condiciones sobre las variables de entrada. Cada nodo del árbol representa una condición, cada rama muestra una posible respuesta y cada hoja entrega el resultado final del modelo.

De esta manera, el árbol va separando los datos paso a paso hasta obtener una predicción. Este método puede utilizarse tanto para problemas de clasificación como

de regresión. Además, una de sus principales ventajas es que su funcionamiento es fácil de interpretar, ya que se puede representar de forma gráfica como un diagrama de decisiones Quinlan (1986); Breiman et al. (1984).

Además de su utilidad directa como modelo individual, los árboles de decisión constituyen la base de métodos más avanzados como Random Forest y XGBoost. En estos algoritmos, múltiples árboles se combinan para mejorar la capacidad predictiva, reducir la sensibilidad al ruido y aumentar la robustez del modelo frente a variaciones en los datos.

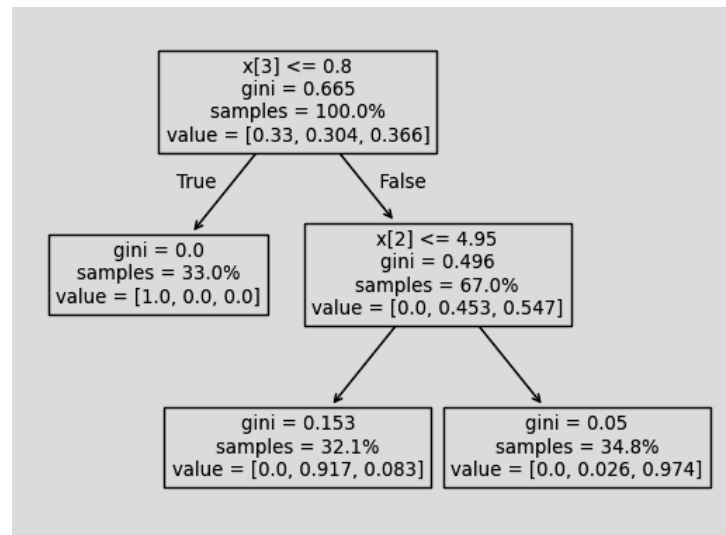


Figura 7.15: Esquema general de un árbol de decisión, donde cada nodo interno representa una regla de decisión, cada rama una condición asociada y cada nodo hoja una salida o predicción. Adaptado de scikit-learn scikit-learn developers (2026).

El aprendizaje no supervisado opera sobre datos sin etiquetas y se centra en descubrir patrones inherentes y estructuras ocultas en los conjuntos de datos. Su propósito principal es segmentar la información y extraer representaciones compactas que faciliten el análisis exploratorio, la visualización y el preprocesamiento de datos para posteriores etapas de modelado.

Entre las técnicas más conocidas de aprendizaje no supervisado se encuentran:

- *Clustering:*
 - K-means
 - DBSCAN
 - Clustering Jerárquico (aglomerativo y divisivo)
 - Mean Shift
 - Gaussian Mixture Models (GMM)

El aprendizaje por refuerzo es un enfoque en el que un agente aprende a tomar decisiones mediante prueba y error, explora acciones en un entorno, recibe señales de recompensa o castigo según los resultados, y ajusta su comportamiento para maximizar la recompensa acumulada a lo largo del tiempo. En cada paso, el agente observa el estado actual, elige una acción y recibe retroalimentación que le indica si esa acción fue adecuada o no, lo que le permite mejorar progresivamente su criterio de decisión.

7.8. Perceptrón

El perceptrón constituye uno de los primeros modelos computacionales de neuro-na artificial y representa una base importante para el desarrollo posterior de las redes neuronales artificiales. Fue propuesto por Rosenblatt como un modelo capaz de relacionar entradas, pesos sinápticos y una regla de decisión para resolver problemas de reconocimiento de patrones mediante aprendizaje supervisado Rosenblatt (1958). Desde el punto de vista computacional, el perceptrón permite transformar un conjunto de variables de entrada en una salida discreta, a partir de una combinación lineal seguida de una función de activación.

En términos generales, el perceptrón recibe un vector de entrada compuesto por las características del problema analizado. Cada entrada se multiplica por un peso asociado, el cual representa la importancia relativa de dicha variable dentro del modelo. Posteriormente, los productos ponderados se suman junto con un término de sesgo, generando una señal interna que es evaluada mediante una función de activación. Este principio permitió establecer una primera representación matemática del aprendizaje automático basado en ajuste de parámetros Rosenblatt (1958); Widrow and Hoff (1960).

Matemáticamente, el perceptrón puede representarse mediante una suma ponderada de las entradas. Si se considera un vector de entrada $x = [x_1, x_2, \dots, x_n]$, un vector de pesos $w = [w_1, w_2, \dots, w_n]$ y un sesgo b , la entrada neta del modelo se expresa como:

$$z = \sum_{i=1}^n w_i x_i + b \quad (7.2)$$

La salida del perceptrón se obtiene aplicando una función de activación sobre la entrada neta. En su forma clásica, se utiliza una función escalón, la cual asigna una clase en función de si el valor de z supera un determinado umbral:

$$y = \begin{cases} 1, & z \geq 0 \\ 0, & z < 0 \end{cases} \quad (7.3)$$

La Figura 7.16 muestra la representación esquemática del perceptrón, donde las variables de entrada son ponderadas mediante pesos sinápticos, sumadas junto con un término de sesgo y evaluadas mediante una función de activación. A partir de este proceso, el perceptrón genera una salida de clasificación, por lo que puede interpretarse como un modelo lineal capaz de separar clases cuando las características de entrada presentan una frontera de decisión linealmente separable.

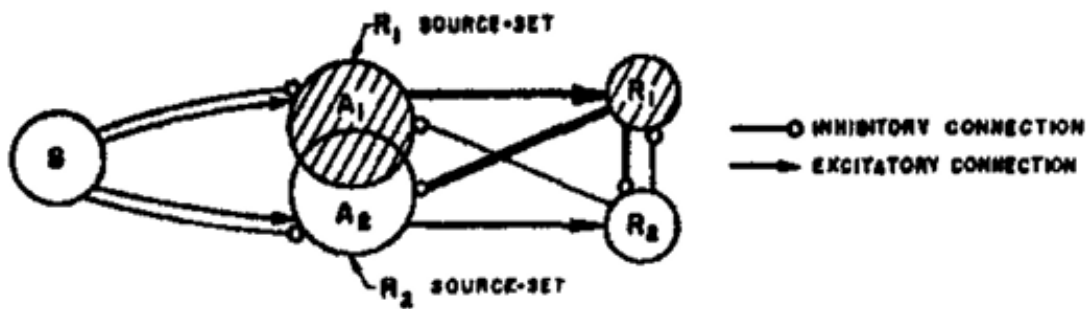


Figura 7.16: Representación esquemática del perceptrón, donde las entradas son ponderadas mediante pesos, sumadas junto con un sesgo y evaluadas mediante una función de activación para obtener la salida del modelo Rosenblatt (1958).

El proceso de aprendizaje del perceptrón consiste en ajustar los pesos del modelo en función del error cometido durante la clasificación. Para cada muestra de entrenamiento, se compara la salida estimada con la salida real, y los pesos se modifican de acuerdo con una regla de corrección. Esta idea fue fundamental para el desarrollo de algoritmos adaptativos posteriores, como los modelos ADALINE y MADALINE propuestos por Widrow y Hoff, los cuales incorporaron criterios de optimización basados en el error Widrow and Hoff (1960).

La actualización de los pesos en el perceptrón clásico puede expresarse como:

$$w_i^{nuevo} = w_i^{actual} + \eta(d - y)x_i \quad (7.4)$$

donde w_i representa el peso asociado a la entrada x_i , η es la tasa de aprendizaje, d es la salida deseada y y es la salida estimada por el modelo. Esta regla permite que el modelo corrija progresivamente sus parámetros durante el entrenamiento, reduciendo los errores de clasificación cuando las clases presentan una separación lineal adecuada.

A pesar de su importancia histórica y conceptual, el perceptrón simple presenta limitaciones cuando el problema no es linealmente separable. Una de las críticas

más conocidas fue desarrollada por Minsky y Papert, quienes demostraron que un perceptrón de una sola capa no puede resolver ciertos problemas lógicos, como el caso XOR Minsky and Papert (1969). Esta limitación impulsó posteriormente el desarrollo de redes neuronales multicapa, capaces de representar relaciones no lineales mediante capas ocultas y funciones de activación diferenciables.

El perceptrón resulta relevante porque constituye la base conceptual de los modelos neuronales empleados para el reconocimiento de patrones a partir de señales. Aunque los modelos modernos utilizan arquitecturas más complejas, el principio fundamental se mantiene, ya que transforman un conjunto de características de entrada en una salida estimada mediante pesos ajustables. Por ello, comprender el funcionamiento del perceptrón permite interpretar de manera más clara la lógica de clasificación utilizada por las redes neuronales aplicadas al diagnóstico del desalineamiento angular.

7.9. Redes neuronales

Las redes neuronales artificiales constituyen una familia de modelos de aprendizaje automático inspirados en el procesamiento distribuido de información. Su estructura se basa en unidades de cálculo interconectadas, denominadas neuronas artificiales, organizadas en capas. Cada neurona recibe señales de entrada, las pondera mediante pesos ajustables, incorpora un término de sesgo y aplica una función de activación para generar una salida. Este principio extiende el concepto del perceptrón, propuesto inicialmente como un modelo computacional para el reconocimiento de patrones, como lo indica Rosenblatt Rosenblatt (1958).

A diferencia del perceptrón simple, cuya capacidad de clasificación está limitada a problemas linealmente separables, una red neuronal multicapa puede representar relaciones no lineales entre las variables de entrada y la salida del sistema. Esta capacidad se logra mediante la incorporación de capas ocultas y funciones de activación no lineales, lo que permite transformar progresivamente la información hasta obtener una representación adecuada para tareas de clasificación o regresión. El entrenamiento de este tipo de redes se fortaleció con el algoritmo de retropropagación del error, como lo señalan Rumelhart, Hinton y Williams Rumelhart et al. (1986).

En el contexto de esta investigación, las redes neuronales se emplearon para relacionar las características extraídas de las señales vibratorias con el nivel de desalineamiento angular del sistema. De esta manera, el modelo aprende patrones presentes en la señal y los utiliza para estimar o clasificar la condición mecánica evaluada. Esta capacidad de aproximar relaciones complejas ha sido sustentada teóricamente en trabajos sobre redes neuronales con funciones de activación no lineales, como lo indica Cybenko Cybenko (1989).

Existen diferentes arquitecturas de redes neuronales, las cuales se seleccionan de acuerdo con el tipo de información disponible y el objetivo del análisis. En aplicaciones de señales, imágenes, audio o datos tabulares, la arquitectura elegida depende de la forma en que se desea representar la información y del tipo de patrones que se busca identificar. En esta investigación se consideraron principalmente redes neuronales completamente conectadas, debido a que las entradas del modelo estuvieron conformadas por características estadísticas extraídas de las señales vibratorias.

7.9.1. Redes neuronales completamente conectadas

Las redes neuronales completamente conectadas, conocidas también como Fully Connected Neural Networks FCNN o perceptrones multicapa MLP, son arquitecturas en las que cada neurona de una capa se conecta con todas las neuronas de la capa siguiente. Esta estructura permite que la información fluya desde la capa de entrada hacia la capa de salida mediante una secuencia de transformaciones ponderadas y funciones de activación.

Una red completamente conectada está formada por una capa de entrada, una o más capas ocultas y una capa de salida. La capa de entrada recibe las variables del problema, que en esta investigación corresponden a las características estadísticas extraídas de las señales vibratorias. Las capas ocultas realizan transformaciones internas de la información, permitiendo identificar relaciones entre las variables. Finalmente, la capa de salida entrega el resultado del modelo, el cual puede corresponder a una estimación del ángulo de desalineamiento o a una clase asociada a un nivel de falla.

Desde el punto de vista matemático, cada neurona realiza una operación equivalente a una suma ponderada seguida de una función de activación. Sin embargo, al organizar múltiples neuronas en varias capas, el modelo adquiere una mayor capacidad para representar relaciones no lineales. Por ello, una FCNN puede entenderse como una extensión del perceptrón simple hacia una arquitectura multicapa, donde los pesos se ajustan durante el entrenamiento para reducir el error entre la salida estimada y la salida real, como lo señalan Rumelhart, Hinton y Williams Rumelhart et al. (1986) y Cybenko Cybenko (1989).

La Figura 7.17 muestra una arquitectura típica de una red neuronal completamente conectada. En ella se observa la conexión entre la capa de entrada, las capas ocultas y la capa de salida, lo que representa el flujo de información utilizado por el modelo durante el proceso de predicción.

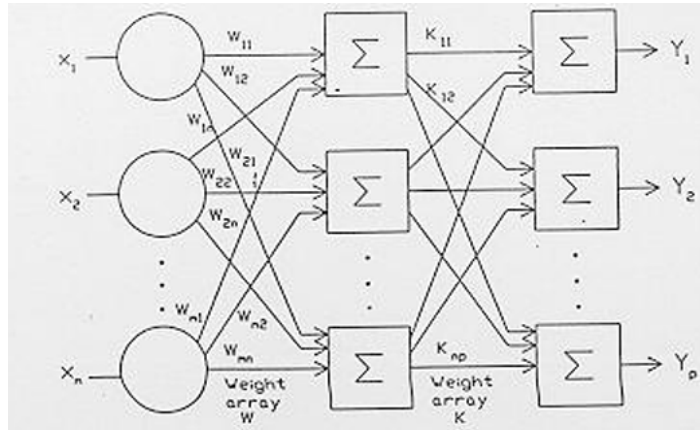


Figura 7.17: Arquitectura de una red neuronal completamente conectada, compuesta por una capa de entrada, capas ocultas y una capa de salida. Adaptado de Basogain Olabe Olabe (s f).

7.10. Transformada de Fourier

La transformada de Fourier (FFT) se define como un algoritmo utilizado para calcular un espectro a partir de una onda de tiempo. En otras palabras, es un cálculo destinado a descomponer una señal en todas sus frecuencias. El FFT convierte una señal del dominio de tiempo al dominio de frecuencia. Mediante el producto punto de la señal en el dominio del tiempo con una serie de ondas de seno y coseno con diferentes frecuencias se determina que frecuencias están más presentes en la señal. Si el producto punto resultante tiene alta amplitud, esto significa que la similitud entre la onda y la señal es alta a una específica frecuencia. El problema de la transformada de Fourier es que a más alta la frecuencia, más baja la resolución en el dominio del tiempo, por lo cual, el FFT no puede indicar que frecuencias se encuentran presentes en determinado instante de tiempo. El siguiente ejemplo demuestra lo dicho:

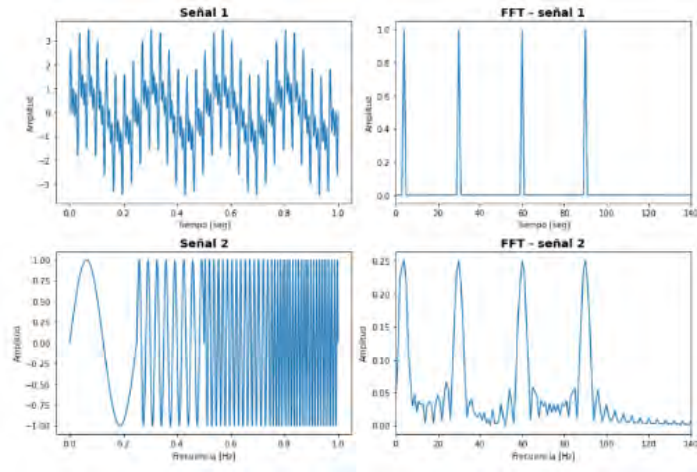


Figura 7.18: Transformada de Fourier de diferentes espectros de vibración.

En la figura anterior. Se puede ver la señal 1 que contiene diferentes frecuencias (4, 30, 60, 90 Hz) que están presentes en toda la señal. En la señal 2, se puede ver que hay 4 frecuencias presentes pero cada frecuencia está presente en un lapso de tiempo de la señal. Al obtener el FFT de cada señal, vemos que ambas contienen 4 picos y comprobamos que, a pesar de la diferente naturaleza de las señales, el FFT resulta en un espectro similar. La transformada de Fourier no puede distinguir en qué instante de tiempo se da una determinada frecuencia, que en este caso sería lo óptimo para analizar la señal 2.

8 | Antecedentes

El diagnóstico de fallas en maquinaria rotativa ha sido ampliamente estudiado mediante técnicas de análisis vibracional, procesamiento de señales, modelamiento dinámico y métodos de aprendizaje automático. En el caso particular del desalineamiento angular de ejes, los antecedentes revisados permiten establecer una base técnica para comprender el fenómeno físico, seleccionar métodos de adquisición de señales y justificar el empleo de modelos computacionales para su identificación automática.

La revisión de antecedentes se organiza considerando tres aspectos principales. Primero, los estudios relacionados con el diagnóstico vibracional en equipos industriales. Segundo, los trabajos orientados al diseño de bancos o módulos experimentales para reproducir condiciones de falla. Tercero, las investigaciones recientes que incorporan procesamiento de señales, sensores alternativos y algoritmos de aprendizaje automático.

8.1. Estudios relacionados con análisis vibracional en maquinaria rotativa

8.1.1. Determinación de fallas mediante análisis vibracional en equipos críticos

Moreno desarrolló una investigación orientada a la determinación de fallas mecánicas en equipos críticos de las áreas de envasado, pasteurización y homogenización de la empresa Prolac. El propósito principal fue conocer el estado real de los equipos mediante mediciones de vibración y criterios de severidad basados en la norma ISO 10816 Moreno (2018).

■ Objetivo del estudio

Determinar el estado operativo de máquinas críticas mediante análisis vibracional, con el fin de identificar fallas asociadas al funcionamiento mecánico de los equipos.

■ Metodología empleada

Se establecieron rutas de medición para un colector de vibraciones, se adquirieron datos en puntos específicos de los equipos y se compararon los niveles obtenidos con criterios de alarma. El análisis se apoyó en la observación de espectros cuando los valores alcanzaban niveles inadmisibles o críticos.

■ Resultados relevantes

Se identificaron fallas como desalineamiento angular, desalineamiento combinado, flexibilidad transversal y pie cojo. Estos problemas evidencian que las condiciones de montaje y alineación afectan directamente la respuesta vibratoria de las máquinas.

- **Aporte para la presente investigación**

Este antecedente respalda el uso de señales dinámicas para detectar desalineamiento, ya que demuestra que esta falla produce variaciones medibles en el comportamiento vibracional de equipos rotativos.

8.1.2. Plan de mantenimiento predictivo basado en análisis vibracional en sistemas de bombeo

Aguilar propuso un plan de mantenimiento predictivo basado en el análisis vibracional aplicado al sistema de bombeo de la unidad de espesadores del proyecto Toromochó. El estudio estuvo orientado a diagnosticar fallas mecánicas mediante el análisis de espectros de vibración y su comparación con patrones típicos de falla Aguilar (2015).

- **Objetivo del estudio**

Implementar una estrategia de mantenimiento predictivo mediante análisis vibracional para detectar fallas en equipos de bombeo.

- **Metodología empleada**

Se realizaron mediciones de vibración en los equipos, se analizaron bandas de frecuencia y se compararon los espectros obtenidos con referencias típicas de fallas mecánicas.

- **Resultados relevantes**

Se diagnosticaron problemas de desbalance, desalineación y soltura mecánica. El análisis de bandas espectrales permitió asociar determinadas zonas de frecuencia con fallas específicas.

- **Aporte para la presente investigación**

El trabajo muestra la utilidad del análisis espectral como herramienta para reconocer fallas en sistemas rotativos. Además, refuerza la importancia de automatizar el diagnóstico para mejorar la respuesta frente a fallas incipientes.

8.2. Estudios relacionados con bancos experimentales

8.2.1. Diseño e implementación de un banco didáctico para alineación de elementos rotativos

Álvarez desarrolló un banco didáctico para la alineación de elementos rotativos. El trabajo tuvo como finalidad fortalecer las prácticas académicas e investigativas relacionadas con el montaje, alineación y comportamiento de elementos rotativos Álvarez (2005).

- **Objetivo del estudio**

Diseñar e implementar un banco experimental que permita reproducir condiciones de alineación y desalineación en elementos rotativos.

- **Metodología empleada**

El desarrollo comprendió el diseño mecánico del banco, la selección de componentes, la implementación del sistema y su aplicación en prácticas de laboratorio.

- **Resultados relevantes**

El banco permitió representar condiciones reales de montaje y estudiar el comportamiento de los elementos rotativos bajo configuraciones controladas.

- **Aporte para la presente investigación**

Este antecedente justifica la construcción de un módulo de pruebas propio, debido a que un sistema experimental permite controlar el nivel de desalineamiento,

repetir ensayos y generar una base de datos confiable para el entrenamiento de modelos.

8.3. Estudios sobre el fenómeno de desalineamiento en ejes

8.3.1. Desalineamiento como falla en maquinaria rotativa

El desalineamiento en ejes es una de las fallas mecánicas más comunes en sistemas rotativos industriales. Esta condición aparece cuando los ejes acoplados no comparten una misma línea de rotación, generando configuraciones de desalineamiento paralelo, angular o combinado.

Guan y Xu estudiaron los efectos del desalineamiento sobre el comportamiento dinámico de sistemas rotativos, destacando que esta falla incrementa los niveles de vibración y genera esfuerzos adicionales sobre rodamientos, acoplamientos y soportes Guan et al. (2017); Xu and Marangoni (1994).

- **Objetivo de los estudios**

Analizar el efecto del desalineamiento sobre la respuesta dinámica de sistemas rotativos y su influencia en la generación de vibraciones.

- **Metodología empleada**

Se desarrollaron análisis experimentales y modelos dinámicos para estudiar la interacción entre eje, acoplamiento y elementos de soporte.

- **Resultados relevantes**

Se observó que el desalineamiento genera incremento de vibraciones, cargas adicionales y reducción de la vida útil de los componentes mecánicos.

- **Aporte para la presente investigación**

Estos trabajos proporcionan el fundamento físico que relaciona el desalineamiento angular con cambios medibles en la respuesta dinámica del sistema.

8.3.2. Análisis vibracional para detección de desalineamiento

El análisis vibracional permite identificar patrones asociados al desalineamiento mediante el estudio de la señal en el dominio del tiempo y de la frecuencia. En el dominio espectral, esta falla se relaciona frecuentemente con componentes en la frecuencia de rotación del eje y sus armónicos.

Palomino y Guan señalan que el desalineamiento puede manifestarse mediante componentes dominantes en 1X y 2X, aunque la presencia de estos componentes depende de la rigidez, carga, montaje y configuración del sistema Palomino et al. (2023); Guan et al. (2017).

- **Objetivo de los estudios**

Establecer criterios de identificación del desalineamiento mediante el análisis de componentes espectrales.

- **Metodología empleada**

Se analizaron señales vibracionales y se evaluó la presencia de componentes armónicas asociadas a la velocidad de rotación.

- **Resultados relevantes**

El desalineamiento angular puede producir incrementos en la componente 2X, además de variaciones en la frecuencia fundamental y en otros armónicos.

- **Aporte para la presente investigación**

Este antecedente permite justificar la extracción de características de señales adquiridas en el módulo experimental, ya que el desalineamiento modifica el contenido dinámico de la señal.

8.4. Modelamiento dinámico y procesamiento de señales

8.4.1. Modelamiento dinámico del desalineamiento

Xu estudió el comportamiento dinámico de sistemas rotor-acoplamiento sometidos a desalineamiento. Su trabajo permitió explicar la aparición de respuestas no lineales y componentes armónicas en sistemas rotativos afectados por esta condición Xu and Marangoni (1994).

- **Objetivo del estudio**

Representar matemáticamente el efecto del desalineamiento sobre la respuesta vibratoria de un sistema rotativo.

- **Metodología empleada**

Se desarrollaron modelos dinámicos que consideran la interacción entre el eje, el acoplamiento y los elementos flexibles del sistema.

- **Resultados relevantes**

El desalineamiento puede generar respuestas no lineales, incremento de vibraciones y aparición de armónicos relacionados con la velocidad de rotación.

- **Aporte para la presente investigación**

Este enfoque permite interpretar las variaciones observadas en las señales como consecuencia del cambio en la condición dinámica del sistema.

8.4.2. Técnicas de procesamiento de señales para diagnóstico de desalineamiento

Szewczyk abordó el diagnóstico de desalineamiento mediante técnicas avanzadas de procesamiento de señales. Estas técnicas buscan mejorar la sensibilidad del diagnóstico en condiciones donde el análisis espectral clásico puede presentar limitaciones Szewczyk et al. (2025).

- **Objetivo del estudio**

Mejorar la detección de desalineamiento mediante técnicas de procesamiento capaces de analizar señales con ruido, variabilidad o comportamiento no estacionario.

- **Metodología empleada**

Se aplicaron herramientas como densidad espectral de potencia, análisis tiempo-frecuencia y métodos estadísticos para representar el comportamiento de las señales.

- **Resultados relevantes**

Las técnicas avanzadas permitieron obtener una descripción más completa de la señal y mejorar la detección de cambios asociados a fallas.

- **Aporte para la presente investigación**

Este antecedente respalda la etapa de segmentación, filtrado y extracción de características aplicada a las señales CKP adquiridas en el módulo experimental.

8.5. Enfoques multisensoriales y aprendizaje automático

8.5.1. Enfoques multisensoriales para detección de desalineamiento

Navarro-Navarro estudió la detección de desalineamiento mediante la combinación de señales de vibración y variables eléctricas del motor. Esta línea de trabajo busca incrementar la confiabilidad del diagnóstico mediante el uso de múltiples fuentes de información Navarro-Navarro et al. (2025).

- **Objetivo del estudio**

Evaluar si la integración de señales mecánicas y eléctricas mejora la detección del desalineamiento en sistemas rotativos.

- **Metodología empleada**

Se adquirieron señales físicas asociadas al funcionamiento del sistema y se analizó su capacidad para representar condiciones de desalineamiento.

- **Resultados relevantes**

La combinación de señales permitió mejorar la robustez del diagnóstico, especialmente en condiciones donde una sola fuente de información puede ser limitada.

- **Aporte para la presente investigación**

Este trabajo respalda el empleo de sensores alternativos al acelerómetro convencional. En la presente tesis, el sensor CKP se utiliza como fuente de información relacionada con el comportamiento rotativo del eje.

8.5.2. Aprendizaje automático para diagnóstico de fallas en maquinaria rotativa

Lei analizó la aplicación de técnicas de aprendizaje automático en el diagnóstico de fallas en maquinaria rotativa. Este enfoque permite automatizar la identificación de fallas mediante el reconocimiento de patrones en señales experimentales Lei et al. (2020).

- **Objetivo del estudio**

Automatizar el diagnóstico de fallas en maquinaria rotativa mediante modelos capaces de aprender patrones a partir de datos experimentales.

- **Metodología empleada**

El proceso general incluye adquisición de señales, preprocesamiento, extracción de características, construcción de bases de datos, entrenamiento de modelos y evaluación mediante métricas de desempeño.

- **Resultados relevantes**

Los métodos de aprendizaje automático permiten identificar relaciones complejas y no lineales presentes en las señales, mejorando la precisión del diagnóstico frente a métodos basados únicamente en interpretación manual.

- **Aporte para la presente investigación**

Este antecedente fundamenta el uso de algoritmos como Random Forest, XG-Boost y redes neuronales para estimar automáticamente el nivel de desalineamiento angular.

8.6. Aporte global de los antecedentes a la presente investigación

A partir de los antecedentes revisados se identifican los siguientes aportes principales.

1. El desalineamiento angular modifica el comportamiento dinámico de los sistemas rotativos y genera variaciones medibles en las señales adquiridas.
2. El análisis vibracional permite detectar fallas como desbalance, soltura mecánica y desalineamiento mediante componentes espectrales y niveles de severidad.
3. Los bancos experimentales permiten reproducir condiciones controladas de falla, repetir ensayos y generar bases de datos para estudios posteriores.
4. El procesamiento de señales es una etapa necesaria para transformar los registros adquiridos en información útil para el diagnóstico.
5. Los sensores alternativos pueden complementar o reemplazar mediciones vibracionales tradicionales cuando entregan información relacionada con el comportamiento rotativo del sistema.
6. Los algoritmos de aprendizaje automático permiten automatizar la identificación de fallas a partir de patrones presentes en los datos experimentales.

9 | DISEÑO Y CONSTRUCCIÓN DE MÓDULO DE PRUEBAS

9.1. DISEÑO CONCEPTUAL

9.1.1. Metodología de diseño

Como parte de un diseño estructurado, el presente trabajo de investigación se basa en las etapas organizadas del procedimiento de diseño basado en la norma VDI 2221.

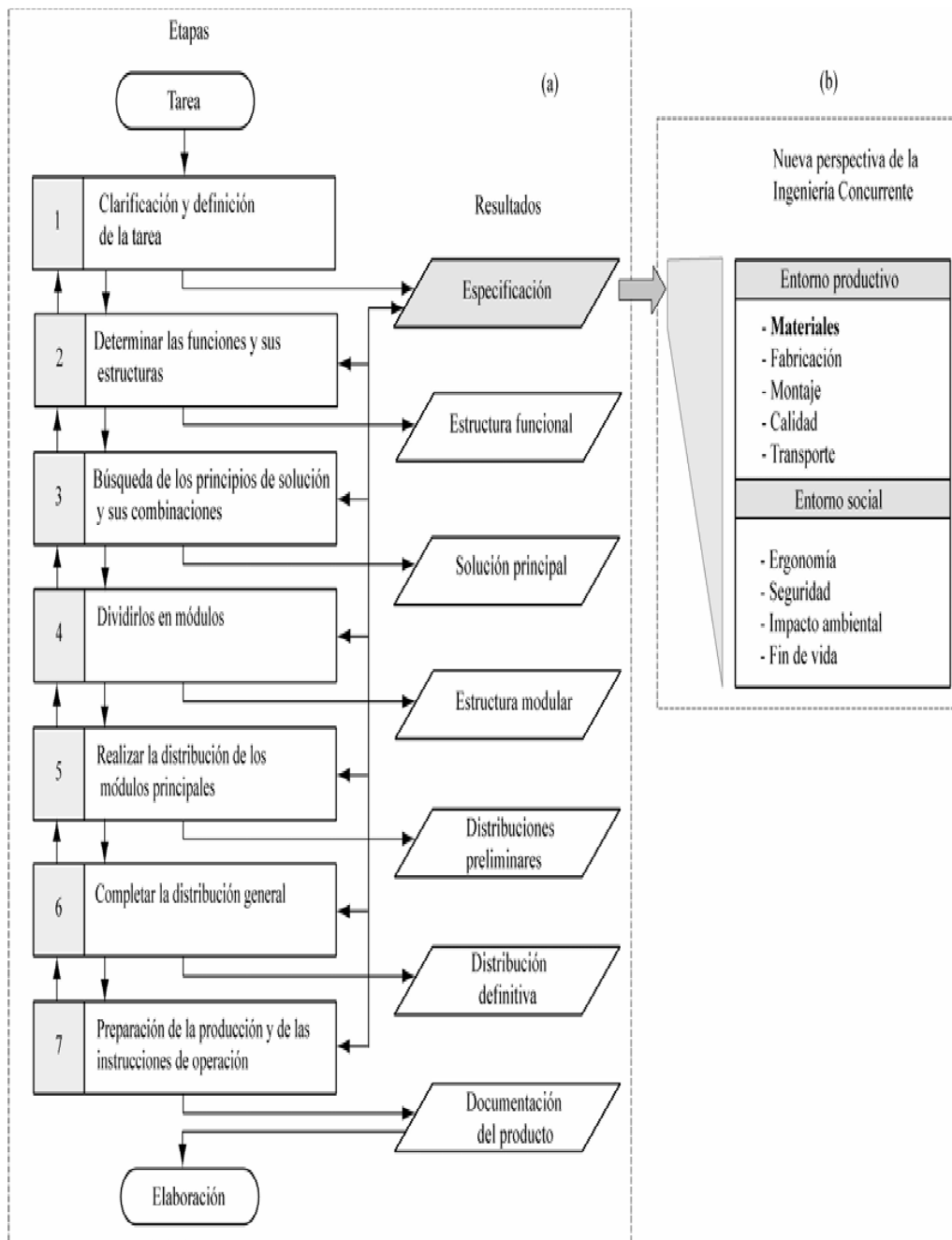


Figura 9.1: Procedimiento de diseño Adaptado: VDI 2221.

9.1.2. Lista de exigencias

En la lista de exigencias se definieron claramente los parámetros técnicos, económicos y de seguridad que se deberán satisfacer en el procedimiento de diseño. Estos requerimientos se resumen en la tabla siguiente:

LISTA DE EXIGENCIAS			
PROYECTO:		DISEÑO DE UN MÓDULO DE PRUEBAS PARA LA OBTENCIÓN DE SEÑALES DE VIBRACIÓN	Fecha -----
CLIENTE:		UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO	Revisado -----
Fecha (modificación)	Deseo o Exigencia	Descripción	Responsable
04/10/2024	E	Función principal: El módulo de pruebas tiene por finalidad generar un movimiento rotatorio en un eje motriz conectado a un eje conducido desalineado y, mediante un sensor acoplado a éste, generar señales de vibración.	W.O.E.A – H.A.C.S
04/10/2024	E	Mecanismo para lograr el desalineamiento: Se seleccionará un acople de tipo flexible que permita tener varios niveles de desalineamiento entre eje motriz y conducido.	W.O.E.A – H.A.C.S
04/10/2024	E	Adquisición de data: Los sensores y sistemas de adquisición de data deberán seleccionarse de modo que la sensibilidad y tasa de muestreo sean las adecuadas.	W.O.E.A – H.A.C.S
04/10/2024	E	Procesamiento de data: La data vibracional generada deberá de ser procesada y evaluada.	W.O.E.A – H.A.C.S
04/10/2024	E	Energía: El módulo de pruebas utilizará energía eléctrica monofásica, la cual se rectificará para alimentar los diferentes sistemas del equipo.	W.O.E.A – H.A.C.S
04/10/2024	E	Costos: El proyecto es financiado integralmente por los investigadores a cargo del estudio, por lo que se toma en consideración que el ratio costo/eficiencia de los subsistemas sea el menor posible.	W.O.E.A – H.A.C.S

Figura 9.2: Lista de exigencias.

9.1.3. Caja negra

En el diseño del módulo de pruebas, se usa el término de caja negra para describir los componentes, sistemas o procesos que operan sin que el diseñador necesite conocer los detalles internos de su funcionamiento. El diseñador solo está interesado en las variables de entrada y salida del sistema, sin necesidad de comprender cómo se lleva a cabo el proceso interno. Esto permite la clarificación de los resultados esperados y su interacción con las entradas disponibles.

En la caja negra se definen las entradas y salidas esperadas del sistema, como se muestra a continuación:

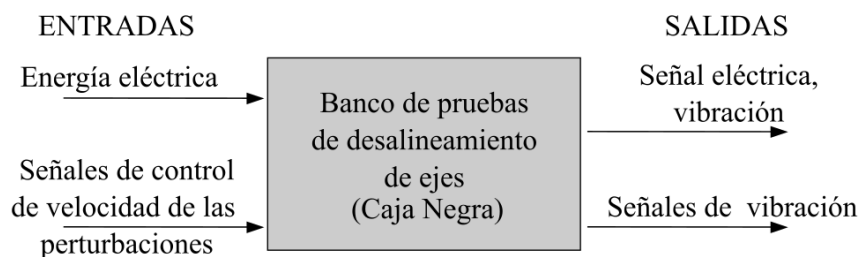


Figura 9.3: Caja negra.

Entradas:

- Energía: Se le suministra al sistema energía eléctrica monofásica.
- Señal: Se tendrá una señal de encendido y apagado del sistema motriz y sistema de adquisición de datos.

Salidas:

- Energía: Energía cinética en forma del movimiento del eje que acciona al mecanismo generador de señales.
- Señal: Se obtendrán las señales electricas porporcionales a la variacion de velocidad en el eje de salida.

9.1.4. Estructura de funciones

En el contexto de la norma de diseño VDI 2221, la estructura funcional hace referencia a la organización y relación de las funciones que deberá de cumplir un producto o sistema. En el diseño del módulo de pruebas, el enfoque de estructura funcional implica dividir el producto en sus funciones esenciales y definir cómo estas funciones interactúan entre sí para cumplir con los objetivos planteados en la lista de exigencias.

La estructura de funciones adicionalmente proporciona una visión clara de las interacciones y dependencias entre las funciones, lo que permite una asignación eficiente de recursos y la identificación de posibles puntos de fallo o áreas de mejora del módulo de pruebas.

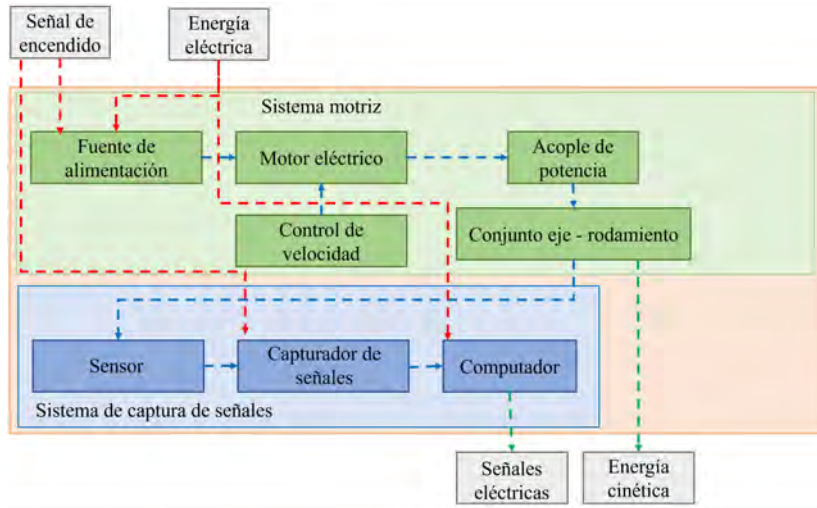


Figura 9.4: Estructura de funciones del módulo de pruebas desarrollada según la metodología VDI 2221. Se consideran como funciones principales el control de velocidad, la rotación del eje, la transmisión de potencia, la medición de la señal y la captura de la señal.

9.1.5. Matriz morfológica

A continuación se presenta la matriz morfológica elaborada para el desarrollo del módulo de pruebas, en la cual se plantean tres posibles conceptos de solución. En esta matriz se muestran las principales funciones del sistema junto con sus respectivos componentes y variantes consideradas, con el fin de visualizar de manera ordenada las diferentes combinaciones que pueden dar lugar a una solución conceptual del diseño, tal como se observa en la Figura 9.5.

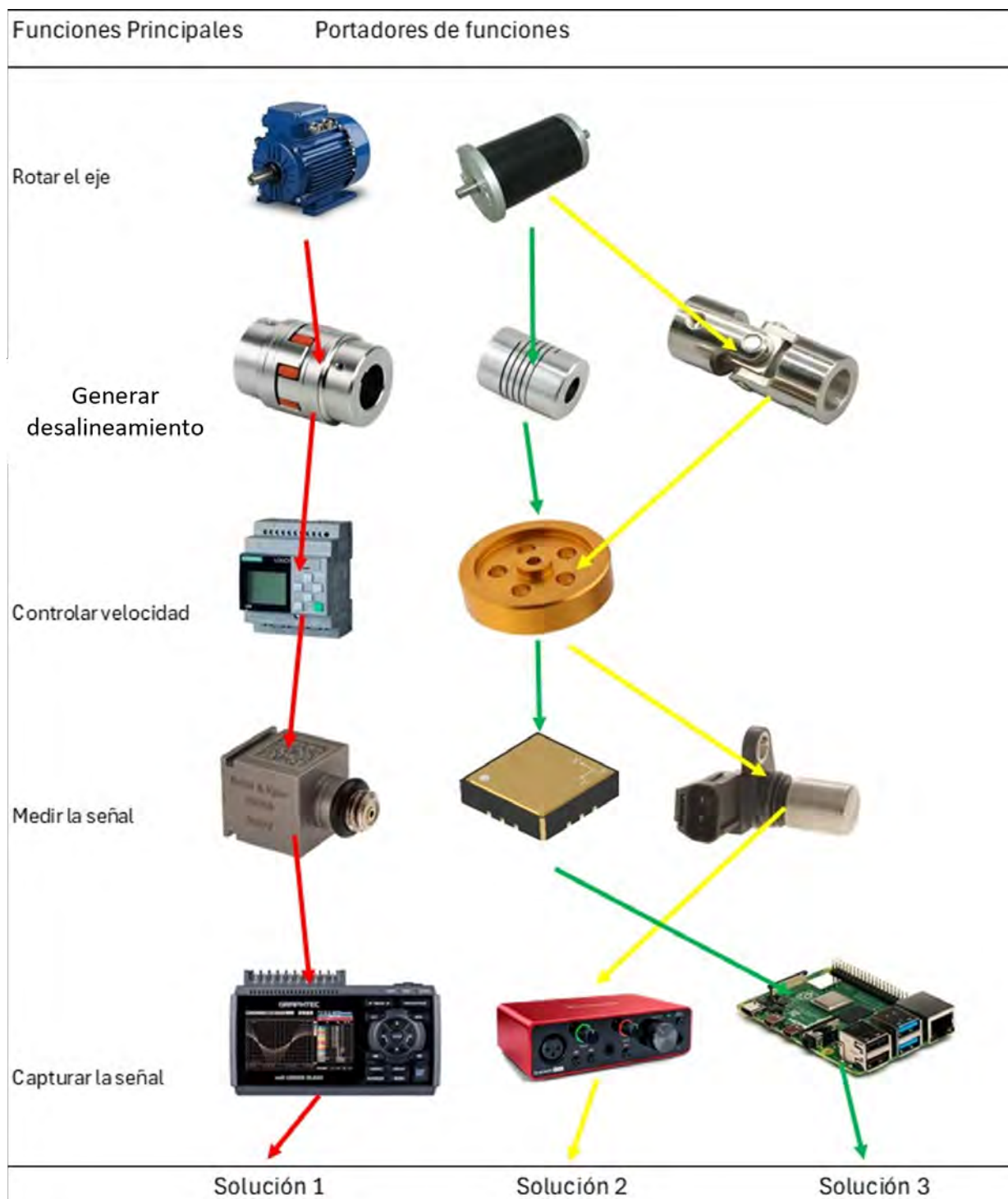


Figura 9.5: Matriz morfológica elaborada para la selección conceptual del módulo de pruebas.

Solución 1

La solución 1 consta de un motor eléctrico de corriente alterna monofásico el cual tiene como acople entre ejes un acople de tipo Oldham. El control de velocidad del motor consiste en un sistema mediante un variador de frecuencia. Para la captura de datos se seleccionó un acelerómetro analógico piezoeléctrico con un equipo de adquisición de datos.

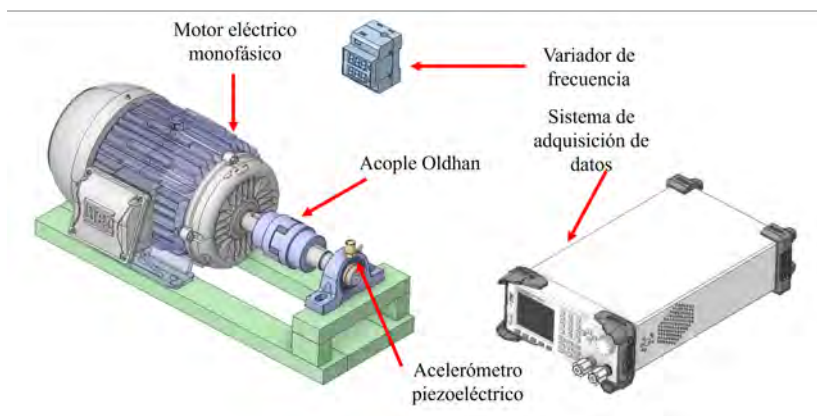


Figura 9.6: Solución 1

Solución 2

La solución 2 consta de un motor eléctrico de corriente continua, el cual cuenta con un volante de inercia como método de regulación de velocidad. El acoplamiento entre ejes se realiza mediante un acople universal tipo cardán. Para la medición de la velocidad de rotación se utilizó un sensor CKP (sensor de posición del cigüeñal magnético), cuyos datos son adquiridos mediante una interfaz de audio conectada a un equipo de cómputo.

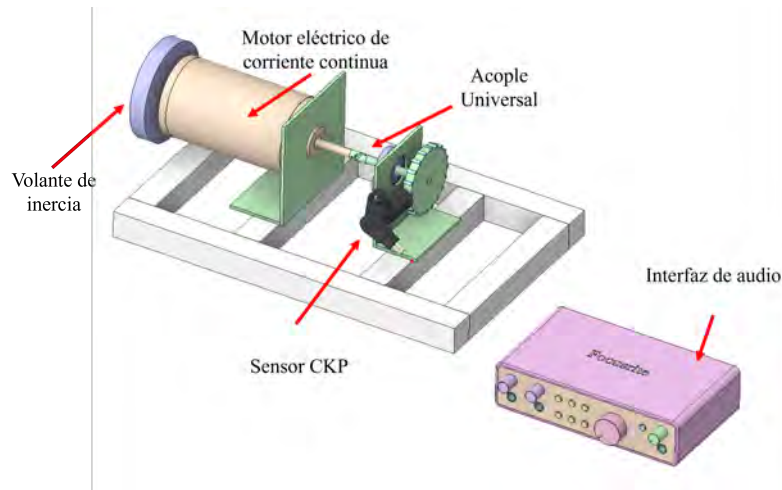


Figura 9.7: Solución 2

Solución 3

La solución 3 está compuesta por un motor de corriente continua, el cual también incorpora un volante de inercia para la regulación de la velocidad. El acoplamiento entre el motor y la carga se efectúa mediante un acople flexible de aluminio. Para la adquisición de datos se empleó un acelerómetro MEMS, cuyos datos son procesados en tiempo real mediante un sistema embebido basado en una plataforma ESP32 o Raspberry Pi.

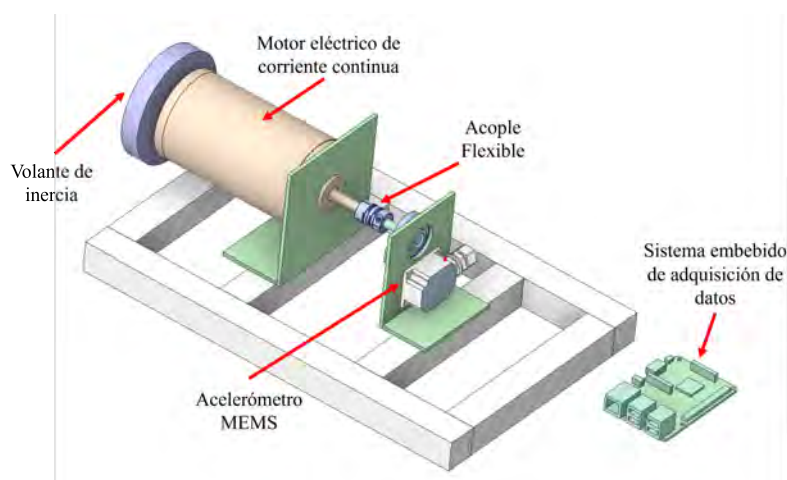


Figura 9.8: Solución 3

9.1.6. Valoración técnica económica

Para seleccionar la alternativa más adecuada del módulo de pruebas, se realizó una valoración técnica y económica de las tres soluciones propuestas. En las Tablas 9.1 y 9.2 se presentan los criterios evaluados, sus respectivos porcentajes de importancia y los puntajes asignados a cada solución según una escala de 0 a 4. Esta evaluación permitió comparar las alternativas considerando aspectos económicos, como costos de diseño, fabricación, mantenimiento y operación, así como aspectos técnicos relacionados con la calidad de los datos, precisión del alineamiento, seguridad, adquisición de señales y facilidad de mantenimiento.

Cuadro 9.1: Evaluación de criterios económicos

Escala de valoración (puntajes entre 0 - 4)					
0=No satisface, 1=Aceptable, 2=Suficiente, 3=Bien, 4=Ideal					
Nro	Criterios Economicos	Importancia (i)	Soluciones		
			S1	S2	S3
1	Costos de diseño	8 %	1	2	2
2	Costos de material	13 %	1	3	3
3	Costos de fabricacion	18 %	3	3	3
4	Costos de mantenimiento	8 %	2	3	3
5	Facilidad de montaje	13 %	1	3	2
6	Costos de operacion	10 %	1	3	3
7	Costos de hardware adicional	16 %	1	3	3
8	Costos de software	14 %	1	3	3
Puntaje máximo $PM = \sum (i) / 100$		100 %	34 %	72 %	69 %
Puntaje unitario $PU = PM / 4$			1.38	2.88	2.75

Cuadro 9.2: Evaluación de criterios técnicos.

Escala de valoración (puntajes entre 0 - 4)					
0=No satisface, 1=Aceptable, 2=Suficiente, 3=Bien, 4=Ideal					
Nro	Criterios Económicos	Importancia (i)	Soluciones		
			S1	S2	S3
1	Calidad de data recolectada	17 %	4	3	4
2	Sistema de adquisición de datos	13 %	3	3	3
3	Precisión en el alineamiento de ejes	11 %	1	3	2
4	Seguridad en la operación	10 %	1	3	1
5	Facilidad de montaje y desmontaje	13 %	2	3	3
6	Facilidad de mantenimiento	10 %	2	3	2
7	Uso energético	9 %	1	3	3
8	Software de procesamiento de data	17 %	3	3	3
Puntaje maximo $PM = \sum (\%i)/100$		100 %	53 %	75 %	66 %
Puntaje unitario $PU = PM/4$			2.13	3.00	2.63

A partir de los puntajes porcentuales obtenidos en ambas evaluaciones, se elaboró el gráfico técnico-económico mostrado en la Figura 9.9. En este gráfico, el eje horizontal representa el desempeño técnico y el eje vertical el desempeño económico de cada alternativa. Los resultados muestran que la Solución 1 alcanzó un 53 % en criterios técnicos y 34 % en criterios económicos, la Solución 2 obtuvo 75 % y 72 %, respectivamente, mientras que la Solución 3 alcanzó 66 % y 69 %. Por lo tanto, la Solución 2 presentó el mejor equilibrio técnico-económico, al ubicarse con los mayores puntajes en ambos criterios y más próxima a una condición favorable de desempeño global.

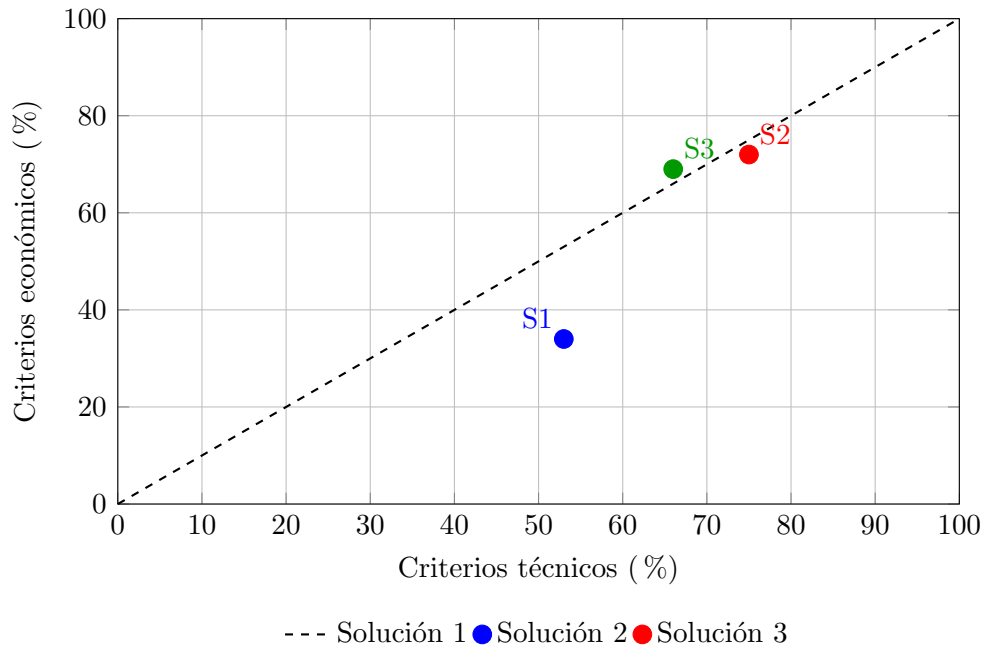


Figura 9.9: Gráfico de evaluación técnico-económica de las soluciones propuestas. El eje horizontal representa el puntaje técnico porcentual y el eje vertical representa el puntaje económico porcentual. La línea diagonal indica una condición de equilibrio entre ambos criterios.

9.2. Diseño modular

El diseño final del módulo de pruebas se dividirá en un diagrama de bloques, mediante el cual se puedan descomponer los subsistemas, y consistirá en dos componentes básicos:

El componente mecánico consistirá en un mecanismo capaz de ejercer un movimiento rotatorio al eje motriz. El componente electrónico del sistema consistirá en un sensor CKP encargado de medir las variaciones de velocidad producidas por el desalineamiento del eje de salida y un sistema colector de datos, el cual deberá poder leer los datos que proporcione el sensor y transmitir los datos a un computador.

9.2.1. Sistema motriz

De acuerdo a la estructura de funciones los primeros módulos a seleccionar corresponderán al sistema motriz, el cual consiste en un motor de corriente continua y una fuente de alimentación.

Se seleccionó un servomotor de corriente continua de la marca YAMAMOTO ELECTRIC CORPORATION modelo YA-010-1. Se seleccionó este motor debido al tamaño reducido, capacidad de alimentarlo con fuentes de corriente continua comunes y bajos niveles de vibración.



Figura 9.10: Modelo de motor seleccionado

9.2.2. Placa base de soporte

9.2.2.1. Diseño por resistencia

Se analiza una viga simplemente apoyada sometida a una carga distribuida del peso propio y a la carga puntual producida por el motor montado en voladizo sobre la estructura. La viga está fabricada a partir de una placa de acero de espesor constante y se apoya de manera simple en ambos extremos, permitiendo el giro pero restringiendo el desplazamiento vertical y axial, como se especifica en el diagrama siguiente:

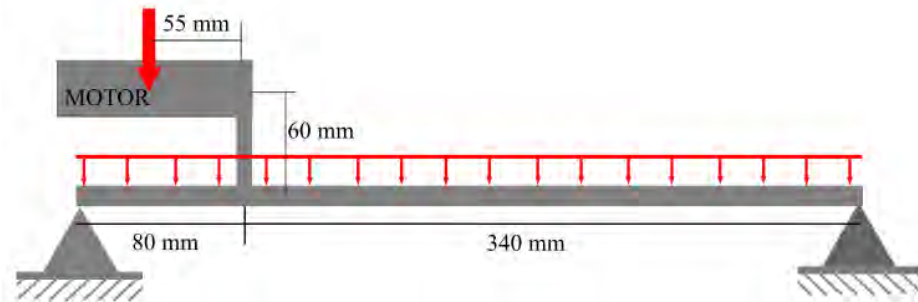


Figura 9.11: Modelo conceptual estático del módulo de pruebas.

Se tienen los parámetros de entrada en el cuadro siguiente:

Cuadro 9.3: Parámetros geométricos y físicos del modelo

Parámetro	Símbolo	Valor
Longitud total de la viga	L	420 mm
Posición del motor desde el apoyo izquierdo	a	25 mm
Excentricidad del motor	e	55 mm
Ancho de la placa	b	190 mm
Espesor de la placa	t	6 mm
Densidad del acero	ρ	7850 kg/m ³
Aceleración de la gravedad	g	9,81 m/s ²
Masa del motor	m	2,366 kg

La carga puntual producida por el peso del motor se calcula a partir de la masa del mismo y la aceleración de la gravedad, según la ecuación.

$$P = mg \quad (9.1)$$

El momento concentrado transmitido por el motor a la viga se debe a la excentricidad existente entre el centro de gravedad del motor y el punto de apoyo estructural.

$$M_0 = Pe \quad (9.2)$$

El peso propio de la viga se modela como una carga distribuida uniforme. Para ello, primero se calcula el área de la sección transversal de la placa.

$$A = bt \quad (9.3)$$

Posteriormente, la intensidad de la carga distribuida se obtiene a partir de la densidad del material, el área de la sección transversal y la aceleración de la gravedad.

$$w = \rho Ag \quad (9.4)$$

Las reacciones en los apoyos se determinan aplicando las ecuaciones de equilibrio estático al diagrama de cuerpo libre de la viga. La condición de equilibrio de fuerzas verticales.

$$\sum F_y = 0 \quad (9.5)$$

Asimismo, la condición de equilibrio de momentos se establece mediante la ecuación (9.6).

$$\sum M = 0 \quad (9.6)$$

El cuadro siguiente muestra los resultados obtenidos del cálculo estático:

Cuadro 9.4: Resultados (según DCL y diagramas obtenidos)

Resultado	Valor
Reacción en apoyo izquierdo R_A [N]	40.265
Reacción en apoyo derecho R_B [N]	19.816
Carga puntual del motor P [N]	23.210
Carga distribuida w [N/m]	87.790
Momento aplicado por excentricidad M_0 [N m]	1.277
Momento flector máximo $M_{\text{máx}}$ [N m]	2.94

A partir del diagrama de cuerpo libre se identificaron las reacciones en los apoyos y las solicitaciones externas actuantes sobre la viga. Con estos resultados se construyeron los diagramas de fuerza cortante y momento flector.



Figura 9.12: Diagrama de cuerpo libre estático del módulo de pruebas.

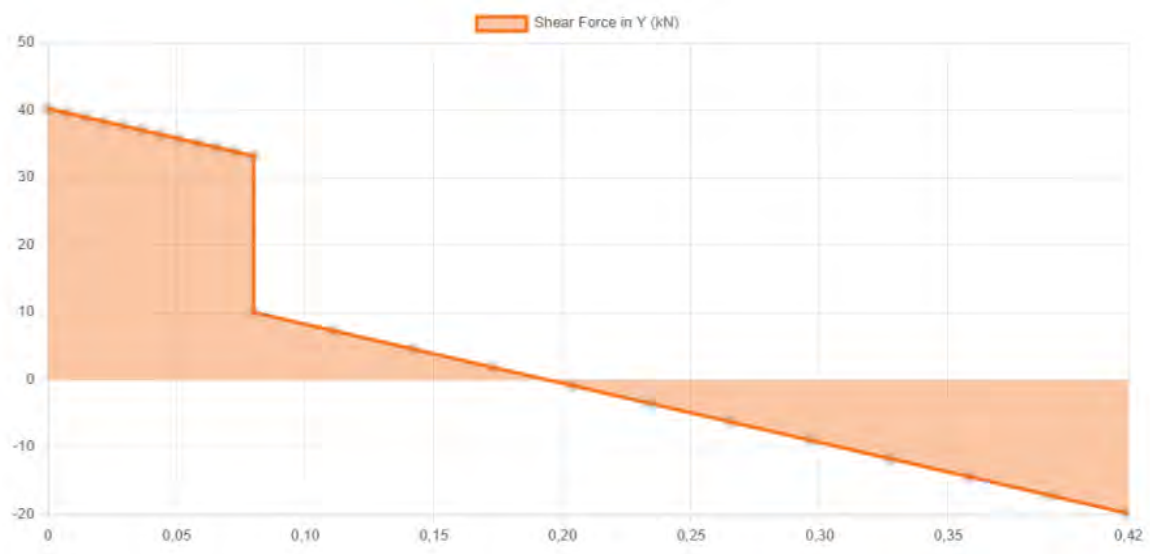


Figura 9.13: Diagrama de fuerzas cortantes.

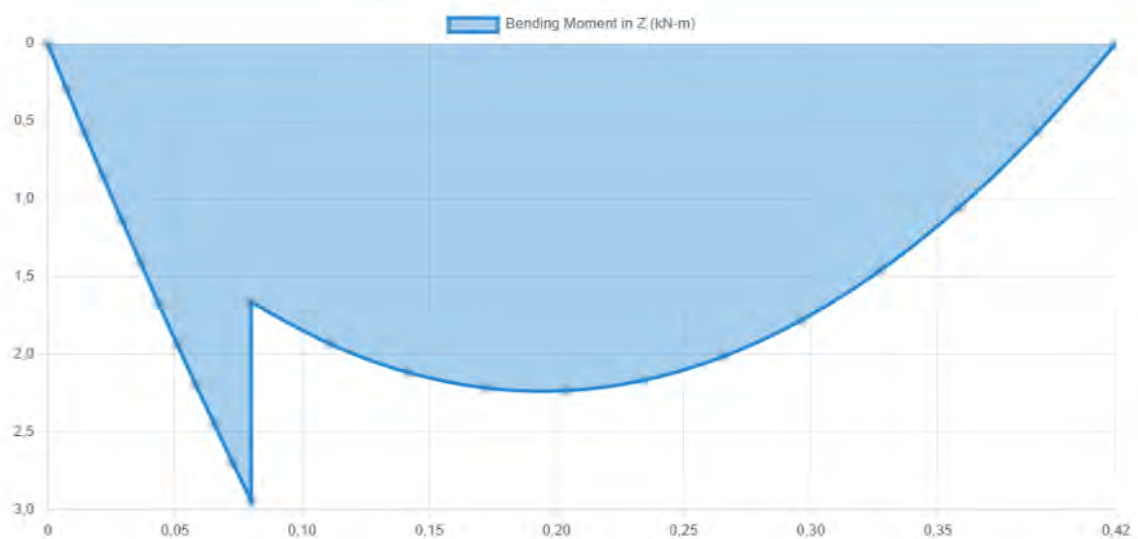


Figura 9.14: Diagrama de momentos flectores.

Se calcula el momento de inercia de la placa de soporte.

$$I = \frac{b h^3}{12} \quad (9.7)$$

$$I = \frac{(6 \text{ mm})(190 \text{ mm})^3}{12} = 3\,429\,500 \text{ mm}^4$$

$$c = \frac{h}{2} \tag{9.8}$$

$$c = \frac{190 \text{ mm}}{2} = 95 \text{ mm}$$

Una vez obtenido el momento flector máximo a partir del diagrama de momento flector, el esfuerzo normal máximo se calcula con la ecuación de flexión:

$$\sigma_{\text{máx}} = \frac{M_{\text{máx}} c}{I} \tag{9.9}$$

Para mantener consistencia de unidades, el momento se expresa en N·mm. Con $M_{\text{máx}} = 2,94 \text{ N}\cdot\text{m} = 2\,940 \text{ N}\cdot\text{mm}$, se obtiene:

$$\sigma_{\text{máx}} = \frac{(2\,940 \text{ N}\cdot\text{mm})(95 \text{ mm})}{3\,429\,500 \text{ mm}^4} = 0,0814 \text{ N/mm}^2 = 0,0814 \text{ MPa}$$

El valor obtenido del esfuerzo normal máximo indica que la viga trabaja dentro de un rango seguro para los materiales estructurales usuales, sin comprometer su integridad bajo las condiciones de carga consideradas.

9.2.2.2. Diseño dinámico

La estimación de la primera frecuencia natural del sistema se realiza mediante el método de Rayleigh, el cual es un método energético ampliamente utilizado para la aproximación de la frecuencia fundamental de sistemas continuos. Este método se basa en la igualdad entre la energía cinética máxima y la energía de deformación máxima durante la vibración libre no amortiguada del sistema Rao (2017); Inman (2014).

En el modelo dinámico se considera una viga simplemente apoyada con un motor rígidamente unido a la estructura. Se incluyen la masa distribuida de la viga, la masa concentrada del motor ubicada en su centro de gravedad y la inercia rotacional del motor, la cual se acopla a la pendiente de la viga en el punto de unión.

La luz total de la viga es:

$$L = 80 \text{ mm} + 340 \text{ mm} = 420 \text{ mm} = 0,42 \text{ m}$$

El centro de gravedad del motor se encuentra a una distancia:

$$a = 80 \text{ mm} - 55 \text{ mm} = 25 \text{ mm} = 0,025 \text{ m}$$

La viga se modela como una placa de sección rectangular de ancho $b = 190 \text{ mm}$ y espesor $t = 6 \text{ mm}$. El momento de inercia de la sección respecto al eje neutro se calcula como:

$$I = \frac{bt^3}{12} \tag{9.10}$$

$$I = \frac{(0,19)(0,006)^3}{12} = 3,42 \times 10^{-9} \text{ m}^4$$

El área de la sección y la masa por unidad de longitud de la viga son:

$$A = bt = 1,14 \times 10^{-3} \text{ m}^2$$

$$\mu = \rho A = 8,949 \text{ kg/m}$$

El motor se idealiza como un cilindro macizo de diámetro $d = 53$ mm y longitud $\ell = 110$ mm. La inercia rotacional del motor respecto a un eje que pasa por su centro de gravedad y es perpendicular a su eje longitudinal se expresa como Blevins (1979):

$$J_m = \frac{1}{12} m_m (3r^2 + \ell^2) \quad (9.11)$$

$$J_m = 2,80 \times 10^{-3} \text{ kg}\cdot\text{m}^2$$

Para la aplicación del método de Rayleigh se asume como forma modal el primer modo de vibración de una viga simplemente apoyada:

$$\phi(x) = \sin\left(\frac{\pi x}{L}\right)$$

La expresión aproximada para la frecuencia natural angular del primer modo, incluyendo masa distribuida, masa puntual e inercia rotacional, resulta:

$$\omega_1^2 \approx \frac{\int_0^L EI (\phi''(x))^2 dx}{\int_0^L \mu (\phi(x))^2 dx + m_m \phi(a)^2 + J_m (\phi'(a))^2} \quad (9.12)$$

Al evaluar las integrales correspondientes para la forma modal adoptada, se obtiene la siguiente expresión compacta:

$$\omega_1^2 \approx \frac{EI \frac{\pi^4}{2L^3}}{\mu \frac{L}{2} + m_m \sin^2\left(\frac{\pi a}{L}\right) + J_m \left(\frac{\pi}{L}\right)^2 \cos^2\left(\frac{\pi a}{L}\right)}$$

Finalmente, la primera frecuencia natural del sistema se calcula como:

$$f_1 = \frac{\omega_1}{2\pi} \quad (9.13)$$

Sustituyendo los valores geométricos, mecánicos y de masa del sistema analizado, se obtiene:

$$f_1 \approx 72 \text{ Hz}$$

El valor obtenido se encuentra alejado de la frecuencia de excitación asociada a la velocidad de giro del motor. Para una velocidad de 940 rpm, la frecuencia de operación es aproximadamente 15,67 Hz, por lo que no se presentan condiciones de resonancia en el sistema bajo esta condición de trabajo.

9.2.3. Volante de inercia

Se seleccionó un volante de inercia como el elemento de máquina encargado de proporcionar estabilidad rotacional al eje durante el funcionamiento del módulo de pruebas.

Un volante de inercia es un dispositivo mecánico utilizado para almacenar energía cinética rotacional. Su función principal es reducir las variaciones de velocidad angular del eje, aportando una rotación más uniforme durante la operación del sistema. En este caso, el volante permite estabilizar el movimiento del eje cuando el motor alcanza su velocidad de trabajo.

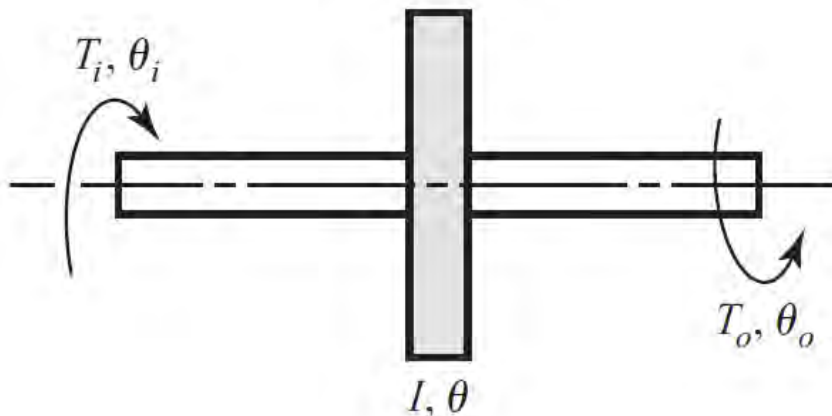


Figura 9.15: Representación matemática de un volante de inercia. Richard G. Budynas (2012)

La ecuación dinámica del volante de inercia se expresa como:

$$\sum M = T_i(\theta_i, \dot{\theta}_i) - T_o(\theta_o, \dot{\theta}_o) - I\ddot{\theta} = 0 \quad (9.14)$$

Donde:

- $T_i(\theta_i, \dot{\theta}_i)$: torque de ingreso.
- $T_o(\theta_o, \dot{\theta}_o)$: torque de salida.
- I : momento de inercia.
- θ : desplazamiento angular.
- $\ddot{\theta}$: aceleración angular.

Simplificando la ecuación anterior, se obtiene:

$$I\ddot{\theta} = T_i(\theta_i, \omega_i) - T_o(\theta_o, \omega_o) \quad (9.15)$$

Para una estimación teórica del diseño, se considera la relación entre el torque, la potencia y la velocidad angular:

$$P = T\omega \quad (9.16)$$

Despejando el torque:

$$T = \frac{P}{\omega} \quad (9.17)$$

La potencia eléctrica consumida por el motor DC se calcula a partir del voltaje y la corriente de operación:

$$P = VI \quad (9.18)$$

El motor utilizado trabaja con un voltaje de 5 V y una corriente aproximada de 1,3 A, por lo tanto:

$$P = 5(1,3)$$

$$P = 6,5 \text{ W}$$

La velocidad de operación del eje es de 940 rpm. Esta velocidad se convierte a radianes por segundo mediante:

$$\omega = N \left(\frac{2\pi}{60} \right) \quad (9.19)$$

$$\omega = 940 \left(\frac{2\pi}{60} \right)$$

$$\omega = 98,44 \text{ rad/s}$$

Por lo tanto, el torque teórico del motor se calcula como:

$$T_{\text{motor}} = \frac{P}{\omega} \quad (9.20)$$

$$T_{\text{motor}} = \frac{6,5}{98,44}$$

$$T_{\text{motor}} = 0,066 \text{ N} \cdot \text{m}$$

Para calcular la aceleración angular, se considera que el sistema pasa del estado de reposo hasta la velocidad de operación de 940 rpm en un tiempo de 0,23 s:

$$\alpha = \frac{\omega_f - \omega_i}{t} \quad (9.21)$$

$$\alpha = \frac{98,44 - 0}{0,23}$$

$$\alpha = 428 \text{ rad/s}^2$$

El momento de inercia mínimo requerido se obtiene a partir de la relación dinámica:

$$T = I\alpha \quad (9.22)$$

Despejando el momento de inercia:

$$I_{\text{mín}} = \frac{T_{\text{motor}}}{\alpha} \quad (9.23)$$

Reemplazando los valores obtenidos:

$$I_{\text{mín}} = \frac{0,066}{428}$$

$$I_{\text{mín}} = 1,54 \times 10^{-4} \text{ kg} \cdot \text{m}^2$$

Considerando un margen adicional del 10 % para compensar pérdidas, variaciones de carga y efectos no considerados en la estimación teórica, el momento de inercia requerido se calcula como:

$$I_{\text{req}} = 1,10I_{\text{mín}} \quad (9.24)$$

$$I_{\text{req}} = 1,10(1,54 \times 10^{-4})$$

$$I_{\text{req}} = 1,69 \times 10^{-4} \text{ kg} \cdot \text{m}^2$$

Las dimensiones del volante de inercia seleccionado se muestran en la Figura 9.16.

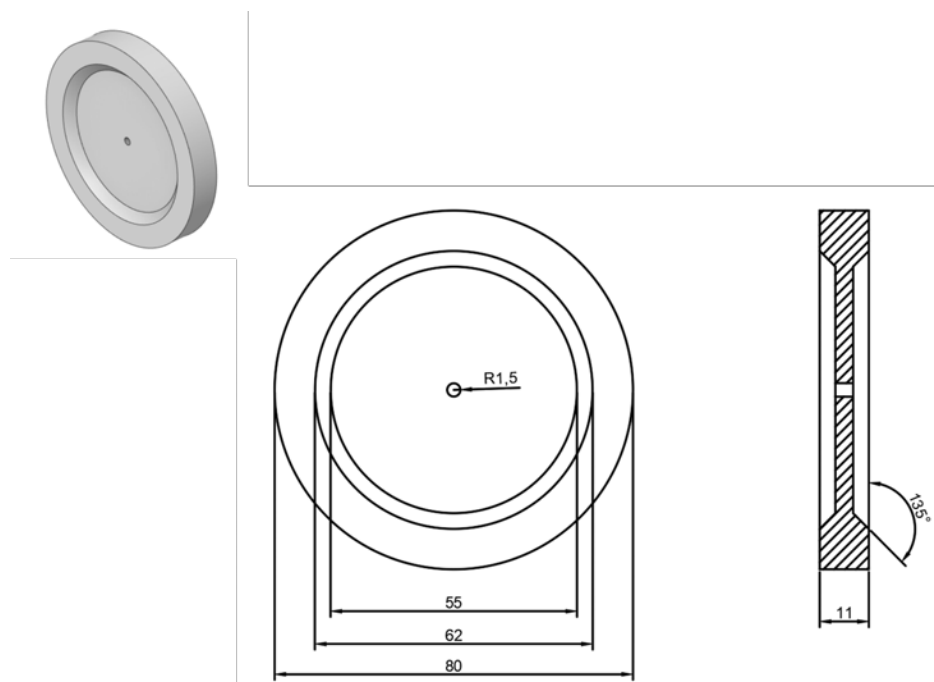


Figura 9.16: Volante de inercia seleccionado

Las propiedades del volante de inercia son:

- Masa: 287 g.
- Momento de inercia: $3,35 \times 10^{-4} \text{ kg} \cdot \text{m}^2$.
- Volumen: 36426 mm^3 .
- Radio exterior: 80 mm.
- Espesor: 11 mm.

Comparando el momento de inercia requerido con el momento de inercia del volante seleccionado:

$$I_{\text{volante}} > I_{\text{req}} \quad (9.25)$$

$$3,35 \times 10^{-4} > 1,69 \times 10^{-4}$$

Por lo tanto, el volante de inercia seleccionado cumple con los requerimientos del sistema, debido a que su momento de inercia es mayor que el momento de inercia mínimo requerido.

9.2.4. Acople universal

Se seleccionó un acople universal o cardán de acero de diámetro de 8 mm.

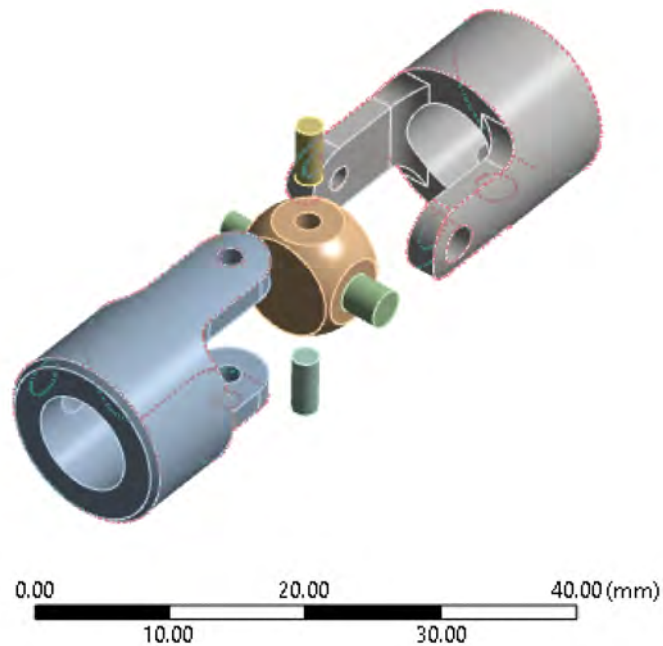


Figura 9.17: Acople universal tipo cardán de 8 mm de diámetro de eje.

Debido a que el acople no es una parte estandarizada y no cuenta con una hoja de datos en la cual se especifiquen las cargas admisibles, se realizará la verificación de esfuerzos máximos mediante el método de los elementos finitos.

	A	B	C	D	E
1	Property	Value	Unit		
2	Material Field Variables	Table			
3	Density	2770	kg m ⁻³		
4	Isotropic Secant Coefficient of Thermal Expansion				
5	Coefficient of Thermal Expansion	2.3E-05	C ⁻¹		
6	Isotropic Elasticity				
7	Derive from	Young's Modulus and Poisson's Ratio			
8	Young's Modulus	7.1E+10	Pa		
9	Poisson's Ratio	0.33			
10	Bulk Modulus	6.9608E+10	Pa		
11	Shear Modulus	2.6692E+10	Pa		
12	S-N Curve	Tabular			
16	Tensile Yield Strength	2.8E+08	Pa		
17	Compressive Yield Strength	2.8E+08	Pa		
18	Tensile Ultimate Strength	3.1E+08	Pa		
19	Compressive Ultimate Strength	0	Pa		
20	Isotropic Thermal Conductivity	Tabular			
23	Specific Heat Constant Pressure, C _p	875	J kg ⁻¹ C ⁻¹		
24	Isotropic Relative Permeability	1			
25	Isotropic Resistivity	Tabular			

Figura 9.18: Propiedades del aluminio. Fuente: ANSYS Materials.

9.2.5. Simulación computacional del acople

Se toma como parámetro de medida de calidad de malla la ortogonalidad entre elementos, donde los elementos que se aproximan mas al valor unitario son los de mayor calidad.

Los controles globales de malla establecen una calidad objetivo de 5 mm, por lo que se procede a realizar un refinamiento.

Se ha seleccionado una malla de elementos hexaédricos para el refinamiento del modelo numérico, dada su capacidad para representar de manera eficiente las geometrías regulares y su desempeño superior en términos de precisión y estabilidad computacional en comparación con elementos tetraédricos. Se ha establecido un tamaño de refinamiento uniforme de 2.5 mm. Esta configuración ha sido definida tras un proceso iterativo de verificación de la convergencia, asegurando que las variaciones en los resultados entre refinamientos sucesivos sean mínimas, manteniendo un balance entre precisión numérica y tiempo de procesamiento computacional.

La calidad ortogonal para un tamaño de malla de 2.5 mm revela una buena

distribución de los elementos. Teniendo la mayor cantidad de componentes de malla con valor de calidad 0.88 por lo que se considera que el mallado es el adecuado para la pieza.

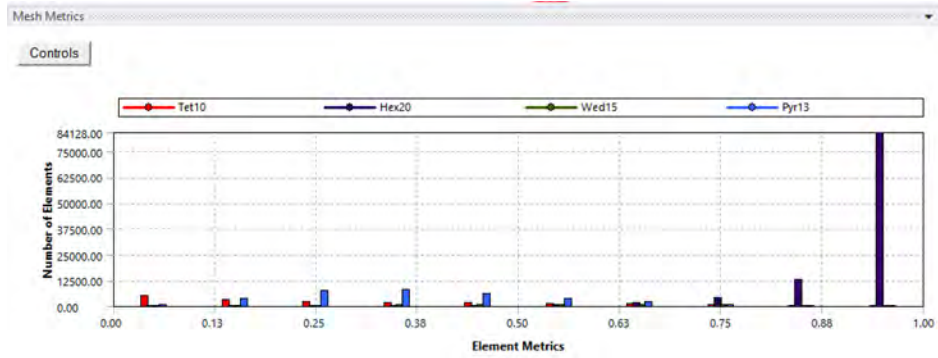


Figura 9.19: Propiedades de calidad del mallado inicial del acople flexible.

9.2.6. Análisis de esfuerzos

Los esfuerzos equivalentes de Von Mises en el acople de acero estructural se muestran a continuación:

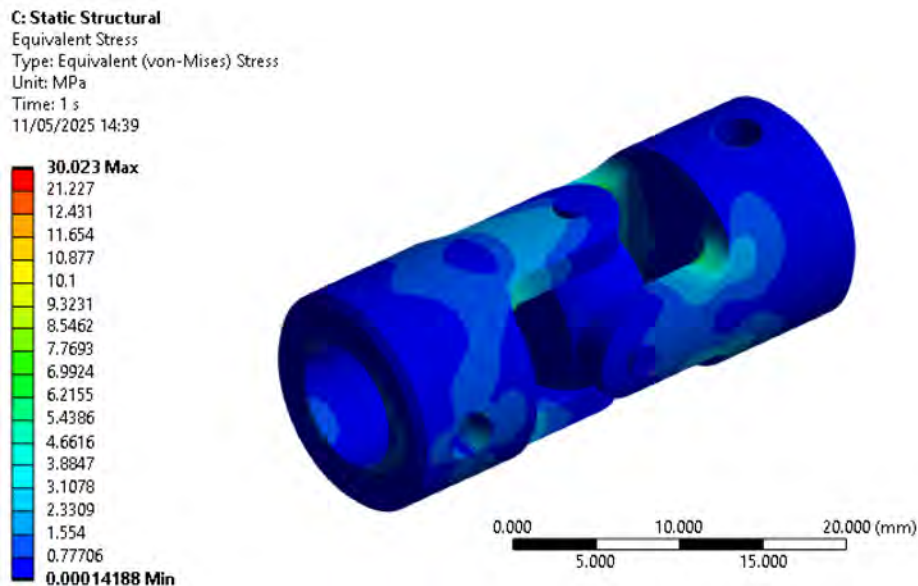


Figura 9.20: Esfuerzos equivalentes en acople con malla refinada de 2.5mm.

Se observa que la distribución de esfuerzos se da en mayor parte en la zona interna

del acople en unos de los pines de la cruceta y tiene como valor máximo 30.023 MPa. Las propiedades del acero estructural ASTM A-36 indican una resistencia a la fluencia de 250 MPa, por lo que el factor de seguridad para el acople flexible es de 8.3, porque la selección del componente es la adecuada y cumple con los parámetros de diseño.

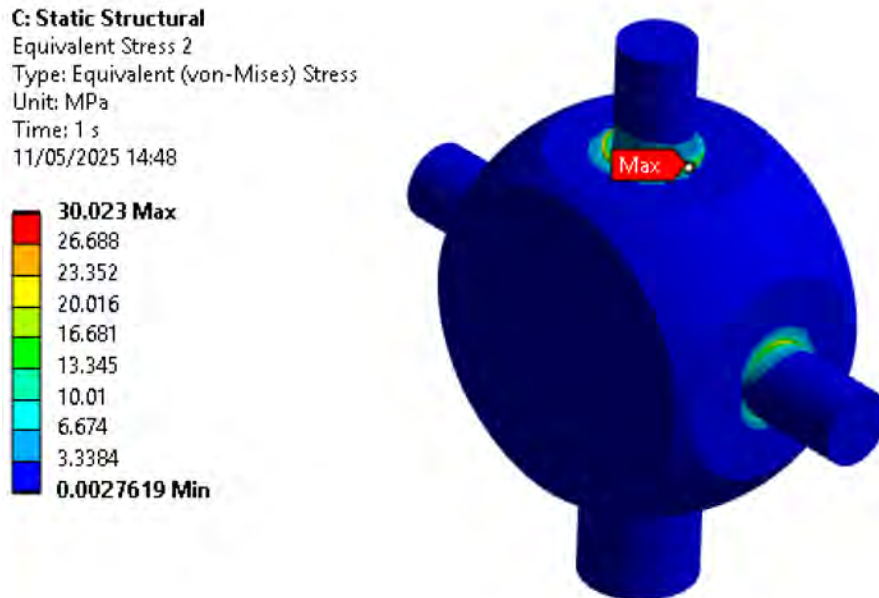


Figura 9.21: Esfuerzos equivalentes en pines internos

9.2.7. Sensor de campo magnético

El sensor CKP (Crankshaft Position Sensor) es un dispositivo diseñado para medir la posición angular y la velocidad de giro del cigüeñal en motores de combustión interna. Este sensor pertenece a la familia de dispositivos basados en principios electromagnéticos o de efecto Hall, lo que significa que utiliza tecnologías de detección magnética o inductiva para realizar sus funciones de medición.

Se describen a continuación algunas características del sensor CKP:

- El sensor CKP generalmente se comunica con la unidad de control del motor (ECU) a través de una señal de voltaje pulsante, la cual es generada por el

movimiento del cigüeñal frente a un rotor dentado o una rueda reluctora. Esta señal permite determinar tanto la posición angular del cigüeñal como la velocidad de rotación del motor.

- Dependiendo del tipo de sensor CKP, puede emplear tecnologías basadas en efecto Hall o en inducción electromagnética. Los sensores de efecto Hall generan pulsos eléctricos al detectar cambios en el campo magnético, mientras que los inductivos generan una señal de voltaje al pasar el rotor frente a una bobina.
- Este tipo de sensor es esencial en motores de combustión interna para garantizar la sincronización de la inyección de combustible y el encendido. Se utiliza comúnmente en vehículos automotrices, sistemas de gestión de motores y aplicaciones donde el monitoreo preciso de la posición del cigüeñal es fundamental.



Figura 9.22: Sensor de posición de cigüeñal CKP.

El sensor CKP magnético es seleccionado para el módulo de pruebas debido a su capacidad para entregar una señal analógica en forma de onda sinusoidal, lo que permite un monitoreo continuo y preciso de la posición angular del eje bajo condiciones de desalineamiento. En este contexto, la señal analógica generada por inducción electromagnética al pasar los dientes de la rueda reluctora frente a la bobina del sensor puede revelar variaciones en la amplitud y frecuencia, las cuales son indicativas del nivel de desalineamiento del eje. Esta característica facilita el análisis detallado de las irregularidades en la rotación del eje, permitiendo detectar y cuantificar distintos grados de desalineamiento.

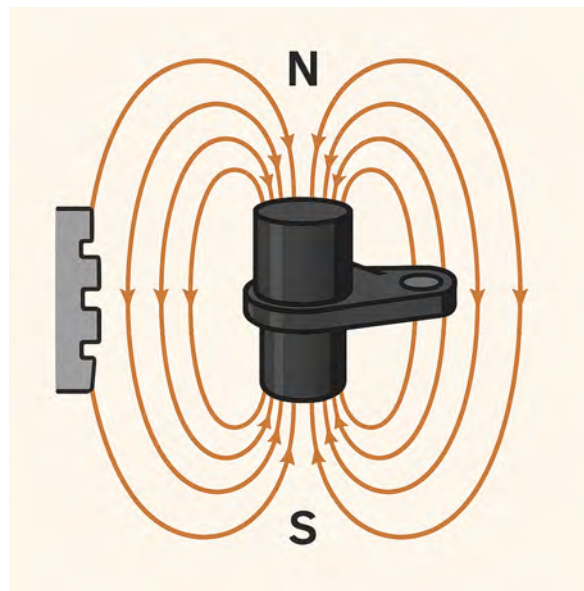


Figura 9.23: Modo de funcionamiento de un sensor CKP.

Para facilitar la generación de señales en el módulo de pruebas, se imprimirá un aditamento diseñado específicamente para simular los dientes del cigüeñal. Este componente incluirá pines metálicos dispuestos a intervalos regulares, los cuales interactuarán con el campo magnético del sensor CKP magnético, generando variaciones en la señal analógica. De esta forma, será posible replicar diferentes grados de desalineamiento al ajustar la posición o separación de los pines, permitiendo un análisis más detallado del comportamiento dinámico del sistema bajo condiciones controladas.

9.2.8. Sistema de adquisición de datos

En lugar de utilizar una unidad de control electrónico (ECU) convencional para la adquisición de la señal del sensor CKP magnético, se implementará una interfaz de audio PreSonus Studio24c, la cual permitirá capturar y digitalizar la señal analógica generada por el sensor. Esta interfaz está equipada con entradas analógicas de alta fidelidad, capaces de recibir señales de bajo voltaje con alta resolución, lo que resulta ideal para registrar la señal sinusoidal generada por el sensor durante la simulación del desalineamiento del eje. Además, su conexión USB facilita la transferencia de datos al sistema de procesamiento para su posterior análisis y almacenamiento. El equipo descrito se muestra en la figura 9.24



Figura 9.24: Interfaz de audio PreSonus Studio 24c Pre (2019)

El análisis de la señal generada por el sensor CKP magnético se realizará capturando primero la señal analógica mediante la interfaz de audio PreSonus Studio24c, que estará conectada al módulo de pruebas. La señal se digitalizará y almacenará en una computadora, donde se utilizará el software Studio One 6 para la captura y edición inicial de los datos. El software seleccionado permitirá visualizar la forma de onda,

aplicar filtros básicos para reducir ruido y exportar los datos en formatos compatibles para análisis posteriores.

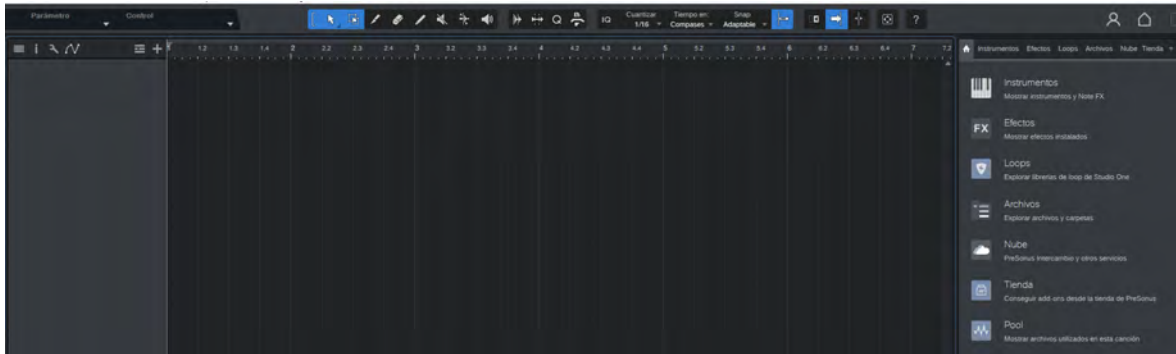


Figura 9.25: Entorno de trabajo de software de captura de audio (Digital Audio Workstation) Studio One.

Posteriormente, los datos procesados serán analizados para identificar variaciones en amplitud, frecuencia y forma de la señal, que son indicativas del nivel de desalineamiento en el eje. Esta metodología facilita un análisis detallado y no invasivo, permitiendo evaluar la calidad de la rotación del eje a partir de la señal capturada sin necesidad de sistemas complejos de adquisición tradicional.

9.2.9. Rodamiento de bolas

Para el soporte del eje del módulo de pruebas se empleó un rodamiento de bolas con chumacera tipo KFL08 por cada bancada. Esta configuración permite sostener el eje de forma estable durante el giro, manteniendo una adecuada alineación del conjunto y reduciendo posibles juegos mecánicos que puedan afectar la medición de las señales de vibración. El rodamiento KFL08 fue seleccionado por su disponibilidad comercial, bajo costo, facilidad de montaje y compatibilidad con el diámetro del eje utilizado en el sistema.

El módulo fue diseñado para trabajar en condiciones de laboratorio y no para transmitir cargas elevadas. Por ello, la selección del rodamiento se realizó principalmente por criterios de estabilidad, montaje y compatibilidad dimensional, más que por una alta exigencia de carga.

De acuerdo con la hoja técnica del rodamiento KFL08, presentada en el Anexo K, este rodamiento tiene un diámetro interior de 8 mm, una capacidad dinámica de carga de $C = 4,2 \text{ kN}$, una capacidad estática de carga de $C_0 = 1,6 \text{ kN}$ y una masa aproximada de 50 g. Además, el catálogo indica que el rodamiento puede compensar pequeños desalineamientos durante el montaje y que se encuentra lubricado de por vida bajo condiciones normales de operación Mädler GmbH (sf).

Los datos considerados para la verificación son los siguientes:

- Velocidad de rotación del eje: 940 rpm.
- Masa aproximada del conjunto eje–rueda fónica: 65 g.
- Carga axial considerada: 0 N.
- Configuración de montaje: un rodamiento por bancada.
- Número total de rodamientos: 2.
- Tipo de operación: intermitente.
- Ambiente de trabajo: laboratorio.
- Lubricación: grasa.

La carga radial principal corresponde al peso del conjunto eje–rueda fónica. Esta carga se calcula mediante:

$$W = mg \tag{9.26}$$

Donde W es el peso del conjunto rotativo, m es la masa del conjunto eje–rueda fónica y g es la aceleración de la gravedad.

Reemplazando:

$$W = 0,065(9,81) = 0,638 \text{ N}$$

Como el sistema utiliza dos bancadas y un rodamiento por bancada, la carga radial se reparte aproximadamente entre dos rodamientos. La carga radial por rodamiento se calcula como:

$$F_{r,\text{rod}} = \frac{W}{N_r} \quad (9.27)$$

donde N_r es el número total de rodamientos que soportan la carga radial.

Para el caso ideal, considerando dos rodamientos:

$$F_{r,\text{rod}} = \frac{0,638}{2} = 0,319 \text{ N}$$

Para realizar una verificación más conservadora, se considera que el peso total del conjunto eje–rueda fónica es soportado por una sola bancada, es decir, por un solo rodamiento:

$$F_{r,\text{rod,cons}} = 0,638 \text{ N}$$

Debido a que no se considera carga axial en el sistema, la carga estática equivalente se calcula mediante:

$$P_0 = F_r + 0,6F_a \quad (9.28)$$

donde P_0 es la carga estática equivalente, F_r es la carga radial y F_a es la carga axial.

Como:

$$F_a = 0 \text{ N}$$

entonces, usando el caso conservador:

$$P_0 = 0,638 + 0,6(0)$$

$$P_0 = 0,638 \text{ N}$$

La capacidad estática del rodamiento KFL08, según el catálogo, es:

$$C_0 = 1,6 \text{ kN} = 1600 \text{ N}$$

El factor de seguridad estático se calcula como:

$$FS_0 = \frac{C_0}{P_0} \quad (9.29)$$

Reemplazando:

$$FS_0 = \frac{1600}{0,638} = 2508$$

Este resultado muestra que la carga aplicada sobre el rodamiento es muy baja en comparación con la capacidad estática indicada por el fabricante.

También se verifica que:

$$P_0 = 0,638 \text{ N} \ll C_0 = 1600 \text{ N}$$

Por tanto, se cumple la condición de diseño estática del rodamiento de acuerdo con el catálogo. Además, como la carga axial considerada es nula, la condición de trabajo corresponde al caso indicado en la hoja técnica para carga radial con fuerza axial igual a cero Mädlar GmbH (sf).

Para la verificación dinámica, al no existir carga axial apreciable, la carga dinámica equivalente puede aproximarse como:

$$P \approx F_r \tag{9.30}$$

donde P es la carga dinámica equivalente.

Por tanto, para el caso conservador:

$$P = 0,638 \text{ N}$$

La capacidad dinámica del rodamiento KFL08, según el catálogo del fabricante, es:

$$C = 4,2 \text{ kN} = 4200 \text{ N}$$

Comparando la carga dinámica equivalente con la capacidad dinámica del rodamiento:

$$P = 0,638 \text{ N} \ll C = 4200 \text{ N}$$

Por lo tanto, también se cumple la condición de diseño dinámica. La carga de trabajo es mucho menor que la capacidad nominal del rodamiento.

La vida nominal básica de un rodamiento de bolas se puede estimar mediante:

$$L_{10h} = \frac{10^6}{60n} \left(\frac{C}{P} \right)^3 \quad (9.31)$$

donde L_{10h} es la vida nominal en horas, C es la capacidad dinámica, P es la carga dinámica equivalente y n es la velocidad de giro en rpm. Para rodamientos de bolas se utiliza el exponente $p = 3$ SKF (2025).

Reemplazando:

$$C = 4200 \text{ N}$$

$$P = 0,638 \text{ N}$$

$$n = 940 \text{ rpm}$$

se obtiene:

$$L_{10h} = \frac{10^6}{60(940)} \left(\frac{4200}{0,638} \right)^3$$

$$L_{10h} \approx 5,06 \times 10^{12} \text{ h}$$

El valor obtenido es muy alto debido a que la carga aplicada es pequeña en comparación con la capacidad del rodamiento. Por ello, en este módulo la vida del rodamiento no está limitada por la carga radial, sino por factores como montaje, lubricación, limpieza, balanceo del eje y condiciones reales de operación.

La velocidad de operación del eje es:

$$n = 940 \text{ rpm}$$

Esta velocidad es baja para una aplicación de laboratorio con este tipo de rodamiento. Por tanto, el rodamiento KFL08 trabaja dentro de condiciones adecuadas para el módulo de pruebas.

El rodamiento KFL08 cumple con las condiciones de diseño de acuerdo con los datos del catálogo. Las cargas estáticas y dinámicas calculadas son mucho menores que las capacidades nominales del rodamiento. Por ello, su selección es adecuada para el módulo, ya que permite un giro estable del eje, es compatible con el diámetro de 8 mm, facilita el montaje y soporta sin problema las cargas presentes durante la medición Mädler GmbH (sf).

La Figura 9.26 muestra el rodamiento de bolas con chumacera utilizado en el módulo de pruebas. La hoja técnica completa del rodamiento KFL08 se presenta en el Anexo K.

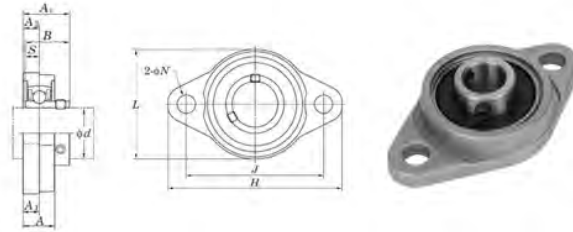


Figura 9.26: Rodamiento de bolas con chumacera tipo KFL08 utilizado en el módulo de pruebas.

9.3. Análisis por Elementos Finitos del módulo de pruebas

El análisis se realizará en el laboratorio de la escuela profesional de Ingeniería Mecánica de la Universidad Nacional de San Antonio Abad del Cusco y se utilizará el software ANSYS versión 19.2.

9.3.1. Propiedades de la geometría

- Masa total del equipo: 8 Kg.
- Material de fabricación: Acero estructural ASTM A-36.

9.3.2. Modelamiento tridimensional

El modelo numérico desarrollado para la simulación por el método de los elementos finitos se basa en una representación tridimensional del módulo de pruebas. La

geometría fue discretizada utilizando elementos sólidos tridimensionales, los cuales permiten describir de manera adecuada el comportamiento estructural del sistema en las tres direcciones del espacio. Este tipo de modelamiento posibilita la correcta captura de los efectos de rigidez, deformación y distribución de esfuerzos en componentes con geometría compleja, garantizando una representación fiel de las condiciones físicas del módulo bajo las cargas y restricciones consideradas en el análisis.

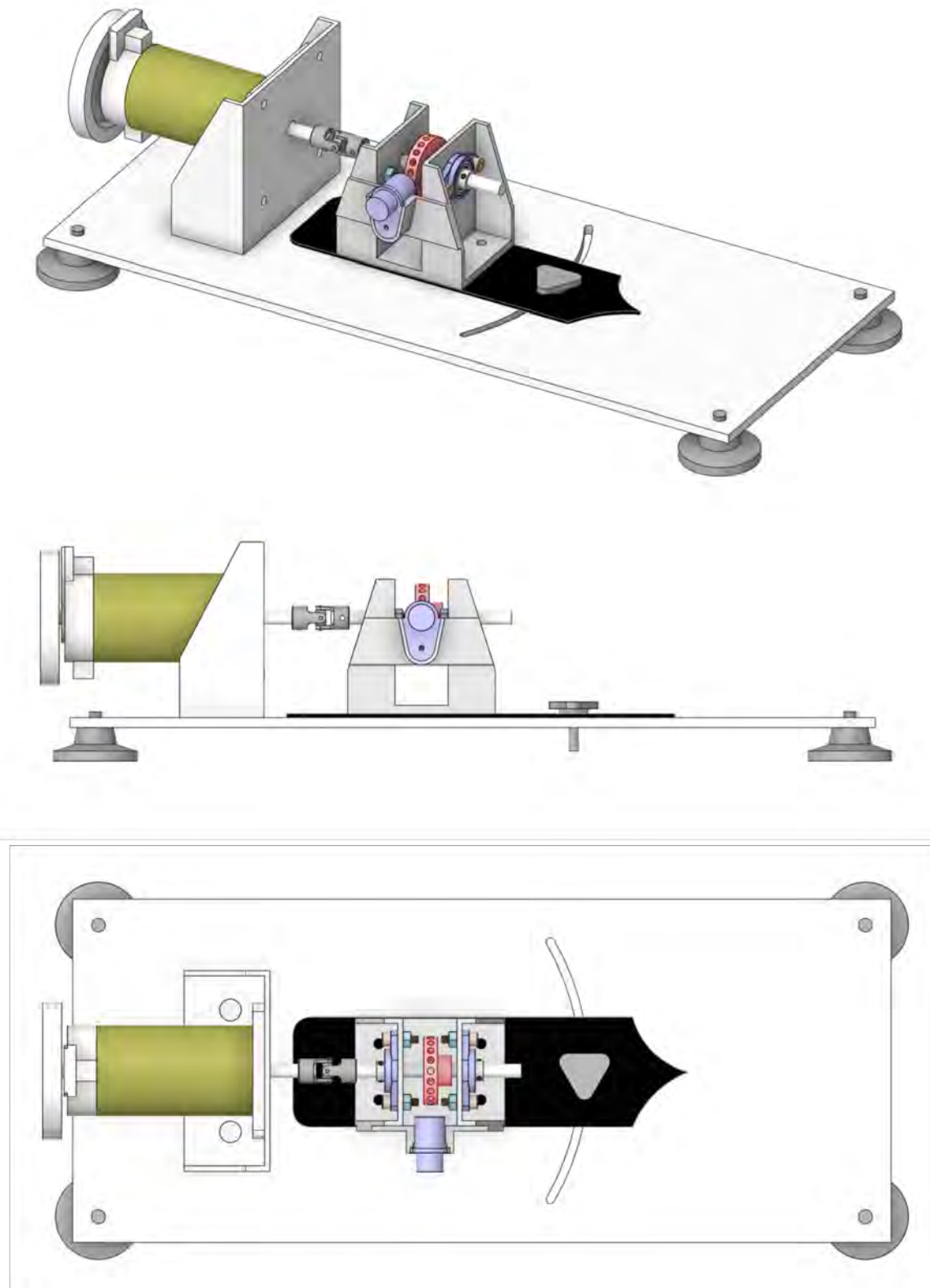


Figura 9.27: Modelo tridimensional del módulo de pruebas.

9.3.3. Mallado

Se realizó un procedimiento de mallado selectivo a fin de aproximar la geometría real con el modelo simulado de manera eficiente y con un costo computacional bajo. Se utilizó una malla de 3 mm (que corresponde al espesor de las placas plegadas) de tipo tetraédrico en los soportes de motor y chumacera. También se utilizó una malla de inicial de 10mm de tipo hexaédrico en la placa de soporte, ya que es el elemento más flexible y, por tanto, el que tiene mayor incidencia en las frecuencias naturales del sistema.

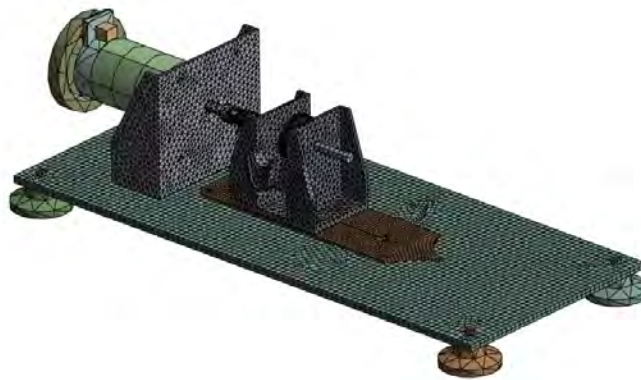


Figura 9.28: Modelo mallado.

Se utilizó la herramienta de visualización de calidad del software y el criterio de ortogonalidad de la malla. Se tienen un total de 76110 nodos y 38635 elementos. El criterio de evaluación de ortogonalidad tiene como valor más óptimo la unidad, como se muestra en la Figura 9.29 los elementos de malla se distribuyen alrededor de la calidad ortogonal de 0.65, por lo cual un refinamiento es necesario.

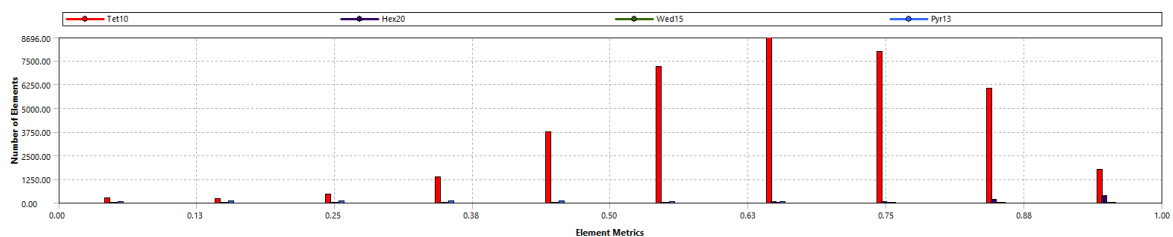


Figura 9.29: Calidad inicial de malla.

9.3.4. Análisis modal

Se realizó un análisis paramétrico modal en el módulo de pruebas, esto con la finalidad de tener un panorama general de los modos de vibración y frecuencias naturales con tal de que no se presenten condiciones dinámicas de pulsaciones o resonancia que puedan interferir con la calidad de las mediciones o la integridad estructural del sistema.

El análisis paramétrico de sensibilidad o convergencia de malla tiene como objetivo evaluar cómo varían los resultados obtenidos mediante simulación por elementos finitos al modificar el tamaño y la calidad de la malla. Esto se debe a la naturaleza aproximada del método de elementos finitos, donde la precisión de los resultados depende en gran medida de la discretización del dominio.

Para ello se consideró la placa base del módulo como elemento de estudio, ya que es la que soporta todos los equipos y aporta mayor rigidez al sistema. La placa base tiene un espesor de 7 mm y para lo cual se eligió un tamaño de malla inicial de 10 mm como parámetro de referencia para una malla gruesa la cual se refinará. Se consideró una malla de tipo hexaédrica cuadrática para el estudio.

El proceso de refinamiento de malla se muestra en la tabla 9.5:

Cuadro 9.5: Refinamiento de malla.

Tamaño de elementos (mm)	Mesh nodes	Mesh elements
10	77392	38396
8	80343	39012
6	87006	40997
5	119200	49816

Se llevó a cabo un análisis modal, siendo los valores de las 6 primeras frecuencias naturales los que se estudiaron.

La Figura 9.30 muestra el estudio de sensibilidad de malla.

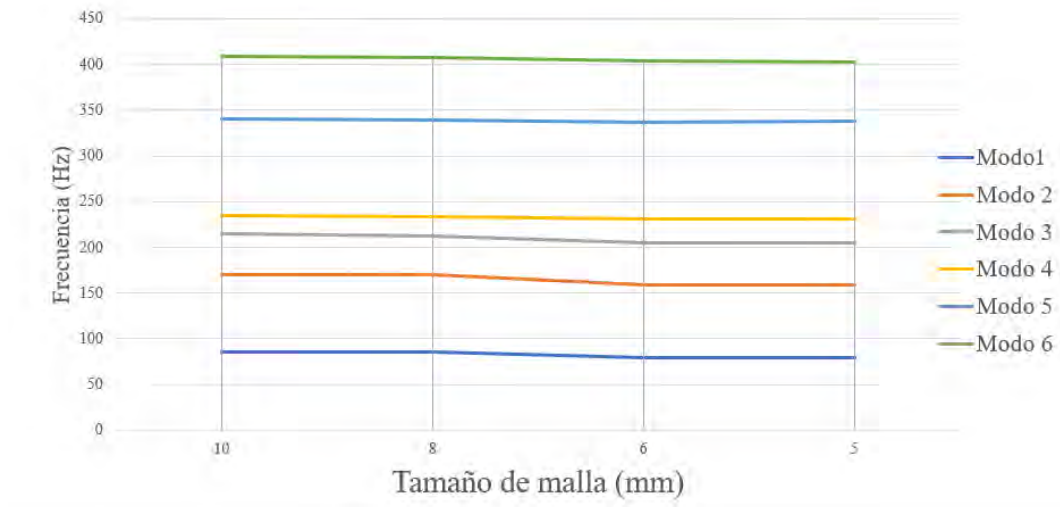


Figura 9.30: Convergencia de malla.

Los resultados del estudio de independencia de malla muestran que las frecuencias naturales de los modos 1 al 6 presentan una tendencia clara a la convergencia a medida que se reduce el tamaño del elemento. En particular, para los modos inferiores (1 a 3), las variaciones entre mallas sucesivas disminuyen progresivamente, evidenciando una adecuada estabilidad numérica del modelo.

A partir de un tamaño de elemento de 6 mm, las diferencias en las frecuencias calculadas para todos los modos analizados se vuelven reducidas, especialmente en el modo 1, cuya frecuencia converge hacia un valor cercano a 80 Hz, en concordancia con la estimación analítica previa. La diferencia relativa entre las mallas de 6 mm y 5 mm resulta mínima, lo que indica que el refinamiento adicional tiene un impacto limitado en la precisión de los resultados.

Los modos intermedios (modos 4 a 6) presentan una mayor sensibilidad al tamaño de malla en comparación con los modos fundamentales; sin embargo, esta sensibilidad disminuye significativamente para tamaños de elemento iguales o menores a 6 mm. Este comportamiento es característico de modos de mayor frecuencia, los cuales requieren discretizaciones más finas debido a la mayor complejidad espacial de sus formas modales.

Primer modo de vibración: En la Figura 9.33, movimiento predominante de la placa base en el plano vertical.

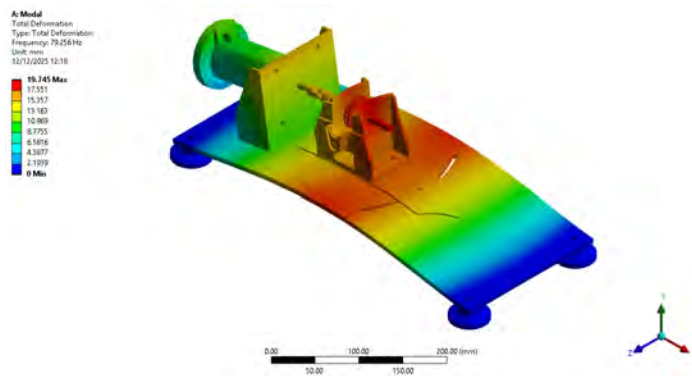


Figura 9.31: Primer modo de vibración.

Segundo modo de vibración: En la Figura 9.32, rotación de la placa base.

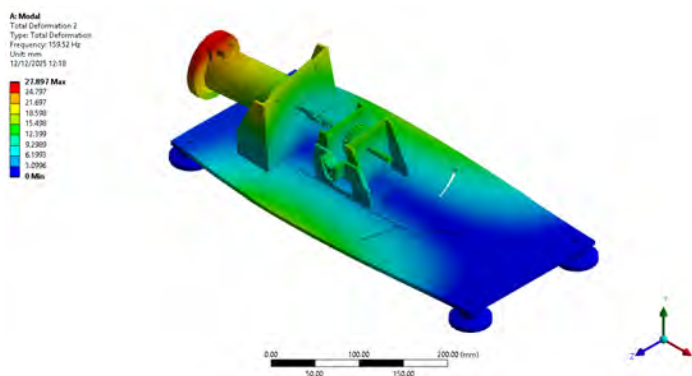


Figura 9.32: Segundo modo de vibración.

Tercer modo de vibración: En la Figura 9.33, movimiento del motor en sentido vertical.

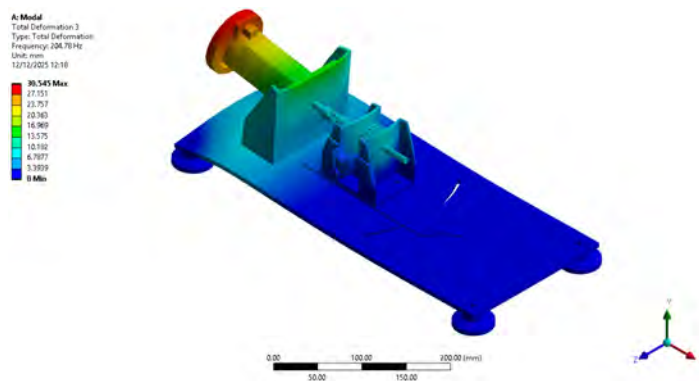


Figura 9.33: Tercer modo de vibración.

Cuarto modo de vibración: En la Figura 9.34, primer armónico de la placa base en sentido vertical.

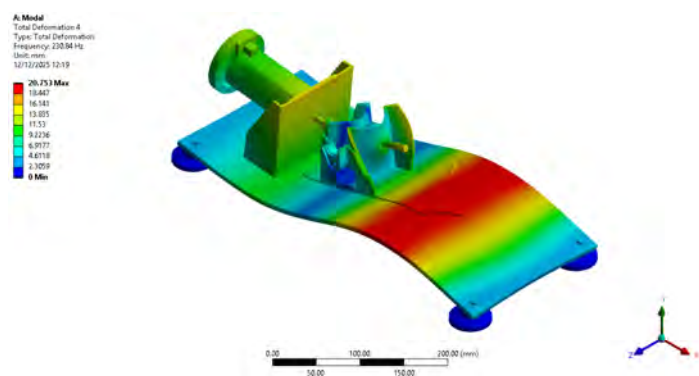


Figura 9.34: Cuarto modo de vibración.

Quinto modo de vibración: En la Figura 9.35, rotación de la placa base en sentido axial.

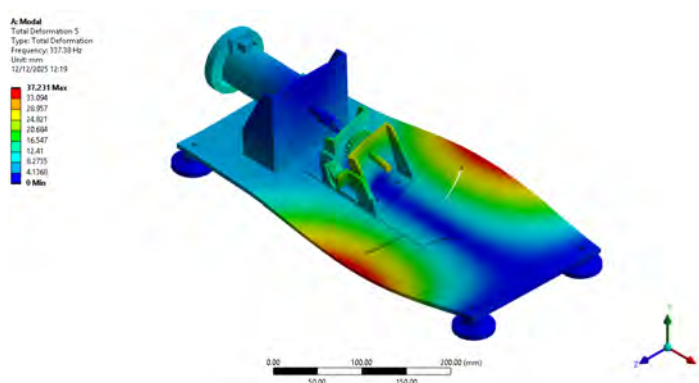


Figura 9.35: Quinto modo de vibración.

Sexto modo de vibración: En la Figura 9.36, segundo armónico de la placa base en sentido vertical.

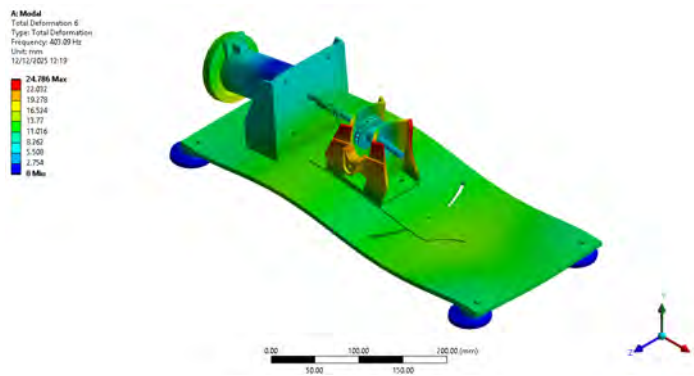


Figura 9.36: Sexto modo de vibración.

Se resumen las frecuencias +de acuerdo a cada modo en el Cuadro 9.6.

Cuadro 9.6: Frecuencias naturales.

Modo	Frecuencia	
	Hz	CPM
1	79.256	4755
2	159.52	9571
3	204.78	12286
4	230.84	13850
5	337.38	20242
6	403.09	24185

La primera frecuencia natural obtenida mediante el análisis modal numérico presenta una buena concordancia con el valor calculado de forma analítica mediante el método de Rayleigh, el cual resulta $f_1 \approx 72$ Hz. Este valor se encuentra en el mismo orden de magnitud que el obtenido numéricamente, lo que indica que las hipótesis empleadas en el modelo analítico describen de manera adecuada el comportamiento dinámico global del sistema y validan su uso como una estimación preliminar confiable.

Asimismo, las frecuencias naturales del sistema se encuentran suficientemente alejadas de la frecuencia de operación del motor, el cual gira aproximadamente a 940 rpm, equivalente a una frecuencia de excitación de 15,67 Hz. Esta separación significativa reduce el riesgo de resonancia durante la operación normal, por lo que el comportamiento dinámico del sistema puede estable bajo las condiciones de funcionamiento analizadas.

9.4. Componentes del equipo

La Figura 9.37 muestra la vista explosionada del módulo de pruebas desarrollado. En esta representación se identifican los principales elementos mecánicos que conforman el sistema, numerados de acuerdo con la lista de materiales presentada en el Cuadro 9.7. Esta vista permite observar la ubicación relativa de cada componente antes del ensamblaje final.

El módulo está compuesto por una placa base principal sobre la cual se montan los soportes de goma, el soporte del motor, el sistema de transmisión, el mecanismo de regulación angular y los elementos de medición. El motor eléctrico se acopla al eje mediante una cruceta tipo cardán, la cual permite generar la condición de desalineamiento angular durante los ensayos. En el extremo del sistema se ubican el eje de transmisión, los rodamientos KFL de 8 mm, la rueda fónica y el sensor CKP inductivo, empleados para obtener la referencia del movimiento rotacional.

Asimismo, se observa el conjunto de placas y elementos de ajuste que permiten regular el ángulo de desalineamiento. La placa indicadora, el dial de regulación y la perilla con espárrago permiten fijar diferentes posiciones angulares de forma controlada. Los pernos, tuercas, arandelas y separadores aseguran la unión de los componentes y permiten mantener la rigidez del conjunto durante la operación.

En conjunto, la vista explosionada facilita la comprensión del orden de montaje del módulo, la función de cada componente y la relación entre el sistema de accionamiento, el sistema de soporte del eje, el sistema de regulación y el sistema de medición. Esta representación es útil para verificar que todos los elementos considerados en el diseño estén correctamente identificados antes del proceso de fabricación y ensamblaje.

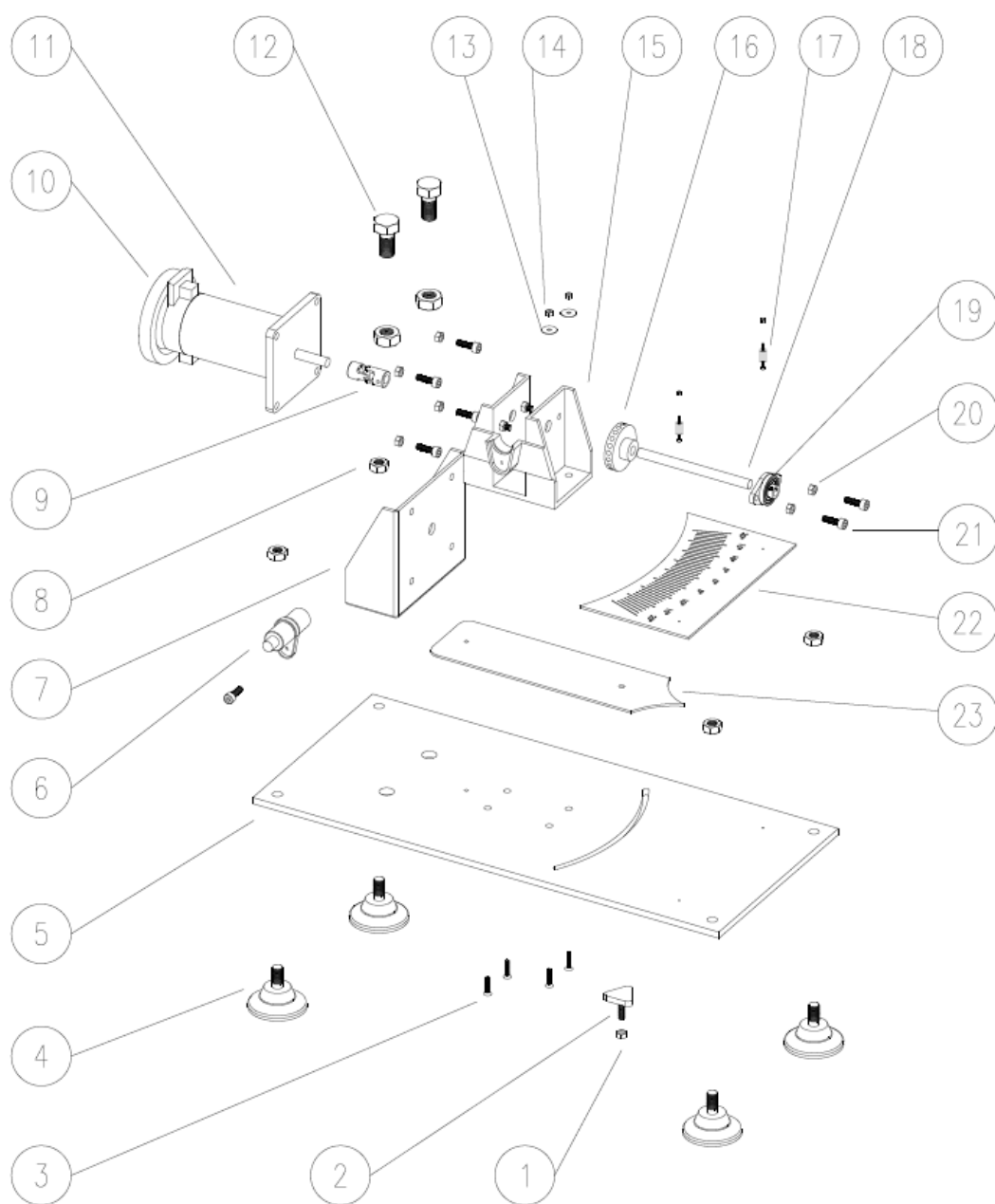


Figura 9.37: Vista de explosión del módulo de pruebas con identificación de componentes.

Cuadro 9.7: Lista de materiales del módulo de pruebas

N.º	Descripción	Cant.	Norma	Material	Long./Dim.	Masa/Peso
23	PL. N°03, e = 2 mm	1	ASTM	A-36	180.0	230.0
22	PL. N°02, e = 2 mm	1	ASTM	D-3935	180.0	30.0
21	PERNO, M4 × 12 mm	9	ASTM	A-36	15.0	3.5
20	TUERCA, M4	9	ASTM	A-36	4.0	1.2
19	RODAMIENTO KLF, = 8 mm	2	ASTM	B-209	90.0	42.3
18	EJE l = 90 mm, = 8 mm	1	ASTM	A-276	90.0	42.3
17	SEPARADOR HM23	2	ASTM	B-124	10.0	3.0
16	RUEDA FÓNICA	1	ISO-1043	PLA	40.0	11.0
15	SOPORTE DE SENSOR	1	ISO-1043	PLA	77.0	79.1
14	TUERCA, M3	2	ASTM	A-36	3.0	0.7
13	ARANDELA, M3	2	ASTM	A-36	8.0	0.8
12	PERNO l = $\frac{25}{32}$ " , = $\frac{3}{8}$ "	2	ASTM	A-36	30.0	24.0
11	MOTOR ELÉCTRICO YA-010	1	-	-	111.0	1579.0
10	VOLANTE DE INERCIA	1	ASTM	A-36	80.0	286.6
9	CRUCETA TIPO CARDÁN	1	ASTM	A-36	35.0	24.0
8	TUERCA = $\frac{3}{8}$ "	6	ASTM	A-36	8.0	8.0
7	SOPORTE MOTOR, e = $\frac{3}{32}$ "	1	ASTM	A-36	1000.0	560.1
6	SENSOR CKP INDUCTIVO	1	ASTM	D-1784	63.0	38.0
5	PL. N°01, e = $\frac{1}{4}$ "	1	ASTM	A-36	920.0	4181.0
4	SOPORTE GOMA CON ESPÁRRAGO = $\frac{3}{8}$ "	4	ASTM	A-36	40.0	40.0
3	PERNO AVELLANADO M3	4	ASTM	A-36	10.00	3.0
2	PERILLA CON ESPÁRRAGO	1	ASTM	A-36	25.00	20.0
1	TUERCA HEXAGONAL M4	1	ASTM	A-36	2.00	1.0

10 | CAPTURA DE SEÑALES DE VIBRACIÓN

La captura de señales es una etapa fundamental en la presente investigación, ya que a partir de ella se obtiene la información experimental necesaria para el análisis del desalineamiento angular en ejes. En este capítulo se describe la metodología empleada para la adquisición de datos durante los ensayos realizados en el módulo de pruebas, así como la forma en que se organizaron las mediciones para generar una base de datos útil para las etapas posteriores de procesamiento y entrenamiento del modelo.

Asimismo, se presenta la selección de los elementos electrónicos y mecánicos involucrados en la medición, considerando su función dentro del sistema de adquisición y su aporte a la obtención de señales consistentes bajo condiciones controladas de ensayo. La captura de datos fue planteada para registrar la respuesta del sistema en diferentes niveles de desalineamiento angular, con incrementos de 1° en 1° , a fin de contar con una representación más detallada del fenómeno estudiado.

De manera complementaria, en este capítulo se describe el procedimiento seguido para la captura de datos durante los ensayos, iniciando desde la preparación del módulo de pruebas y el ajuste del nivel de desalineamiento angular, hasta la adquisición y almacenamiento de las señales obtenidas por el sensor. Este procedimiento permitió realizar mediciones de manera ordenada y repetitiva para cada condición evaluada,

asegurando la generación de una base de datos experimental consistente para las etapas posteriores de procesamiento y análisis.

10.1. Parámetros de operación generales

El motor eléctrico de corriente continua opera con un voltaje de alimentación de 5 a 24 voltios, con lo que se obtiene una frecuencia de giro medida experimentalmente en el eje en el rango de los 880 RPM - 1822 RPM.



Figura 10.1: Medición de frecuencia de operación con tacómetro digital.

De este parámetro se derivan las frecuencias que se generan a medida que los distintos elementos que conforman el sistema interactúan entre sí.

10.2. Selección de frecuencia de muestreo según teorema de Nyquist

En el contexto de la adquisición de señales para el análisis del comportamiento dinámico de sistemas rotativos, como el que se aborda en la investigación, es fundamental garantizar que las señales registradas conserven toda la información relevante del fenómeno físico medido. Para ello, se debe cumplir el teorema de muestreo de Nyquist-Shannon, el cual establece que una señal analógica puede ser reconstruida completamente a partir de sus muestras digitales siempre que la frecuencia de muestreo sea, como mínimo, el doble de la mayor frecuencia presente en la señal original.

Este principio es crucial para evitar el fenómeno de Aliasing, en el que componentes de alta frecuencia se superponen con los de baja frecuencia, distorsionando el análisis y generando resultados incorrectos como se muestra en la Figura 10.2. En el caso del presente estudio, donde se analizan señales generadas por un sensor CKP en función de la rotación del eje, se estima que los componentes de interés se encuentran dentro del rango de 20 Hz a 20 kHz. Por tanto, una frecuencia de muestreo de 44100 muestras por segundo, como la que ofrece la interfaz de audio Presonus Studio 24c, resulta adecuada para cumplir con el criterio de Nyquist y garantizar una digitalización fiel de las señales. Las especificaciones técnicas de la interfaz de audio se muestran en la Figura 10.3.

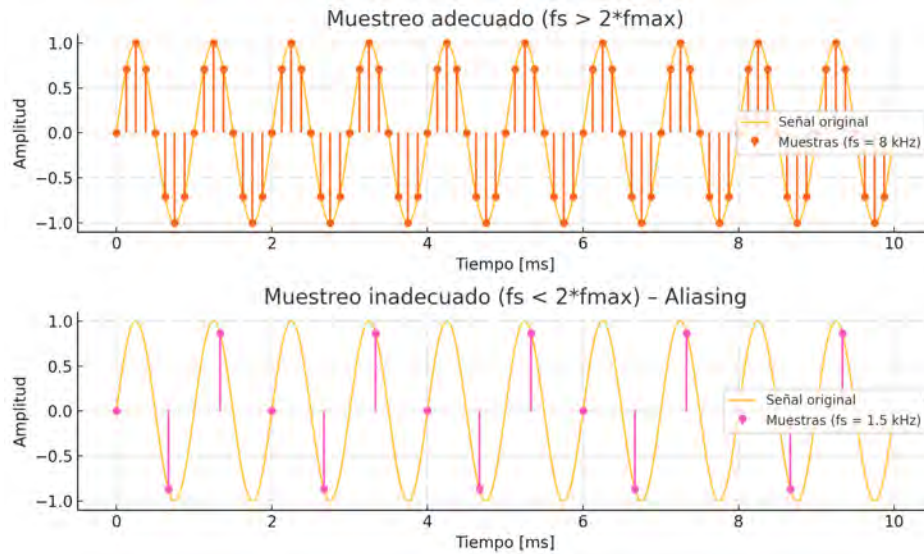


Figura 10.2: Principio de Aliasing.

5 Technical Information	
5.1 Specifications	
General	
Sample Rates	44.1, 48, 88.2, 96, 176.4, or 192 kHz
Converter Resolution	24 bit
Converter Dynamic Range	108 dB
Converter Dynamic Range	108 dB

Figura 10.3: Frecuencia de muestreo de interfaz de audio PreSonus Studio 24cPre (2019)

Observando el comportamiento dinámico del sistema bajo diferentes condiciones de operación y las capacidades del sistema de adquisición de datos, se realizarán pruebas experimentales variando la frecuencia de funcionamiento. Esta variación se logrará modificando el número de pines metálicos en las ruedas fónicas acopladas al eje. El número de pines determina la frecuencia de la señal generada por el sensor CKP, lo cual permite simular distintos escenarios de velocidad angular sin necesidad de alterar la configuración mecánica completa del sistema.

Para la captura de señales representativas del comportamiento dinámico del eje en estudio, se implementó un montaje experimental que incluye un sensor tipo CKP

magnético (Crankshaft Position Sensor) orientado a detectar los pulsos generados por una rueda fónica acoplada al eje de salida. El sensor se fijó de manera rígida a una estructura sólida mediante un soporte metálico, asegurando una distancia constante entre el extremo del sensor y los pines metálicos de la rueda. Esta disposición garantiza la estabilidad de la señal capturada y minimiza las interferencias mecánicas. La rueda fónica se diseñó con configuraciones intercambiables de 4 y 8 dientes, lo que permite modificar la frecuencia de la señal generada sin alterar la velocidad de rotación, facilitando así el estudio del sistema bajo diferentes condiciones de frecuencia angular. La señal generada por el sensor CKP fue posteriormente digitalizada mediante una interfaz de audio de alta resolución para su posterior análisis.

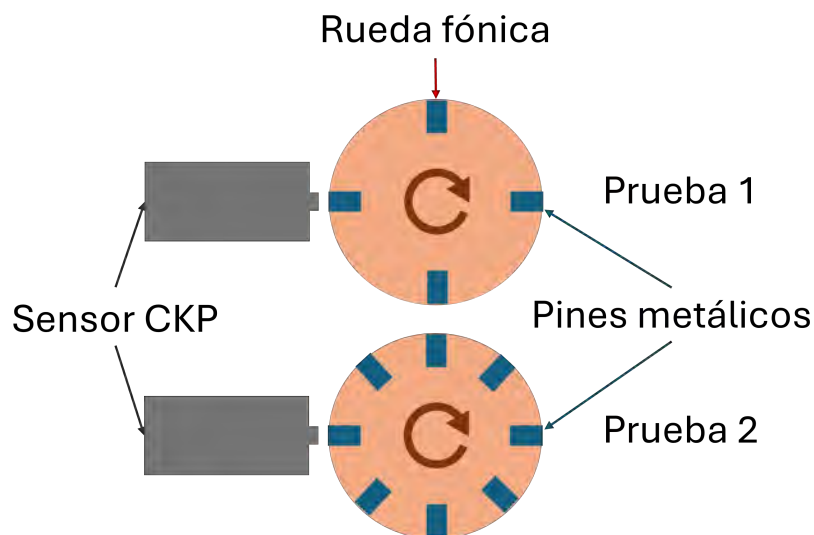


Figura 10.4: Arreglo general de las diferentes pruebas realizadas.

En caso de que los requerimientos experimentales así lo exijan, también se procederá a ajustar directamente la velocidad de rotación del motor. Esta alternativa permite ampliar el rango de frecuencias analizadas y verificar la respuesta del sistema ante variaciones más amplias de régimen dinámico. De esta forma, se garantiza un conjunto de pruebas más completo y representativo del comportamiento real de la maquinaria rotativa.

10.3. Diseño y desarrollo experimental

En este capítulo se describe el diseño experimental implementado para el estudio del desalineamiento en ejes y la obtención ordenada de datos de vibración bajo condiciones controladas. A diferencia de los capítulos posteriores, en los que se desarrolla con mayor detalle la captura de señales, el procesamiento de datos y la aplicación de técnicas de inteligencia artificial, en esta sección se presenta la lógica experimental que permitió generar un conjunto de datos consistente, trazable y representativo de los distintos niveles de desalineamiento evaluados.

De manera general, la secuencia metodológica del estudio puede apreciarse en la Figura 10.5, donde se resume el flujo completo seguido desde la definición de las condiciones de ensayo hasta la obtención del modelo final. Asimismo, para facilitar la interpretación visual del procedimiento, la Figura 10.6 muestra la etapa correspondiente a la adquisición experimental, la segmentación de señales y la extracción de características, mientras que la Figura 10.7 presenta la organización del conjunto de datos, el entrenamiento de los algoritmos y la evaluación del modelo desarrollado.

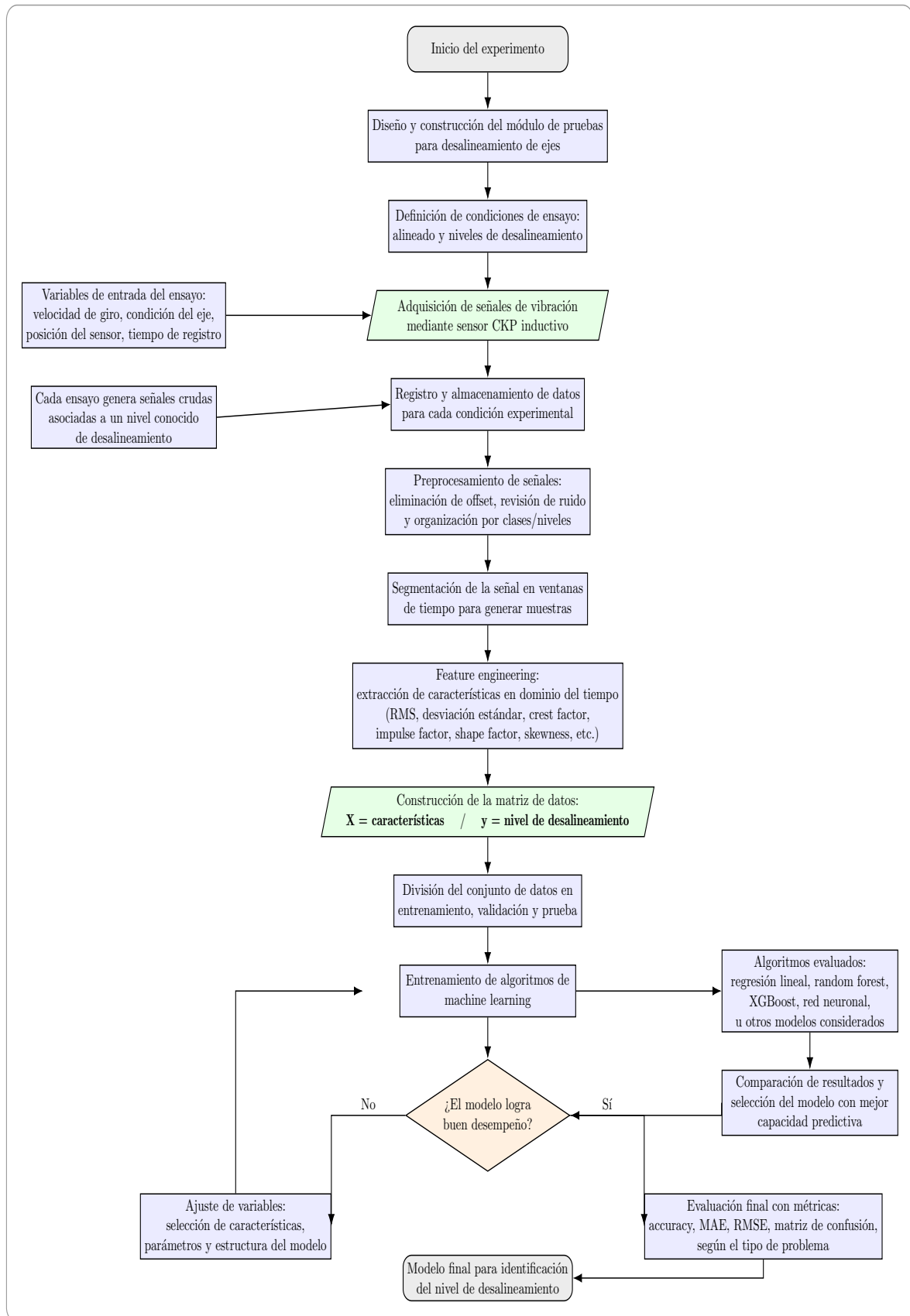


Figura 10.5: Diagrama de flujo de la metodología experimental y computacional empleadas.

El desarrollo experimental se planteó con el propósito de reproducir de manera controlada diferentes estados de desalineamiento angular en un sistema rotativo, de modo que cada condición pudiera asociarse a una respuesta específica. Para ello, se definieron niveles de desalineamiento conocidos, se estableció una secuencia ordenada de ensayos y se controlaron las principales variables involucradas en la medición, con el fin de minimizar fuentes externas de variabilidad y asegurar la comparabilidad entre pruebas.

10.3.1. Planteamiento experimental

El desarrollo experimental de la presente investigación se basó en la necesidad de obtener señales de vibración asociadas a distintos niveles de desalineamiento angular en ejes, bajo condiciones de operación controladas. Para ello, se diseñó un procedimiento experimental que permitiera modificar de manera progresiva el desalineamiento del sistema y registrar la respuesta dinámica correspondiente en cada estado.

El enfoque adoptado fue de tipo experimental, debido a que se intervino directamente sobre la variable de interés, correspondiente al ángulo de desalineamiento, con la finalidad de observar su influencia sobre las señales medidas. De esta manera, fue posible construir un conjunto de datos etiquetado, en el que cada registro quedó asociado a una condición conocida del sistema mecánico.

La estrategia general del estudio consistió en partir de una condición alineada de referencia y, a continuación, introducir variaciones graduales del desalineamiento angular hasta alcanzar el valor máximo definido para el ensayo. Esta lógica de trabajo puede apreciarse de forma resumida en la Figura 10.5, y de forma más visual en la Figura 10.6, donde se observa la relación entre cada nivel de desalineamiento, la señal adquirida y su posterior segmentación en ventanas temporales.

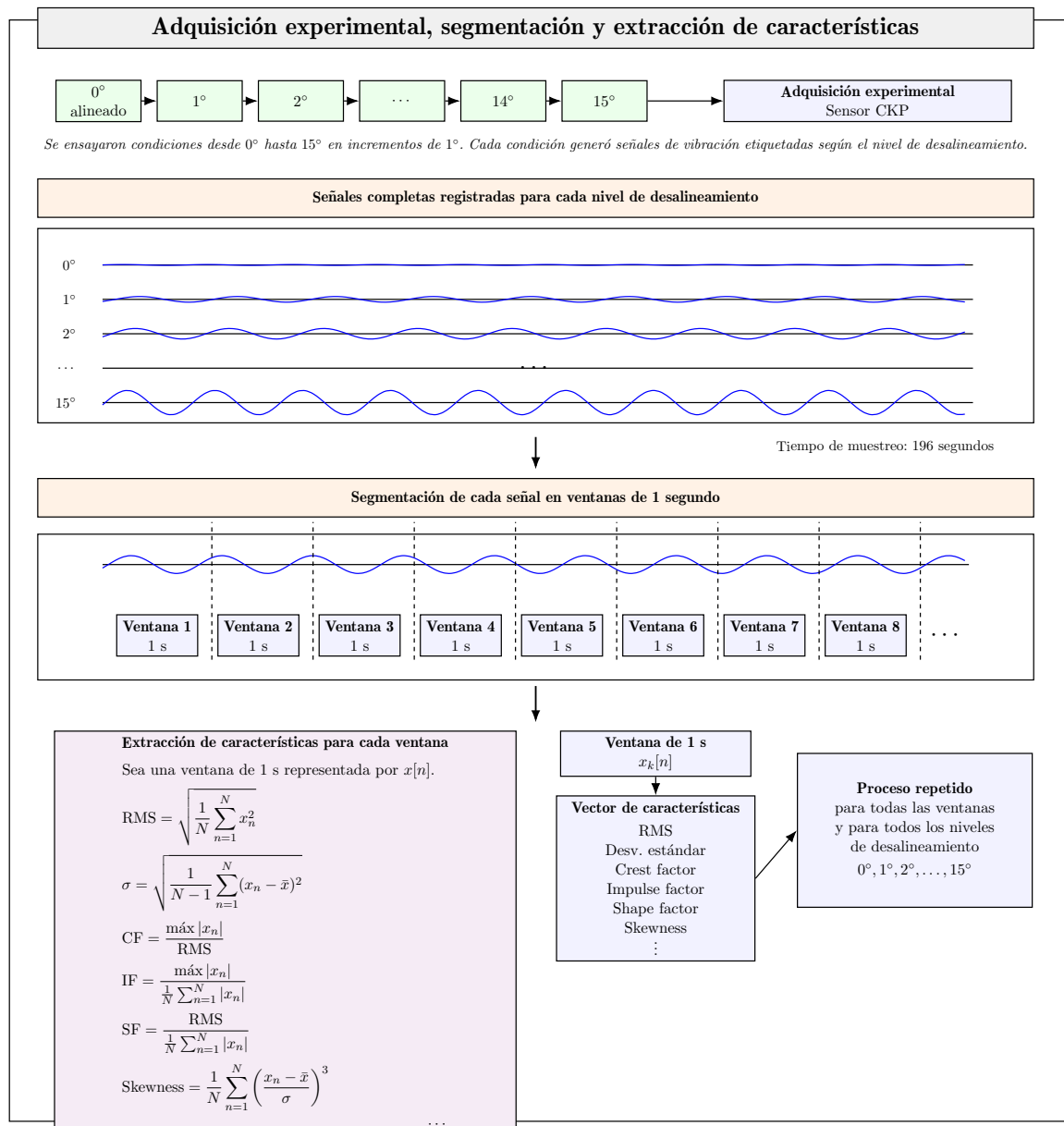


Figura 10.6: Resumen visual de la adquisición experimental, la generación de ventanas de 1 segundo y la extracción de características a partir de las señales de vibración medidas para cada nivel de desalineamiento.

10.3.2. Definición de los niveles de desalineamiento

Con el fin de analizar la evolución de la respuesta vibratoria frente al incremento del desalineamiento, se definieron distintos estados del sistema comprendidos entre la condición alineada y un valor máximo de desalineamiento angular previamente estable-

cido. La condición de referencia correspondió a un alineamiento nominal de 0° , mientras que las condiciones restantes fueron generadas mediante incrementos sucesivos de 1° hasta alcanzar 15° .

La selección de incrementos de un grado respondió a la necesidad de disponer de una resolución suficientemente fina para observar cambios graduales en el comportamiento vibratorio del sistema. Esta decisión permitió evaluar no solo diferencias marcadas entre condiciones extremas, sino también variaciones intermedias que resultan relevantes desde el punto de vista del diagnóstico.

Cada nivel de desalineamiento fue tratado como una condición experimental independiente. En consecuencia, las señales obtenidas en cada ensayo pudieron asociarse de manera directa con el valor angular correspondiente, lo cual se representa de manera esquemática en la Figura 10.6. Esta asociación entre condición mecánica y señal registrada constituyó la base para la estructuración posterior del conjunto de datos.

10.3.3. Procedimiento general de ensayo

El procedimiento experimental se inició con la preparación del sistema en condición alineada, la verificación del montaje y la comprobación del funcionamiento general del banco de pruebas. Una vez establecida la condición de referencia, se realizaron las mediciones correspondientes y posteriormente se introdujo de manera progresiva el desalineamiento angular, siguiendo la secuencia definida para el estudio.

Para cada nivel de desalineamiento se ejecutó un ensayo individual, durante el cual se registró la respuesta vibratoria del sistema durante un intervalo de tiempo suficiente para garantizar la obtención de datos representativos. Finalizada una condición de prueba, el sistema era reajustado al siguiente nivel angular y se repetía el mismo procedimiento de medición.

Tal como se observa en la Figura 10.6, cada señal completa obtenida durante el ensayo fue posteriormente dividida en ventanas temporales de un segundo, con el objetivo de generar múltiples muestras a partir de un mismo registro. Esta estrategia permitió mejorar la organización del conjunto de datos y preparar la información para las etapas posteriores de extracción de características y análisis.

10.3.4. Estructuración del conjunto de datos experimental

Como resultado del desarrollo experimental, se obtuvo un conjunto de señales de vibración correspondientes a múltiples condiciones de desalineamiento angular. La organización de esta información fue planteada desde el inicio del estudio con un criterio de trazabilidad, de modo que cada registro pudiera identificarse de acuerdo con la condición de ensayo bajo la cual fue generado.

La base experimental quedó conformada por señales etiquetadas según el nivel de desalineamiento aplicado, lo que permitió disponer de datos ordenados y comparables entre sí. Esta estructura resultó especialmente importante para las etapas posteriores del trabajo, puesto que hizo posible relacionar de manera explícita cada respuesta vibratoria con su respectiva condición mecánica.

Posteriormente, a partir de las ventanas obtenidas de cada señal, se generaron vectores de características que fueron organizados en una matriz de datos, tal como se muestra en la Figura 10.7. En dicha matriz, cada fila representa una muestra individual y cada etiqueta corresponde al nivel de desalineamiento asociado. Esta organización permitió establecer una base de datos adecuada para el entrenamiento, validación y prueba de los modelos estudiados.

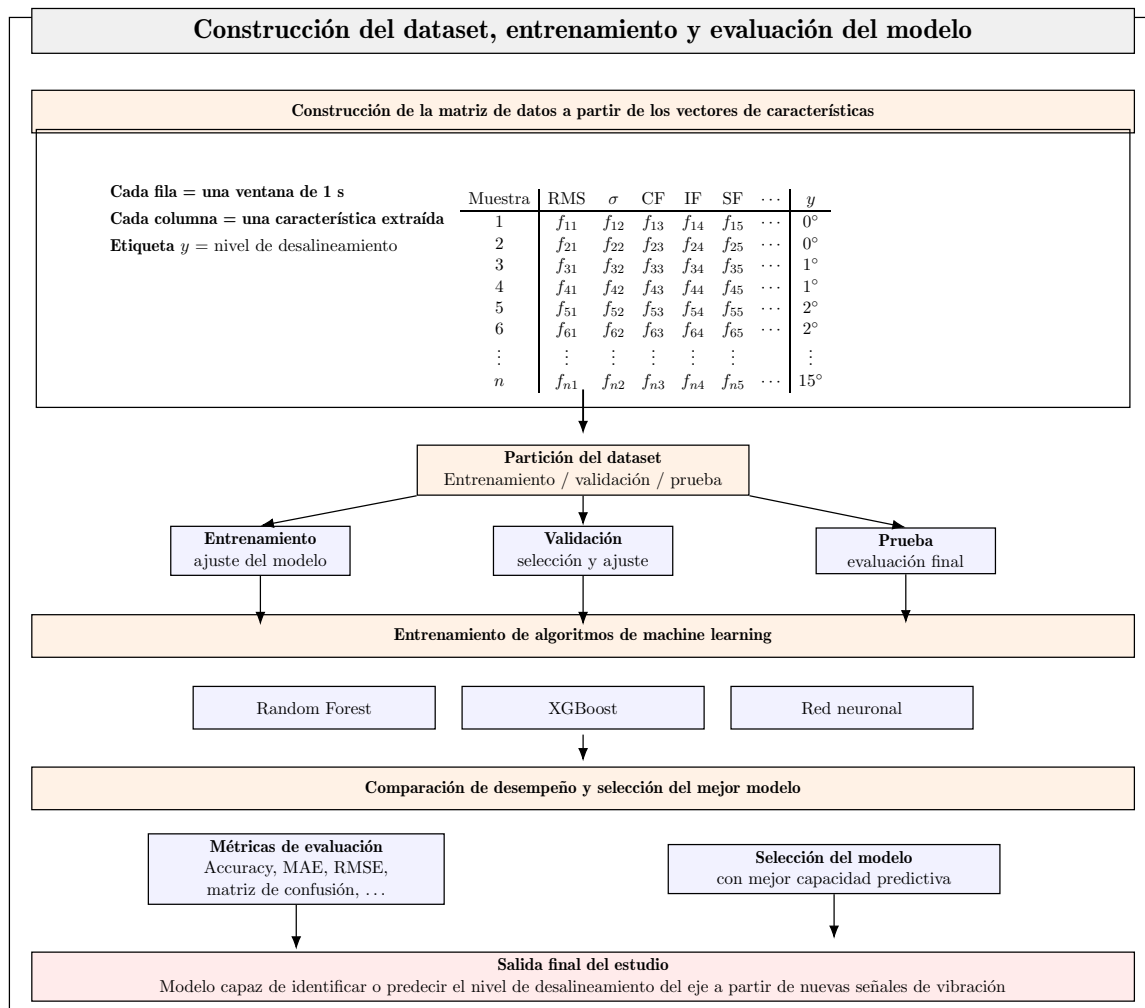


Figura 10.7: Resumen visual de la construcción del conjunto de datos, la partición en subconjuntos, el entrenamiento de los algoritmos y la evaluación del modelo final para la identificación del nivel de desalineamiento.

10.4. Captura de señales de vibración

Para ilustrar visualmente los principios de montaje que servirán de referencia en este trabajo, la Figura 10.8 presenta un sensor CKP magnético anclado a un soporte maquinado en aluminio y alineado con una rueda dentada acoplada al cigüeñal. La figura destaca tres buenas prácticas clave, un entrehierro constante en el rango de 1–3 mm, necesario para una conmutación magnética limpia, la orientación del eje sensible perpendicular al flanco del diente, lo que maximiza la variación de flujo magnético

captada, y la fijación mediante pernos que mitigan vibraciones relativas entre sensor y rueda, reduciendo así el riesgo de saltos en la señal.

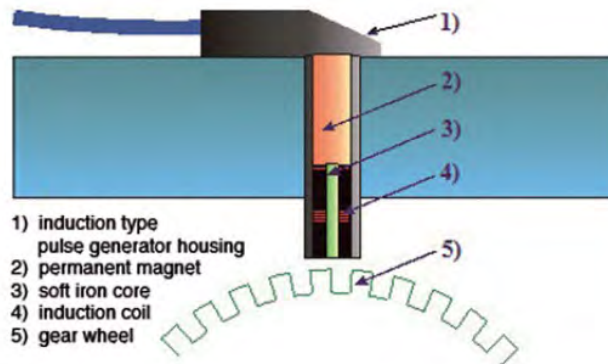


Figura 10.8: Montaje de sensor con referencia a la rueda fónica Castellanos Martínez, José Francisco (2015).

En el módulo de pruebas para desalineamiento propuesto en esta tesis se adoptará la misma filosofía en el montaje del sensor, pero con una adaptación manufacturable en laboratorio, se diseñará una rueda fónica impresa en 3D sobre la cual se insertarán pines metálicos cilíndricos que funcionarán como dientes magnéticos. El número óptimo de pines y, por ende, la resolución angular se determinará experimentalmente, partiendo de configuraciones de 4 y 10 pines, y se evaluará la robustez de la señal del sensor frente a vibraciones y velocidad variable, seleccionando el arreglo que ofrezca mejor compromiso entre resolución temporal y robustez de lectura.

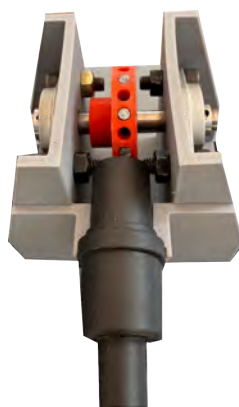


Figura 10.9: Montaje del sensor.

La adquisición de datos provenientes del sensor se realizó utilizando el software Studio One 6, el cual permitió registrar con precisión las señales generadas durante las pruebas de rotación del sistema. Este entorno de grabación fue seleccionado por su capacidad de capturar señales de alta resolución en formato de audio, lo que facilitó el posterior análisis en el dominio del tiempo y la frecuencia. Gracias a su compatibilidad con la interfaz de audio empleada y su tasa de muestreo adecuada, se aseguró una correcta digitalización de la señal.

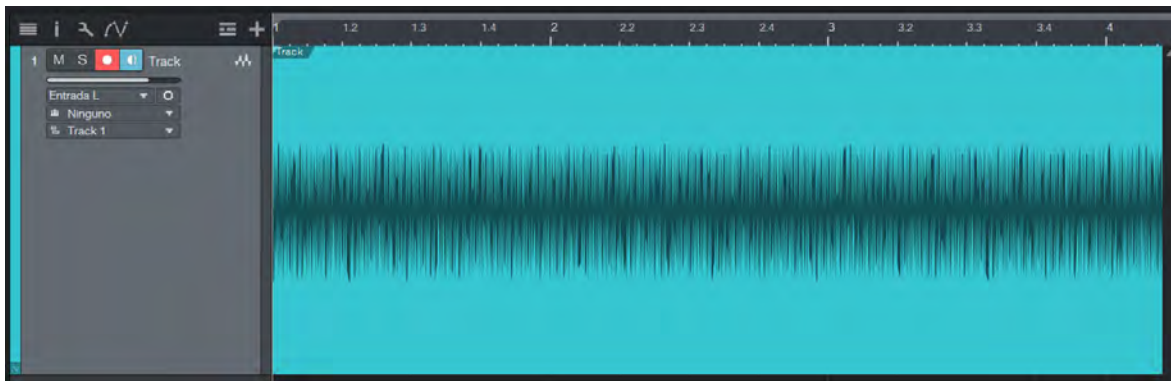


Figura 10.10: Ejemplo de la señal capturada durante el proceso de adquisición de datos en el módulo de pruebas.

La captura de datos se realizó para distintos niveles de desalineamiento angular, comenzando en 0° y finalizando en 15° , con incrementos de 1° en 1° . En la Figura 10.11 se muestran las condiciones de desalineamiento consideradas para la categorización de los estados evaluados.

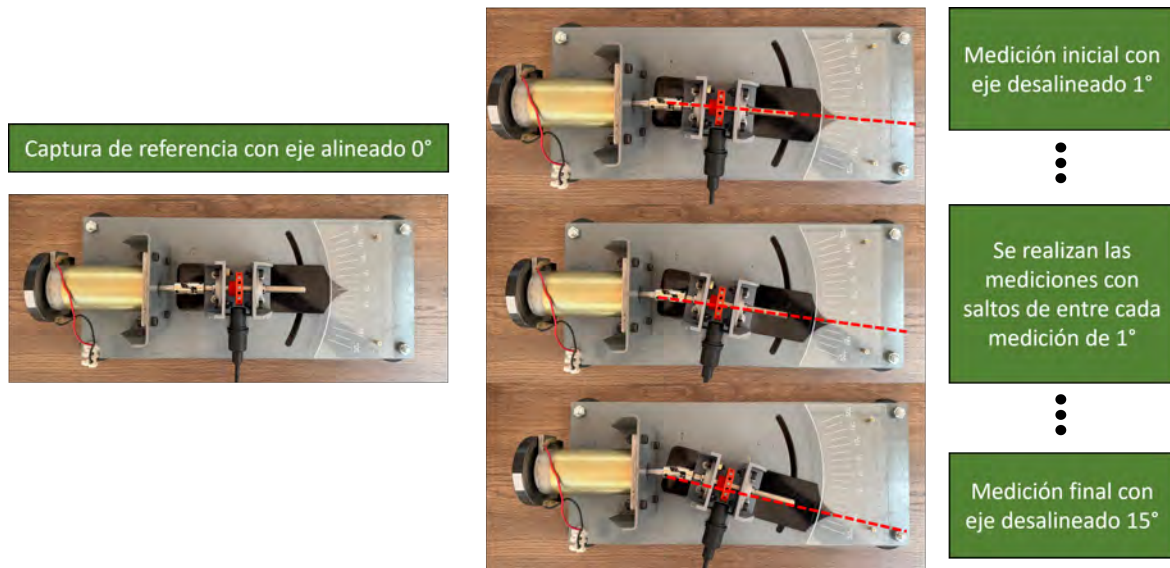


Figura 10.11: Niveles de desalineamiento en ejes.

Se llevaron a cabo pruebas de desalineamiento angular en el eje considerando incrementos de 1° en 1° , abarcando los distintos niveles de desalineamiento establecidos para el estudio. En cada una de estas condiciones se registraron 197 segundos de señal proveniente del sensor por cada prueba, con una tasa de muestreo de 44.1 kHz; estos datos, adquiridos de forma continua y bajo condiciones controladas, permiten comparar directamente cómo varían las características de la señal conforme se incrementa gradualmente el ángulo de desalineamiento.

Finalmente, con la data recolectada, se construyó una base de datos estructurada a partir de la señal continua de 197 segundos, segmentando cada registro en 197 fragmentos de 1 segundo de duración. Cada uno de estos fragmentos temporales se procesó y exportó como un fichero de texto en formato CSV, de modo que cada archivo contiene las muestras correspondientes a un intervalo de 1 segundo. Esta organización permitió disponer de múltiples muestras por cada nivel de desalineamiento, facilitando su posterior procesamiento y su utilización en el entrenamiento del algoritmo de aprendizaje automático.

11 | PROCESAMIENTO DE SEÑALES

11.1. Análisis de señales en el dominio del tiempo

En el monitoreo de vibraciones, el análisis en el dominio del tiempo se utiliza para evaluar la periodicidad, la duración y la amplitud de eventos vibracionales. Esto es crucial para identificar patrones característicos asociados con posibles fallas en rodamientos, como la presencia de impactos, desalineaciones o desgastes anómalos.

Las ventajas de este enfoque incluyen su simplicidad y claridad en la interpretación de eventos temporales. Sin embargo, sus limitaciones se centran en la capacidad limitada para proporcionar información detallada sobre la frecuencia y la duración de los eventos, lo que motiva la exploración de técnicas más avanzadas.

En la Figura 11.1, se muestra la señal en el dominio del tiempo para diferentes condiciones de desalineamiento angular de nuestra base de datos. Se destaca que la diferencia más notable reside en las amplitudes de vibración. Sin embargo, estos resultados no brindan información sobre las frecuencias, aspecto crucial para la detección del nivel de desalineamiento angular.

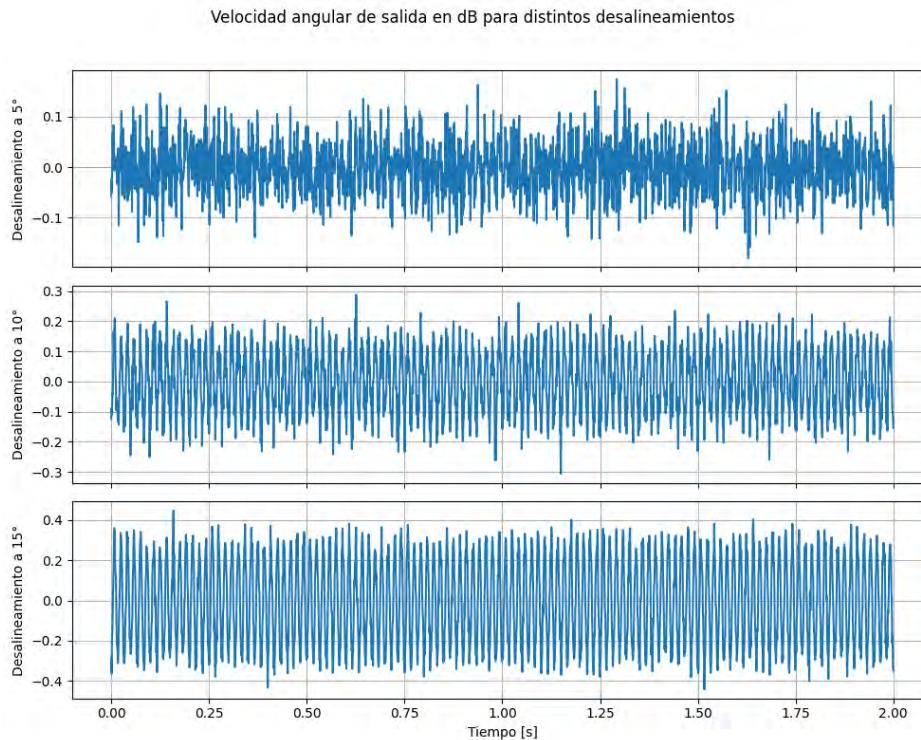


Figura 11.1: Señal en el tiempo capturada de distintos niveles de desalineamiento angular.

11.2. Procesamiento de señales mediante la Transformada Rápida de Fourier FFT

Se realizó un estudio espectral mediante la Transformada Rápida de Fourier FFT con el objetivo de observar el comportamiento frecuencial de las señales obtenidas durante los ensayos preliminares del módulo de pruebas. Este análisis permitió comparar la respuesta de la señal ante diferentes configuraciones de la rueda fónica y distintas condiciones de alineamiento del sistema.

El estudio se desarrolló considerando seis situaciones experimentales. Para cada configuración de la rueda fónica, con 4 y 10 pines, se evaluaron las condiciones de eje alineado y desalineamiento angular de 5° , 10° y 15° . Las señales fueron procesadas en el software *enDAQ VibrationData*, a partir del cual se obtuvieron los espectros en frecuencia para comparar la distribución de amplitudes y la presencia de componentes dominantes en cada caso.

La Figura 11.2 presenta la comparación espectral obtenida mediante FFT para las condiciones evaluadas. Esta representación permitió identificar diferencias en la respuesta frecuencial entre ambas configuraciones de rueda fónica y establecer una base técnica para la selección de la alternativa de medición más adecuada en la adquisición de datos experimentales.



Figura 11.2: Comparación espectral mediante FFT de las señales obtenidas con rueda fónica de 4 y 10 pines, considerando condiciones de eje alineado y desalineamientos angulares de 5°, 10° y 15°.

De la Figura 11.2 que compara los resultados obtenidos con distintos números de insertos metálicos en la rueda fónica, se observa que con la frecuencia de operación del motor de 940 RPM - 15.67 Hz y con 4 insertos se evidencia que en la respuesta en frecuencia se tienen picos máximos en el rango de 58 Hz, con armónicos claramente visibles a 2X, 3X, 4X y 5X, lo cual evidencia que el espectro se encuentra en un rango

no lineal que puede ser producido por condiciones magnéticas, eléctricas, mecánicas o de muestreo. Comparando con los resultados obtenidos con la rueda fónica con 10 pines, que presenta una respuesta que se puede considerarse como lineal, con picos máximos a 161 Hz y sin presentar armónicos significativos, se deduce que la no linealidad se da en el equipo de medición que presenta una respuesta no lineal en la captura de data a bajas frecuencias.

11.2.1. Extracción de características de la base de datos

La extracción de características nos que permitió transformar la gran cantidad de datos crudos en un conjunto reducido de indicadores representativos del comportamiento dinámico del sistema. Esto se logra a través del cálculo de estadísticas de dominio temporal. Estas métricas son especialmente útiles para entrenar modelos de inteligencia artificial, ya que facilitan la distinción entre diferentes estados de operación del sistema, como los distintos niveles de desalineamiento del eje. Las fórmulas detalladas de cada parámetro se muestran a continuación:

Cuadro 11.1: Resumen de fórmulas estadísticas utilizadas en el análisis de señales

Nombre	Fórmula
Media (\bar{x})	$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$
Desviación estándar (Std)	$\text{std} = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}$
RMS (Root Mean Square)	$\text{RMS} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}$
PP (Pico a Pico)	$\text{PP} = x_{\text{máx}} - x_{\text{mín}}$
Factor de impulso (IF)	$\text{IF} = \frac{x_{\text{máx}}}{\bar{x}}$
Skewness	$S = \frac{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^3}{\sigma^3}$
Kurtosis	$K = \frac{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^4}{\sigma^4}$
Crest factor	$C = \frac{ x_{\text{máx}} }{\text{RMS}}$
Shape factor	$\text{SF} = \frac{1}{\frac{1}{N} \sum_{i=1}^N x_i}$

Donde en todos los casos x es el valor de amplitud de la señal de audio.

Se utilizaron los registros experimentales de vibración obtenidos y procesados, para posteriormente etiquetarlos en los distintos niveles de desalineamiento medidos de 0° , 5° , 10° y 15° . Cada prueba se realizó durante 1000 segundos, con una tasa de muestreo de 44.1 kHz. Para preparar la base de datos que alimentará los algoritmos de clasificación y regresión, los datos de cada prueba fueron segmentados en intervalos de 1 segundo, generando así 1000 muestras por cada nivel de desalineamiento. Esta división permite mantener una resolución temporal adecuada y facilita el análisis estadístico de corto plazo de las vibraciones.

Posteriormente, para cada segmento de 1 segundo de duración se aplicó el conjunto de fórmulas de extracción de parámetros previamente definidas. Esto nos devuelve una matriz de 1000 filas por 9 columnas en cada estado de desalineamiento. Se eliminaron la primera y última fila para evitar datos que puedan ser interferidos por el inicio y la parada de la medición, reduciendo la matriz a 998 filas por 9 columnas. El resultado se muestra en la Figura 11.3.

```

1 track, file, rms, impulse_factor, shape_factor, peak_to_peak, crest_factor, mean_std, skewness
2 pista_0_0_0000.csv, 693748576.1904937, 2.0584627864476577, 1.134208626161593, 2444741888.0, 1.8078356478268616, -7.16228874362245e-08, 693748576.1904937, 0.0171884384057301
3 pista_0_0_0001.csv, 693360590.0092138, 2.155543464103816, 1.1358280101978402, 2534571264.0, 1.89777276562176, -1.0968335902069161e-07, 693360590.0092138, 0.0193461028025885
4 pista_0_0_0002.csv, 693011446.6357121, 2.2139936032372605, 1.1365211515212548, 2591726336.0, 1.94880443459180576, -5.8820728812358284e-08, 693011446.6357121, 0.0176102672979983
5 pista_0_0_0003.csv, 694454448.0459394, 2.132254812139246, 1.1347022982807356, 2511701248.0, 1.8789990074481615, -2.9929370836876415e-08, 694454448.0459394, 0.0166905002866334
6 pista_0_0_0004.csv, 692040353.67577, 2.031451543616817, 1.13461303317924, 2442073600.0, 1.7904355795425626, -1.4186175772392291e-08, 692040353.67577, 0.0174915812381325
7 pista_0_0_0005.csv, 692715607.9947805, 2.1412350592670872, 1.135450815385104, 2524981248.0, 1.8858016835725795, -4.1866518742913833e-08, 692715607.9947805, 0.021107555313764
8 pista_0_0_0006.csv, 697918822.8268058, 2.191152086498384, 1.13608424842821, 2606360576.0, 1.9286880260243695, -1.411697491496599e-07, 697918822.8268058, 0.0175931526657427
9 pista_0_0_0007.csv, 695288748.3197448, 2.1606814572706035, 1.1363222357546785, 25242457600.0, 1.901468957734164, 7.871597532242063e-08, 695288748.3197448, 0.009253973904198
10 pista_0_0_0008.csv, 694368001.3835969, 2.067213132582119, 1.13475559442236, 2467683584.0, 1.8217254944300176, -9.065312322845804e-08, 694368001.3835969, 0.009995632550588
11 pista_0_0_0009.csv, 695379864.9133111, 2.067487159659592, 1.135703616839092, 2462823424.0, 1.820659669730871, 2.734338683390024e-08, 695379864.9133111, 0.0110167688783936
12 pista_0_0_0010.csv, 695345521.2917385, 2.131145055738924, 1.1361247634703683, 2519075584.0, 1.8758019578873693, 3.338941370819161e-08, 695345521.2917385, 0.0203895920769676
13 pista_0_0_0011.csv, 695187744.1288638, 2.20558851596691, 1.1361744935654672, 2599456612.0, 1.9412410051977744, -1.4532180059523811e-08, 695187744.1288638, 0.0144743132286596
14 pista_0_0_0012.csv, 696424340.2275518, 2.138848180156139, 1.1351721197480666, 2524084584.0, 1.8841620155636156, -8.390683062030242e-08, 696424340.2275518, 0.0072694121086664
15 pista_0_0_0013.csv, 694138258.1179427, 2.122630610421863, 1.134649953201694, 2536846336.0, 1.8707360842278609, -1.6954210069444444e-08, 694138258.1179427, 0.0182176019886246
16 pista_0_0_0014.csv, 693386078.6281621, 2.1065825153247326, 1.1352505569720734, 2490244864.0, 1.8556102010934144, -1.1080437402565192e-07, 693386078.6281621, 0.0131453692963697
17 pista_0_0_0015.csv, 696863320.7632557, 2.174503267606314, 1.136408459095301, 2565041408.0, 1.9134873998896, 2.5604317247732425e-08, 696863320.7632557, 0.0251085244808902
18 pista_0_0_0016.csv, 694696615.2159201, 2.1315545490701564, 1.1359859256870248, 2519626496.0, 1.8763916883750371, -4.203952088647959e-08, 694696615.2159201, 0.0083902836551288
19 pista_0_0_0017.csv, 693045845.948584, 2.0955052254404403, 1.1352221117134798, 2496849408.0, 1.84589888007727943, -9.98222368374433e-08, 693045845.948584, 0.01357473657969
20 pista_0_0_0018.csv, 696263856.3825378, 2.0683879259723573, 1.13574948553150063, 2488352000.0, 1.8211480196257224, -1.83382721797052e-08, 696263856.3825378, 0.0156915125085195
21 pista_0_0_0019.csv, 696361182.7981291, 2.181858784692287, 1.1354376999190374, 2576252160.0, 1.921601448197912, 6.88548513917233e-08, 696361182.7981291, 0.0186088046206697
22 pista_0_0_0020.csv, 699051335.9415523, 2.1488106784691703, 1.1360491462617803, 2541902496.0, 1.8914768057147751, -1.3926672557043654e-07, 699051335.9415523, 0.020061684022953
23 pista_0_0_0021.csv, 698327368.62005, 2.181415590557508, 1.136485820866779, 2574931456.0, 1.910439340995932, -1.4635081345663266e-07, 698327368.62005, 0.0156951147471567

```

Figura 11.3: Base de datos obtenida de la extracción de parámetros.

Debido a la multidimensionalidad de la matriz formada por la extracción de características y con la finalidad de visualizar la dispersión de la data, se realiza el análisis de componentes principales (PCA), el cual nos permite reducir la matriz multidimensional a una gráfica bidimensional, como se observa en la Figura 11.4.

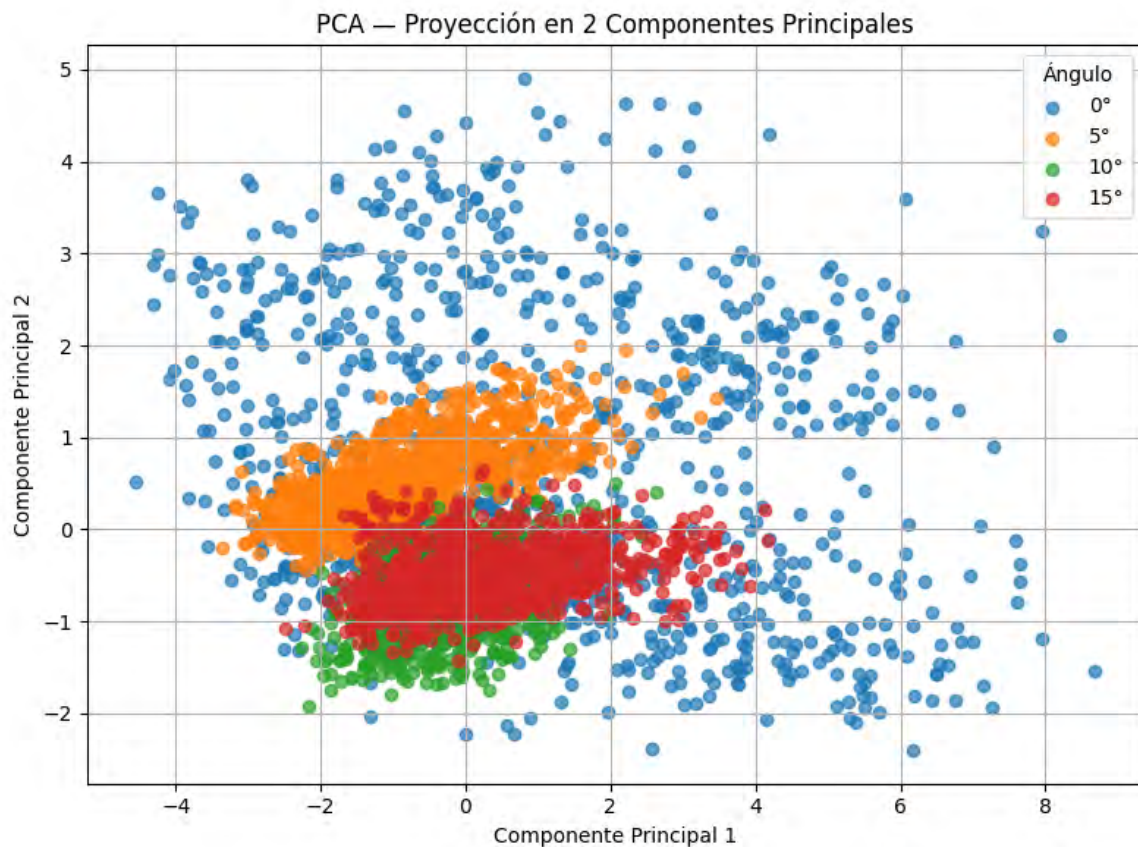


Figura 11.4: Dispersión de pruebas a distintos niveles de desalineamiento analizado por un análisis de componentes principales PCA.

Como se muestra en la Figura 11.4, aunque hay tendencias de agrupamiento por cada ángulo de desalineamiento, existe solapamiento entre las clases, sobre todo entre 5°, 10° y 15°, y por el contrario, los puntos de 0° son muy dispersos. Esto indica que en la matriz reducida por PCA no hay separación lineal perfecta entre todos los niveles de desalineamiento; por ende, se necesita un algoritmo de regresión más sofisticado.

12 | DISEÑO E IMPLEMENTACIÓN DEL ALGORITMO DE APREN- DIZAJE AUTOMÁTICO

El presente capítulo describe detalladamente el proceso de diseño e implementación del algoritmo de aprendizaje automático desarrollado para la detección del nivel de desalineamiento en ejes rotativos. Este componente constituye el núcleo del sistema propuesto, ya que permite transformar datos adquiridos experimentalmente en información útil para la identificación y clasificación automática del estado del sistema mecánico evaluado. Se adopta un enfoque sistemático que combina la preparación de datos, el modelado del problema, la selección de técnicas de aprendizaje adecuadas y la validación rigurosa del modelo obtenido. El flujo de trabajo para la implementación del sistema de aprendizaje automático se ve en la Figura 12.1.

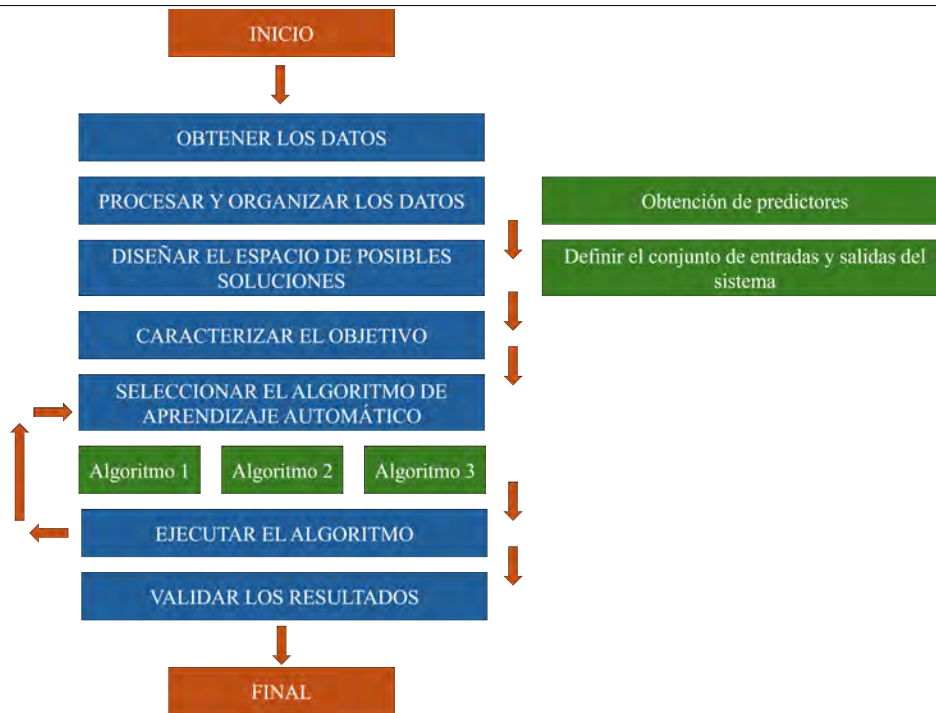


Figura 12.1: Flujograma para la implementación de un sistema inteligente.

El desarrollo del algoritmo se estructura siguiendo una secuencia metodológica compuesta por seis etapas principales. Estas etapas comprenden la obtención de datos, el procesamiento y organización de la información, el diseño del espacio de posibles soluciones, la caracterización del objetivo, la selección del algoritmo de aprendizaje automático, la ejecución del algoritmo y la validación de los resultados.

Esta secuencia permite que el desarrollo sea ordenado, reproducible y coherente con el tratamiento de datos experimentales y la aplicación de modelos de aprendizaje supervisado. Además, este enfoque facilita que el algoritmo implementado tenga una base sólida desde el punto de vista experimental y computacional, permitiendo su posible aplicación en sistemas de monitoreo y diagnóstico de fallas en maquinaria rotativa.

12.1. Diseño del espacio de posibles soluciones

En esta etapa se definió la forma en que el problema de detección del desalineamiento angular sería representado desde un enfoque computacional. Para ello, se establecieron las variables de entrada, la variable objetivo y el tipo de salida esperada del sistema predictivo. Esta formulación permitió transformar el fenómeno físico, asociado a la respuesta vibracional del módulo de pruebas, en un problema de aprendizaje automático estructurado.

El espacio de posibles soluciones se planteó a partir de las señales adquiridas durante los ensayos experimentales. Dichas señales fueron procesadas y segmentadas en ventanas de análisis, a partir de las cuales se extrajeron características estadísticas representativas del comportamiento vibracional. Estas características constituyeron el vector de entrada de los modelos, permitiendo describir cuantitativamente la amplitud, dispersión, energía, forma y presencia de picos en la señal.

La variable de salida se definió en función del nivel de desalineamiento angular impuesto durante cada ensayo. De esta manera, el sistema predictivo fue formulado para relacionar las características extraídas de la señal con el ángulo de desalineamiento correspondiente. Esta representación permitió evaluar diferentes alternativas de modelamiento, tanto para estimar el valor del ángulo como para identificar clases asociadas a distintos niveles de desalineamiento.

El diseño del espacio de posibles soluciones permitió delimitar el problema, establecer la estructura de los datos de entrada y salida, y definir la base metodológica para la implementación de los modelos de inteligencia artificial utilizados en la investigación.

12.1.1. Modelamiento matemático del fenómeno

La transmisión de movimiento rotatorio mediante una junta universal presenta un comportamiento cinemático particular cuando existe un ángulo entre el eje de entrada y el eje de salida. En este tipo de mecanismo, aun cuando la velocidad angular de entrada sea constante, la velocidad angular de salida puede variar de forma periódica debido a la geometría de transmisión de la junta. Este fenómeno es ampliamente descrito en la teoría de máquinas y mecanismos, donde se establece que el análisis cinemático permite relacionar desplazamientos, velocidades y aceleraciones en sistemas mecánicos articulados Uicker et al. (2017); Norton (2020).

Una junta universal está compuesta por dos horquillas unidas mediante una cruzeta, lo que permite transmitir movimiento entre dos ejes cuyos centros de rotación se intersectan formando un ángulo β . Para el presente análisis se considera θ_i como el ángulo de giro del eje de entrada y θ_o como el ángulo de giro del eje de salida. La relación geométrica entre ambos ángulos se expresa como Uicker et al. (2017):

$$\tan \theta_o = \frac{\sin \theta_i}{\cos \theta_i \cdot \cos \beta}$$

donde β representa el ángulo de desalineamiento entre el eje de entrada y el eje de salida.

Simplificando la expresión anterior, se obtiene:

$$\tan \theta_o = \frac{\tan \theta_i}{\cos \beta} \quad (12.1)$$

Para obtener la relación entre las velocidades angulares, se deriva la expresión anterior con respecto al tiempo. Este procedimiento es consistente con el análisis ci-

nemático de mecanismos, donde las velocidades se obtienen a partir de la derivación temporal de las coordenadas angulares Norton (2020):

$$\frac{d}{dt}(\tan \theta_o) = \frac{d}{dt} \left(\frac{\tan \theta_i}{\cos \beta} \right)$$

Aplicando la derivada de la función tangente:

$$\sec^2 \theta_o \cdot \frac{d\theta_o}{dt} = \frac{1}{\cos \beta} \cdot \sec^2 \theta_i \cdot \frac{d\theta_i}{dt}$$

Se definen las velocidades angulares de entrada y salida como:

$$\omega_i = \frac{d\theta_i}{dt}$$

$$\omega_o = \frac{d\theta_o}{dt}$$

Por tanto, la ecuación anterior puede escribirse como:

$$\sec^2 \theta_o \cdot \omega_o = \frac{\sec^2 \theta_i}{\cos \beta} \cdot \omega_i$$

A partir de la identidad trigonométrica:

$$\sec^2 \theta_o = 1 + \tan^2 \theta_o$$

y usando la relación geométrica de la junta universal:

$$\tan^2 \theta_o = \frac{\tan^2 \theta_i}{\cos^2 \beta}$$

se obtiene:

$$\sec^2 \theta_o = 1 + \frac{\tan^2 \theta_i}{\cos^2 \beta} = \frac{\cos^2 \beta + \tan^2 \theta_i}{\cos^2 \beta}$$

Reemplazando esta expresión en la ecuación de velocidades:

$$\frac{\cos^2 \beta + \tan^2 \theta_i}{\cos^2 \beta} \cdot \omega_o = \frac{\sec^2 \theta_i}{\cos \beta} \cdot \omega_i$$

Además, considerando que:

$$\sec^2 \theta_i = 1 + \tan^2 \theta_i$$

se obtiene:

$$\frac{\cos^2 \beta + \tan^2 \theta_i}{\cos^2 \beta} \cdot \omega_o = \frac{1 + \tan^2 \theta_i}{\cos \beta} \cdot \omega_i$$

Multiplicando ambos lados por $\cos^2 \beta$:

$$(\cos^2 \beta + \tan^2 \theta_i) \cdot \omega_o = (1 + \tan^2 \theta_i) \cos \beta \cdot \omega_i$$

Despejando la velocidad angular de salida:

$$\omega_o = \frac{(1 + \tan^2 \theta_i) \cos \beta}{\cos^2 \beta + \tan^2 \theta_i} \cdot \omega_i$$

Finalmente, mediante identidades trigonométricas, la expresión puede escribirse como:

$$\omega_o = \omega_i \cdot \frac{\cos \beta}{1 - \sin^2 \beta \cdot \cos^2 \theta_i} \quad (12.2)$$

Esta expresión muestra que la velocidad angular de salida ω_o depende de la velocidad angular de entrada ω_i , del ángulo de desalineamiento β y de la posición angular instantánea del eje de entrada θ_i . Este comportamiento coincide con el principio cinemático de una junta universal simple, en la cual la relación de velocidades deja de ser constante cuando existe un ángulo entre los ejes acoplados Uicker et al. (2017); Norton (2020).

Cuando $\beta = 0$, se cumple que $\cos \beta = 1$ y $\sin^2 \beta = 0$, por lo que la ecuación se reduce a:

$$\omega_o = \omega_i$$

Este resultado indica que, en condición alineada, la velocidad angular de salida sigue directamente a la velocidad angular de entrada. En cambio, cuando β aumenta, el denominador de la ecuación varía periódicamente con θ_i , generando fluctuaciones en la velocidad angular de salida. Estas fluctuaciones son propias de la junta universal simple y explican la aparición de componentes dinámicas asociadas al desalineamiento angular Norton (2020).

Para visualizar este comportamiento, se puede graficar la velocidad angular de salida ω_o en función del tiempo para diferentes valores del ángulo de desalineamiento, considerando una velocidad de entrada constante de 940 rpm. En la Figura 12.2 se observa que, para $\beta = 0^\circ$, la velocidad de salida permanece constante y coincide con

la velocidad de entrada. Sin embargo, cuando el ángulo aumenta a 5° , 10° y 15° , la velocidad de salida empieza a presentar una variación periódica alrededor del valor de referencia. Esta variación demuestra que el desalineamiento angular modifica la transmisión de velocidad en la junta universal simple, generando una fluctuación que puede reflejarse posteriormente en la respuesta vibracional del sistema.

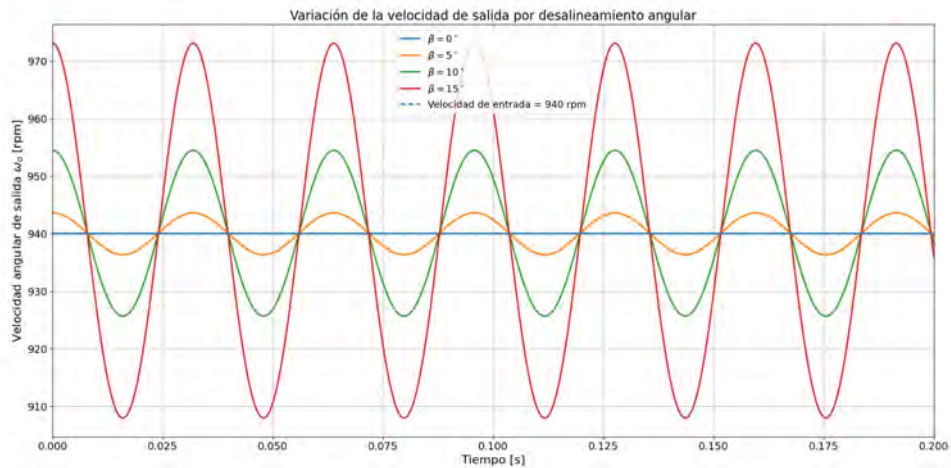


Figura 12.2: Variación temporal de la velocidad angular de salida para diferentes ángulos de desalineamiento, considerando una velocidad de entrada constante de 940 rpm.

12.2. Selección del Algoritmo de Aprendizaje Automático

En esta investigación se utilizaron modelos de aprendizaje automático supervisado para estimar el ángulo de desalineamiento del eje a partir de las características extraídas de las señales medidas. Las variables de entrada corresponden a parámetros estadísticos y temporales de la señal, mientras que la variable de salida corresponde al ángulo de desalineamiento expresado en grados.

Debido a que el objetivo del modelo es entregar un valor numérico del ángulo, se emplearon algoritmos de regresión. Este enfoque permite estimar directamente el nivel de desalineamiento y evaluar el error de predicción en unidades de grados.

Para esta investigación se seleccionaron tres algoritmos de regresión:

- Random Forest Regressor
- XGBoost Regressor
- Red neuronal MLP Regressor

La selección de estos modelos se realizó porque permiten representar relaciones no lineales entre las características extraídas de las señales y el ángulo de desalineamiento. Esta consideración es importante debido a que la respuesta dinámica del sistema no necesariamente varía de forma lineal con el incremento del ángulo. Además, los tres modelos son adecuados para trabajar con bases de datos tabulares generadas a partir de características extraídas de señales experimentales.

12.2.1. Random Forest Regressor

Random Forest Regressor es un algoritmo basado en un conjunto de árboles de decisión. Cada árbol realiza una estimación del ángulo de desalineamiento a partir de las características de entrada. Posteriormente, el modelo combina los resultados de todos los árboles y obtiene la predicción final mediante el promedio de sus salidas. Este enfoque reduce la dependencia de un único árbol de decisión y mejora la estabilidad de la predicción, como indica Breiman (2001).

Este algoritmo fue seleccionado porque trabaja adecuadamente con datos tabulares obtenidos a partir de características de señales. Además, no requiere asumir una relación lineal entre las variables de entrada y la salida, por lo que resulta apropiado para analizar un fenómeno mecánico donde las características vibracionales pueden cambiar de forma no lineal con el desalineamiento.

CAPÍTULO 12. DISEÑO E IMPLEMENTACIÓN DEL ALGORITMO DE APRENDIZAJE AUTOMÁTICO

En el caso de regresión, Random Forest no utiliza una salida de tipo discreta, sino que promedia las predicciones numéricas generadas por los diferentes árboles. De esta manera, cada árbol aporta una estimación individual y la salida final corresponde a un valor promedio más estable. Este principio se muestra de manera esquemática en la Figura 12.4.

Cuadro 12.1: Funcionamiento general de Random Forest Regressor.

Etapa	Descripción
Entrada del modelo	El modelo recibe las características extraídas de la señal.
Construcción de árboles	Se generan varios árboles de decisión, cada uno con una estimación propia del ángulo.
Combinación de resultados	Las predicciones individuales de los árboles se promedian.
Salida final	Se obtiene el ángulo de desalineamiento estimado.

La predicción final del modelo se expresa como:

$$\hat{y} = \frac{1}{n} \sum_{i=1}^n \hat{y}_i \quad (12.3)$$

Donde \hat{y} es el ángulo de desalineamiento estimado, \hat{y}_i es la predicción entregada por el árbol i , y n es el número total de árboles del modelo.

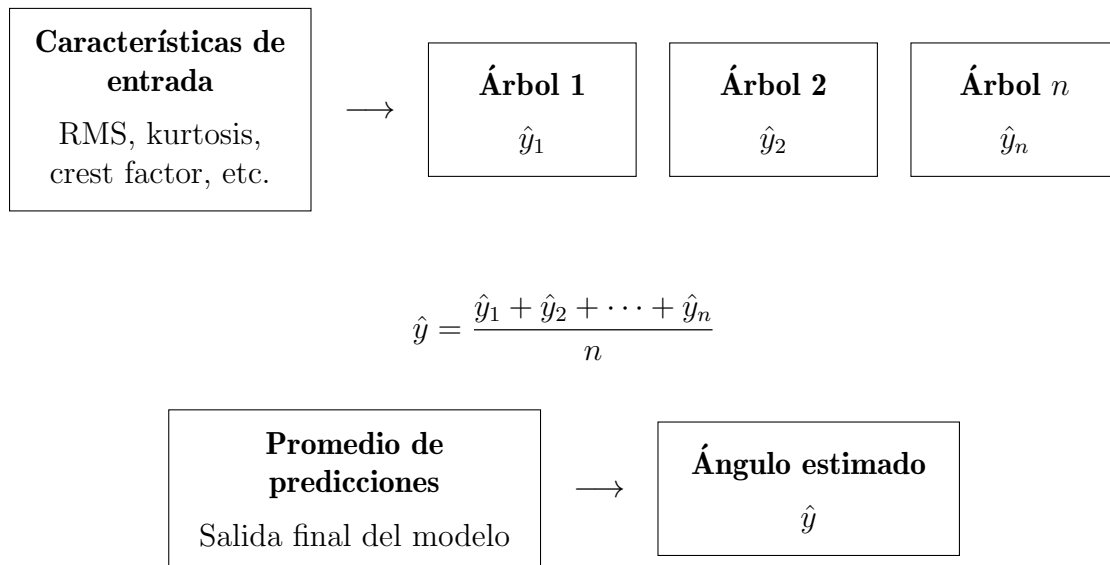


Figura 12.3: Esquema general del funcionamiento de Random Forest Regressor para la estimación del ángulo de desalineamiento.

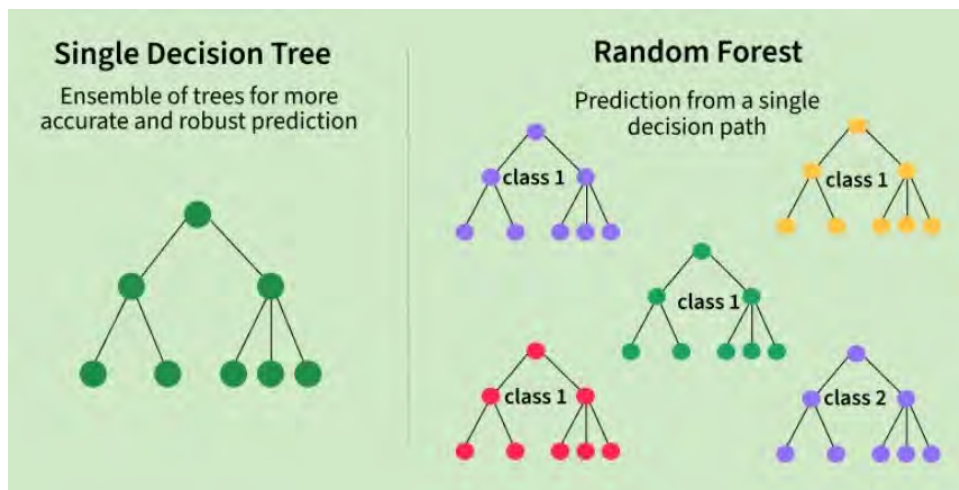


Figura 12.4: Diagrama simplificado del funcionamiento de Random Forest, donde se observa el uso de múltiples árboles de decisión para generar una predicción más robusta que la obtenida con un solo árbol. Adaptado de GeeksforGeeks (2026).

12.2.2. XGBoost Regressor

XGBoost Regressor también se basa en árboles de decisión, pero su forma de aprendizaje es distinta a la de Random Forest. En este caso, los árboles se construyen de manera secuencial. Cada nuevo árbol se ajusta para reducir el error que permanece después de los árboles anteriores. Por esta razón, XGBoost pertenece a los métodos de *boosting*, los cuales buscan mejorar progresivamente el modelo mediante la corrección de errores sucesivos, como indica Chen and Guestrin (2016).

Este algoritmo fue seleccionado porque suele presentar buen desempeño en problemas de regresión con variables numéricas derivadas de datos experimentales. Además, su capacidad para combinar múltiples árboles de forma secuencial permite representar interacciones entre características que podrían estar asociadas al incremento del ángulo de desalineamiento.

En este método, el modelo no depende de un único árbol, sino de una suma progresiva de contribuciones. Cada árbol nuevo intenta corregir los residuos o errores que no fueron explicados correctamente por los árboles anteriores. Este funcionamiento se representa de forma simplificada en la Figura 12.6.

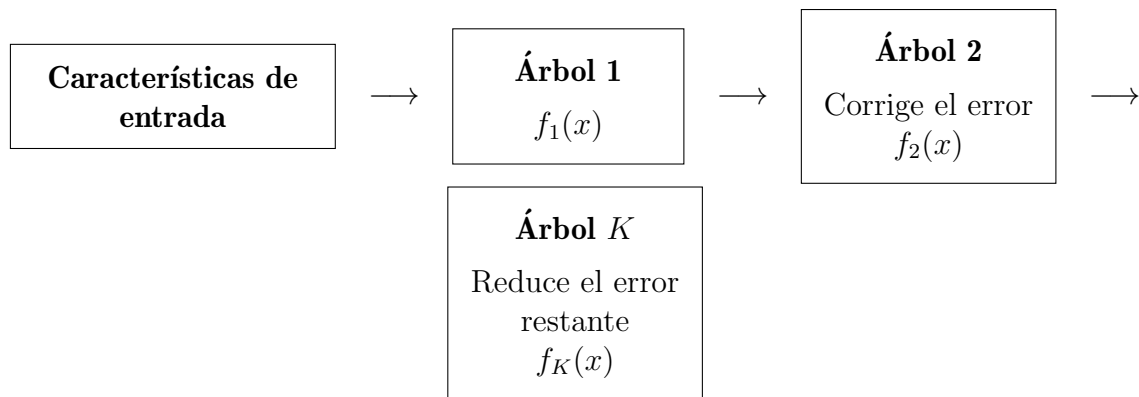
Cuadro 12.2: Funcionamiento general de XGBoost Regressor.

Etapas	Descripción
Entrada del modelo	El modelo recibe las características extraídas de la señal.
Primer árbol	Se obtiene una primera estimación del ángulo de desalineamiento.
Árboles posteriores	Cada árbol nuevo intenta reducir el error dejado por los árboles anteriores.
Salida final	La predicción final se obtiene mediante la suma de las contribuciones de los árboles.

De forma general, la predicción de XGBoost puede representarse como la suma de las contribuciones de varios árboles:

$$\hat{y} = \sum_{k=1}^K f_k(x) \tag{12.4}$$

donde x representa el vector de características de entrada, $f_k(x)$ corresponde a la contribución del árbol k , K es el número total de árboles y \hat{y} es el ángulo de desalineamiento estimado.



$$\hat{y} = f_1(x) + f_2(x) + \dots + f_K(x)$$

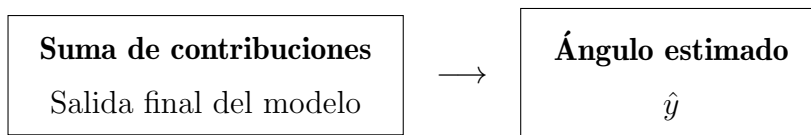


Figura 12.5: Esquema general del funcionamiento de XGBoost Regressor para la estimación del ángulo de desalineamiento.

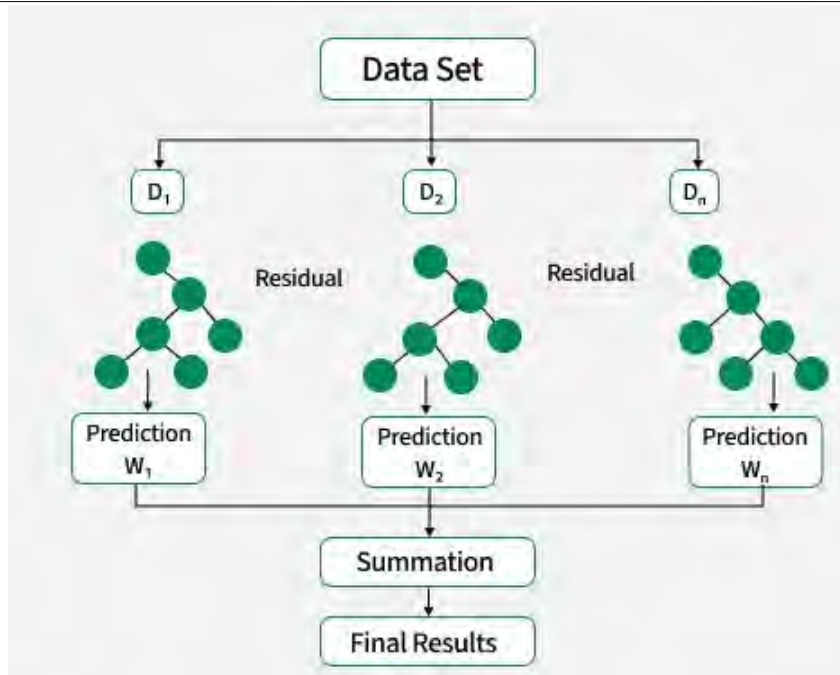


Figura 12.6: Diagrama simplificado del funcionamiento de XGBoost, donde se muestra la construcción secuencial de árboles y la suma de sus contribuciones para obtener el resultado final. Adaptado de GeeksforGeeks (2025).

12.2.3. Perceptrón Multicapa: MLPRegressor

El MLPRegressor es una red neuronal artificial tipo perceptrón multicapa. Este modelo está formado por una capa de entrada, una o más capas ocultas y una capa de salida. La capa de entrada recibe las características extraídas de la señal, las capas ocultas procesan la información mediante pesos y funciones de activación, y la capa de salida entrega el ángulo de desalineamiento estimado.

Durante el entrenamiento, la red ajusta sus pesos internos para reducir la diferencia entre el ángulo real y el ángulo estimado. Este ajuste se realiza mediante el proceso de retropropagación del error, ampliamente utilizado en el entrenamiento de redes neuronales, como indica Rumelhart et al. (1986).

Este algoritmo fue seleccionado porque permite aproximar funciones no lineales entre las características de entrada y la variable de salida. A diferencia de los métodos

basados en árboles, el MLPRegressor aprende una función continua a partir de los datos, lo cual resulta conveniente para estimar una magnitud física como el ángulo de desalineamiento.

Cuadro 12.3: Funcionamiento general de MLPRegressor.

Etapas	Descripción
Capa de entrada	Recibe las características extraídas de la señal.
Capas ocultas	Procesan la información mediante pesos, sesgos y funciones de activación.
Entrenamiento	Los pesos internos se ajustan para reducir el error entre el ángulo real y el estimado.
Capa de salida	Entrega el ángulo de desalineamiento estimado.

La operación básica de una neurona puede expresarse como:

$$a = \varphi \left(\sum_{i=1}^m w_i x_i + b \right) \quad (12.5)$$

donde x_i representa cada característica de entrada, w_i es el peso asociado a dicha característica, b es el sesgo, φ es la función de activación y a es la salida de la neurona.

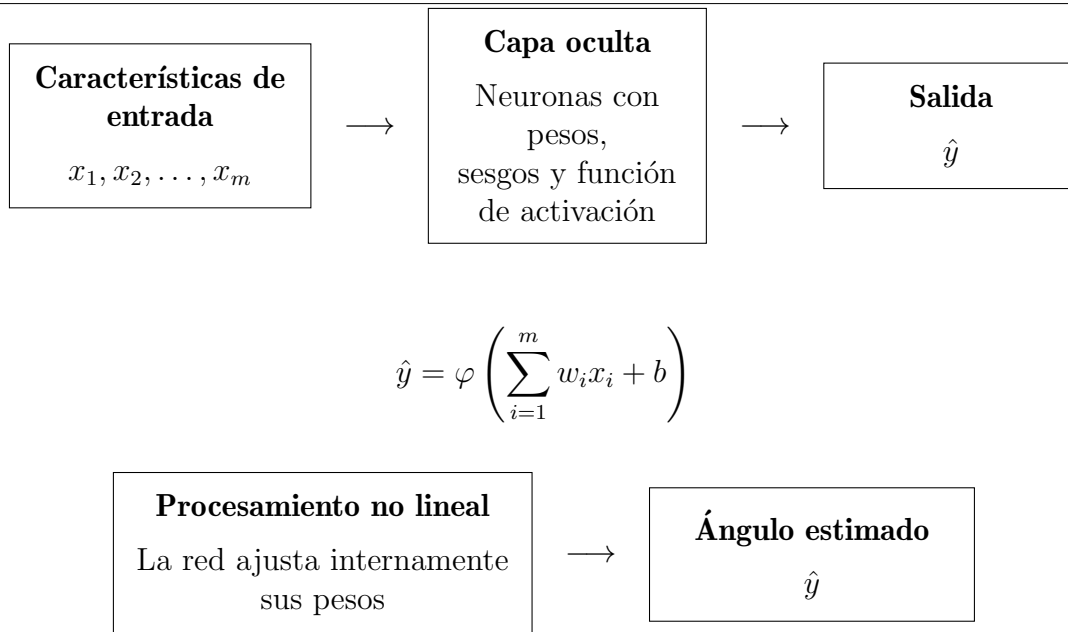


Figura 12.7: Esquema general del funcionamiento de MLPRegressor para la estimación del ángulo de desalineamiento.

12.2.4. Criterio de selección de los modelos

Los tres algoritmos fueron seleccionados porque representan enfoques diferentes de regresión supervisada. Random Forest Regressor permite obtener una estimación robusta mediante el promedio de múltiples árboles de decisión. XGBoost Regressor emplea una construcción secuencial de árboles para reducir el error de predicción de forma progresiva. MLPRegressor permite evaluar un modelo basado en redes neuronales, capaz de aproximar relaciones continuas y no lineales entre las características de la señal y el ángulo de desalineamiento.

La comparación de estos tres modelos permite evaluar el desempeño de métodos basados en árboles y de un modelo neuronal sobre la misma base de datos experimental. Esta selección es adecuada para el objetivo del trabajo, debido a que las características extraídas de las señales forman una base de datos tabular y la salida requerida es un valor numérico correspondiente al ángulo de desalineamiento.

Cuadro 12.4: Criterio de selección de los algoritmos de regresión evaluados.

Algoritmo	Tipo de modelo	Motivo de selección
Random Forest Regressor	Conjunto de árboles de decisión	Modelo robusto para datos tabulares y relaciones no lineales entre características y salida.
XGBoost Regresor	Árboles secuenciales con corrección de error	Modelo de alta capacidad predictiva para regresión con variables numéricas experimentales.
MLPRegressor	Red neuronal tipo perceptrón multicapa	Modelo capaz de aproximar funciones continuas no lineales entre las características de señal y el ángulo.

Los tres modelos fueron evaluados utilizando la misma base de datos y una partición de entrenamiento y prueba del 80 % y 20 %, respectivamente, con una semilla fija de 42 para asegurar la reproducibilidad de los resultados. Como métricas de evaluación se emplearon el Error Absoluto Medio (MAE), la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación R^2 , ya que estas permiten cuantificar la precisión de la predicción del ángulo de desalineamiento en grados.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (12.6)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (12.7)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (12.8)$$

donde y_i representa el ángulo real de desalineamiento, \hat{y}_i representa el ángulo estimado por el modelo, \bar{y} representa el promedio de los valores reales y n corresponde al número total de muestras evaluadas.

12.3. Ejecución del algoritmo

Durante la etapa de ejecución del algoritmo se implementaron tres modelos de regresión con el propósito de estimar el ángulo de desalineamiento a partir de las características extraídas de las señales experimentales. Para ello, se emplearon los algoritmos Random Forest Regressor, XGBoost Regressor y una red neuronal tipo perceptrón multicapa implementada mediante `MLPRegressor`.

La base de datos utilizada por los algoritmos fue construida a partir de los archivos de características almacenados en la carpeta `features_1s_dco`. Cada archivo corresponde a una pista de medición, y el ángulo de desalineamiento fue obtenido directamente del nombre del archivo mediante la identificación del valor numérico asociado a `pista_`. Posteriormente, se asignó este valor como variable de salida `angle_deg`. Para formar la matriz de entrada, se eliminaron columnas auxiliares como `angle_deg`, `track` y `file`, conservando únicamente las variables numéricas extraídas de las señales.

Los datos fueron divididos en un conjunto de entrenamiento y un conjunto de prueba utilizando una proporción de 80 % y 20 %, respectivamente. En los tres casos se empleó una semilla fija de `random_state=42`, con el fin de mantener la misma partición de datos y permitir una comparación directa entre los modelos. La evaluación del desempeño se realizó mediante el Error Absoluto Medio (MAE), la Raíz del Error Cuadrático Medio (RMSE) y el coeficiente de determinación R^2 .

En el caso de Random Forest Regressor, se utilizó un modelo basado en un conjunto de árboles de decisión. Cada árbol genera una estimación del ángulo de desali-

neamiento y la predicción final se obtiene mediante el promedio de las predicciones individuales. Este algoritmo permite capturar relaciones no lineales entre las características de entrada y la salida, además de presentar buena estabilidad para datos experimentales derivados de señales.

Cuadro 12.5: Parámetros utilizados para Random Forest Regressor

Parámetro	Descripción	Valor
<code>n_estimators</code>	Número de árboles del modelo	800
<code>random_state</code>	Semilla para reproducibilidad	42
<code>n_jobs</code>	Uso de los núcleos disponibles para el procesamiento	-1
<code>max_depth</code>	Profundidad máxima de los árboles	Por defecto
<code>min_samples_split</code>	Número mínimo de muestras para dividir un nodo	Por defecto
<code>min_samples_leaf</code>	Número mínimo de muestras en una hoja terminal	Por defecto

Para el modelo XGBoost Regressor se empleó una configuración basada en árboles de decisión construidos de manera secuencial. A diferencia de Random Forest, donde los árboles se combinan por promedio, en XGBoost cada nuevo árbol se incorpora para reducir el error residual de los árboles anteriores. Esta característica permite mejorar progresivamente la predicción del ángulo de desalineamiento.

Cuadro 12.6: Parámetros utilizados para XGBoost Regressor

Parámetro	Descripción	Valor
<code>n_estimators</code>	Número de árboles de boosting	1200
<code>learning_rate</code>	Tasa de aprendizaje del modelo	0.03
<code>max_depth</code>	Profundidad máxima de cada árbol	5
<code>subsample</code>	Proporción de muestras utilizadas en cada árbol	0.9
<code>colsample_bytree</code>	Proporción de características utilizadas por árbol	0.9
<code>reg_lambda</code>	Regularización L2 aplicada al modelo	1.0
<code>random_state</code>	Semilla para reproducibilidad	42
<code>n_jobs</code>	Uso de los núcleos disponibles para el procesamiento	-1

Por otro lado, para la red neuronal se utilizó un perceptrón multicapa mediante `MLPRegressor`. En este caso, el modelo fue implementado dentro de una estructura `Pipeline`, incorporando previamente una etapa de estandarización mediante `StandardScaler`. Esta etapa es necesaria debido a que las redes neuronales son sensibles a la escala de las variables de entrada. De esta manera, las características se normalizan antes de ingresar al modelo neuronal.

La arquitectura utilizada considera tres capas ocultas con 128, 64 y 32 neuronas, respectivamente. La función de activación empleada fue ReLU y el entrenamiento se realizó mediante el optimizador Adam. Esta configuración permite que la red aprenda relaciones no lineales entre las características de las señales y el ángulo de desalineamiento.

Cuadro 12.7: Parámetros utilizados para la red neuronal MLPRegressor

Parámetro	Descripción	Valor
StandardScaler	Estandarización previa de las variables de entrada	Sí
hidden_layer_sizes	Arquitectura de capas ocultas	(128, 64, 32)
activation	Función de activación	ReLU
solver	Algoritmo de optimización	Adam
alpha	Regularización L2	1×10^{-4}
learning_rate_init	Tasa de aprendizaje inicial	0.001
max_iter	Número máximo de iteraciones	800
random_state	Semilla para reproducibilidad	42

Después del entrenamiento de cada modelo, se estimó el ángulo de desalineamiento para el conjunto de prueba. Los valores predichos fueron comparados con los valores reales mediante las métricas MAE, RMSE y R^2 , lo que permitió evaluar el error de predicción en grados y la capacidad de ajuste de cada algoritmo.

12.4. Validación de resultados experimentales

Para interpretar estos resultados, se consideraron tres métricas principales de evaluación. El MAE representa el error absoluto medio entre el ángulo real y el ángulo estimado, por lo que permite conocer la desviación promedio del modelo en grados. El RMSE corresponde a la raíz del error cuadrático medio y permite evaluar la magnitud global del error, penalizando con mayor intensidad las predicciones con desviaciones elevadas. Por su parte, el coeficiente de determinación R^2 indica qué proporción de la variabilidad de los ángulos reales es explicada por el modelo y por ello, valores cercanos a 1 evidencian una mayor capacidad predictiva.

La Tabla 12.8 presenta los resultados obtenidos con los tres modelos de regresión evaluados. En general, los tres algoritmos mostraron un desempeño satisfactorio, ya

CAPÍTULO 12. DISEÑO E IMPLEMENTACIÓN DEL ALGORITMO DE APRENDIZAJE AUTOMÁTICO

que alcanzaron valores de R^2 superiores a 0.95, lo que indica un buen ajuste entre los valores predichos y los ángulos reales de desalineamiento.

El modelo Random Forest presentó el mejor desempeño global, con un MAE de 0.4733° , un RMSE de 1.0105° y un coeficiente $R^2 = 0,9540$. Esto indica que, en promedio, el error absoluto de predicción fue menor a medio grado, lo cual representa un resultado favorable para la estimación del nivel de desalineamiento.

La red neuronal también mostró un comportamiento competitivo, con un MAE de 0.5352° y un R^2 de 0.9535, mientras que XGBoost obtuvo un MAE de 0.5893° y un R^2 de 0.9519. Aunque las diferencias entre los tres modelos no son muy grandes, los resultados muestran que Random Forest fue el algoritmo que logró la mejor precisión dentro de las condiciones evaluadas.

En consecuencia, los resultados obtenidos permiten afirmar que el uso de modelos de regresión aplicados a las características extraídas de las señales medidas constituye una alternativa viable para estimar automáticamente el ángulo de desalineamiento en ejes.

Cuadro 12.8: Resultados obtenidos con los modelos de regresión evaluados

Modelo	MAE [°]	RMSE [°]	R^2
Red neuronal	0.5352	1.0158	0.9535
XGBoost Regressor	0.5893	1.0331	0.9519
Random Forest Regressor	0.4733	1.0105	0.9540

12.5. Validación del modelo con conjunto de datos externo

Con el propósito de evaluar la capacidad de generalización del modelo desarrollado a partir del módulo experimental, se realizó una validación externa empleando el conjunto de datos *Shaft Angular Misalignment Dataset* Kechik et al. (2021). Este conjunto de datos fue utilizado como referencia externa para contrastar el comportamiento del modelo en un dominio distinto al del banco experimental desarrollado en la presente investigación.

De acuerdo con la documentación del conjunto de datos, cada archivo contiene tres canales de vibración correspondientes a las direcciones radial horizontal, radial vertical y axial. Asimismo, para el caso de desalineamiento angular horizontal, los archivos se agrupan en tres niveles de severidad:

Archivos 1 a 24 para 0,5 mm/100 mm, archivos 25 a 48 para 0,8 mm/100 mm y archivos 49 a 72 para 1,12 mm/100 mm. En esta investigación se seleccionó el subconjunto de desalineamiento angular horizontal y, dentro de este, se empleó el canal radial horizontal por ser el más compatible con el enfoque de análisis adoptado en la etapa experimental.

12.5.1. Referencia del conjunto de datos empleado

La Figura 12.8 presenta una imagen referencial del procedimiento experimental asociado al desalineamiento angular, en la cual se ilustra de manera esquemática la condición de desalineamiento entre ejes. Esta imagen se incluye con fines descriptivos para facilitar la comprensión del fenómeno físico considerado en la validación externa. Asimismo, permite contextualizar el principio mecánico sobre el cual se construyó el

conjunto de datos real empleado en esta etapa de la investigación. La imagen fue tomada del recurso publicado por Kechik, Aslamov y Davydov Kechik et al. (2021).

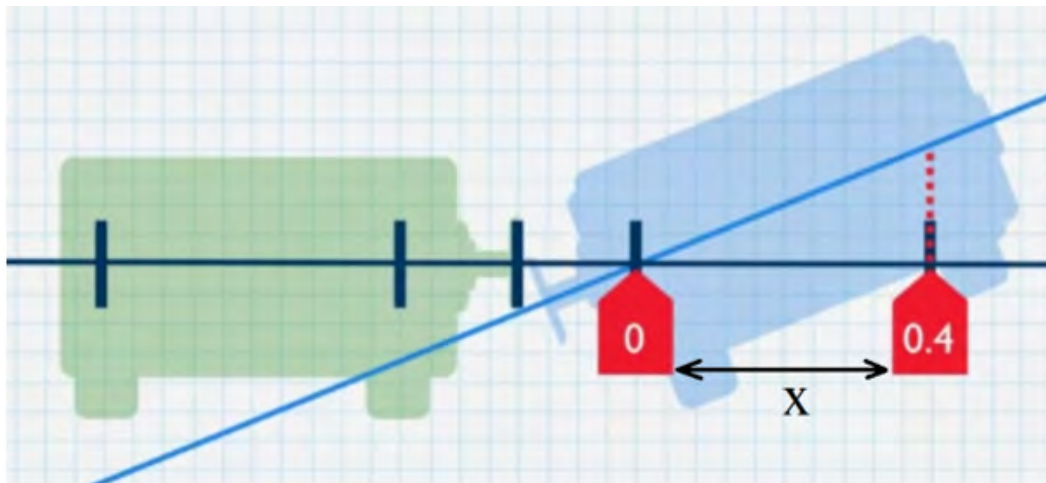


Figura 12.8: Diagrama esquemático del desalineamiento angular empleado como referencia conceptual del procedimiento experimental. Fuente: Kechik et al. (2021).

12.5.2. Correspondencia de clases para el preentrenamiento y la validación

Para permitir la compatibilidad entre los datos obtenidos en el módulo experimental y el conjunto de datos real, se construyó un esquema de tres clases para el preentrenamiento del modelo. Con el fin de evitar fronteras ambiguas entre categorías contiguas, se agruparon ángulos separados del banco experimental, dejando zonas intermedias de transición.

Cuadro 12.9: Agrupación de clases empleada en el preentrenamiento con datos del módulo experimental.

Clase	Ángulos del módulo experimental	Interpretación
1	1°, 2°, 3°, 4°	Severidad baja
2	6°, 7°, 8°, 9°	Severidad media
3	11°, 12°, 13°, 14°	Severidad alta

En el caso del *dataset* real, la correspondencia de clases utilizada durante la

validación externa fue la indicada en la Tabla 12.10.

Cuadro 12.10: Correspondencia de clases en el *dataset* real de desalineamiento angular horizontal.

Clase	Severidad real	Archivos asociados
1	0,5 mm/100 mm	1–24
2	0,8 mm/100 mm	25–48
3	1,12 mm/100 mm	49–72

12.5.3. Procedimiento de comprobación con el conjunto de datos real

El procedimiento seguido para la validación externa se resume en las siguientes etapas:

1. Selección del subconjunto Horizontal angular del conjunto de datos real.
2. Selección del canal radial horizontal como variable de entrada para la validación.
3. Segmentación de cada señal en ventanas de 1 segundo.
4. Eliminación del componente DC en cada ventana.
5. Extracción de las mismas características estadísticas empleadas en el módulo experimental.
6. Construcción de un esquema de tres clases para el preentrenamiento con datos del módulo.
7. Preentrenamiento de la red neuronal con la base experimental.
8. Ajuste fino (*fine-tuning*) con datos reales.
9. Comparación contra un modelo entrenado únicamente con datos reales desde cero.

CAPÍTULO 12. DISEÑO E IMPLEMENTACIÓN DEL ALGORITMO DE APRENDIZAJE AUTOMÁTICO

La Figura 12.9 presenta de manera esquemática el procedimiento seguido para la comprobación con el conjunto de datos real.

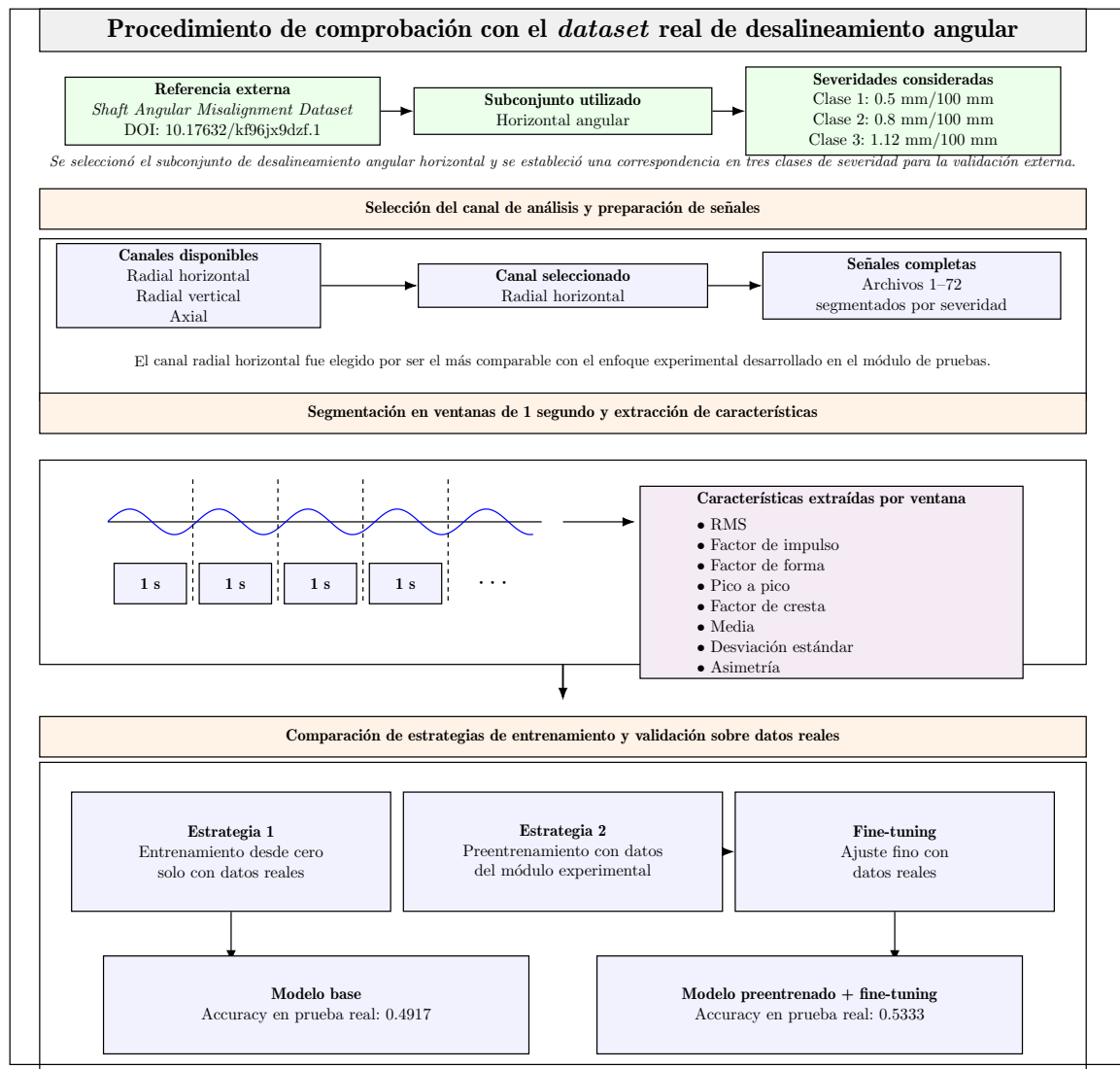


Figura 12.9: Resumen visual del procedimiento de comprobación con el conjunto de datos real, incluyendo la selección del subconjunto de desalineamiento angular horizontal, la segmentación en ventanas de 1 segundo, la extracción de características y la comparación entre entrenamiento desde cero y preentrenamiento con ajuste fino.

12.5.4. Compatibilización de señales y extracción de características

A fin de mantener la coherencia metodológica entre la fase experimental y la validación externa, las señales del conjunto real fueron procesadas con el mismo flujo utilizado para las señales obtenidas en el módulo experimental. En primer lugar, cada archivo fue segmentado en ventanas de 1 segundo. Posteriormente, a cada ventana se le removió el componente DC, con el objetivo de reducir el efecto de desplazamientos de nivel medio que no aportan información relevante a la clasificación. Finalmente, se extrajeron las mismas características estadísticas definidas en la fase de entrenamiento experimental.

12.5.5. Metodología de preentrenamiento y ajuste fino con conjunto de datos externo

El procedimiento de aprendizaje por transferencia se aplicó con el objetivo de evaluar si el conocimiento adquirido por una red neuronal a partir de las señales del módulo experimental podía mejorar su desempeño al ser ajustada y validada con un conjunto de datos externo. Para ello, se empleó una red neuronal tipo *Multilayer Perceptron* MLP, manteniendo una misma estructura de características de entrada para ambas fuentes de información.

La metodología se desarrolló en dos etapas. En la primera etapa, el modelo fue preentrenado con los datos obtenidos en el módulo experimental propio, permitiendo que la red aprendiera patrones iniciales asociados al desalineamiento angular. En la segunda etapa, el modelo preentrenado fue ajustado con el conjunto de datos externo *Shaft Angular Misalignment Dataset*, desarrollado por Kechik, Aslamov y Davydov Kechik et al. (2021). Este ajuste permitió adaptar los parámetros internos del modelo a una fuente experimental distinta.

Posteriormente, el desempeño del modelo ajustado fue comparado con el de una red neuronal entrenada directamente con el conjunto externo. Esta comparación permitió determinar el efecto del preentrenamiento sobre la precisión final, el tiempo de ajuste y la capacidad de generalización del modelo.

La Figura 12.10 resume el procedimiento general aplicado para la validación mediante aprendizaje por transferencia.

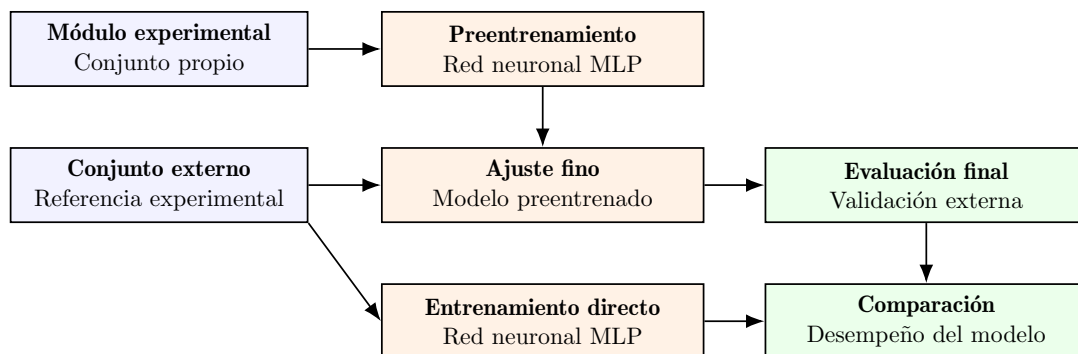


Figura 12.10: Procedimiento general de aprendizaje por transferencia aplicado al modelo de red neuronal, considerando el preentrenamiento con datos del módulo experimental, el ajuste fino con el conjunto externo y la comparación frente al entrenamiento directo.

12.5.6. Resultados de la validación externa

Con el fin de evaluar la capacidad de generalización del modelo entrenado con los datos obtenidos en el módulo experimental, se realizó una validación externa empleando un dataset real. Para ello, se compararon dos estrategias de entrenamiento, la primera consistió en entrenar el modelo únicamente con los datos reales disponibles, mientras que la segunda utilizó un preentrenamiento previo con los datos generados en el módulo de pruebas, seguido de un ajuste final con datos reales.

Los resultados obtenidos muestran que el uso del preentrenamiento permitió mejorar el desempeño del modelo en la etapa de validación externa. En términos prácticos, esto indica que el conocimiento aprendido a partir del módulo experimental aportó in-

formación útil para reconocer patrones presentes en un caso real, incluso cuando las condiciones de ambos conjuntos de datos no son exactamente iguales.

La Tabla 12.11 presenta la comparación entre ambas estrategias sobre el conjunto de prueba del *dataset* real.

Cuadro 12.11: Comparación del desempeño en la validación externa.

Estrategia evaluada	Precisión	Interpretación
Entrenamiento solo con datos reales	0.4917	Se toma como referencia base para evaluar el desempeño del modelo sin conocimiento previo del módulo experimental.
Preentrenamiento con datos del módulo + ajuste con datos reales	0.5333	Presenta el mejor resultado, lo que evidencia una mejora en la capacidad de identificación al aprovechar el aprendizaje previo obtenido con el módulo de pruebas.

De acuerdo con la Tabla 12.11, el modelo que incorporó preentrenamiento alcanzó una precisión de 0.5333, mientras que el modelo entrenado únicamente con datos reales obtuvo una precisión de 0.4917. Esto representa una mejora absoluta de 0.0416 puntos de precisión y una mejora relativa de 8.47%, calculada con respecto al modelo base entrenado solo con datos reales.

Con el propósito de facilitar la interpretación de estos resultados, en la Tabla 12.12 se resume la mejora obtenida al utilizar la técnica de preentrenamiento con datos del módulo experimental.

Cuadro 12.12: Mejora obtenida mediante la técnica de preentrenamiento.

Indicador	Valor	Descripción
Precisión sin preentrenamiento	0.4917	Resultado obtenido al entrenar el modelo únicamente con datos reales.
Precisión con preentrenamiento	0.5333	Resultado obtenido al aplicar la técnica de preentrenamiento con datos del módulo y ajuste posterior con datos reales.
Mejora absoluta	0.0416	Incremento directo en la precisión al aplicar la técnica propuesta.
Mejora relativa	8.47 %	Porcentaje de mejora logrado mediante la técnica de preentrenamiento respecto al entrenamiento sin preentrenamiento.

Desde el punto de vista ingenieril, estos resultados indican que el módulo de pruebas no solo permitió generar una base de datos útil para entrenar el modelo, sino que además aportó información previa que mejoró la respuesta del algoritmo cuando fue aplicado a un caso real. En consecuencia, la técnica de preentrenamiento puede considerarse una estrategia favorable para fortalecer la etapa de validación externa, ya que permitió incrementar la precisión del modelo en un 8.47 % respecto al entrenamiento realizado únicamente con datos reales.

La Figura 12.11 muestra de manera gráfica la mejora obtenida en la validación externa al incorporar el preentrenamiento con datos del módulo experimental.

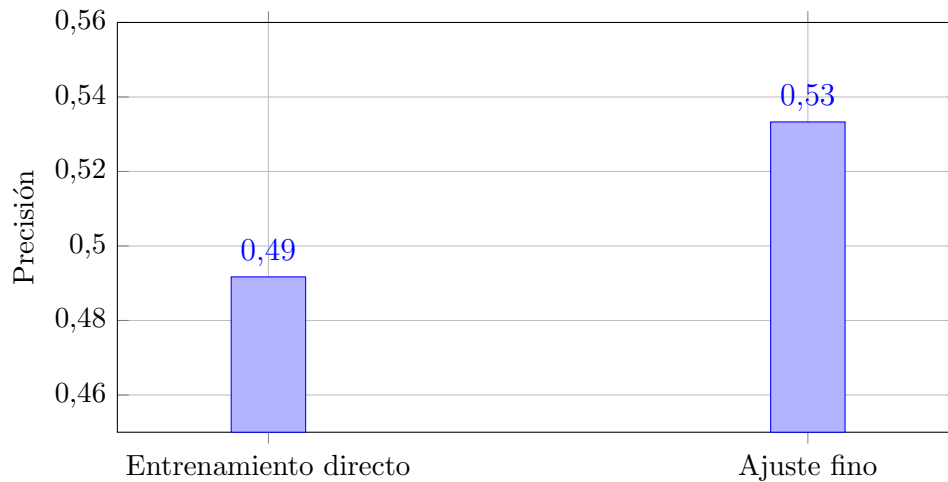


Figura 12.11: Comparación de la precisión obtenida en la validación externa. La estrategia basada en preentrenamiento y ajuste fino presentó una mejora relativa de 8.47% respecto al entrenamiento directo.

Como se aprecia en la Figura 12.11, el modelo que empleó preentrenamiento con datos provenientes del módulo de pruebas alcanzó una mayor precisión que el modelo entrenado únicamente con datos reales. Este resultado respalda la utilidad del módulo experimental como herramienta para generar conocimiento previo transferible hacia escenarios de validación más cercanos a condiciones reales.

Con respecto al tiempo de entrenamiento, se observó que la estrategia basada en preentrenamiento permitió reducir el tiempo requerido en la etapa de ajuste con datos reales. En efecto, el tiempo hasta alcanzar la mejor época durante el entrenamiento desde cero fue de 0.0366 s, mientras que en la etapa de ajuste posterior al preentrenamiento fue de 0.0272 s, lo que representa una reducción aproximada del 25.68%.

No obstante, al considerar el tiempo total del procedimiento, incluyendo la etapa previa de preentrenamiento sobre los datos del módulo experimental, el costo computacional acumulado del enfoque transferido fue mayor que el del entrenamiento realizado únicamente con datos reales. Por ello, el principal aporte de la técnica propuesta

CAPÍTULO 12. DISEÑO E IMPLEMENTACIÓN DEL ALGORITMO DE APRENDIZAJE AUTOMÁTICO

no radica en disminuir el tiempo total de cómputo, sino en mejorar el desempeño del modelo en la validación externa y en reducir el tiempo requerido específicamente en la etapa de ajuste con datos reales.

De acuerdo con la Tabla 12.13, la técnica de preentrenamiento permitió disminuir el tiempo requerido en la fase de ajuste con datos reales. Sin embargo, al incorporar la etapa previa de aprendizaje con el módulo experimental, el tiempo total del procedimiento se incrementa. En consecuencia, el beneficio principal de esta estrategia no se encuentra en la reducción del tiempo total de entrenamiento, sino en la mejora de la precisión alcanzada durante la validación con datos reales.

Cuadro 12.13: Comparación de tiempos de entrenamiento en la validación externa.

Etapa evaluada	Tiempo [s]	Interpretación
Entrenamiento desde cero con datos reales	0.0366	Tiempo requerido por el modelo base para alcanzar su mejor desempeño.
Ajuste con datos reales usando preentrenamiento	0.0272	Tiempo requerido en la etapa final de ajuste; representa una reducción del 25.68 % respecto al entrenamiento desde cero.
Tiempo total del enfoque con preentrenamiento	0.2990	Incluye el preentrenamiento y el ajuste final; por ello resulta mayor que el entrenamiento directo con datos reales.

13 | Presupuesto

En esta sección se describirán los recursos más importantes que se usaron para el desarrollo del proyecto de investigación.

Cuadro 13.1: Recursos para el desarrollo del proyecto de investigación.

Tipo	Descripción	Costo Soles
Consumible	Cinta de cobre	10.00
	Placa de acero	60.00
	Cable	5.00
Equipo	Sensor CKP	60.00
	Interfaz de audio	460.00
	Motor eléctrico	60.00
	Acople universal	6.50
	Rodamientos de pared	20.00
	Fuente de alimentación	30.00
	Pernos	10.00
Energía	Electricidad	20.00
Mano de obra	Construcción de prototipo	600.00
Movilidad	Combustible	90.00
	Pasajes transporte público	10.00
Total		1486.5

14 | Conclusiones

1. La metodología de diseño VDI 2221 aplicada permitió desarrollar un módulo de pruebas funcional y adecuado para el estudio del desalineamiento angular en ejes. El sistema construido cumplió con los requerimientos mecánicos, eléctricos y de sensorización necesarios para reproducir condiciones controladas de desalineamiento, permitiendo una operación estable durante los ensayos y constituyéndose en una plataforma experimental útil para la adquisición de datos y la validación de algoritmos de diagnóstico.
2. La comparación entre el cálculo analítico y el análisis modal computacional permitió validar el comportamiento dinámico del modelo simulado, ya que la frecuencia natural estimada se encuentra en el mismo orden y cercana al primer modo obtenido por simulación, cuyo valor fue de 79,256 Hz. Además, esta frecuencia se encuentra suficientemente alejada de la frecuencia de operación del motor, correspondiente a 940 rpm o 15,67 Hz, por lo que no se presentan condiciones de resonancia durante el funcionamiento del módulo. Esta concordancia brinda confianza en los resultados de la simulación y permite afirmar que los componentes cumplirán adecuadamente su función bajo las condiciones evaluadas, tanto en el análisis estático de esfuerzos como en el análisis dinámico. En consecuencia, los resultados obtenidos respaldan que el diseño es adecuado y que la máquina podrá operar correctamente durante los ensayos experimentales.

-
3. La etapa de captura y procesamiento de señales permitió construir una base de datos experimental consistente a partir de los registros obtenidos con el sensor CKP y la interfaz de audio. La segmentación de las señales en ventanas de un segundo y la posterior extracción de características estadísticas y espectrales permitieron transformar las señales crudas en información cuantificable y útil para el entrenamiento de modelos de aprendizaje automático. En este sentido, el procesamiento de datos constituyó una etapa clave, ya que permitió mejorar la calidad de la información de entrada y favorecer el desempeño de los algoritmos utilizados.
 4. El análisis espectral mediante la transformada rápida de Fourier permitió evaluar la influencia de la configuración de la rueda fónica en la calidad de las señales adquiridas, considerando la comparación entre las configuraciones de 4 y 10 pines. En la configuración de 4 pines se observó una mayor presencia de armónicos y componentes adicionales, lo que afectaba la limpieza del espectro y dificultaba una representación clara del fenómeno de desalineamiento. En cambio, la configuración de 10 pines presentó una respuesta FFT más definida, estable y con menor presencia de componentes indeseados, tal como se muestra en la Figura 11.2. En esta evaluación se consideró la respuesta espectral global de la señal, sin incluir resultados específicos de amplitud para los armónicos 2X, 3X, 4X y 5X. Por tanto, se concluye que la geometría de la rueda fónica influye directamente en la calidad de la data adquirida y que la configuración de 10 pines constituye la alternativa técnicamente más adecuada para la generación de la base de datos experimental empleada en la investigación.
 5. En la evaluación de los modelos de regresión sobre la base de datos experimental, los tres algoritmos implementados mostraron un desempeño satisfactorio. Random Forest presentó el mejor resultado global, con un MAE de $0,4733^\circ$, un RMSE de $1,0105^\circ$ y un coeficiente de determinación de $R^2 = 0,9540$. La red neuronal obtuvo un MAE de $0,5352^\circ$, un RMSE de $1,0158^\circ$ y un $R^2 = 0,9535$, mientras que

XGBoost alcanzó un MAE de $0,5893^\circ$, un RMSE de $1,0331^\circ$ y un $R^2 = 0,9519$. Estos resultados demuestran que los modelos aplicados a las características extraídas de las señales permiten estimar el ángulo de desalineamiento con un nivel de precisión favorable.

6. En la validación externa mediante redes neuronales, empleando como conjunto de referencia el *Shaft Angular Misalignment Dataset* de Kechik, Aslamov y Davydov Kechik et al. (2021), se compararon dos estrategias de entrenamiento. La primera correspondió al entrenamiento directo, donde la red neuronal fue entrenada únicamente con el conjunto externo, alcanzando una precisión de 49.17%. La segunda correspondió al ajuste fino con aprendizaje por transferencia, donde el modelo fue previamente entrenado con los datos obtenidos en el módulo experimental y posteriormente ajustado con el conjunto externo, alcanzando una precisión de 53.33%. Esta diferencia representa una mejora absoluta de 0.0416 puntos de precisión y una mejora relativa de 8.47% respecto al entrenamiento directo. En consecuencia, se concluye que el ajuste fino con aprendizaje por transferencia permitió mejorar el desempeño del modelo frente al entrenamiento directo, evidenciando una mayor capacidad de adaptación a un conjunto de datos externo.
7. Respecto al tiempo de ajuste, se compararon dos estrategias aplicadas sobre el conjunto externo *Shaft Angular Misalignment Dataset* de Kechik, Aslamov y Davydov Kechik et al. (2021). La primera correspondió al entrenamiento directo, en el cual la red neuronal fue entrenada únicamente con el conjunto externo, alcanzando su mejor resultado en 0,0366 s. La segunda correspondió al ajuste fino con aprendizaje por transferencia, donde el modelo fue previamente entrenado con los datos del módulo experimental y posteriormente adaptado al conjunto externo, alcanzando su mejor resultado en 0,0272 s. Esta diferencia representa una reducción de 0,0094 s, equivalente a una mejora relativa de 25,68% en la eta-

pa de ajuste respecto al entrenamiento directo. Al considerar el proceso completo de aprendizaje por transferencia, incluyendo la etapa previa de preentrenamiento con los datos del módulo experimental, el tiempo acumulado fue de 0,2990 s. Por tanto, el aporte principal del ajuste fino con aprendizaje por transferencia se evidenció en la reducción del tiempo de adaptación al conjunto externo, el incremento de la precisión final y el fortalecimiento de la capacidad de generalización del modelo frente a datos externos.

8. En conjunto, los resultados obtenidos permiten concluir que la metodología desarrollada en esta tesis, basada en el diseño y construcción de un módulo de pruebas, la captura controlada de señales, el procesamiento de datos y la aplicación de algoritmos de aprendizaje automático, sí permiten identificar de manera automática el nivel de desalineamiento angular en ejes. Asimismo, el uso de técnicas de adquisición no convencionales para este tipo de aplicaciones, como el empleo de una interfaz de audio y sensores de bajo costo, demostró ser una alternativa viable e innovadora para la captura de señales asociadas al desalineamiento. Los resultados alcanzados evidencian que es posible obtener información útil para el diagnóstico de fallas con un costo considerablemente menor al de los equipos comerciales especializados en adquisición y análisis de vibraciones. Además, la incorporación de una validación externa con un conjunto de referencia fortaleció la propuesta, al demostrar que el conocimiento aprendido en el entorno experimental puede aportar una mejora medible cuando se transfiere a un escenario distinto y más representativo del fenómeno de desalineamiento angular en ejes.

15 | Recomendaciones

1. Ampliar la base de datos experimental mediante un mayor número de ensayos, niveles de desalineamiento y condiciones de operación, con el propósito de fortalecer el entrenamiento de los modelos y mejorar su capacidad de generalización.
2. Continuar la investigación incorporando otros tipos de fallas mecánicas, tales como desalineamiento paralelo, desbalanceo u holguras, para evaluar la capacidad del modelo al diferenciar distintos fenómenos dinámicos presentes en sistemas rotativos.
3. Optimizar el módulo de pruebas para incrementar su flexibilidad experimental, considerando configuraciones de acople intercambiables, variación controlada de parámetros mecánicos y adaptación a diferentes condiciones de funcionamiento.
4. Validar la metodología desarrollada con conjuntos de datos de referencia más amplios y obtenidos en configuraciones experimentales cercanas a condiciones reales de operación, con la finalidad de evaluar la robustez del modelo y el desempeño del aprendizaje por transferencia.
5. Implementar, como trabajo futuro, una interfaz de visualización o herramienta de apoyo al diagnóstico que permita presentar de forma clara los resultados entregados por el modelo de aprendizaje automático y facilite su interpretación técnica.

Bibliografía

- (2019). *Studio 24c: Interfaz de Audio USB-C a 24-bit / 192 kHz*. PreSonus Audio Electronics, Inc., Baton Rouge, LA, USA. Manual del usuario, Part No. 70-12000127-A.
- Aguilar (2015). Conclusiones. In *Sistema de detección de fallas en las pistas de rodamientos, usando el análisis espectral de frecuencias en señales acústicas*, pages 76–77. Universidad Nacional de Trujillo.
- Alvarez (2005). Conclusiones. In *Técnicas de diagnóstico de daños incipiente en rodamientos*, pages 104–108. Universidad de Magallanes.
- Aviles, C. (2003). Ortogonalidad y series de fourier. In *Análisis de señales*, pages 19–21. Universidad Autónoma Metropolitana.
- Bendat, J. S. and Piersol, A. G. (2010). *Random Data: Analysis and Measurement Procedures*. Wiley, Hoboken, NJ, USA, 4th edition.
- Blevins, R. D. (1979). *Formulas for Natural Frequency and Mode Shape*. Van Nostrand Reinhold, New York, NY, USA.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.
- Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees*. Wadsworth International Group, Belmont, California.

- Castellanos Martínez, José Francisco (2015). *Sensores de Giro del Motor CKP y CMP*. Centro Internacional de Instrucción Técnica Automotriz (CITEC), El Salvador. Código del curso: P-SFI&OBDII-V01.
- Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM.
- Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314.
- Damalini AB (2014). *Easy-Laser Extreme: Sistema de alineación de ejes. Manual*. Damalini AB. Manual N.º art. 05-0320 Rev3, versión en español.
- GeeksforGeeks (2025). Implementation of xgboost (extreme gradient boosting). Última actualización: 5 de septiembre de 2025.
- GeeksforGeeks (2026). Random forest algorithm in machine learning. Última actualización: 2 de mayo de 2026.
- Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press, Cambridge, MA, USA.
- Guan, Z., Chen, P., Zhang, X., Zhou, X., and Li, K. (2017). Vibration analysis of shaft misalignment and diagnosis method of structure faults for rotating machinery. *International Journal of Performability Engineering*, 13(4):337–347.
- Inman, D. J. (2014). *Engineering Vibration*. Pearson Education, Boston, MA, USA, 4th edition.
- Kechik, D., Aslamov, Y., and Davydov, I. (2021). Shaft angular misalignment dataset.
- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

- Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N., and Nandi, A. K. (2020). Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mechanical Systems and Signal Processing*, 138:106587.
- Marvasti, F. (2001). Nonuniform sampling: Theory and practice. *Kluwer Academic Publishers*.
- Minsky, M. and Papert, S. (1969). *Perceptrons: An Introduction to Computational Geometry*. MIT Press, Cambridge, Massachusetts.
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, New York, NY, USA.
- Mitra, S. K. (2011). *Digital Signal Processing: A Computer-Based Approach*. McGraw-Hill, 4th edition.
- Moreno (2018). Conclusiones. In *Diagnóstico de fallas en rodamientos utilizando vibraciones mecánicas*, pages 69–70. Universidad de Antioquía.
- Mädler GmbH (s.f.). *Ball Flange Bearings KFL and SSKFL, Light Series*. Mädler GmbH. Ficha técnica del rodamiento KFL08, diámetro interior de 8 mm. Capacidad dinámica $C = 4.2$ kN y capacidad estática $C_0 = 1.6$ kN.
- Navarro-Navarro, A., Biot-Monterde, V., Ruiz-Sarrio, J. E., and Antonino-Daviu, J. A. (2025). Current- and vibration-based detection of misalignment faults in synchronous reluctance motors. *Machines*, 13(4):319.
- Norton, R. L. (2020). *Design of Machinery: An Introduction to the Synthesis and Analysis of Mechanisms and Machines*. McGraw-Hill Education, 6 edition.
- Nyquist, H. (1928). Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644.
- Olabe, X. B. (s. f.). *Redes neuronales artificiales y sus aplicaciones*. Escuela Superior de Ingeniería de Bilbao, Universidad del País Vasco/Euskal Herriko Unibertsitatea, Bilbao, España. Material del curso.

- Oppenheim, A. V., Willsky, A. S., and Nawab, S. H. (1997). *Signals and Systems*. Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition.
- Palomino, E., Taveras Lay, F., Cabrera Gómez, J., and Castillo Serpa, A. M. d. (2023). Influence of misalignment on functional dynamic parameters of a 250 kw direct current motor. *Revista Técnica*.
- Papoulis, A. and Pillai, S. U. (2002). *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, New York, NY, USA, 4th edition.
- Proakis, J. G. and Manolakis, D. G. (2007). *Digital Signal Processing: Principles, Algorithms, and Applications*. Pearson Education, Upper Saddle River, NJ, USA, 4th edition.
- Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, 1(1):81–106.
- Rao, S. S. (2017). *Mechanical Vibrations*. Pearson Education, Upper Saddle River, NJ, USA, 6th edition.
- Richard G. Budynas, J. K. N. (2012). Embragues, frenos, coples y volantes de inercia. In *Diseño en ingeniería mecánica*, pages 795–847. Mc Graw Hill.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536.
- scikit-learn developers (2026). Understanding the decision tree structure. Documentación oficial de scikit-learn.
- Shannon, C. E. (1949). Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21.

- Simm, A., Wang, Q., Huang, S., and Zhao, W. (2016). Laser based measurement for the monitoring of shaft misalignment. *Measurement*, 87:104–116.
- SKF (2025). Skf rating life. Consultado en 2026.
- Szewczyk, S., Barczewski, R., Staszewski, W. J., Janiga, D., and Dao, P. B. (2025). Cointegration approach for vibration-based misalignment detection in rotating machinery under varying load conditions. *Sensors*, 25(21):6764.
- Uicker, J. J., Pennock, G. R., and Shigley, J. E. (2017). *Theory of Machines and Mechanisms*. Oxford University Press, 5 edition.
- Unser, M. (2000). Sampling—50 years after shannon. *Proceedings of the IEEE*, 88(4):569–587.
- White (2010). Desalineación. In *Introducción al Análisis de Vibraciones*, pages 109–11. Azima DLI.
- Widrow, B. and Hoff, M. E. (1960). Adaptive switching circuits. In *IRE WESCON Convention Record*, volume 4, pages 96–104. Institute of Radio Engineers.
- Xu, M. and Marangoni, R. D. (1994). Vibration analysis of a motor-flexible coupling-rotor system subject to misalignment. *Journal of Sound and Vibration*, 176(5):663–679.

A | Anexo A: CÓDIGO FUENTE RANDOM FOREST

```
1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #ALGORITMO DE REGRESIÓN POR EL MÉTODO "RANDOM FOREST"---#
8 import re
9 import numpy as np
10 import pandas as pd
11 from pathlib import Path
12 from sklearn.model_selection import train_test_split
13 from sklearn.ensemble import RandomForestRegressor
14 from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
15
16 FEATURES_DIR = Path(r"D:\Data tesis\nueva medición\features_1s_dco")
17
18 def load_dataset(features_dir: Path):
19     files = [p for p in features_dir.iterdir() if p.is_file() and
20 p.name.lower().startswith("pista_") and "features" in p.name.lower()]
21     if not files:
22         raise FileNotFoundError(f"No encontré archivos pista_*_features en:
23 {features_dir}")
24
25     rows = []
26     for f in sorted(files):
27         m = re.search(r"pista_(\d+)", f.name.lower())
28         if not m:
29             continue
30         angle = float(m.group(1))
31         df = pd.read_csv(f)
32         df["angle_deg"] = angle
33         rows.append(df)
34
35     data = pd.concat(rows, ignore_index=True)
36     drop_cols = {"angle_deg", "track", "file"}
37     X = data.drop(columns=[c for c in drop_cols if c in data.columns],
38 errors="ignore")
39     X = X.select_dtypes(include=[np.number])
40     y = data["angle_deg"].to_numpy()
41     return X, y
42
43 X, y = load_dataset(FEATURES_DIR)
44 print(f"Dataset: {X.shape[0]} filas, {X.shape[1]} features, y únicos =
45 {sorted(set(y))}")
46
47 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
48 random_state=42)
49
50 rf = RandomForestRegressor(
51     n_estimators=800,
52     random_state=42,
53     n_jobs=-1
54 )
55 rf.fit(X_train, y_train)
56 pred = rf.predict(X_test)
57
58 mae = mean_absolute_error(y_test, pred)
59 mse = mean_squared_error(y_test, pred)
```

```
56 rmse = float(np.sqrt(mse))
57 r2 = r2_score(y_test, pred)
58
59 print("=== Random Forest Regressor ===")
60 print(f"MAE = {mae:.4f} deg")
61 print(f"RMSE = {rmse:.4f} deg")
62 print(f"R2 = {r2:.4f}")
63
```

B | Anexo B: CÓDIGO FUENTE XG BOOST

```
1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #ALGORITMO DE REGRESIÓN POR EL MÉTODO "xg boost"---#
8 import re
9 import numpy as np
10 import pandas as pd
11 from pathlib import Path
12 from sklearn.model_selection import train_test_split
13 from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
14
15 try:
16     from xgboost import XGBRegressor
17 except ImportError:
18     raise SystemExit("Falta xgboost. Instala con: pip install xgboost")
19
20 FEATURES_DIR = Path(r"D:\Data tesis\nueva medición\features_1s_dco")
21
22 def load_dataset(features_dir: Path):
23     files = [p for p in features_dir.iterdir() if p.is_file() and
24             p.name.lower().startswith("pista_") and "features" in p.name.lower()]
25     if not files:
26         raise FileNotFoundError(f"No encontré archivos pista_*_features en:
27 {features_dir}")
28
29     rows = []
30     for f in sorted(files):
31         m = re.search(r"pista_(\d+)", f.name.lower())
32         if not m:
33             continue
34         angle = float(m.group(1))
35         df = pd.read_csv(f)
36         df["angle_deg"] = angle
37         rows.append(df)
38
39     data = pd.concat(rows, ignore_index=True)
40     drop_cols = {"angle_deg", "track", "file"}
41     X = data.drop(columns=[c for c in drop_cols if c in data.columns],
42 errors="ignore")
43     X = X.select_dtypes(include=[np.number])
44     y = data["angle_deg"].to_numpy()
45     return X, y
46
47 X, y = load_dataset(FEATURES_DIR)
48 print(f"Dataset: {X.shape[0]} filas, {X.shape[1]} features, y únicos =
49 {sorted(set(y))}")
50
51 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
52 random_state=42)
53
54 xgb = XGBRegressor(
55     n_estimators=1200,
56     learning_rate=0.03,
57     max_depth=5,
58     subsample=0.9,
59     colsample_bytree=0.9,
60     reg_lambda=1.0,
```

```
56     random_state=42,  
57     n_jobs=-1  
58 )  
59  
60 xgb.fit(X_train, y_train)  
61 pred = xgb.predict(X_test)  
62  
63 mae = mean_absolute_error(y_test, pred)  
64 mse = mean_squared_error(y_test, pred)  
65 rmse = float(np.sqrt(mse))  
66 r2 = r2_score(y_test, pred)  
67  
68 print("=== XGBoost Regressor ===")  
69 print(f"MAE = {mae:.4f} deg")  
70 print(f"RMSE = {rmse:.4f} deg")  
71 print(f"R2 = {r2:.4f}")  
72
```

C | Anexo C: CÓDIGO FUENTE RED NEURONAL

```
1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #ALGORITMO DE REGRESIÓN POR EL MÉTODO "RED NEURONAL"---#
8 import re
9 import numpy as np
10 import pandas as pd
11 from pathlib import Path
12 from sklearn.model_selection import train_test_split
13 from sklearn.pipeline import Pipeline
14 from sklearn.preprocessing import StandardScaler
15 from sklearn.neural_network import MLPRegressor
16 from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
17
18 FEATURES_DIR = Path(r"D:\Data tesis\nueva medición\features_1s_dco")
19
20 def load_dataset(features_dir: Path):
21     files = [p for p in features_dir.iterdir() if p.is_file() and
22 p.name.lower().startswith("pista_") and "features" in p.name.lower()]
23     if not files:
24         raise FileNotFoundError(f"No encontré archivos pista_*_features en:
25 {features_dir}")
26
27     rows = []
28     for f in sorted(files):
29         m = re.search(r"pista_(\d+)", f.name.lower())
30         if not m:
31             continue
32         angle = float(m.group(1))
33         df = pd.read_csv(f)
34         df["angle_deg"] = angle
35         rows.append(df)
36
37     data = pd.concat(rows, ignore_index=True)
38     drop_cols = {"angle_deg", "track", "file"}
39     X = data.drop(columns=[c for c in drop_cols if c in data.columns],
40 errors="ignore")
41     X = X.select_dtypes(include=[np.number])
42     y = data["angle_deg"].to_numpy()
43     return X, y
44
45 X, y = load_dataset(FEATURES_DIR)
46 print(f"Dataset: {X.shape[0]} filas, {X.shape[1]} features, y únicos =
47 {sorted(set(y))}")
48
49 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
50 random_state=42)
51
52 nn = Pipeline([
53     ("scaler", StandardScaler()),
54     ("mlp", MLPRegressor(
55         hidden_layer_sizes=(128, 64, 32),
56         activation="relu",
57         solver="adam",
58         alpha=1e-4,
59         learning_rate_init=1e-3,
60         max_iter=800,
```

```
56         random_state=42
57     ))
58 ])
59
60 nn.fit(X_train, y_train)
61 pred = nn.predict(X_test)
62
63 mae = mean_absolute_error(y_test, pred)
64 mse = mean_squared_error(y_test, pred)
65 rmse = float(np.sqrt(mse))
66 r2 = r2_score(y_test, pred)
67
68 print("=== Neural Network: MLPRegressor ===")
69 print(f"MAE = {mae:.4f} deg")
70 print(f"RMSE = {rmse:.4f} deg")
71 print(f"R2 = {r2:.4f}")
72
```

**D | Anexo D: CÓDIGO FUENTE DE
EXTRACCIÓN DE CARACTERÍS-
TICAS**

```
1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #EXTRACCIÓN DE CARACTERÍSTICAS#
8 import os
9 import pandas as pd
10 import numpy as np
11 from scipy.stats import skew
12
13 # --- Funciones de extracción de features ---
14 def rms(x):
15     return np.sqrt(np.mean(x**2))
16
17 def impulse_factor(x):
18     denom = np.mean(np.abs(x))
19     return np.max(np.abs(x)) / denom if denom != 0 else np.nan
20
21 def shape_factor(x):
22     denom = np.mean(np.abs(x))
23     return rms(x) / denom if denom != 0 else np.nan
24
25 def peak_to_peak(x):
26     return np.max(x) - np.min(x)
27
28 def crest_factor(x):
29     r = rms(x)
30     return np.max(np.abs(x)) / r if r != 0 else np.nan
31
32 def mean_val(x):
33     return np.mean(x)
34
35 def std_val(x):
36     return np.std(x, ddof=0)
37
38 def skewness(x):
39     return skew(x)
40
41
42 # --- Rutas base ---
43 base_dir = r"C:\Users\intel\OneDrive\Documentos\mediciones_tesis\nueva medición"
44 src_dir = os.path.join(base_dir, "csv_1s_dco") # <-- data sin DC
45 out_dir = os.path.join(base_dir, "features_1s_dco") # salida features
46
47 os.makedirs(out_dir, exist_ok=True)
48
49 # --- Buscar pistas ---
50 track_folders = sorted(
51     d for d in os.listdir(src_dir)
52     if os.path.isdir(os.path.join(src_dir, d)) and d.lower().startswith("pista_")
53 )
54
55 if not track_folders:
56     raise FileNotFoundError(f"No se encontraron subcarpetas 'pista_*' en: {src_dir}")
57
58 # --- Procesamiento ---
59 for track_folder in track_folders:
60     track_path = os.path.join(src_dir, track_folder)
```

```
61     records = []
62
63     csv_files = sorted(f for f in os.listdir(track_path) if
64 f.lower().endswith(".csv"))
65     if not csv_files:
66         print(f"⚠ {track_folder}: no tiene CSV, se omite.")
67         continue
68
69     for fname in csv_files:
70         df = pd.read_csv(os.path.join(track_path, fname))
71
72         if "amplitude" in df.columns:
73             x = df["amplitude"].to_numpy()
74         else:
75             x = df.iloc[:, -1].to_numpy()
76
77         records.append({
78             "track":      track_folder,
79             "file":      fname,
80             "rms":       rms(x),
81             "impulse_factor": impulse_factor(x),
82             "shape_factor": shape_factor(x),
83             "peak_to_peak": peak_to_peak(x),
84             "crest_factor": crest_factor(x),
85             "mean":     mean_val(x),
86             "std":     std_val(x),
87             "skewness": skewness(x)
88         })
89
90     df_feats = pd.DataFrame(records)
91     out_csv = os.path.join(out_dir, f"{track_folder}_features.csv")
92     df_feats.to_csv(out_csv, index=False)
93     print(f"✓ {track_folder}: Exportado {len(df_feats)} filas en: {out_csv}")
94 # --- Consolidado total ---
95 all_feats = []
96 for f in sorted(os.listdir(out_dir)):
97     if f.lower().endswith("_features.csv"):
98         all_feats.append(pd.read_csv(os.path.join(out_dir, f)))
99
100 if all_feats:
101     df_all = pd.concat(all_feats, ignore_index=True)
102     out_all = os.path.join(out_dir, "ALL_tracks_features.csv")
103     df_all.to_csv(out_all, index=False)
104     print(f"✅ Consolidado total: {len(df_all)} filas en: {out_all}")
105
```

**E | Anexo E: CÓDIGO FUENTE DEL
ACONDICIONAMIENTO DE SE-
ÑAL**

```
1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #ACONDICIONAMIENTO DE DATASET EXPERIMENTAL-#
8 import json
9 import numpy as np
10 import pandas as pd
11 from pathlib import Path
12
13 REAL_FILE = Path(r"D:\Data tesis\nueva
medición\features_1s_dco_exp\ALL_tracks_features.csv")
14 OUT_DIR = Path(r"D:\Data tesis\nueva medición\pretrain_transfer_learning")
15 OUT_DIR.mkdir(parents=True, exist_ok=True)
16
17 REAL_SEVERITY_TO_CLASS = {
18     0.5: 1,
19     0.8: 2,
20     1.12: 3
21 }
22
23 CHANNEL_TO_USE = "radial_horizontal"
24 RANDOM_STATE = 42
25
26 def split_by_source_within_class(df_class, train_ratio=0.6, val_ratio=0.2):
27     sources = sorted(df_class["source_file"].unique())
28     rng = np.random.default_rng(RANDOM_STATE)
29     sources = np.array(sources)
30     rng.shuffle(sources)
31
32     n = len(sources)
33     n_train = max(1, int(round(n * train_ratio)))
34     n_val = max(1, int(round(n * val_ratio)))
35
36     if n_train + n_val >= n:
37         n_train = max(1, n - 2)
38         n_val = 1
39
40     train_sources = set(sources[:n_train])
41     val_sources = set(sources[n_train:n_train + n_val])
42     test_sources = set(sources[n_train + n_val:])
43
44     # respaldo si algo queda vacío
45     if len(test_sources) == 0:
46         moved = next(iter(val_sources))
47         val_sources.remove(moved)
48         test_sources.add(moved)
49
50     df_train = df_class[df_class["source_file"].isin(train_sources)].copy()
51     df_val = df_class[df_class["source_file"].isin(val_sources)].copy()
52     df_test = df_class[df_class["source_file"].isin(test_sources)].copy()
53
54     return df_train, df_val, df_test
55
56 def main():
57     df = pd.read_csv(REAL_FILE)
58
59     if "channel_name" in df.columns:
```

```
60     df = df[df["channel_name"].astype(str).str.lower() == CHANNEL_TO_USE].copy()
61     else:
62         df = df[df["track"].astype(str).str.lower() == "pista_0"].copy()
63
64     if df.empty:
65         raise RuntimeError("No hay filas del canal radial_horizontal en el dataset
real.")
66
67     df["class_label"] = df["severity_mm_per_100mm"].map(REAL_SEVERITY_TO_CLASS)
68     df = df[df["class_label"].notna()].copy()
69     df["class_label"] = df["class_label"].astype(int)
70
71     if "source_file" not in df.columns:
72         raise RuntimeError("El dataset real necesita la columna source_file para
partir por archivo.")
73
74     train_parts, val_parts, test_parts = [], [], []
75
76     for cls in [1, 2, 3]:
77         dfc = df[df["class_label"] == cls].copy()
78         tr, va, te = split_by_source_within_class(dfc)
79         train_parts.append(tr)
80         val_parts.append(va)
81         test_parts.append(te)
82
83     df_train = pd.concat(train_parts, ignore_index=True)
84     df_val = pd.concat(val_parts, ignore_index=True)
85     df_test = pd.concat(test_parts, ignore_index=True)
86
87     df_train.to_csv(OUT_DIR / "real_train.csv", index=False)
88     df_val.to_csv(OUT_DIR / "real_val.csv", index=False)
89     df_test.to_csv(OUT_DIR / "real_test.csv", index=False)
90
91     summary = {
92         "train_counts":
df_train["class_label"].value_counts().sort_index().to_dict(),
93         "val_counts": df_val["class_label"].value_counts().sort_index().to_dict(),
94         "test_counts": df_test["class_label"].value_counts().sort_index().to_dict(),
95         "train_rows": int(len(df_train)),
96         "val_rows": int(len(df_val)),
97         "test_rows": int(len(df_test))
98     }
99
100     with open(OUT_DIR / "real_split_summary.json", "w", encoding="utf-8") as f:
101         json.dump(summary, f, indent=2)
102
103     print("✅ Guardados real_train.csv, real_val.csv y real_test.csv")
104     print(summary)
105
106 if __name__ == "__main__":
107     main()
```

**F | Anexo F: CÓDIGO FUENTE DEL
PREENTRENAMIENTO DEL MO-
DELO**

```
1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #PREENTRENAMIENTO MEDIANTE ENSAYOS EN MODULO DE PRUEBAS #
8 import json
9 import time
10 import joblib
11 import numpy as np
12 import pandas as pd
13 from pathlib import Path
14 from sklearn.preprocessing import StandardScaler
15 from sklearn.neural_network import MLPClassifier
16 from sklearn.metrics import accuracy_score, f1_score
17
18 BASE_DIR = Path(r"D:\Data tesis\nueva medición\pretrain_transfer_learning")
19 MODEL_DIR = BASE_DIR / "models_v2"
20 MODEL_DIR.mkdir(parents=True, exist_ok=True)
21
22 MODULE_FILE = BASE_DIR / "module_pretrain_3classes.csv"
23
24 EPOCHS = 40
25 RANDOM_STATE = 42
26 FEATURE_EXCLUDE = {"class_label", "angle_deg", "source_name"}
27
28 def main():
29     df = pd.read_csv(MODULE_FILE)
30
31     feature_cols = [
32         c for c in df.columns
33         if c not in FEATURE_EXCLUDE and pd.api.types.is_numeric_dtype(df[c])
34     ]
35
36     X = df[feature_cols].copy().to_numpy(dtype=float)
37     y = df["class_label"].to_numpy(dtype=int)
38
39     scaler = StandardScaler()
40     Xs = scaler.fit_transform(X)
41
42     clf = MLPClassifier(
43         hidden_layer_sizes=(64, 32),
44         activation="relu",
45         solver="adam",
46         alpha=1e-4,
47         learning_rate_init=1e-3,
48         max_iter=1,
49         warm_start=True,
50         random_state=RANDOM_STATE
51     )
52
53     classes = np.array([1, 2, 3], dtype=int)
54
55     history = []
56     t0 = time.perf_counter()
57
58     for epoch in range(1, EPOCHS + 1):
59         clf.partial_fit(Xs, y, classes=classes)
60         pred = clf.predict(Xs)
```

```
61
62     acc = accuracy_score(y, pred)
63     f1m = f1_score(y, pred, average="macro")
64     elapsed = time.perf_counter() - t0
65
66     history.append({
67         "epoch": epoch,
68         "acc_train": acc,
69         "f1_macro_train": f1m,
70         "cumulative_time_s": elapsed
71     })
72
73     if epoch % 5 == 0 or epoch == 1:
74         print(f"Epoch {epoch:03d} | acc={acc:.4f} | f1={f1m:.4f} | t=
75 {elapsed:.2f}s")
76
77     pd.DataFrame(history).to_csv(MODEL_DIR / "pretrain_history.csv", index=False)
78     joblib.dump(scaler, MODEL_DIR / "pretrain_scaler.pkl")
79     joblib.dump(feature_cols, MODEL_DIR / "feature_cols.pkl")
80     joblib.dump(clf, MODEL_DIR / "pretrained_mlp.pkl")
81
82     metrics = {
83         "final_epoch": EPOCHS,
84         "final_acc_train": float(history[-1]["acc_train"]),
85         "final_f1_macro_train": float(history[-1]["f1_macro_train"]),
86         "total_pretrain_time_s": float(history[-1]["cumulative_time_s"])
87     }
88
89     with open(MODEL_DIR / "pretrain_metrics.json", "w", encoding="utf-8") as f:
90         json.dump(metrics, f, indent=2)
91
92     print("✅ Preentrenamiento v2 guardado.")
93     print(metrics)
94 if __name__ == "__main__":
95     main()
```

**G | Anexo G: CÓDIGO FUENTE DEL
AJUSTE FINO DEL MODELO**

```
1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #AJUSTE FINO MEDIANTE PREENTRENAMIENTO#
8 import json
9 import time
10 import copy
11 import joblib
12 import numpy as np
13 import pandas as pd
14 from pathlib import Path
15 from sklearn.preprocessing import StandardScaler
16 from sklearn.metrics import accuracy_score, f1_score, classification_report,
17 confusion_matrix
18 BASE_DIR = Path(r"D:\Data tesis\nueva medición\pretrain_transfer_learning")
19 MODEL_DIR = BASE_DIR / "models_v2"
20 OUT_DIR = BASE_DIR / "finetuned_v2_real_scaler"
21 OUT_DIR.mkdir(parents=True, exist_ok=True)
22
23 TRAIN_FILE = BASE_DIR / "real_train.csv"
24 VAL_FILE = BASE_DIR / "real_val.csv"
25 TEST_FILE = BASE_DIR / "real_test.csv"
26
27 EPOCHS_REAL = 120
28
29 def load_xy(df, feature_cols):
30     X = df[feature_cols].copy().to_numpy(dtype=float)
31     y = df["class_label"].to_numpy(dtype=int)
32     return X, y
33
34 def main():
35     df_train = pd.read_csv(TRAIN_FILE)
36     df_val = pd.read_csv(VAL_FILE)
37     df_test = pd.read_csv(TEST_FILE)
38
39     feature_cols = joblib.load(MODEL_DIR / "feature_cols.pkl")
40     clf = joblib.load(MODEL_DIR / "pretrained_mlp.pkl")
41
42     X_train, y_train = load_xy(df_train, feature_cols)
43     X_val, y_val = load_xy(df_val, feature_cols)
44     X_test, y_test = load_xy(df_test, feature_cols)
45
46     scaler = StandardScaler()
47     X_train_s = scaler.fit_transform(X_train)
48     X_val_s = scaler.transform(X_val)
49     X_test_s = scaler.transform(X_test)
50
51     classes = np.array([1, 2, 3], dtype=int)
52
53     history = []
54     best_val_acc = -1.0
55     best_epoch = -1
56     best_clf = None
57
58     t0 = time.perf_counter()
59
```

```

60     for epoch in range(1, EPOCHS_REAL + 1):
61         clf.partial_fit(X_train_s, y_train, classes=classes)
62
63         pred_val = clf.predict(X_val_s)
64         val_acc = accuracy_score(y_val, pred_val)
65         val_f1 = f1_score(y_val, pred_val, average="macro")
66         elapsed = time.perf_counter() - t0
67
68         history.append({
69             "epoch": epoch,
70             "val_acc": val_acc,
71             "val_f1_macro": val_f1,
72             "cumulative_time_s": elapsed
73         })
74
75         if val_acc > best_val_acc:
76             best_val_acc = val_acc
77             best_epoch = epoch
78             best_clf = copy.deepcopy(clf)
79
80         if epoch % 10 == 0 or epoch == 1:
81             print(f"Epoch {epoch:03d} | val_acc={val_acc:.4f} | val_f1={val_f1:.4f} |
t={elapsed:.2f}s")
82
83         pred_test = best_clf.predict(X_test_s)
84         test_acc = accuracy_score(y_test, pred_test)
85         test_f1 = f1_score(y_test, pred_test, average="macro")
86         cm = confusion_matrix(y_test, pred_test, labels=[1, 2, 3])
87         report = classification_report(y_test, pred_test, labels=[1, 2, 3], digits=4,
zero_division=0)
88
89         pd.DataFrame(history).to_csv(OUT_DIR / "finetune_history.csv", index=False)
90         joblib.dump(best_clf, OUT_DIR / "finetuned_mlp.pkl")
91         joblib.dump scaler, OUT_DIR / "real_scaler.pkl")
92
93         time_at_best = float(pd.DataFrame(history).loc[
94             pd.DataFrame(history)["epoch"] == best_epoch, "cumulative_time_s"
95         ].iloc[0])
96
97         metrics = {
98             "best_epoch": int(best_epoch),
99             "best_val_acc": float(best_val_acc),
100             "time_at_best_epoch_s": time_at_best,
101             "total_finetune_time_s": float(history[-1]["cumulative_time_s"]),
102             "test_acc": float(test_acc),
103             "test_f1_macro": float(test_f1),
104             "scaler_mode": "real_train_scaler"
105         }
106
107         with open(OUT_DIR / "finetune_metrics.json", "w", encoding="utf-8") as f:
108             json.dump(metrics, f, indent=2)
109
110         with open(OUT_DIR / "finetune_report.txt", "w", encoding="utf-8") as f:
111             f.write("Confusion matrix:\n")
112             f.write(np.array2string(cm))
113             f.write("\n\nClassification report:\n")
114             f.write(report)
115
116         print("✅ Fine-tuning v2 con scaler real terminado.")
117         print(metrics)

```

```
118     print(report)
119
120 if __name__ == "__main__":
121     main()
```

**H | Anexo H: CÓDIGO FUENTE DEL
ENTRENAMIENTO CON EL CON-
JUNTO DE REFERENCIA**

```
1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #ENTRENAMIENTO MEDIANTE DATA EXPERIMENTAL REAL#
8 import json
9 import time
10 import joblib
11 import numpy as np
12 import pandas as pd
13 from pathlib import Path
14 from sklearn.preprocessing import StandardScaler
15 from sklearn.neural_network import MLPClassifier
16 from sklearn.metrics import accuracy_score, f1_score, classification_report,
    confusion_matrix
17
18 BASE_DIR = Path(r"D:\Data tesis\nueva medición\pretrain_transfer_learning")
19 MODEL_DIR = BASE_DIR / "scratch_real"
20 MODEL_DIR.mkdir(parents=True, exist_ok=True)
21
22 TRAIN_FILE = BASE_DIR / "real_train.csv"
23 VAL_FILE = BASE_DIR / "real_val.csv"
24 TEST_FILE = BASE_DIR / "real_test.csv"
25
26 EPOCHS_REAL = 120
27 RANDOM_STATE = 42
28 EXCLUDE_COLS = {
29     "class_label", "severity_mm_per_100mm", "window_index", "source_file_num",
30     "fs", "n_samples"
31 }
32
33 def get_feature_cols(df):
34     cols = []
35     for c in df.columns:
36         if c in EXCLUDE_COLS:
37             continue
38         if pd.api.types.is_numeric_dtype(df[c]):
39             cols.append(c)
40     return cols
41
42 def load_xy(df, feature_cols):
43     X = df[feature_cols].copy().to_numpy(dtype=float)
44     y = df["class_label"].to_numpy(dtype=int)
45     return X, y
46
47 def main():
48     df_train = pd.read_csv(TRAIN_FILE)
49     df_val = pd.read_csv(VAL_FILE)
50     df_test = pd.read_csv(TEST_FILE)
51
52     feature_cols = get_feature_cols(df_train)
53
54     X_train, y_train = load_xy(df_train, feature_cols)
55     X_val, y_val = load_xy(df_val, feature_cols)
56     X_test, y_test = load_xy(df_test, feature_cols)
57
58     scaler = StandardScaler()
59     X_train_s = scaler.fit_transform(X_train)
```

```

60 X_val_s = scaler.transform(X_val)
61 X_test_s = scaler.transform(X_test)
62
63 clf = MLPClassifier(
64     hidden_layer_sizes=(128, 64, 32),
65     activation="relu",
66     solver="adam",
67     alpha=1e-4,
68     learning_rate_init=1e-3,
69     max_iter=1,
70     warm_start=True,
71     random_state=RANDOM_STATE
72 )
73
74 classes = np.array([1, 2, 3], dtype=int)
75
76 history = []
77 best_val_acc = -1.0
78 best_epoch = -1
79 best_model = None
80
81 t0 = time.perf_counter()
82
83 for epoch in range(1, EPOCHS_REAL + 1):
84     clf.partial_fit(X_train_s, y_train, classes=classes)
85
86     pred_val = clf.predict(X_val_s)
87     val_acc = accuracy_score(y_val, pred_val)
88     val_f1 = f1_score(y_val, pred_val, average="macro")
89     elapsed = time.perf_counter() - t0
90
91     history.append({
92         "epoch": epoch,
93         "val_acc": val_acc,
94         "val_f1_macro": val_f1,
95         "cumulative_time_s": elapsed
96     })
97
98     if val_acc > best_val_acc:
99         best_val_acc = val_acc
100         best_epoch = epoch
101         best_model = joblib.dumps(clf) if hasattr(joblib, "dumps") else None
102         if best_model is None:
103             joblib.dump(clf, MODEL_DIR / "tmp_best_scratch.pkl")
104
105     if epoch % 10 == 0 or epoch == 1:
106         print(f"Epoch {epoch:03d} | val_acc={val_acc:.4f} | val_f1={val_f1:.4f} |
t={elapsed:.2f}s")
107
108     if best_model is not None:
109         best_clf = joblib.loads(best_model)
110     else:
111         best_clf = joblib.load(MODEL_DIR / "tmp_best_scratch.pkl")
112
113     pred_test = best_clf.predict(X_test_s)
114     test_acc = accuracy_score(y_test, pred_test)
115     test_f1 = f1_score(y_test, pred_test, average="macro")
116     cm = confusion_matrix(y_test, pred_test, labels=[1, 2, 3])
117     report = classification_report(y_test, pred_test, labels=[1, 2, 3], digits=4,
zero_division=0)

```

```
118
119 pd.DataFrame(history).to_csv(MODEL_DIR / "scratch_history.csv", index=False)
120 joblib.dump(best_clf, MODEL_DIR / "scratch_mlp.pkl")
121 joblib.dump(scaler, MODEL_DIR / "scratch_scaler.pkl")
122 joblib.dump(feature_cols, MODEL_DIR / "scratch_feature_cols.pkl")
123
124 metrics = {
125     "best_epoch": int(best_epoch),
126     "best_val_acc": float(best_val_acc),
127     "time_at_best_epoch_s": float(pd.DataFrame(history).loc[pd.DataFrame(history)
["epoch"] == best_epoch, "cumulative_time_s"].iloc[0]),
128     "total_scratch_time_s": float(history[-1]["cumulative_time_s"]),
129     "test_acc": float(test_acc),
130     "test_f1_macro": float(test_f1)
131 }
132
133 with open(MODEL_DIR / "scratch_metrics.json", "w", encoding="utf-8") as f:
134     json.dump(metrics, f, indent=2)
135
136 with open(MODEL_DIR / "scratch_report.txt", "w", encoding="utf-8") as f:
137     f.write("Confusion matrix:\n")
138     f.write(np.array2string(cm))
139     f.write("\n\nClassification report:\n")
140     f.write(report)
141
142 print("✅ Entrenamiento desde cero terminado.")
143 print(metrics)
144 print(report)
145
146 if __name__ == "__main__":
147     main()
```

I | Anexo I: CÓDIGO FUENTE DE LA PREPARACIÓN INICIAL DE LOS DATOS

```

1 #
2 # UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO---#
3 #ESCUELA PROFESIONAL DE INGENIERIA MECANICA---#
4 #Autor: ESTRADA ABARCA, Wilfredo Omar---#
5 #Co-Autor: CAMPANA SOTELO, Herbert Augusto---##
6 #
7 #PREPARACION DE DATA DE MODULO DE PRUEBAS PARA AJUSTE FINO#
8 import re
9 import json
10 import numpy as np
11 import pandas as pd
12 from pathlib import Path
13
14 FEATURES_DIR = Path(r"D:\Data tesis\nueva medición\features_1s_dco")
15 OUT_DIR = Path(r"D:\Data tesis\nueva medición\pretrain_transfer_learning")
16 OUT_DIR.mkdir(parents=True, exist_ok=True)
17
18 # Clases separadas
19 ANGLE_TO_CLASS = {
20     1.0: 1, 2.0: 1, 3.0: 1, 4.0: 1,
21     6.0: 2, 7.0: 2, 8.0: 2, 9.0: 2,
22     11.0: 3, 12.0: 3, 13.0: 3, 14.0: 3
23 }
24
25 def main():
26     files = sorted([
27         p for p in FEATURES_DIR.iterdir()
28         if p.is_file() and p.name.lower().startswith("pista_") and "features" in
29         p.name.lower()
30     ])
31
32     if not files:
33         raise FileNotFoundError(f"No encontré archivos pista_*_features en:
34         {FEATURES_DIR}")
35
36     rows = []
37     for f in files:
38         m = re.search(r"pista_(\d+)", f.name.lower())
39         if not m:
40             continue
41
42         angle = float(m.group(1))
43         if angle not in ANGLE_TO_CLASS:
44             continue
45
46         df = pd.read_csv(f)
47         df["angle_deg"] = angle
48         df["class_label"] = ANGLE_TO_CLASS[angle]
49         df["source_name"] = f.name
50         rows.append(df)
51
52     if not rows:
53         raise RuntimeError("No se encontraron ángulos válidos para el
54         preentrenamiento.")
55
56     data = pd.concat(rows, ignore_index=True)
57
58     # eliminar no numéricas salvo target
59     drop_cols = {"track", "file"}

```

```
57 data = data.drop(columns=[c for c in drop_cols if c in data.columns],
errors="ignore")
58
59 out_csv = OUT_DIR / "module_pretrain_3classes.csv"
60 data.to_csv(out_csv, index=False)
61
62 summary = {
63     "angles_used": sorted(ANGLE_TO_CLASS.keys()),
64     "class_counts": data["class_label"].value_counts().sort_index().to_dict(),
65     "rows": int(len(data))
66 }
67
68 with open(OUT_DIR / "module_pretrain_summary.json", "w", encoding="utf-8") as f:
69     json.dump(summary, f, indent=2)
70
71 print(f"✅ Guardado: {out_csv}")
72 print("Conteo por clase:", summary["class_counts"])
73
74 if __name__ == "__main__":
75     main()
```

**J | Anexo J: Especificaciones técnicas
de la interfaz de captura de audio
PRESONUS STUDIO 24C**

5 Información técnica

5.1 Especificaciones

General

Frecuencias de muestreo	44.1, 48, 88.2, 96, 176.4 ó 192 kHz
Resolución del convertidor	24-bit
Rango dinámico del convertidor	108 dB
Rango dinámico del convertidor	108 dB

Entradas de micrófono

Nivel máximo	+10 dBu (balanceadas, min gain)
Rango de ganancia	50 dB
Respuesta en frecuencia	20Hz - 20kHz (+/- 0.1 dB, ganancia mínima, 48kHz)
Rango dinámico	106 dB (A-ponderado, min gain)
THD + N	0.005% (1kHz, -1 dBFS, ganancia unidad)
EIN -128 dBu (ganancia máxima, 150 Ω , A-ponderado)	
Impedancia de entrada	1.4K Ω
Alimentación Phantom +48 VDC (10mA total)	

Entradas de línea/Instrumento

Nivel máximo	+10 dBu (balanceadas, min gain)
Rango de ganancia	50 dB
Respuesta en frecuencia	20Hz - 20kHz (+/- 0.1 dB, ganancia mínima, 48kHz)
Rango dinámico	106 dB (A-ponderado, min gain)
THD + N	0.005% (1kHz, -1 dBFS, min gain)
Impedancia de entrada	750K Ω

Salidas principales

Nivel máximo	+10 dBu (balanceadas)
Respuesta en frecuencia	20Hz - 20kHz (+/- 0.1 dB, ganancia unidad, 48kHz)
Rango dinámico 106 dB (A-ponderado)	
THD+N	0.003% (1kHz, -1 dBFS)

Salidas de auriculares

Potencia máxima	40 mW/channel (56 Ω load)
Respuesta en frecuencia	20Hz - 20kHz (+/- 0.2dB, 56 Ω , 48kHz)
Rango dinámico	83 dB (A-ponderado, 56 Ω load)
THD + N	0.005% (1kHz, -1 dBFS, no load)
Rango de funcionamiento de la impedancia	32 Ω a 300 Ω

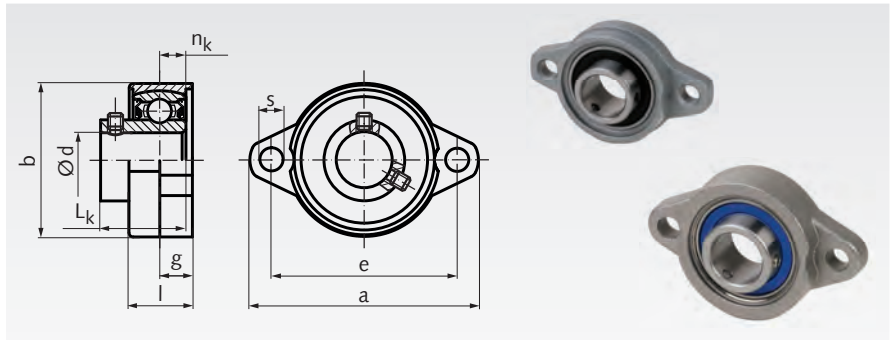
**K | Anexo K: Especificaciones técnicas
del rodamiento de bolas con chu-
macera KFL08**

Ball Flange Bearings KFL and SSKFL, light series

Material KFL: Housing: from Zinc die cast,
Rolling bearing: from bearing steel.

Material SSKFL: Housing: Stainless steel
1.4301 (AISI 304). Rolling bearing:
Stainless steel 1.4125 (AISI 440C), lubricated with
grease FM 222 for food processing machinery, with
registration FDA, CIFA, KPF2K-20, NSF H1.

The rolling bearing can be swiveled when mounting
to compensate shaft misalignment. The shaft will
get fastened with 2 set screws. Lubricated for life at
normal operating conditions.



Ordering Details: e.g.: Product No. 626 608 00, Ball Flange Bearing KFL 08, Bore 8mm

Product No. KFL	Product No. SSKFL	No.	d mm	a mm	b mm	e mm	g mm	l mm	s mm	L _k mm	n _k mm	Bearing-Load Rating ¹⁾				Weight KFL g	Weight SSKFL g
												KFL		SSKFL			
												dyn. C kN	stat.C ₀ kN	dyn. C kN	stat.C ₀ kN		
626 608 00	-	08	8	48	27	37	4,5	8,5	4,8	11,5	3,5	4,2	1,6	-	-	50	-
626 610 00	626 996 10	000	10	60	36	45	5,5	11,5	7	15 ²⁾	4	4,7	2,0	4,0	1,6	60	70
626 612 00	626 996 12	001	12	63	38	48	5,5	11,5	7	15 ²⁾	4	5,2	2,45	4,4	1,95	70	80
626 615 00	626 996 15	002	15	67	42	53	6,5	13	7	16,5	4,5	5,7	2,9	4,85	2,3	80	100
626 617 00	626 996 17	003	17	71	46	56	7	14	7	17,5	5	6,1	3,35	5,2	2,7	100	140
626 620 00	626 996 20	004	20	90	55	71	8	16	10	21	6	9,55	5,15	8,1	4,1	170	210
626 625 00	626 996 25	005	25	95	60	75	8	16	10	22,5	6	10,3	5,95	8,75	4,75	210	270

¹⁾ Maximum radial load if axial force = 0. The axial load rating is approx. 20% of the radial load rating.

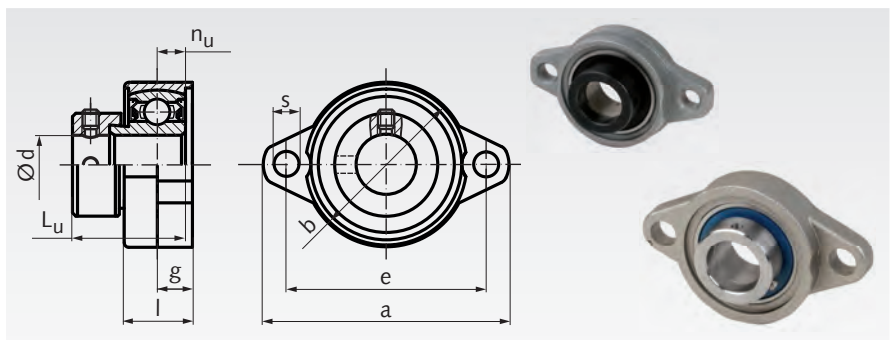
²⁾ For size SSKFL 000 the dimension is L_k 14mm and for size SSKFL 001 the dimension is L_k 14,5mm.

Ball Flange Bearings UFL and SSUFL, light series, with Eccentric Ring

Material UFL: Housing: from Zinc die cast,
Rolling bearing: from bearing steel.

Material SSUFL: Housing: Stainless steel
1.4301 (AISI 304). Rolling bearing:
Stainless steel 1.4125 (AISI 440C), lubricated with
grease FM 222 for food processing machinery, with
registration FDA, CIFA, KPF2K-20, NSF H1.

The rolling bearing can be swiveled when mounting
to compensate shaft misalignment. During
assembly, the eccentric ring and the inner bearing
ring are turned against each other to clamp the
shaft. The eccentric ring is additionally secured on
the shaft with 1 set screw. Lubricated for life at
normal operating conditions.



Ordering Details: e.g.: Product No. 626 710 00, Ball Flange Bearing UFL 000, Bore 10mm

Product No. UFL	Product No. SSUFL	No.	d mm	a mm	b mm	e mm	g mm	l mm	s mm	L _u mm	n _u mm	Bearing-Load Rating ¹⁾				Weight UFL g	Weight SSUFL g
												UFL		SSUFL			
												dyn. C kN	stat.C ₀ kN	dyn. C kN	stat.C ₀ kN		
626 710 00	626 997 10	000	10	60	36	45	5,5	11,5	7	17,5	4	4,6	1,98	4,0	1,6	60	80
626 712 00	626 997 12	001	12	63	38	48	5,5	11,5	7	17,5	4	5,1	2,27	4,4	1,95	70	90
626 715 00	626 997 15	002	15	67	42	53	6,5	13	7	18,5	4,5	5,6	2,55	4,85	2,3	90	120
626 717 00	626 997 17	003	17	71	46	56	7	14	7	21,0 ²⁾	5	6,0	2,84	5,2	2,7	110	140
626 720 00	626 997 20	004	20	90	55	71	8	16	10	25,5 ²⁾	6	9,35	4,55	8,1	4,1	210	250
626 725 00	626 997 25	005	25	95	60	75	8	16	10	25,5	6	10,1	5,05	8,75	4,75	220	280

¹⁾ Maximum radial load if axial force = 0. The axial load rating is approx. 20% of the radial load rating.

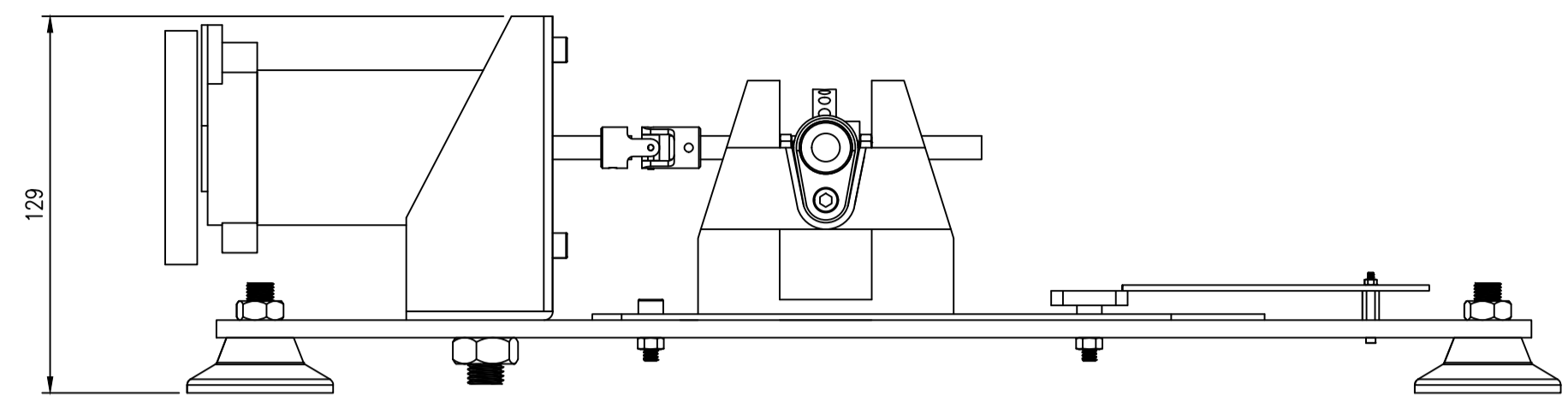
²⁾ For size SSUFL 003 the dimension is L_u 20,5mm and for size SSUFL 004 the dimension is L_u 24,5mm.

Shaft Connection with Eccentric Ring

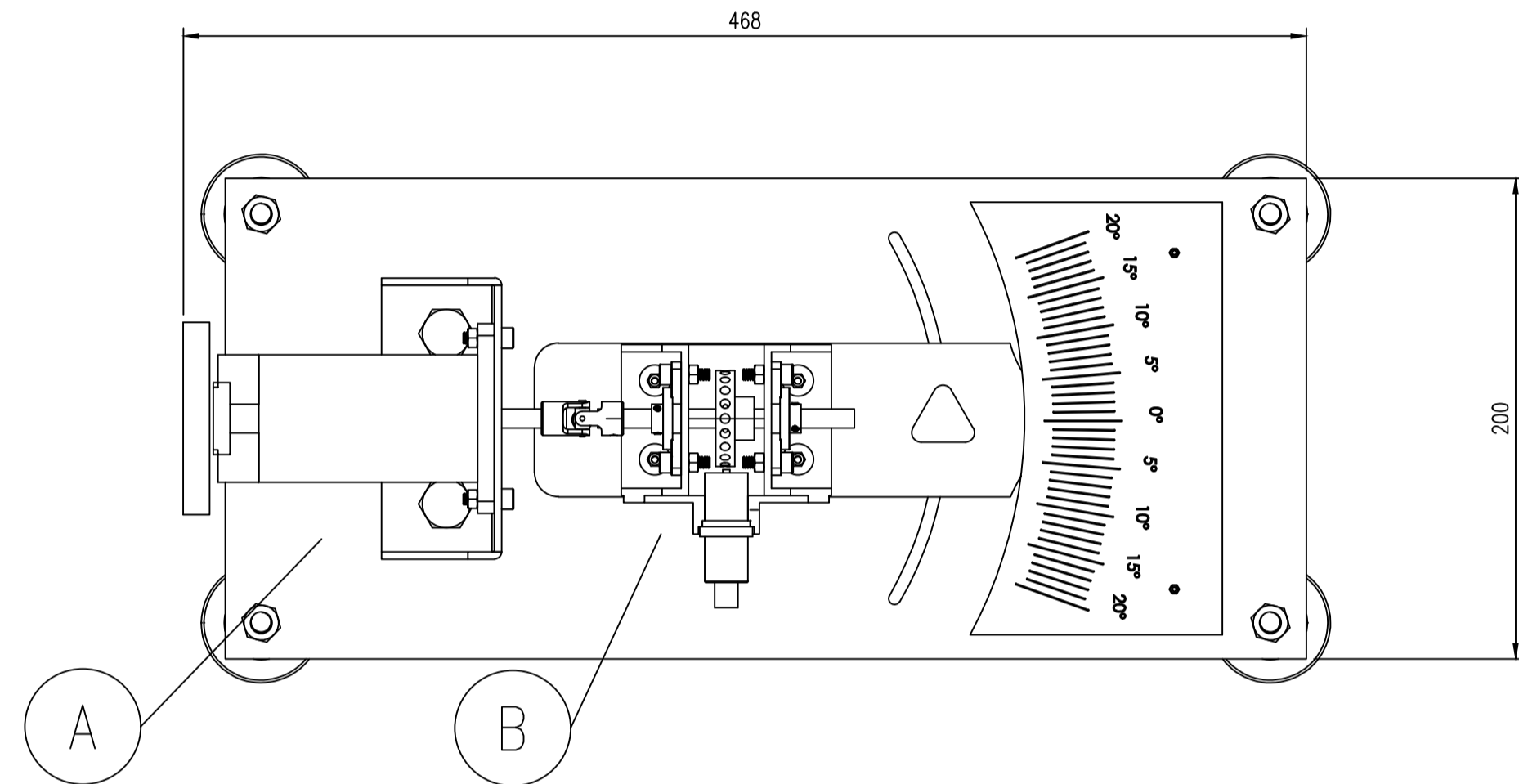
The eccentric ring has an eccentric recess, a radial bore for hook wrench with pin
and a set screw. The inner ring of the bearing insert has an eccentric shoulder. For
assembly, the eccentric ring and the inner bearing ring must be turned against
each other to clamp the shaft. Finally the eccentric ring must be secured with the
set screw.



Anexo L: Planos

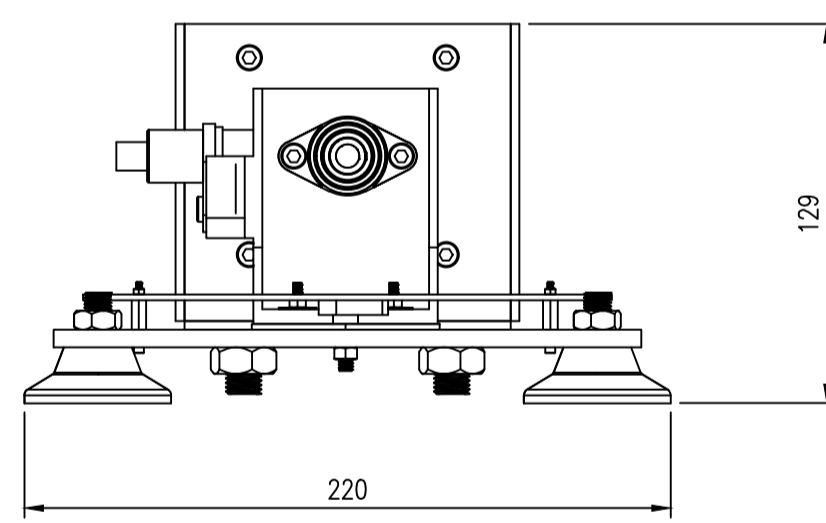


MÓDULO DE PRUEBAS.
VISTA LATERAL
ESCALA 1:2



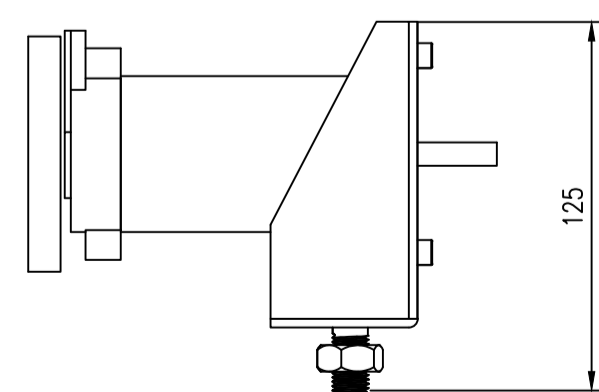
MÓDULO DE PRUEBAS.
VISTA DE PLANTA
ESCALA 1:2

MÓDULO DE PRUEBAS.
VISTA FRONTAL
ESCALA 1:2



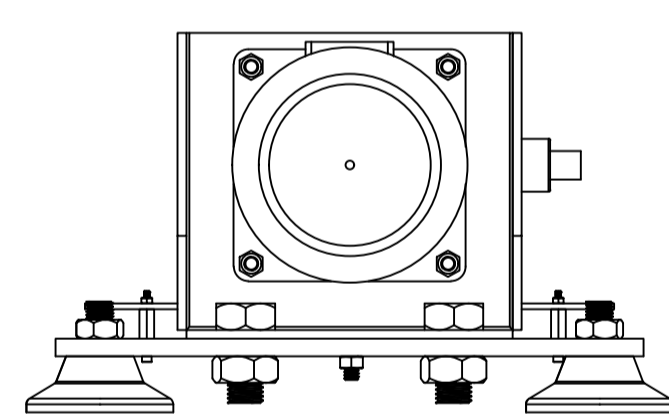
MÓDULO DE PRUEBAS.
VISTA FRONTAL
ESCALA 1:2

DETALLE A



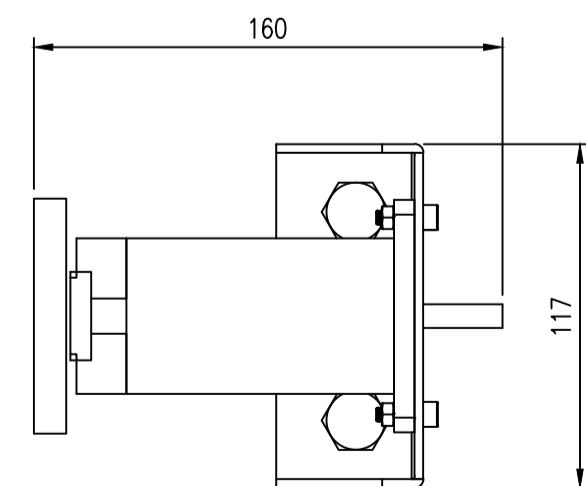
DETALLE DE MONTAJE DE MOTOR.
VISTA LATERAL
ESCALA 1:2

MÓDULO DE PRUEBAS.
VISTA FRONTAL
ESCALA 1:2

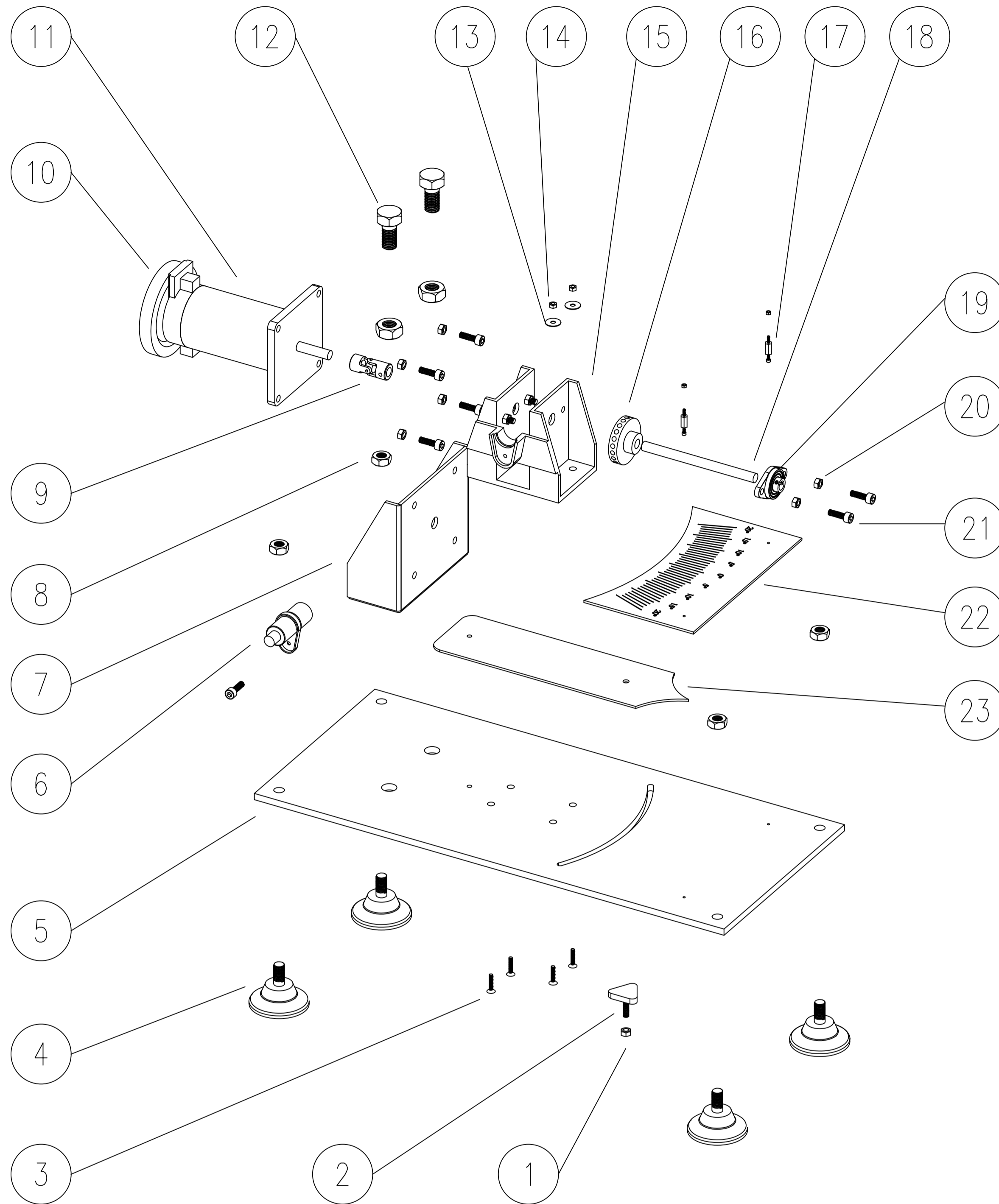


MÓDULO DE PRUEBAS.
VISTA TRASERA
ESCALA 1:2

DETALLE A

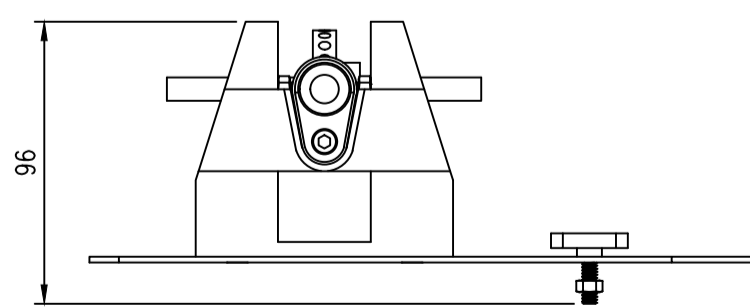


DETALLE DE MONTAJE DE MOTOR.
VISTA DE PLANTA
ESCALA 1:2



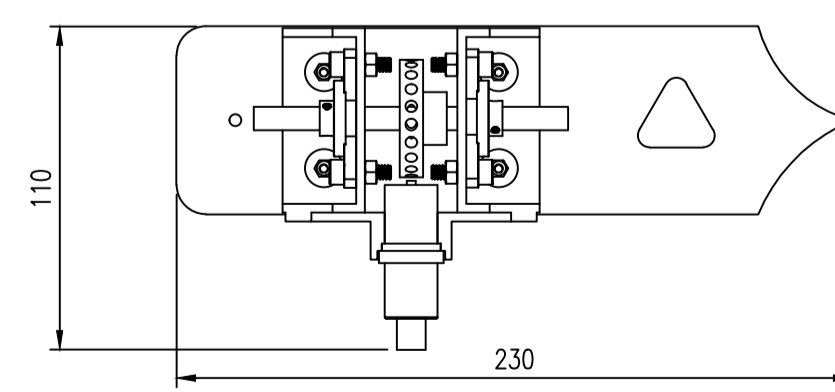
MÓDULO DE PRUEBAS.
VISTA DE EXPLOSIÓN
ESCALA 1:2

DETALLE B

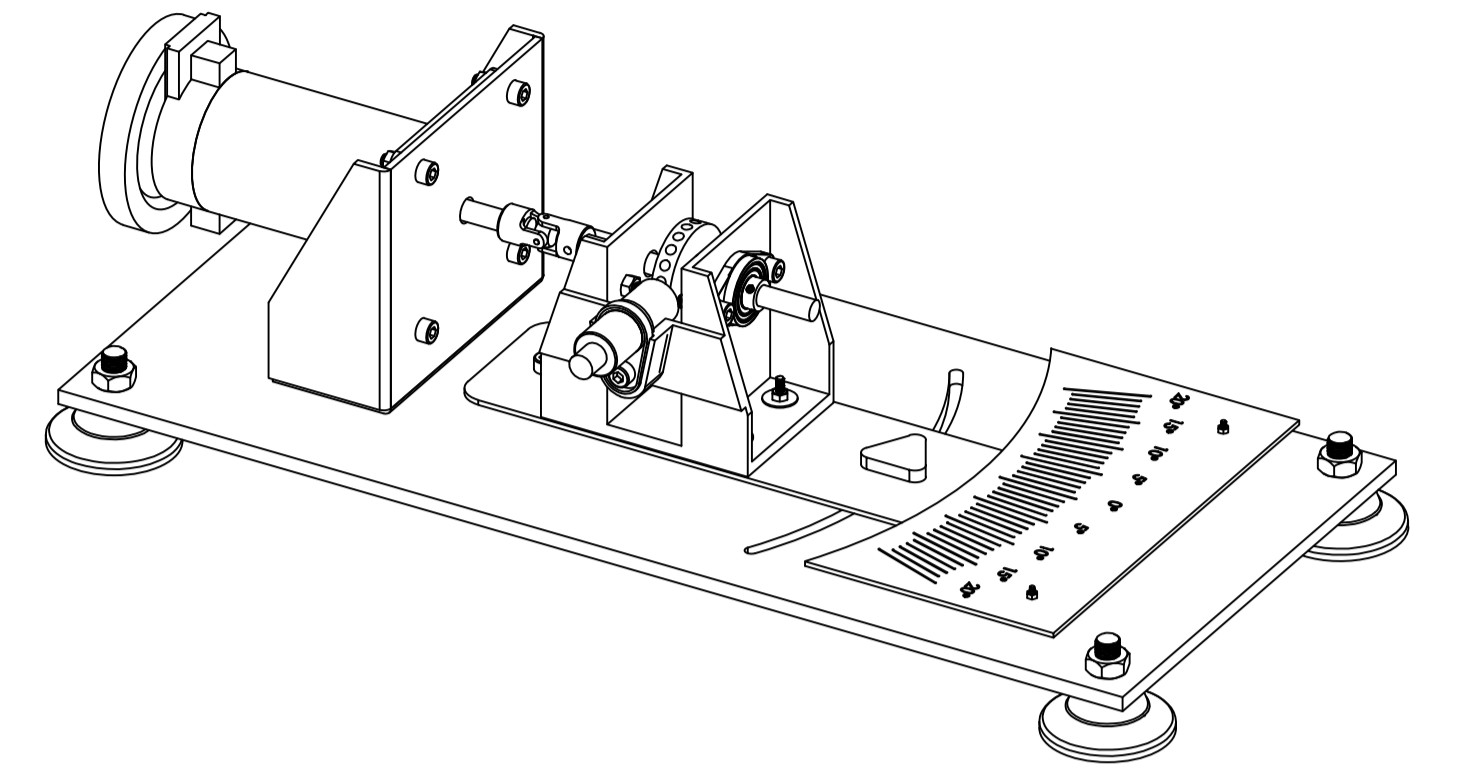


DETALLE DE GRUPO DE MONTAJE DE SENSOR.
VISTA LATERAL
ESCALA 1:2

DETALLE B



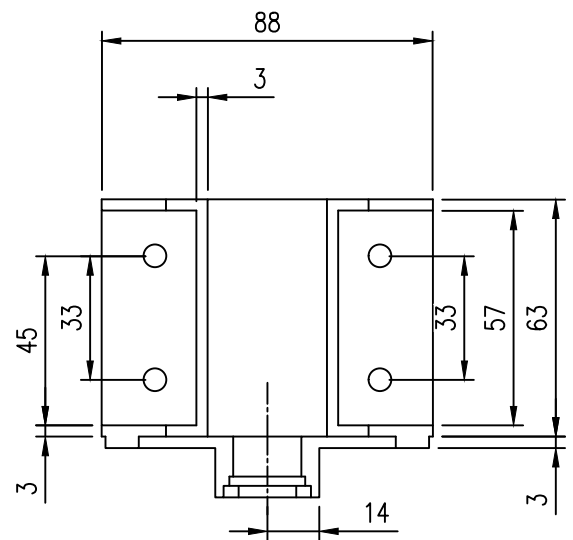
DETALLE DE GRUPO DE MONTAJE DE SENSOR.
VISTA DE PLANTA
ESCALA 1:2



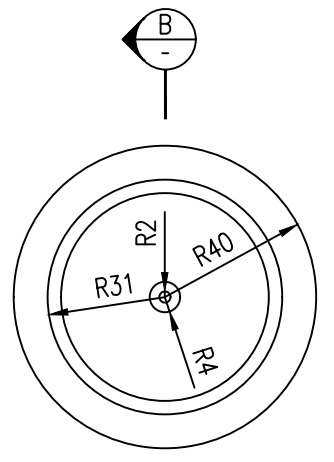
MÓDULO DE PRUEBAS PARA IDENTIFICACIÓN DEL DESALINEAMIENTO DE EJES.
VISTA ISOMÉTRICA
ESCALA 1:2

23	PL. N°03, e=2 mm	1	ASTM	A-36	180.0	230.0
22	PL. N°02, e=2 mm	1	ASTM	D-3935	180.0	30.0
21	PERNO, M4 x 12 mm	9	ASTM	A-36	15.0	3.5
20	TUERCA, M4	9	ASTM	A-36	4.0	1.2
19	RODAMIENTO KLF, $\phi=8$ mm	2	ASTM	B-209	90.0	42.3
18	EJE l=90 mm, $\phi=8$ mm	1	ASTM	A-276	90.0	42.3
17	SEPARADOR HM23	2	ASTM	B-124	10.0	3.0
16	RUEDA FÓNICA	1	ISO-1043	PLA	40.0	11.0
15	SOPORTE DE SENSOR	1	ISO-1043	PLA	77.0	79.1
14	TUERCA, M3	2	ASTM	A-36	3.0	0.7
13	ARANDELA, M3	2	ASTM	A-36	8.0	0.8
12	PERNO l= $\frac{25}{32}$ " , $\phi=\frac{3}{8}$ "	2	ASTM	A-36	30.0	24.0
11	MOTOR ELÉCTRICO YA-010	1	-	-	111.0	1579.0
10	VOLANTE DE INERCIA	1	ASTM	A-36	80.0	286.6
9	CRUCETA TIPO CARDÁN	1	ASTM	A-36	35.0	24.0
8	TUERCA $\phi=\frac{3}{8}$ "	6	ASTM	A-36	8.0	8.0
7	SOPORTE MOTOR e= $\frac{3}{32}$ "	1	ASTM	A-36	1000.0	560.1
6	SENSOR CKP INDUCTIVO	1	ASTM	D-1784	63.0	38.0
5	PL. N°01, e= $\frac{1}{4}$ "	1	ASTM	A-36	920.0	4181.0
4	SOPORTE GOMA CON ESPARRAGO $\phi=\frac{3}{8}$ "	4	ASTM	A-36	40.0	40.0
3	PERNO AVELLANADO M3	4	ASTM	A-36	10.00	3.0
2	PERILLA CON ESPARRAGO	1	ASTM	A-36	25.00	20.0
1	TUERCA HEXAGONAL M4	1	ASTM	A-36	2.00	1.0

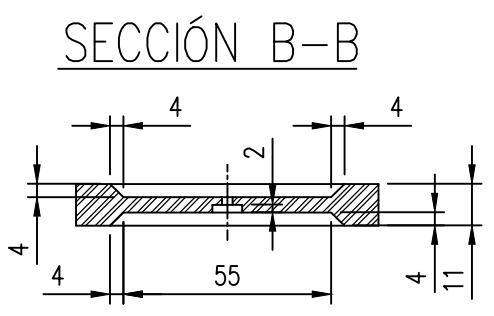
ACABADO		REVISIONES Y ROMPER ARISTAS VIVAS		REVISION	
SI NO SE INDICA LO CONTINÚO LAS COTAS SE EXPRESAN EN MM		UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAID DEL CUSCO		TÍTULO	
ACABADO SUPERFICIAL: TELEFORMADO		NOMBRE: [] FIRMA: [] FECHA: 27/03/26		"MÓDULO DE PRUEBAS PARA IDENTIFICACIÓN DEL DESALINEAMIENTO DE EJES"	
ACABADO ANGULAR:		VERIF: []		N° DE DIBUJO: MC-DWG-001	
ACABADO:		APROB: []		AL: []	
ACABADO:		FABR: []		PESO: []	
ACABADO:		CALIB: []		ESCALA: 1:2	
ACABADO:		MATERIAL: []		FOLIA 1 DE 1	



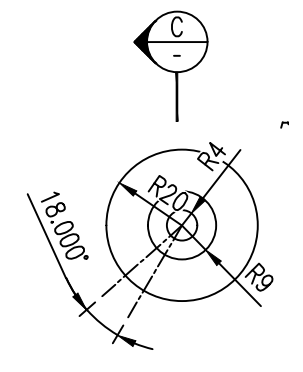
SOPORTE DE SENSOR Y EJE
VISTA DE PLANTA
ESCALA 1:2



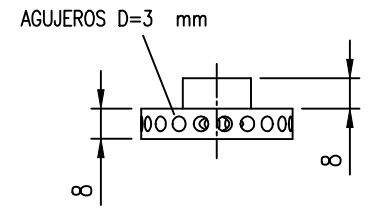
VOLANTE DE INERCIA
VISTA DE PLANTA
ESCALA 1:2



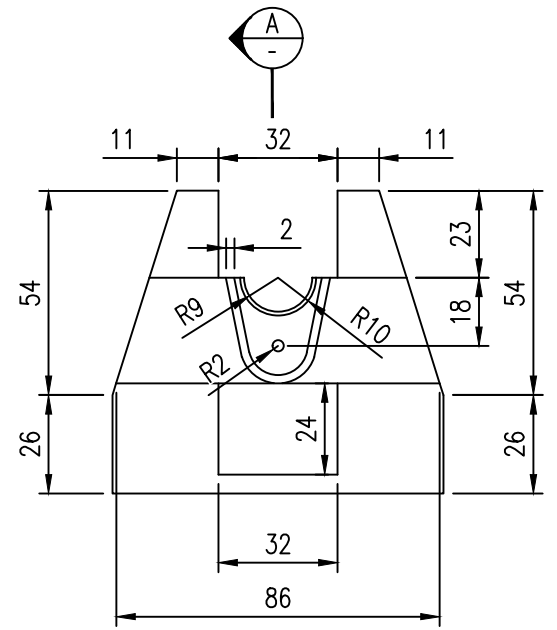
SECCIÓN B-B
VOLANTE DE INERCIA



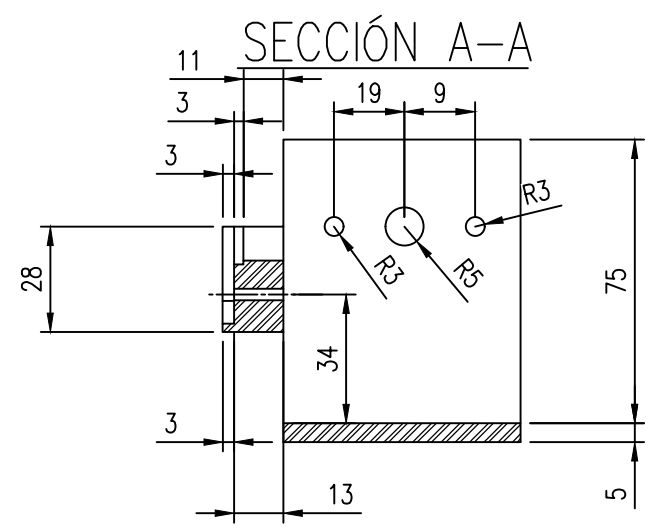
RUEDA FÓNICA
VISTA DE PLANTA
ESCALA 1:2



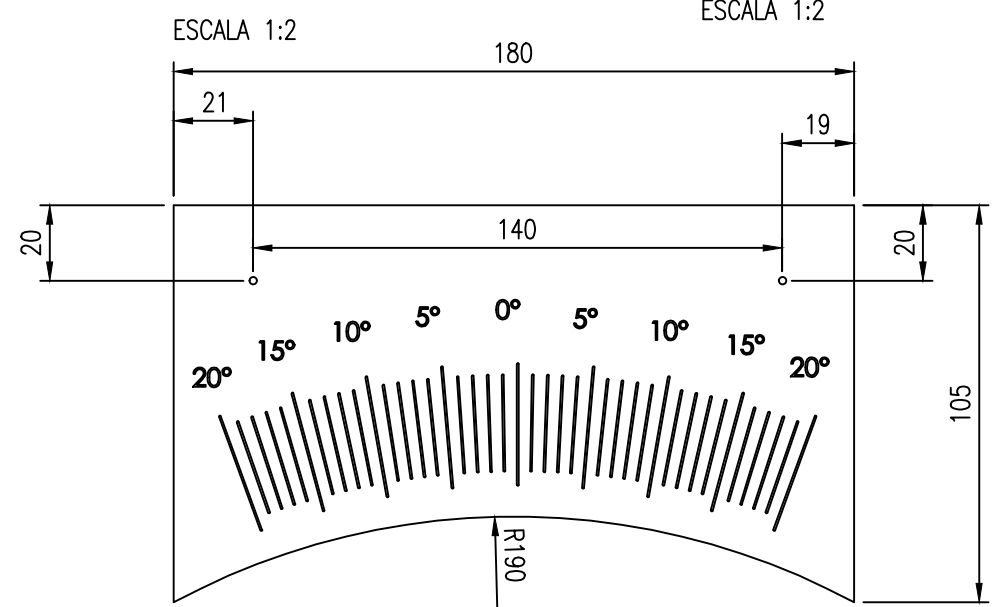
RUEDA FÓNICA
VISTA LATERAL
ESCALA 1:2



SOPORTE DE SENSOR Y EJE
VISTA FRONTAL
ESCALA 1:2

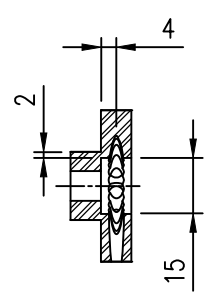


SECCIÓN A-A
SOPORTE DE SENSOR Y EJE
VISTA FRONTAL
ESCALA 1:2

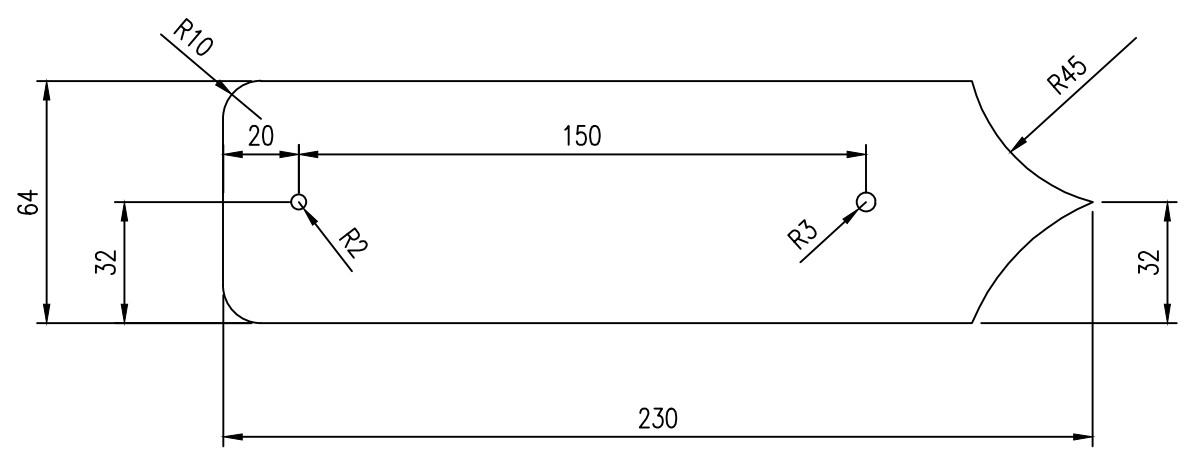


PL N°2 e=2 mm
VISTA DE PLANTA
ESCALA 1:2

SECCIÓN C-C

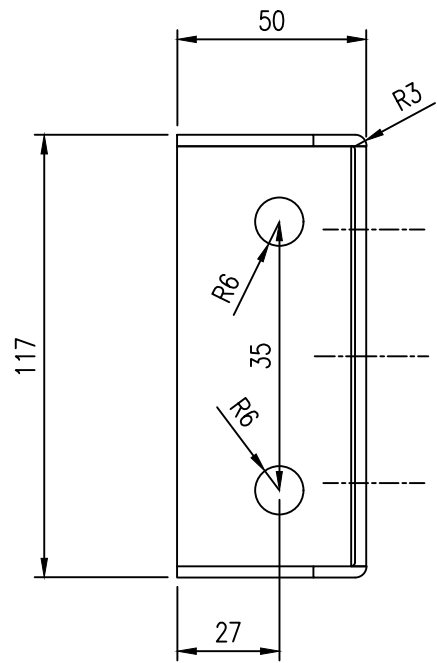


RUEDA FÓNICA
ESCALA 1:2

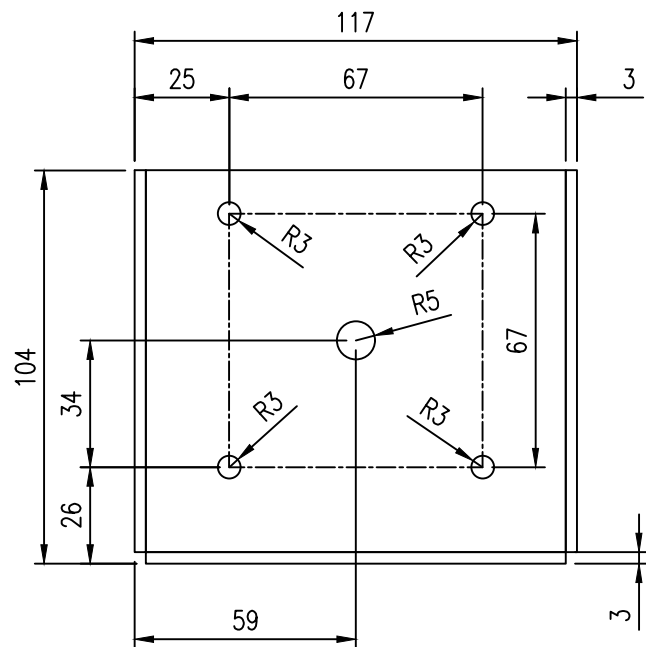


PL N°3 e=2 mm
VISTA DE PLANTA
ESCALA 1:2

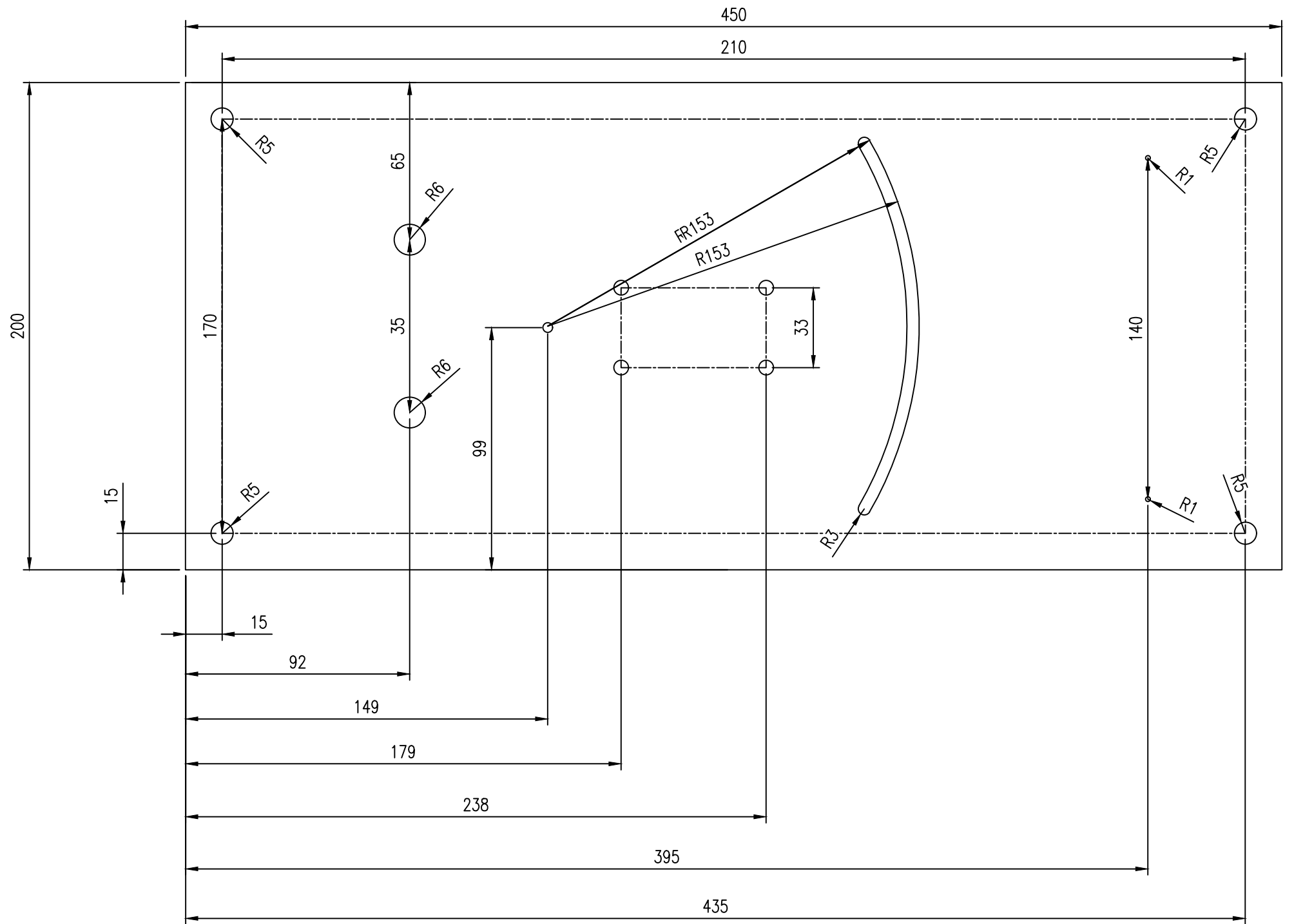
SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			ACABADO:	REBARBAR Y ROMPER ARISTAS VIVAS	REVISIÓN
DIBUJ.	W.D.E.A /H.A.C.S.	FIRMA	FECHA	UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO	
VERIF.	W.D.E.A /H.A.C.S.		27/03/26	TÍTULO: "MÓDULO DE PRUEBAS PARA IDENTIFICACION DEL DESALINEAMIENTO DE EJES"	
APROB.	W.D.E.A /H.A.C.S.		27/03/26	N° DE DIBUJO MC-DWG-002	
FABR.	W.D.E.A /H.A.C.S.		27/03/26	A1	
CALID.				MATERIAL: -	ESCALA: 1:2
				PESO:	HOJA 1 DE 1



SOPORTE DE MOTOR
VISTA DE PLANTA
ESCALA 1:2



SOPORTE DE MOTOR
VISTA TRASERA
ESCALA 1:2



PL N°1 e=1/4"
VISTA DE PLANTA
ESCALA 1:2

SI NO SE INDICA LO CONTRARIO: LAS COTAS SE EXPRESAN EN MM ACABADO SUPERFICIAL: TOLERANCIAS: LINEAL: ANGULAR:			ACABADO:	REBARBAR Y ROMPER ARISTAS VIVAS	REVISIÓN
UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO					TÍTULO:
"MÓDULO DE PRUEBAS PARA IDENTIFICACIÓN DEL DESALINEAMIENTO DE EJES"					N° DE DIBUJO
A-36					MC-DWG-003
PESO:					ESCALA: 1:2
HOJA 1 DE 1					A1