



Universidad de Valladolid

**ESCUELA DE INGENIERÍA INFORMÁTICA
DE SEGOVIA**

**Grado en Ingeniería Informática
de Servicios y Aplicaciones**

**Diagnóstico de fallos incipientes de rodamientos en
motores de inducción mediante técnicas de
clasificación supervisada**

Alumna: Gema de Pablo Ordiales

Tutores: Amelia García Garrosa y Miguel Fernández Temprano

Diagnóstico de fallos incipientes de rodamientos en motores de inducción mediante técnicas de clasificación supervisada

Gema de Pablo Ordiales

Julio, 2023

Índice general

Lista de figuras	V
Lista de tablas	IX
Resumen	XVII
I Descripción del proyecto	1
1. Introducción	3
1.1. Motivación y descripción del problema	3
1.2. Objetivos	5
1.3. Estructura	6
2. Motores de inducción	7
2.1. Crecimiento en la industria	7
2.2. Historia de los motores de inducción	8
2.3. Principios de funcionamiento	10
2.4. Tipos de fallo y detección	12
3. Planificación	15
3.1. Metodología de trabajo	15
3.1.1. Valores	15
3.1.2. Roles	16
3.1.3. Eventos	17
3.1.4. Herramientas	17
3.1.5. Estimación del esfuerzo	18
3.2. Planificación temporal	19
3.2.1. Sprint 1	19
3.2.2. Sprint 2	21
3.2.3. Sprint 3	22
3.2.4. Sprint 4	23
3.3. Balance temporal real del proyecto	24
3.4. Presupuesto	25

3.4.1. Hardware y software	26
3.4.2. Recursos humanos	26
II Metodología, análisis y resultados	29
4. Metodología	31
4.1. Introducción al problema de clasificación	31
4.1.1. Análisis discriminante lineal	31
4.1.2. Árboles de decisión	36
4.2. Construcción y evaluación de modelos	37
4.2.1. Medida del error	38
4.2.2. División del conjunto de datos	39
4.2.3. Búsqueda de hiperparámetros	40
4.3. Análisis de componentes principales	40
4.4. Análisis de la varianza	43
5. Datos	45
5.1. Descripción del conjunto de datos	45
5.2. Filtrado de los datos	47
5.3. Análisis exploratorio de los datos	49
6. Análisis y resultados	53
6.1. Experimento	54
6.2. Análisis 1. Grupo 1 contra grupo 2	54
6.2.1. Análisis discriminante lineal	54
6.2.2. Árboles de decisión	62
6.2.3. Análisis de la varianza de los resultados del grupo 1 contra el grupo 2	64
6.3. Análisis 2. Grupo 1 contra grupo 2+3+4+5+6	66
6.3.1. Análisis discriminante lineal	66
6.3.2. Árboles de decisión	68
6.3.3. Análisis de la varianza de los resultados del grupo 1 contra el grupo	
2+3+4+5+6	69
7. Conclusiones y trabajo futuro	73
III Apéndices	75
A. Gráficos y tablas complementarios	77
A.1. Análisis exploratorio. Diagramas de caja	77
A.2. Resultados de los ACP	83
A.2.1. Análisis 1. Grupo 1 contra grupo 2. Resultados ACP.	83
A.2.2. Análisis 2. Grupo 1 contra grupo 2+3+4+5+6. Resultados ACP. . .	93

A.3. Resultados gráficos adicionales del Análisis 1	102
A.3.1. Resultados en Carga Alta	103
A.3.2. Resultados en Carga Baja	108
A.4. Resultados gráficos y numéricos del Análisis 2	118
A.4.1. Resultados en Carga Alta	118
A.4.2. Resultados en Carga Baja	127
B. Código	137
B.1. Diagramas de caja	137
B.2. Distancia de Mahalanobis por Grupos de Respuesta	140
B.3. Análisis Algoritmo LDA	141
B.4. Árboles de decisión	151
B.5. Métricas	155
Bibliografía	161

Índice de figuras

2.1. Batería eléctrica de Volta.	8
2.2. Motorhomopolar de Faraday.	9
2.3. Motor eléctrico funcional de Jacobi.	9
2.4. Motor de corriente alterna de Tesla.	10
2.5. Partes de un motor de jaula de ardilla.	12
3.1. Organización de tareas en Trello.	18
3.2. Cartas de planning poker.	19
3.3. Subobjetivos alcanzados durante el Sprint 1.	20
3.4. Diagrama de Gantt del Sprint 1.	20
3.5. Subobjetivos alcanzados durante el Sprint 2.	21
3.6. Diagrama de Gantt del Sprint 2.	22
3.7. Subobjetivos alcanzados durante el Sprint 3.	22
3.8. Diagrama de Gantt del Sprint 3.	23
3.9. Subobjetivos alcanzados durante el Sprint 4.	23
3.10. Diagrama de Gantt del Sprint 4.	24
3.11. Diagrama de Gantt Completo - Planificación inicial	24
3.12. Diagrama de Gantt Completo - Balance temporal real	25
3.13. Gastos materiales	26
3.14. Gastos personal	27
4.1. Clasificación. Con una sola característica se produce superposición. Fuente: [31].	33
4.2. Diagrama de dispersión bidimensional. Fuente: [31].	33
4.3. Proyecciones de los puntos con clasificación perfecta. Fuente: [31].	34
4.4. Árbol de decisión y valor del error de clasificación total. Fuente: [28]	36
4.5. Ejemplo de construcción de árbol. Fuente: [28]	37
4.6. Matriz de Confusión. Fuente: [4]	38
4.7. Validación cruzada de K iteraciones con K=4. Fuente: [36]	40
4.8. Determinación del plano formado por las dos primeras componentes principales. Fuente: [14]	42
4.9. Gráfico de Proporción de Varianza Explicada Acumulada (PVEA) mediante ACP de Corriente en Carga Alta. Análisis 1 contra 2	42

5.1. Matriz de dispersión de las variables preliminares evaluadas por pares de corriente en carga baja del análisis del grupo 1 contra el grupo 2+3+4+5+6.	51
5.2. Mapa de calor de las variables preliminares de corriente en carga baja del análisis 1 contra 2+3+4+5+6 evaluadas por pares.	52
6.1. Resumen del procedimiento. Adaptado de [3]	54
6.2. Métricas Análisis 1-CA_LDA_Corr	57
6.3. Métricas Análisis 1-CA_LDA_Sonido	58
6.4. Métricas Análisis 1-CA_LDA_Vibración eje x	59
6.5. Métricas Análisis 1-CA_LDA_Vibración eje y	60
6.6. Métricas Análisis 1-CA_LDA_Vibración eje z	61
6.7. Gráfico de interacciones del modelo ANOVA final del análisis 2	71
A.1. Gráficas de Caja y Bigotes de Corriente a Carga Alta en fase 1	77
A.2. Gráficas de Caja y Bigotes de Corriente a Carga Alta en fase 3	78
A.3. Gráficas de Caja y Bigotes de Corriente a Carga Alta en fase 3	78
A.4. Gráficas de Caja y Bigotes de Corriente a Carga Alta en fase 3	78
A.5. Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje x	79
A.6. Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje x	79
A.7. Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje x	79
A.8. Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje y	80
A.9. Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje y	80
A.10. Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje z	80
A.11. Gráficas de Caja y Bigotes de Sonido a Carga Alta	81
A.12. Gráficas de Caja y Bigotes de Corriente a Carga Baja	81
A.13. Gráficas de Caja y Bigotes de Corriente a Carga Baja en fase 3	81
A.14. Gráficas de Caja y Bigotes de Vibraciones a Carga Baja en eje x	82
A.15. Gráficas de Caja y Bigotes de Vibraciones a Carga Baja en eje y	82
A.16. Métricas Análisis 1-CA_AD_Corriente	103
A.17. Métricas Análisis 1-CA_AD_Sonido	104
A.18. Métricas Análisis 1-CA_AD_Vibración eje x	105
A.19. Métricas Análisis 6-CA_AD_Vibración eje y	106
A.20. Métricas Análisis 1-CA_AD_Vibración eje z	107
A.21. Gráfico 3D. Análisis 1-CB_LDA_Corriente	108
A.22. Gráfico 3D. Análisis 1-CB_LDA_Sonido	109
A.23. Gráfico 3D. Análisis 1-CB_LDA_Vibraciones eje x	110
A.24. Gráfico 3D. Análisis 1-CB_LDA_Vibraciones eje y	111
A.25. Gráfico 3D. Análisis 1-CB_LDA_Vibraciones eje z	112
A.26. Gráfico 3D. Análisis 1-CB_AD_Corriente	113
A.27. Gráfico 3D. Análisis 1-CB_AD_Sonido	114
A.28. Gráfico 3D. Análisis 1-CB_AD_Vibraciones eje x	115
A.29. Gráfico 3D. Análisis 1-CB_AD_Vibraciones eje y	116
A.30. Gráfico 3D. Análisis 1-CB_AD_Vibraciones eje z	117

A.31.Gráfico 3D. Análisis 2-CA_LDA_Corriente	118
A.32.Gráfico 3D. Análisis 2-CA_LDA_Sonido	119
A.33.Gráfico 3D. Análisis 2-CA_LDA_Vibración eje x	120
A.34.Gráfico 3D. Análisis 2-CA_LDA_Vibración eje z	121
A.35.Gráfico 3D. Análisis 2-CA_AD_Corriente	122
A.36.Gráfico 3D. Análisis 2-CA_AD_Sonido	123
A.37.Gráfico 3D. Análisis 2-CA_AD_Vibraciones eje x	124
A.38.Gráfico 3D. Análisis 2-CA_AD_Vibraciones eje y	125
A.39.Gráfico 3D. Análisis 2-CA_AD_Vibraciones eje z	126
A.40.Gráfico 3D. Análisis 2-CB_LDA_Corriente	127
A.41.Gráfico 3D. Análisis 2-CB_LDA_Sonido	128
A.42.Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje x	129
A.43.Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje y	130
A.44.Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje z	131
A.45.Gráfico 3D. Análisis 2-CB_AD_Corriente	132
A.46.Gráfico 3D. Análisis 2-CB_LDA_Sonido	133
A.47.Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje x	134
A.48.Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje y	135
A.49.Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje z	136

Índice de cuadros

5.1. Variables predictoras	46
5.2. Posibles puntos anómalos detectados mediante análisis bivariantes	48
5.3. Puntos anómalos finales detectados mediante Distancia de Mahalanobis	49
5.4. Parejas de variables con correlaciones superiores a 0,9 para las variables de corriente en carga baja del análisis 1 contra 2+3+4+5+6.	50
6.1. Error obtenido en el análisis 1 para cada fold y promedio para el Análisis Discriminante Lineal	62
6.2. Error obtenido en el análisis 1 para cada fold y promedio para los árboles de decisión	63
6.3. Tabla variables para los Árboles de decisión en el análisis 1	64
6.4. Tabla ANOVA con interacciones correspondiente al análisis 1	65
6.5. Tabla ANOVA sin interacciones correspondiente al análisis 1	65
6.6. Tabla ANOVA final correspondiente al análisis 1	66
6.7. Análisis post-hoc mediante el método de Tukey de los resultados correspondientes al análisis 1	66
6.8. Error obtenido en el análisis 2 para cada fold y promedio para los análisis discriminantes lineales	67
6.9. Error obtenido en el análisis 2 para cada fold y promedio para los árboles de decisión	68
6.10. Tabla variables para los árboles de decisión en el análisis 2	69
6.11. Tabla ANOVA con todas las interacciones correspondiente al análisis 2	70
6.12. Tabla ANOVA con solo las interacciones significativas correspondiente al análisis 2	70
6.13. Tabla ANOVA final correspondiente al análisis 2	71
A.1. Tabla de Autovalores Corriente en Carga Alta	83
A.2. Tabla de Autovectores Corriente en Carga Alta	83
A.3. Tabla de Autovalores Sonido en Carga Alta	84
A.4. Tabla de Autovectores Sonido en Carga Alta	84
A.5. Tabla de Autovalores Vibraciones eje x en Carga Alta	85
A.6. Tabla de Autovectores Vibraciones eje x en Carga Alta	85
A.7. Tabla de Autovalores Vibraciones eje y en Carga Alta	86

A.8. Tabla de Autovectores Vibraciones eje y en Carga Alta	86
A.9. Tabla de Autovalores Vibraciones eje z en Carga Alta	87
A.10. Tabla de Autovectores Vibraciones eje z en Carga Alta	87
A.11. Tabla de Autovalores Corriente eje x en Carga Baja	88
A.12. Tabla de Autovectores Corriente en Carga Baja	88
A.13. Tabla de Autovalores Sonido en Carga Baja	89
A.14. Tabla de Autovectores Sonido en Carga Baja	89
A.15. Tabla de Autovalores Vibraciones eje x en Carga Baja	90
A.16. Tabla de Autovectores Vibraciones eje x en Carga Baja	90
A.17. Tabla de Autovalores Vibraciones eje y en Carga Baja	91
A.18. Tabla de Autovectores Vibraciones eje y en Carga Baja	91
A.19. Tabla de Autovalores Vibraciones eje z en Carga Baja	92
A.20. Tabla de Autovectores Vibraciones eje z en Carga Baja	92
A.21. Tabla de Autovalores Corriente en Carga Alta	93
A.22. Tabla de Autovectores Corriente en Carga Alta	93
A.23. Tabla de Autovalores Sonido en Carga Alta	94
A.24. Tabla de Autovectores Sonido en Carga Alta	94
A.25. Tabla de Autovalores Vibraciones eje x en Carga Alta	95
A.26. Tabla de Autovectores Vibraciones eje x en Carga Alta	95
A.27. Tabla de Autovalores Vibraciones eje y en Carga Alta	96
A.28. Tabla de Autovectores Vibraciones eje y en Carga Alta	96
A.29. Tabla de Autovalores Vibraciones eje z en Carga Alta	97
A.30. Tabla de Autovectores Vibraciones eje z en Carga Alta	97
A.31. Tabla de Autovalores Corriente en Carga Baja	98
A.32. Tabla de Autovectores Corriente en Carga Baja	98
A.33. Tabla de Autovalores Sonido en Carga Baja	99
A.34. Tabla de Autovectores Sonido en Carga Baja	99
A.35. Tabla de Autovalores Vibraciones x en Carga Baja	100
A.36. Tabla de Autovectores Vibraciones x en Carga Baja	100
A.37. Tabla de Autovalores Vibraciones y en Carga Baja	101
A.38. Tabla de Autovectores Vibraciones y en Carga Baja	101
A.39. Tabla de Autovalores Vibraciones z en Carga Baja	102
A.40. Tabla de Autovectores Vibraciones z en Carga Baja	102
A.41. Métricas. Análisis 1-CB_LDA_Corriente	108
A.42. Métricas. Análisis 1-CB_LDA_Sonido	109
A.43. Métricas. Análisis 1-CB_LDA_Vibraciones eje x	110
A.44. Métricas. Análisis 1-CB_LDA_Vibraciones eje y	111
A.45. Métricas. Análisis 1-CB_LDA_Vibraciones eje z	112
A.46. Métricas. Análisis 1-CB_AD_Corriente	113
A.47. Métricas. Análisis 1-CB_AD_Sonido	114
A.48. Métricas. Análisis 1-CB_AD_Vibraciones eje x	115
A.49. Métricas. Análisis 1-CB_AD_Vibraciones eje y	116
A.50. Métricas. Análisis 1-CB_AD_Vibraciones eje z	117

A.51.Métricas. Análisis 2-CA_LDA_Corriente	118
A.52.Métricas. Análisis 2-CA_LDA_Sonido	119
A.53.Métricas. Análisis 2-CA_LDA_Vibración eje x	120
A.54.Métricas. Análisis 2-CA_LDA_Vibración eje z	121
A.55.Métricas. Análisis 2-CA_AD_Corriente	122
A.56.Métricas. Análisis 2-CA_AD_Sonido	123
A.57.Métricas. Análisis 2-CA_AD_Vibraciones eje x	124
A.58.Métricas. Análisis 2-CA_AD_Vibraciones eje y	125
A.59.Métricas. Análisis 2-CA_AD_Vibraciones eje z	126
A.60.Métricas. Análisis 2-CB_LDA_Corriente	127
A.61.Métricas. Análisis 2-CB_LDA_Sonido	128
A.62.Métricas. Análisis 2-CB_LDA_Vibraciones eje x	129
A.63.Métricas. Análisis 2-CB_LDA_Vibraciones eje y	130
A.64.Métricas. Análisis 2-CB_LDA_Vibraciones eje z	131
A.65.Métricas. Análisis 2-CB_AD_Corriente	132
A.66.Métricas. Análisis 2-CB_LDA_Sonido	133
A.67.Métricas. Análisis 2-CB_LDA_Vibraciones eje x	134
A.68.Métricas. Análisis 2-CB_LDA_Vibraciones eje y	135
A.69.Métricas. Análisis 2-CB_LDA_Vibraciones eje z	136

Agradecimientos

Quiero dar las gracias, en primer lugar, a mis tutores, Miguel y Amelia, quienes me han dado el privilegio de ser mis guías en este camino, compartiendo todo su conocimiento y experiencia con absoluta dedicación a través de sus consejos y sus rigurosas correcciones. Sois una fuente de inspiración para mí.

Agradezco al departamento de Ingeniería Eléctrica de la Universidad de Valladolid su indispensable contribución proporcionando los datos con los que se ha podido realizar este trabajo .

También quiero darle las gracias a mi hermana Silvia, a mis tíos y a mis abuelos, por apoyarme y acompañarme durante mi etapa académica.

Hay personas con las que estás y hay personas con las que eres, así que gracias a mis amigos, por ser y dejarme ser a vuestro lado. Nunca imaginé cuando entré por la puerta de la universidad que me llevaría una familia tan bonita.

Gracias Papá, por recordarme a través de tu ejemplo que con ganas se sueña y con esfuerzo se logra. Como decía Séneca, “la suerte es donde confluye la preparación y la oportunidad”. Tú me has enseñado que hay que prepararse, porque el que se prepara y se forma es capaz de percibir mucho mejor las oportunidades.

Por último, gracias Mamá por enseñarme a dar mis primeros pasos, guiarme en todos los que doy y dejarme seguir los tuyos. Gracias por enseñarme a luchar en lo que creo y confiar en mí cuando ni yo lo hacía. Hay veces que no se necesita ver para creer sino creer para ver y tú eres quien siempre ha creído en mí.

Resumen

En este trabajo fin de Grado se trata el diagnóstico de fallos de rodamientos en motores de inducción alimentados directamente de la red eléctrica durante el estado estacionario. Se aborda no solo la detección de fallos incipientes en los motores, de especial interés en la industria por el coste que puede suponer una parada en un motor, sino la detección de cualquier tipo de fallo.

Teniendo en cuenta buenos resultados obtenidos en contextos similares en trabajos previos utilizando técnicas muy sofisticadas de última generación, aquí se pretende averiguar si técnicas más clásicas, como el análisis discriminante lineal o los árboles de decisión, permiten obtener también buenos resultados evitando los problemas tanto computacionales como de interpretación de esas otras técnicas más complejas. Asimismo, se abordan otras cuestiones de interés desde el punto de vista industrial, como si el nivel de carga del motor influye en la capacidad de los procedimientos a la hora de detectar los fallos, o si el tipo de datos empleado en el procedimiento es relevante en esa capacidad de detección.

Las respuestas obtenidas a las cuestiones planteadas son muy relevantes tanto desde el punto de vista metodológico como práctico, puesto que indican que, en el caso que nos ocupa, efectivamente las técnicas clásicas tienen una alta capacidad de diagnóstico de los fallos, en qué casos es necesario programar mantenimientos en los que se baje la carga del motor para detectar mejor los fallos, o que los datos obtenidos de la corriente eléctrica están entre los que permiten una mejor capacidad de detección, lo que es muy interesante puesto que son los datos que pueden obtenerse de forma menos invasiva durante el trabajo del motor.

Estos resultados abren también líneas de trabajo futuro para intentar saber si este tipo de comportamiento aparece también para otros tipos de fallo del motor, como fallos en el estator, o en otras condiciones de funcionamiento, como cuando el motor no está alimentado directamente de la red sino a través de variadores.

Palabras claves: motores de inducción, diagnóstico de fallos, análisis discriminante, árboles de decisión, validación cruzada.

Abstract

In this end-of-degree project, the diagnosis of bearing faults in induction motors directly powered by the electrical grid during steady-state operation is addressed. It covers not only the detection of incipient faults in motors, which is of special interest to the industry due to the cost associated with motor downtime, but also the detection of any type of fault.

Taking into account good results obtained in similar contexts in previous works using highly sophisticated techniques, this study aims to determine whether more classical techniques, such as linear discriminant analysis or decision trees, can also yield good results, avoiding the computational and interpretational problems of those other more complex techniques. Additionally, other industrial-relevant issues are addressed, such as whether the motor load level influences the capability of the procedures to detect faults, or if the type of data used in the procedure is relevant to this detection capability.

The answers obtained to the questions posed are highly relevant from both a methodological and practical standpoint. They indicate that, in the case at hand, classical techniques indeed have a high diagnostic capacity for faults. They also highlight the cases where scheduling maintenance with reduced motor load is necessary for better fault detection, and that the data obtained from electrical current are among those that provide better detection capability, which is very interesting since they can be obtained in a less invasive manner during motor operation.

These results also open up avenues for future research to determine if this type of behavior also occurs for other types of motor faults, such as faults in the stator, or under other operating conditions, such as when the motor is not directly powered by the grid but through variable frequency drives.

Parte I

Descripción del proyecto

Capítulo 1

Introducción

En la introducción detallaremos la casuística e importancia del problema que queremos tratar. Así mismo expondremos los objetivos que se persiguen y que son claramente definidos mediante la metodología más adecuada para este proyecto, resaltando el componente original que pretende aportar a investigaciones anteriores. Por último, describiremos cómo está organizada esta memoria de Trabajo de Fin de Grado (TFG).

1.1. Motivación y descripción del problema

Una máquina eléctrica es un dispositivo donde se lleva a cabo la conversión electromecánica de energía, es decir, que convierten la energía eléctrica que se les entrega en energía mecánica para accionar otras máquinas o dispositivos. El objeto de estudio de este proyecto son los motores de inducción, máquinas eléctricas en las que, tanto en el estator como en el rotor, circulan corrientes alternas.

Los motores de inducción son un componente crítico en la industria ya que proporcionan más del 80% de la fuerza motriz (potencia mecánica) empleada en el presente en los procesos industriales (véase García-Escudero et al. [12]). Tanta es su presencia que se emplean en la mayoría de maquinaria, principalmente porque son más potentes y ecológicos que otros motores convencionales que encontramos en el mercado (véase Siyambalapitiya et al. [33])

En vista de que los motores de inducción están cobrando cada vez más importancia en el sector industrial, la detección temprana de fallos es muy importante para evitar el fallo completo del motor y pérdidas de producción inesperadas en la industria.

Debido al habitual ambiente áspero y corrosivo del entorno industrial, los motores eléctricos están potencialmente expuestos a un elevado riesgo de fallos mecánicos, químicos y eléctricos inesperados y agravados por el hecho de que es habitual que sufran de fatiga mecánica debido a mayores tiempos de uso que los estimados y con ausencia de mantenimientos periódicos.

Asimismo, el mantenimiento de los sistemas eléctricos no puede alcanzar gran potencial si solo se limita a enfoques preventivos y correctivos. Esto se debe a que estos mantenimientos no aprovechan al máximo la vida útil de los equipos sustituyéndolos, o

bien mucho antes de que se produzca el fallo, o cuando ya ha aparecido la avería. Es mucho más conveniente el denominado mantenimiento predictivo. El diagnóstico predictivo de una falla en desarrollo permite que el personal de mantenimiento programe reparaciones en el instante óptimo, es decir, justo en el momento antes de que empiece a fallar. Para anticiparnos al fallo se realiza un control de determinadas variables de funcionamiento, que van a precisar el momento adecuado para intervenir y así corregir lo que en un futuro sería una probable avería.

Por tanto, resulta evidente que para mantener el estado del motor, técnicas como la monitorización, detección, clasificación y diagnóstico de fallos son cada vez más esenciales debido a que la mayoría de los fallos de los motores eléctricos interrumpen el proceso, reducen la producción y pueden dañar otras máquinas relacionadas. Por consiguiente, el mantenimiento predictivo permite disminuir el coste de reparación.

Cabe señalar que los fallos mecánicos tales como desequilibrios, barras rotas del rotor o defectos en rodamientos suelen manifestarse mediante la generación de vibraciones y sonido excesivo que en condiciones ideales no deberían existir. Por este motivo, estas perturbaciones proporcionan información útil del estado de salud del motor. Dichos sonidos y vibraciones tienen asociados unas frecuencias características o armónicos que nos permiten determinar el estado del motor mediante el análisis de dicha señal.

Puesta en consideración la motivación del asunto tenemos que acotar el problema que trataremos. La tesis central de este trabajo trata de abordar la detección incipiente de fallos debido a la contaminación por partículas de los rodamientos de motores de inducción en estado estacionario (cuando la señal está estabilizada) mediante el análisis de datos relativos a la señal de corriente, el sonido y las vibraciones, medidas de carácter no invasivo y de bajo coste que nos permiten predecir el estado del motor sin interrumpir el funcionamiento del mismo.

Uno de los objetivos principales de este estudio, que desarrollamos con más detalle en la siguiente sección, es de tipo metodológico puesto que se pretenden construir modelos válidos con los que realizar un diagnóstico exitoso investigando, para comenzar, el desempeño de técnicas de aprendizaje estadístico como el análisis discriminante lineal o los árboles de decisión en este tipo de fallos. El propósito es dar una respuesta tentativa a la pregunta de si métodos más opacos como boosting y redes neuronales (apoyándonos en los hallazgos del estudio de Manuel Astorgano [2]) proporcionan una mejora sustancial en el diagnóstico de los fallos que atenúe la falta de interpretabilidad de estos procedimientos más opacos, o si, por el contrario, métodos estadísticos tradicionales y más simples tienen una capacidad predictiva similar a los más complejos a la vez que nos permiten adquirir y entender el conocimiento aprendido por el clasificador (información interpretable).

El interés de esta investigación también reside en un enfoque industrial, ya que los resultados obtenidos en este estudio podrían suponer una importante contribución al diagnóstico predictivo de fallos industriales en motores de inducción dado que al contrastar el estado del motor sano con el estado del motor levemente dañado somos capaces de determinar con exactitud cuándo comienza a fallar el motor (fallo incipiente). Con ello evitaríamos la falla catastrófica del sistema que podría suponer la destrucción total del motor, acarrear consecuencias para el resto del sistema de producción, e incluso poner en

riesgo la vida de las personas.

1.2. Objetivos

A grandes rasgos, la cuestión central discutida en detalle en este documento es la construcción de un sistema de diagnóstico de fallos incipientes en rodamientos para motores de inducción mediante aprendizaje supervisado. Dentro del aprendizaje supervisado se engloban un conjunto de técnicas de análisis de datos que utilizan algoritmos que consideran un conjunto de datos para los que se conocen tanto las variables predictoras como el estado del motor (que se denominan datos de entrenamiento) para construir reglas de clasificación diseñadas para poder predecir el estado del motor a partir de las variables predictoras en situaciones en las que este estado no es conocido.

Nuestro estudio se lleva a cabo bajo los siguientes supuestos:

- El motor se alimenta directamente de la red eléctrica de distribución (sin variadores).
- El motor está operando en estado estacionario, no en la fase de arranque.
- Se dispone de datos del motor para dos niveles de carga (alto y bajo).
- Las magnitudes físicas empleadas para caracterizar el fallo (variables predictoras) se obtienen a partir de tres fuentes de información: la corriente, la vibración y el sonido del motor.

En cuanto a la variable respuesta (estado del motor), en un principio en los ensayos se consideraron 6 niveles de fallo crecientes, desde motor sano a motor completamente dañado. Nuestro banco de datos está basado en un número pequeño de ensayos y con un equilibrio entre las clases, conteniendo el mismo número de observaciones correspondientes a cada uno de los 6 estados. Inicialmente, en este estudio se considera la detección del fallo incipiente (en principio más difícil de detectar) por lo que se analizará el motor sano contra fallo leve, aunque el estudio también se extiende al caso del motor sano contra todo tipo de fallo.

Las técnicas estadísticas de aprendizaje supervisado que utilizaremos serán el análisis discriminante lineal y los árboles de decisión.

Además, a través de la consecución del objetivo principal la investigación también pretende abordar los siguientes objetivos específicos:

- Estudiar y validar cómo procedimientos estadísticos menos complejos como los anteriormente mencionados clasifican fallos incipientes de motores y compararlos con las metodologías de aprendizaje no supervisado Boosting y Redes neuronales.
- Determinar la influencia de la carga del motor en la calidad de la clasificación de los fallos. Esta cuestión es importante desde el punto de vista industrial porque permitirá saber si es necesario bajar la carga del motor para tener un mejor diagnóstico.

- Conocer qué fuentes de información proporcionan mayor capacidad de clasificación de los fallos. Este punto también es importante desde el punto de vista industrial puesto que hay fuentes de datos (corriente) que son más fáciles de obtener y menos invasivas que otras (sonido y vibraciones).
- Realizar un estudio que permita determinar qué variables son más idóneas en la etapa de clasificación para el desarrollo del diagnóstico. Esto puede proporcionar información sobre qué variables es necesario medir o no en la práctica.

1.3. Estructura

Este trabajo se desarrolla en los siguientes capítulos.

- **Capítulo 1: Introducción.** En este capítulo se introduce el problema a tratar, su motivación y los principales objetivos que se abordan.
- **Capítulo 2: Motores de inducción.** En este capítulo se presenta una introducción a los motores de inducción y su historia, describiendo sus elementos constructivos, sus principios de funcionamiento y los tipos de fallos más comunes por los que se pueden ver afectados.
- **Capítulo 3: Planificación.** Se expondrá el análisis de la metodología elegida y la gestión de la planificación llevada a cabo para el desarrollo del proyecto junto al presupuesto del mismo. Se incluirá el balance temporal real del desarrollo del trabajo.
- **Capítulo 4: Metodología.** Se tratará la metodología empleada en el análisis de las señales, es decir, tanto la explicación teórica como práctica de los distintos métodos y técnicas estadísticas empleadas.
- **Capítulo 5: Datos.** Se abordarán todos los aspectos relacionados con la obtención, descripción y procesamiento de los datos del proyecto, las transformaciones y selección de variables.
- **Capítulo 6: Análisis y resultados.** En este capítulo a partir de los modelos desarrollados con los conjuntos de datos se interpretan y evalúan los resultados para obtener unas conclusiones que aporten respuesta a la motivación de los objetivos específicos del proyecto.
- **Capítulo 7: Conclusiones y trabajos futuros.** En este capítulo se resume el aprendizaje adquirido y se plantean futuras posibilidades de investigación.

Capítulo 2

Motores de inducción

2.1. Crecimiento en la industria

Como ya se ha comentado previamente, el motor de inducción es, en gran medida, una máquina fiable y es esta característica la que ha impulsado su introducción masiva en el mundo industrial, comercial y doméstico. Se utilizan en las industrias de minería, metales y cemento, automotriz, petróleo y gas, salud y manufactura, y forman parte de bombas, elevadores, cintas, montacargas, sistemas de ventilación, grúas, trituradoras, equipos de extracción de petróleo, etc.

La protección ambiental en todas las industrias también contribuye al crecimiento de los motores de inducción, ya que tienen una tasa de emisión baja. Es por ello que la creciente demanda de energía eléctrica para sostener el desarrollo global que requiere fuertes inversiones que consisten en la generación de suministro de energía esté impulsando así la aplicación de motores de inducción en el sector eléctrico. Del mismo modo, el mercado de vehículos eléctricos está impulsando la demanda de motores de inducción en la industria automotriz.

En los últimos años, la demanda de recursos para construir viviendas residenciales y comerciales se ha visto incrementada. Esto se debe a que se prevé que la población mundial crezca de 7700 millones en 2019 a 8500 millones en 2030 (aumento del 10%) y a 9700 millones en 2050 (26%). Por lo tanto, máquinas de inducción como montacargas y grúas son el sustento de la mayoría de los proyectos de construcción. Así bien, según el informe de Mordor Intelligence [17], no sólo el sector de la construcción prevé ser contribuyente al crecimiento del mercado de motores de inducción ya que otras industrias, como la del petróleo, la del gas y la textil están experimentando un crecimiento constante sobre todo en Asia-Pacífico, lo que se espera que ofrezca una gran oportunidad de crecimiento para los actores mundiales del motor de inducción.

2.2. Historia de los motores de inducción

Según se describe en [16], la historia del motor eléctrico se remonta a principios del siglo XIX cuando se desarrollaron los principios básicos del electromagnetismo gracias a los experimentos de Oersted, Faraday, Henry, Lenz, Barlow y Maxwell.



Figura 2.1: Batería eléctrica de Volta.

Aunque a Faraday se le atribuye popularmente la creación del primer motor eléctrico, existen hechos anteriores que se remontan casi un siglo antes (véase, por ejemplo la figura 2.1), y que demuestran que el primer motor eléctrico fue un motor electrostático simple desarrollado en 1740 por un monje benedictino escocés, Andrew Gordon. Más tarde, en 1820, el francés André-Marie Ampere publicó el principio que declara la producción de la fuerza mecánica por las interacciones del campo magnético y de la corriente eléctrica. Es la llamada ley de la fuerza de Ampere, y que constituye también otra unidad de medición, la de la intensidad de corriente eléctrica.

El británico Michael Faraday, en 1821 declaró el principio para convertir la energía eléctrica en energía mecánica utilizando el electromagnetismo. Faraday llevó a cabo el siguiente experimento. Un imán cilíndrico se sostiene colgando verticalmente con uno de los polos sumergido en mercurio. El polo superior se conecta al mercurio mediante un cable. Si el imán se pone en movimiento de rotación, se observa el paso de corriente por el galvanómetro. Si se sustituye el galvanómetro por una batería que suministre corriente al circuito, el imán empieza a girar espontáneamente alrededor de su eje, tenemos entonces un motor que genera movimiento (figura 2.2).

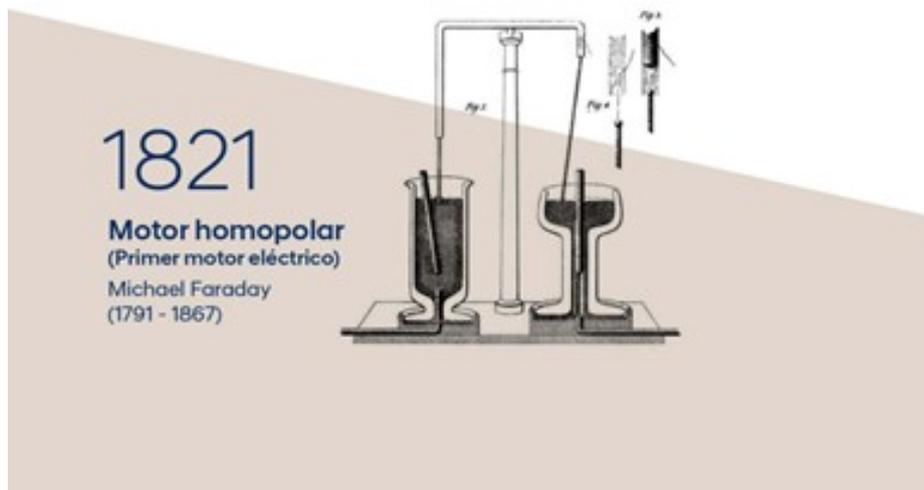


Figura 2.2: Motor homopolar de Faraday.

La ley de inducción electromagnética de Faraday llevó a muchos científicos e ingenieros a intentar crear una máquina que produjera electricidad de formas desconocidas en ese momento. Al ingeniero Moritz von Jacobi se le atribuye la invención del primer motor eléctrico práctico y su presentación al mundo en 1834 (figura 2.3).

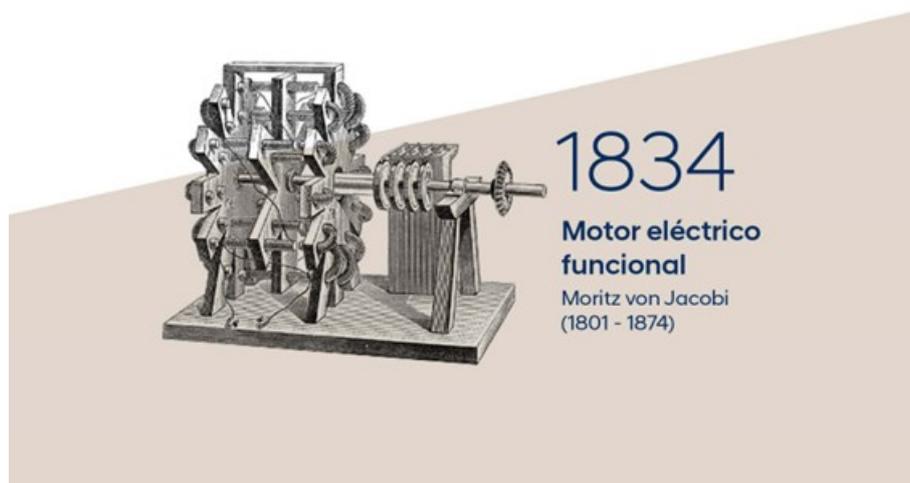


Figura 2.3: Motor eléctrico funcional de Jacobi.

Resulta imprescindible la mención a Nicola Tesla cuando hablamos de la historia de los motores. Y es que en 1888 ideó el primer motor de inducción de corriente alterna (figura 2.4) apoyado en la creación de Galileo Ferraris del año 1885.



Figura 2.4: Motor de corriente alterna de Tesla.

En 1889 un empleado de la empresa alemana AEG llamado Mikhail Dolivo-Dobrowolsky, inventó el motor de inducción trifásico de rotor bobinado además de un generador trifásico y en 1991 se desarrolló el motor de inducción de rotor de jaula de ardilla.

2.3. Principios de funcionamiento

Como ya hemos comentado, las máquinas eléctricas hacen uso fundamentalmente del campo magnético como medio de acoplamiento para la conversión de energía. Por tanto, nos interesará tener campos magnéticos elevados, lo que se consigue utilizando materiales de alta permeabilidad magnética. Esta es la razón por la que se utilizan materiales ferromagnéticos en las máquinas eléctricas. Además, estos materiales poseen unas propiedades mecánicas muy buenas y se utilizan también como soporte de los conductores y sus aislamientos.

Se da el nombre de asíncrono al motor de corriente alterna cuya parte móvil gira a una velocidad distinta a la de sincronismo (véase el libro de Chapman [6]). La velocidad de sincronismo de los motores eléctricos de corriente alterna viene definida por la expresión:

$$n = 60 \cdot \frac{f}{p}$$

donde n es el número de revoluciones por minuto, f es la frecuencia de la red y p es el número de pares de polos de la máquina.

Como todas las máquinas eléctricas, un motor eléctrico está constituido por un circuito magnético y dos eléctricos, uno colocado en la parte fija (estator) y otro en la parte móvil (rotor). El circuito magnético está formado por chapas apiladas en forma de cilindro en el rotor y en forma de anillo en el estator. El cilindro se introduce en el interior del anillo

y, para que pueda girar libremente, hay que dotarlo de un entrehierro constante. El anillo se dota de ranuras en su parte interior para colocar el bobinado inducido (motores de rotor bobinado) o bien se le incorporan conductores de gran sección soldados a anillos del mismo material en los extremos del cilindro (motores de jaula de ardilla, llamados así por su similitud a la rueda de un hamster). El eje se apoya en unos rodamientos de acero para evitar rozamientos y se saca al exterior para transmitir el movimiento, y lleva acoplado un ventilador para refrigeración. Los extremos de los bobinados se sacan al exterior y se conectan a la placa de bornes. El estator que es la parte inmóvil y el rotor, y que está ubicada dentro del estator, que es la parte que rota, lo hace alrededor de un eje situado en su centro y que se encuentra sustentado sobre los rodamientos y de tal forma que pueda girar. Ambos componentes están compuestos por devanados fabricados con algún tipo de conductor eléctrico y protegidos por aislantes. El espacio entre el rotor y el estator se conoce como entrehierro. Además, los devanados quedan protegidos por una carcasa. El funcionamiento es simple, corriente alterna sinusoidal alimenta los devanados del estator son alimentados por corriente alterna sinusoidal generando un campo magnético que penetra en los conductores del devanado del rotor, suscitando una corriente en éste. A causa de la ley de Lenz, en el rotor se producen fuerzas electromotrices que va a intentar seguir al campo giratorio del estator, por lo que el rotor comenzará a girar en el mismo sentido. Para que se produzca este fenómeno, es necesario que haya un movimiento relativo entre el campo magnético y los conductores inducidos. Por lo tanto, el rotor nunca alcanzará la misma velocidad que el campo (velocidad síncrona), ya que esto daría lugar a la anulación de las fuerzas electromotrices. La velocidad del rotor aumentará más o menos dependiendo del par resistente al que esté sometido. Por ejemplo, en el vacío la velocidad del rotor sería casi idéntica. La figura 2.5 muestra las diferentes partes que actúan en un motor jaula de ardilla.

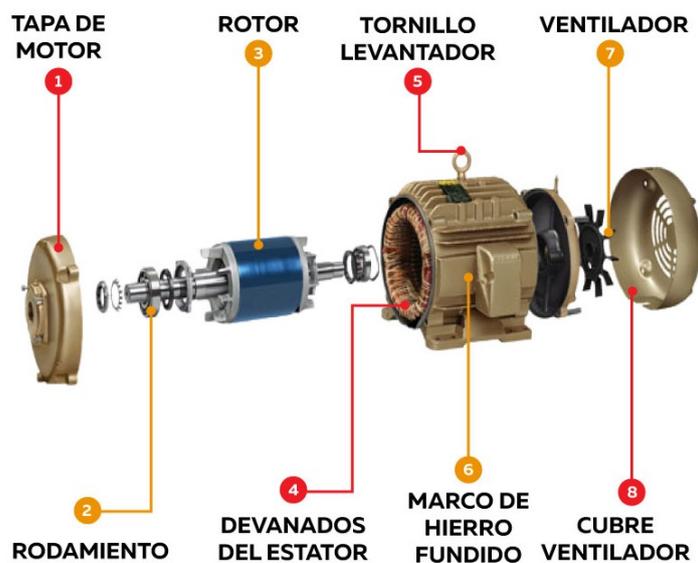


Figura 2.5: Partes de un motor de jaula de ardilla.

2.4. Tipos de fallo y detección

No obstante, aunque el motor de inducción posee un carácter robusto que ha impulsado su introducción masiva en la industria, el comercio y los hogares, éstos están sometidos a un elevado riesgo de averías.

El motor de inducción está concebido para soportar amplia variedad esfuerzos en los devanados del estator, rotor, núcleos magnéticos, jaula, rodamientos y eje. Si estas tensiones permanecen dentro de sus rangos característicos de diseño y el uso del motor es correcto, no debe ocurrir una falla prematura del motor. Sin embargo, exceder cualquiera de estos límites puede reducir significativamente la vida útil del motor y causar una falla catastrófica.

Los principales fallos del motor, (véase [1] y [21]), se pueden clasificar de la siguiente manera:

1. **Fallos del estator que dan lugar al cortocircuito de una bobina o más de un devanado de fase del estator.**

El estator y el aislamiento del devanado del rotor están sujetos durante el funcionamiento a una combinación de tensiones térmicas, eléctricas, vibratorias y termo mecánicas que provocan un envejecimiento y que finalmente conduce a la ruptura del aislamiento.

2. **Conexión anormal de los devanados del estator.**

Cuando el motor trabaja en una sola fase una de las conexiones de línea no está conectada. Una condición monofásica somete al motor a un desequilibrio de voltaje excesivo, lo que generalmente significa altas corrientes y calentamiento del motor.

3. **Barra del rotor rota o anillos terminales del rotor agrietados.**

4. **Irregularidades estáticas y/o dinámicas en la separación del aire.**

La excentricidad es un tipo de falla de asimetría rotórica, la cual se caracteriza por la desigualdad en el entrehierro de la máquina. La presencia de ésta, incrementa los niveles de vibración tanto en dirección axial como radial, debido a las desigualdades magnéticas creadas entre rotor y estator.

5. **Eje doblado (similar a la excentricidad dinámica)**

Esta circunstancia puede provocar un roce entre el rotor y el estator, causando graves daños al núcleo del estator y a los devanados.

6. **Fallos en los rodamientos y la caja de cambios.**

Casi el 40-50 % de todos los fallos están relacionados con rodamientos, siendo el caso de los fallos de este estudio. Cada uno de estos rodamientos consta de dos anillos, uno interno y el otro externo. Un conjunto de bolas o elementos rodantes colocados en las pistas de rodadura giran dentro de estos anillos. Los rodamientos están definidos para unas condiciones de servicio dadas (carga y velocidad). La vida útil de un rodamiento está medida por el número de vueltas u horas de funcionamiento, a velocidad constante, que puede efectuar sin que aparezcan los primeros síntomas de fallo debido a la tensión continua en estos rodamientos.

Los métodos de diagnóstico para identificar las fallas anteriores pueden involucrar varios tipos de campos de la ciencia y la tecnología.

- **Vigilancia del campo electromagnético.** [10, 22] Cuando un motor de inducción funciona en condiciones normales se producen variaciones sinusoidales en el flujo de aire. Sin embargo, los rotores o estatores asimétricos pueden conducir a cambios en las variaciones de la señal.
- **Mediciones de temperatura.** [11, 26, 27] En caso de cortocircuito de la bobina del estator del motor, la corriente en el estator será tan alta que podría generar un calor excesivo y dar lugar a la avería de la bobina del motor. Variaciones en la temperatura pueden ser, por tanto, indicios de posibles problemas en el motor.
- **Vigilancia de vibraciones.** [34, 7, 8] Cuando se producen fallos en los componentes internos de un motor de inducción, se generan señales de vibración cuya detección puede servir para el diagnóstico de posibles fallos futuros.

- **Mediciones de ruido acústico.** [32, 23] El ruido en las máquinas eléctricas se debe a la combinación de señal sonora generada por los rápidos cambios en la presión del aire. Una de las posibles fuentes de este ruido son los fallos mecánicos del motor. En consecuencia, el análisis de este tipo de señal puede conducir a detectar irregularidades en el comportamiento del motor.
- **Análisis de la firma de corriente motriz (MCSA).** [25, 19, 20] MCSA trabaja midiendo corrientes desde los estatores del motor de inducción. Las corrientes medidas se procesan para producir sus perfiles de espectro de potencia y estos perfiles, o resúmenes estadísticos de los mismos, se analizan para detectar cambios en el funcionamiento del motor que pueden ser indicios de fallos.

Este trabajo considera diferentes fuentes de información de las mencionadas anteriormente (corriente, sonido y vibraciones) con el objetivo, entre otros, de valorar, mediante métodos estadísticos de clasificación supervisada, cuál de estas fuentes de información proporciona mejores reglas de clasificación.

Capítulo 3

Planificación

3.1. Metodología de trabajo

Resulta preciso trazar una buena gestión de la planificación para realizar un proyecto con una calidad y unas especificaciones deseadas, ayudándonos a estructurarlo y establecer prioridades en las tareas permitiéndonos alcanzar el mejor control del tiempo y agregando el máximo valor a nuestro proyecto. Nacida en 2001 a consecuencia de los retrasos que conllevaban las metodologías tradicionales en los desarrollos software de la industria, Agile es la metodología que mejora su calidad, incrementa la satisfacción del cliente y la toma de decisiones, acorta los ciclos de producción minimizando los tiempos de reacción y destacando la adaptación al cambio. Este aprendizaje que nos aporta feedback continuo para aprender de las iteraciones pasadas mejorando las iteraciones futuras está basado en el marco de trabajo Scrum. A pesar de que este trabajo no conlleva desarrollo de software y el equipo, formado por la estudiante y los tutores académicos, es reducido se ha elegido la metodología ágil por ser una metodología muy utilizada en la última década en el entorno profesional en el que la alumna deberá desenvolverse en el futuro.

A continuación, describiremos los valores, eventos, roles y artefactos que definen el marco de trabajo, así como las herramientas empleadas en este entorno y cómo se estima el esfuerzo.

3.1.1. Valores

Scrum basa su conocimiento en el empirismo, planteando un enfoque iterativo e incremental para optimizar la predictividad y controlar los cambios que puedan surgir. Cuando el equipo incorpora los valores de compromiso, coraje, enfoque, apertura y respeto, los pilares de transparencia, inspección y adaptación de Scrum cobran vida y crean confianza en todos (véase [24])

3.1.2. Roles

Los equipos Scrum, que son auto-organizados y multifuncionales, están compuestos por tres roles principales:

- El Product Owner es la persona responsable de asegurar que el equipo aporte valor al negocio. Representa las partes interesadas internas y externas, por lo que debe comprender y apoyar las necesidades de todos los usuarios en el negocio, así como también las necesidades y el funcionamiento del Equipo Scrum.
- El Equipo de Desarrollo (compuesto entre 3 y 9 personas) son todas las personas necesarias para hacer que un elemento del backlog del producto se transforme en un incremento del producto potencialmente entregable.
- El Scrum Master es un facilitador de equipos scrum. Ayuda al equipo a mantenerse enfocado en los objetivos del proyecto y elimina los impedimentos que van apareciendo durante el camino.

A continuación, vamos a definir los roles de nuestro proyecto que aunque están fundamentados en los roles de scrum, están ajustados al marco académico de desarrollo del TFG. Los papeles que desempeñan nuestro equipo son:

- El rol del estudiante. Será la alumna autora del TFG, Gema de Pablo Ordiales. Desempeña el papel del equipo de desarrolladores del proyecto (analista, diseñador, implementador, tester y documentador) constituyendo el núcleo de cualquier proyecto Scrum. Este rol es el responsable de entender los requerimientos especificados por el product owner, estimar las Historias de Usuarios y crear los sprints del proyecto realizando las entregas iterativas del producto.
- Tutores. En el presente TFG este rol lo desempeñan Miguel Fernández Temprano y Amelia García Garrosa. Ambos desarrollan los papeles de Scrum Master y de Product Owner. Por una parte, son Scrum Master al actuar como directores del proyecto y encargados de supervisar todo el trabajo para guiarlo hacia la consecución de objetivos. También desempeñan el rol de Product Owner, actuando como el cliente que demanda el proyecto y definiendo los requisitos que espera obtener en el producto a desarrollar. Además, los tutores ayudarán a resolver cualquier bloqueo que le pueda surgir al Estudiante.
- Tribunal. Basado en el rol de cliente se encarga de evaluar el Trabajo Fin de Grado y estará constituido por tres profesores o profesoras titulares y tres suplentes (presidente o presidenta, vocal y secretario o secretaria). Entre sus responsabilidades tendrán que valorar el grado de consecución de los objetivos de aprendizaje previstos para el trabajo a través del balance de la calidad del producto y de la documentación técnica generada.

3.1.3. Eventos

El evento central de la metodología de trabajo 'scrum' es el 'sprint'. Se trata de un miniproyecto de no más de un mes (ciclos de ejecución muy cortos -entre una y cuatro semanas), cuyo objetivo es conseguir un incremento de valor en el producto que estamos construyendo. Todo 'sprint' cuenta con una definición y una planificación que ayudará a lograr las metas marcadas. El resto de las ceremonias que se llevan a cabo dentro del sprint son las siguientes: el Sprint Planning, al comienzo del Sprint para planificar el trabajo a través del sprint backlog, el Daily Meeting, a diario para que en 15 minutos el equipo se sincronice, el Sprint Review, al final del sprint para evaluar el incremento realizado y el Sprint Retrospective, para inspeccionar el equipo y levantar mejoras que se apliquen en el siguiente sprint.

3.1.4. Herramientas

- Trello: es un tablero scrum que permite a los equipos gestionar el flujo de trabajo, así como supervisar las tareas y registrar toda información necesaria para su consecución. Uno de los elementos principales es el Product Backlog, que es una lista priorizada de los requisitos del producto. En la figura 3.1 se muestra un ejemplo de organización de tareas usando esta herramienta.

- Overleaf: es un editor online de documentos en LaTeX que permite compartirlos con otras personas para colaborar en tiempo real sobre el documento y realizar un control de versiones.

- Microsoft Outlook: es la herramienta de correo electrónico mediante la cual se establece la comunicación con los tutores para fijar las reuniones.

- Microsoft teams: es la herramienta que ha permitido programar las reuniones y mantener conversaciones con los tutores durante el desarrollo de este TFG.

- Python: es la herramienta de programación de alto nivel utilizada en el análisis junto con sus librerías principales Scikit-learn, NumPy, Pandas y Matplotlib.

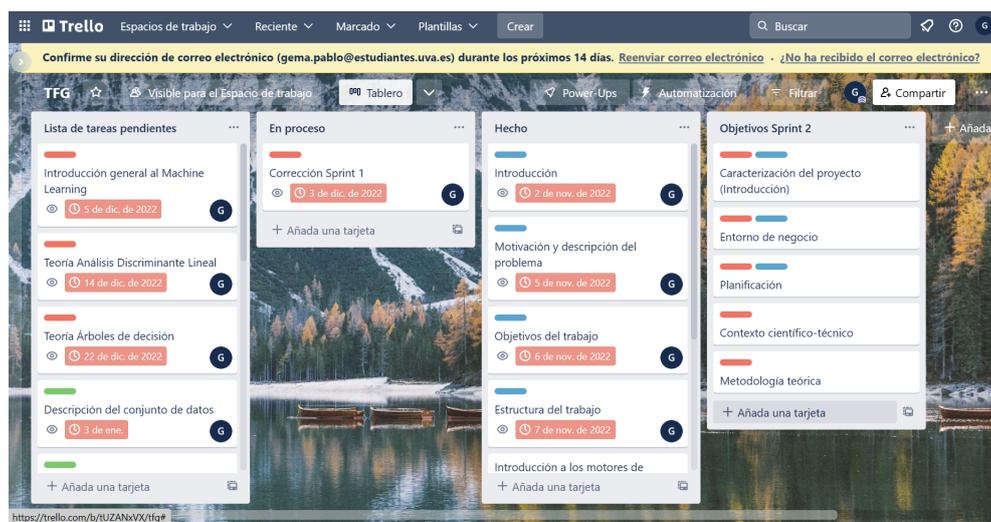


Figura 3.1: Organización de tareas en Trello.

3.1.5. Estimación del esfuerzo

El Planning Poker es una técnica que ayuda a un equipo gestionado con una metodología ágil a estimar el tamaño relativo de las tareas o, en otros términos, la cantidad de trabajo necesario para completar las historias de usuario del Product Backlog. Su finalidad principal es la de facilitar al Scrum Master el comprender exactamente cuánto trabajo se puede realizar en cada sprint.

Consiste en que cada miembro posee una baraja de cartas. Cada una de ellas tendrá un valor distinto fruto de las variaciones redondeadas de la secuencia de Fibonacci. Por lo general, estos valores representan puntos de historia. Sin embargo, algunos equipos pueden usarlos como una estimación basada en el tiempo para saber cuánto tiempo se necesita para finalizar una historia de usuario.

Según se observa en la figura 3.2, adoptaremos el día como unidad de medida temporal. Además, tendremos 3 cartas únicas: una con el icono de ∞ , otra con el de $?$ y una última con una figura de una taza de café, que representa la ausencia de dificultad, hasta 21, que indica el nivel de dificultad más alto. En el caso de ∞ , quiere decir que la tarea debe subdividirse porque si no, es imposible de realizar. La carta con un $?$, quiere decir que no sabemos cuál será la dificultad, y la representación del café se utiliza para realizar una pausa cuando se está planificando.

Finalmente, se hace una lectura en alto de cada tarea y cada persona revela la carta elegida para cada tarea. Si coincide, se registran los puntos de historia en el tablero del proyecto. En caso de que exista desacuerdo entre las personas, se procederá a debatir las razones por las cuales cada individuo ha asignado los puntos correspondientes. Después se tiene que repetir la estimación, y se repetirán estos 2 últimos pasos hasta que se alcance un consenso. Tras realizar la estimación del esfuerzo contabilizamos un total de 121 puntos de historia.



Figura 3.2: Cartas de planning poker.

3.2. Planificación temporal

Este Trabajo de Fin de Grado tiene una carga de 12 ECTS que equivalen a 300 horas de trabajo. Definiremos un plan de actuación, planificando la gestión de tiempos, definiendo las actividades y sus puntos de historia asociados, estimando los recursos y la duración de las actividades para finalmente desarrollar un cronograma. Éste, que tendrá forma de Diagrama de Gantt, mostrará el esfuerzo real del proyecto ya que ciertos retrasos en la realización de las tareas han supuesto una adaptación dinámica en la planificación.

Nuestra propuesta inicial tiene un total de 4 sprints cada uno de 4 semanas y 75 horas de trabajo. El primer Sprint comienza el 31 de octubre y el cuarto Sprint termina el 19 de febrero. Debemos definir las tareas y el esfuerzo para abordarlas (Puntos de Historia) y para ello dividiremos cada sprint en unos subobjetivos:

3.2.1. Sprint 1

Para cumplir el objetivo del Sprint 1 hay que desempeñar los siguientes 3 subobjetivos y 11 tareas

SUBOBJETIVO 1.1 Caracterización del proyecto (Introducción)

SUBOBJETIVO 1.2 Entorno de negocio

SUBOBJETIVO 1.3 Planificación

- TAREA 1.1 Caracterización del proyecto (Introducción) – Introducción
- TAREA 1.2 Caracterización del proyecto (Introducción) – Motivación y descripción del problema
- TAREA 1.3 Caracterización del proyecto (Introducción) – Objetivos del trabajo
- TAREA 1.4 Caracterización del proyecto (Introducción) – Estructura del trabajo
- TAREA 1.5 Entorno de negocio – Introducción a los motores de inducción.

- TAREA 1.6 Entorno de negocio – Historia de los motores de inducción.
- TAREA 1.7 Entorno de negocio – Principios de funcionamiento.
- TAREA 1.8 Entorno de negocio – Tipos de fallo y detección.
- TAREA 1.9 Planificación – Metodología de trabajo
- TAREA 1.10 Planificación – Planificación temporal
- TAREA 1.11 Planificación – Presupuesto

Subobjetivos	S 1.1	S 1.2	S 1.3
Tareas			
T 1.1	✘		
T 1.2	✘		
T 1.3	✘		
T 1.4	✘		
T 1.5		✘	
T 1.6		✘	
T 1.7		✘	
T 1.8		✘	
T 1.9			✘
T 1.10			✘
T 1.11			✘

Figura 3.3: Subobjetivos alcanzados durante el Sprint 1.

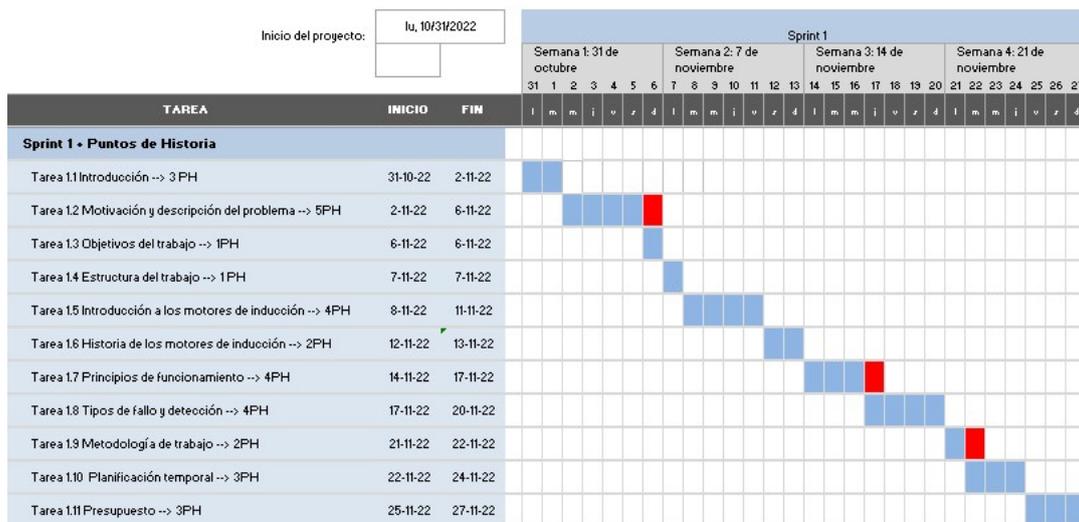


Figura 3.4: Diagrama de Gantt del Sprint 1.

3.2.2. Sprint 2

Para cumplir el objetivo del Sprint 2 hay que desempeñar los siguientes 4 subobjetivos y 8 tareas:

SUBOBJETIVO 2.1 Caracterización del proyecto (Introducción)

SUBOBJETIVO 2.2 Entorno de negocio

SUBOBJETIVO 2.3 Planificación

SUBOBJETIVO 2.4 Metodología teórica

- TAREA 2.1- Correcciones Sprint 1
- TAREA 2.2 Metodología teórica –Introducción al problema de clasificación
- TAREA 2.3 Metodología teórica – Análisis discriminante lineal
- TAREA 2.4 Metodología teórica – Árboles de decisión
- TAREA 2.5 Metodología teórica – Construcción y evaluación de modelos: Medidas del error
- TAREA 2.6 Metodología teórica – Construcción y evaluación de modelos: hold-out y validación cruzada
- TAREA 2.7 Metodología teórica – Construcción y evaluación de modelos: búsqueda de hiperparámetros
- TAREA 2.8 Metodología teórica – Análisis de la varianza (ANOVA)

Subobjetivos	S 2.1	S 2.2	S 2.3	S 2.4
Tareas				
T 2.1	✘	✘	✘	
T 2.2				✘
T 2.3				✘
T 2.4				✘
T 2.5				✘
T 2.6				✘
T 2.7				✘
T 2.8				✘

Figura 3.5: Subobjetivos alcanzados durante el Sprint 2.

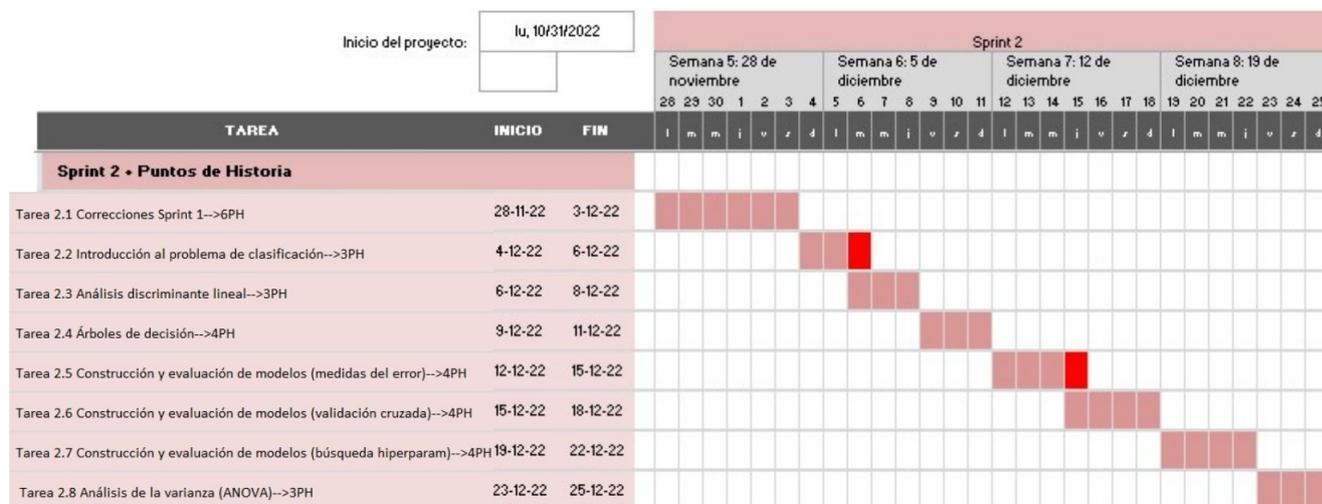


Figura 3.6: Diagrama de Gantt del Sprint 2.

3.2.3. Sprint 3

Para cumplir el objetivo del Sprint 3 hay que desempeñar los siguientes 4 subobjetivos y 6 tareas:

- SUBOBJETIVO 3.1** Metodología teórica
- SUBOBJETIVO 3.2** Conjunto de Datos
- SUBOBJETIVO 3.3** Análisis
- SUBOBJETIVO 3.4** Resultados

- TAREA 3.1 Correcciones Sprint 2
- TAREA 3.2 Conjunto de Datos – Descripción del conjunto de datos
- TAREA 3.3 Conjunto de Datos – Procesamiento de datos
- TAREA 3.4 Análisis – Experimento
- TAREA 3.5 Análisis – Análisis 1 contra 2
- TAREA 3.6 Resultados – Resultados Análisis 1 contra 2

Subobjetivos	S 3.1	S 3.2	S 3.3	S 3.4
Tareas				
T 3.1	✘			
T 3.2		✘		
T 3.3		✘		
T 3.4			✘	
T 3.5			✘	
T 3.6				✘

Figura 3.7: Subobjetivos alcanzados durante el Sprint 3.

Capítulo 3. Planificación



Figura 3.10: Diagrama de Gantt del Sprint 4.

historia entre las jornadas de trabajo concluimos que deberemos de realizar 1,08, es decir, 1 punto de historia por jornada. Además, cada jornada contará con 2,67 horas, lo que suponen aproximadamente 3 horas diarias para que el balance temporal se ajuste a las 300 horas de carga que se corresponden al desarrollo de un Trabajo de Fin de Grado.

En la figura 3.11 presentamos el diagrama de Gantt del proyecto al completo.

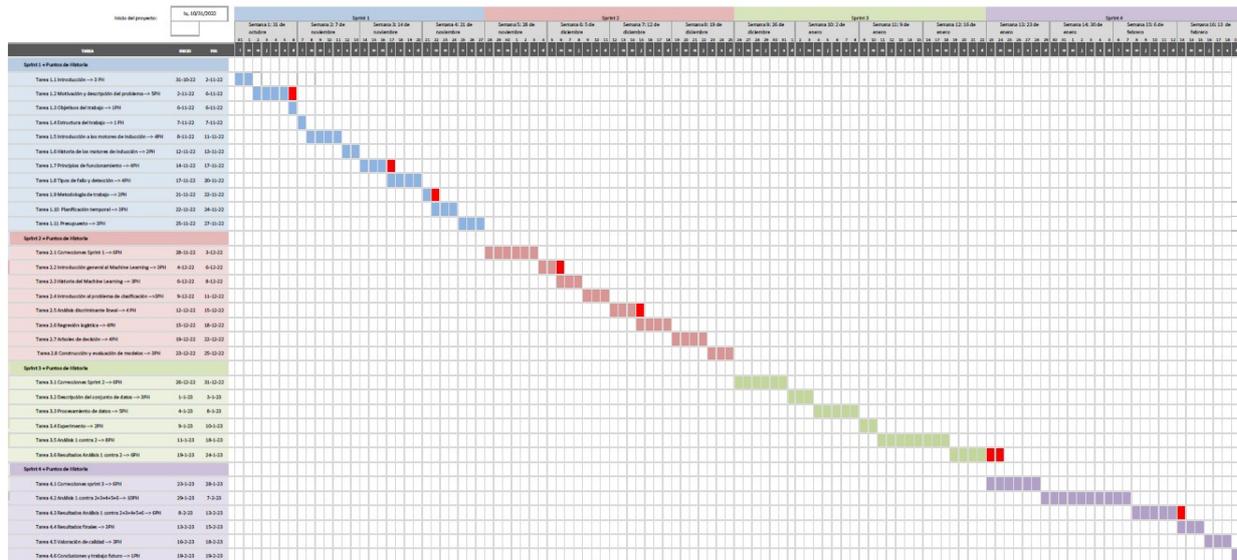


Figura 3.11: Diagrama de Gantt Completo - Planificación inicial

3.3. Balance temporal real del proyecto

En este apartado se expone el análisis temporal real llevado a cabo en el transcurso del trabajo. Se han presentado ciertas dificultades que han dado lugar a una modificación de la planificación temporal previamente establecida. Esto es debido, principalmente, a que la carga docente de las asignaturas del primer cuatrimestre y la convocatoria de exámenes

finales de enero consumieron más tiempo de lo previsto y la dedicación al trabajo fin de grado fue mucho menor de lo deseado en los primeros meses.

Aunque se había iniciado el trabajo en el primer cuatrimestre, una planificación más realista sitúa la fecha de comienzo el 1 de Febrero desplazando la ejecución de las tareas, que finalizan el 31 de Mayo. Sin embargo, el proyecto ha evolucionado de una manera consistente con la planificación inicial en cuanto al tiempo dedicado a cada tarea y sin cambios significativos en su trayectoria. Se pueden observar los retrasos marcados en rojo en la figura 3.11. En la figura 3.12 se presenta un diagrama de Gantt simplificado con la evolución temporal final de los cuatro sprints. Se observa que el último sprint se completa en una semana más y esto se debe a la dilación en la tarea de correcciones del Sprint 4, sobre todo, a contratiempos y ajustes en los análisis.

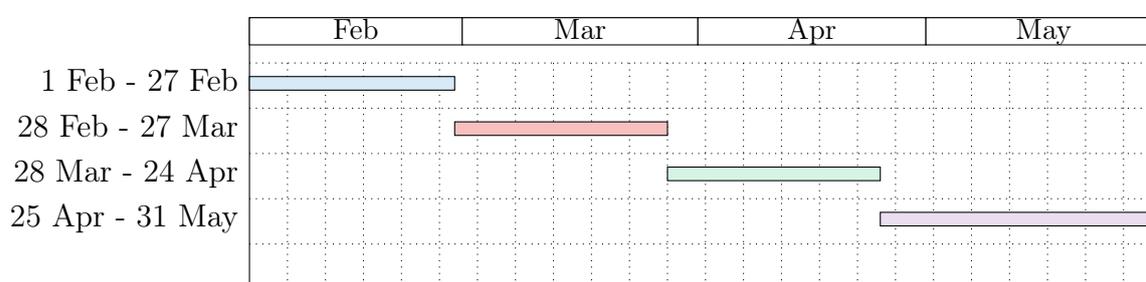


Figura 3.12: Diagrama de Gantt Completo - Balance temporal real

Se ha estimado que desde febrero se emplearon una media de 4 horas debido a que la alumna tuvo que compatibilizar el desarrollo de las prácticas curriculares en empresa con la docencia de las últimas asignaturas y el desarrollo del TFG por lo que en el primer sprint y en el segundo sprint se trabajaron 74,76 horas, en el tercer sprint 84,07 horas y en el último sprint 140 horas. En conclusión, el proyecto ha llevado un total de 373,5 horas.

3.4. Presupuesto

Del mismo modo que resulta imprescindible realizar una planificación temporal para distribuir el esfuerzo estimado a lo largo de la duración prevista del proyecto y asignarlo a los miembros específicos del equipo, también resulta esencial desarrollar un plan presupuestario. Un presupuesto es una estimación detallada de todos los costes necesarios para completar las tareas del proyecto.

Debemos tener en cuenta los gastos en herramientas de hardware y software, en recursos humanos y otros gastos como los debidos a la conexión a Internet. Para que nuestra medida se ajuste lo más posible a la realidad a la hora de estimar el coste del equipo hardware y software, se tendrá en cuenta el tiempo, la frecuencia de uso y la vida útil de las herramientas.

3.4.1. Hardware y software

Un ordenador para proyectos de ciencia de datos necesita una CPU potente con multi-núcleos e hilos para el procesamiento de los datos, por eso elegiremos un equipo de gama media/alta. Dicho dispositivo tiene un valor de 1000 € en el mercado y posee las siguientes especificaciones: Portátil HP 15s-fq5061ns de 15.6” pulgadas, con procesador Intel Core i7-1255U - U15 de 12^a generación, pantalla FHD SVA antirreflectante con microborde, sistema operativo de 64 bits Windows 11, 16 GB de RAM, 256 GB de memoria SSD, y 1 TB de memoria HDD.

Además, necesitaremos conexión a internet para poder redactar la memoria en Overleaf utilizando lenguaje LaTeX y para poder revisar revistas y otros documentos. Aprovechamos la conexión para mantener comunicación con los tutores vía email y reuniones frecuentes mediante Microsoft Teams.

En cuanto a las herramientas de software, utilizaremos Trello, Overleaf, Python (Spyder, anaconda 3) y herramientas de Microsoft 365. Sin embargo, estas herramientas no suponen un incremento en el presupuesto debido a que, o bien tienen licencia gratuita, o bien se pueden utilizar con la licencia de la Universidad de Valladolid mientras la estudiante esté matriculada en dicha universidad.

Servicios	Precio en el mercado	% Uso	Meses de uso	Coste/mes	Coste/mes * % Uso	(Coste/mes * % Uso) * meses de uso
Ordenador Portátil	1000 euros	85 por ciento	4 meses	13.8 euros/mes	11.73 euros/ mes	46.92 euros
Conexión a internet	32 euros/mes	25 por ciento	4 meses	-	-	32 euros
Microsoft Outlook	0 euros	0 euros	0 euros	0 euros	0 euros	0 euros
Trello	0 euros	0 euros	0 euros	0 euros	0 euros	0 euros
Microsoft Excel	0 euros	0 euros	0 euros	0 euros	0 euros	0 euros
Overleaf	0 euros	0 euros	0 euros	0 euros	0 euros	0 euros
Spyder (anaconda 3)	0 euros	0 euros	0 euros	0 euros	0 euros	0 euros
					Total	78.92 euros

Figura 3.13: Gastos materiales

3.4.2. Recursos humanos

Nuestro proyecto requiere de distintos perfiles informáticos que asumirá la alumna al ser la única trabajadora de éste. Para calcular el presupuesto en recursos humanos debemos multiplicar las horas en las cuales se actúa con un cierto rol por el precio medio al que se paga la hora de trabajo a ese trabajador en el mercado. Además, añadiremos el coste adicional de dar de alta en la Seguridad Social a un trabajador que corresponde al 30 % del sueldo bruto.

La alumna desempeñará los roles de desarrollador especializado en Python para realizar la escritura y las pruebas del código, y de científico de datos que explorará y procesará los datos, además de trabajar conjuntamente con el desarrollador en los modelos y posteriormente interpretar los resultados.

Vamos a hacer una estimación de cada sueldo suponiendo que el trabajador realiza un total de 40 horas a la semana (160 horas al mes). En España, el sueldo medio de

un desarrollador Python es de 31.348 € /año, que equivale a 16,33 €/hora y el de un científico de datos de 42.164 € /año, cobrando 21,96 €/hora.

Finalmente se destinará un total de 8934.26 € para pagar al personal.

Gastos en RRHH	Salario (€/hora)	Horas	Coste (€)
Sueldo desarrollador Phyton	16.33 €/hora	125 horas	2041.25 €
Sueldo científico de datos	21.96 €/hora	220 horas	4831.2€
Pago Seguridad Social	-	-	2061.81 €
Total			8934.26 €

Figura 3.14: Gastos personal

Por tanto, sumando los importes de las figuras 3.13 y 3.14 se puede concluir que se necesita un presupuesto total de 9013,18 € (78,92+8934,26) para la realización del proyecto.

Parte II

Metodología, análisis y resultados

Capítulo 4

Metodología

En este capítulo se describe la metodología utilizada en el abordaje del problema a tratar en este trabajo. El problema considerado es un problema de clasificación supervisada por lo que se comentarán las técnicas de clasificación supervisada que se utilizan en este TFG, el análisis discriminante lineal y los árboles de decisión. Adicionalmente, ha sido necesario considerar una técnica multivariante de reducción de dimensión, el análisis en componentes principales, y otra de comparación multivariante de medias, el análisis de la varianza, que también se describen en las secciones correspondientes.

4.1. Introducción al problema de clasificación

Todos los seres vivos requerimos de adquirir información del entorno para tomar decisiones que favorezcan nuestra supervivencia. Es más, el ser humano, como primate mamífero dotado de inteligencia, ha evolucionado gracias a esta capacidad de adquirir conocimiento. La clasificación es una operación elemental para comprender la realidad, y para tomar decisiones, como, por ejemplo, decidir, en función de una serie de variables si a un individuo se le da o no una hipoteca. Es por eso que se han desarrollado una serie de técnicas para poder clasificar individuos y objetos y poder lograr una comprensión más profunda de los fenómenos que nos rodean.

Durante las próximas secciones se expondrán las bases teóricas de los métodos que se van a emplear para abordar el problema de la clasificación.

4.1.1. Análisis discriminante lineal

En el contexto del análisis discriminante, se establece una asociación entre una variable dependiente (respuesta) con un conjunto de variables discriminantes o independientes.

La cuestión fundamental radica en generar una serie de funciones lineales utilizando las variables independientes que permitan clasificar nuevos datos en alguna de los grupos definidos por la variable dependiente e interpretar las diferencias entre los mismos (véase [13]).

El análisis discriminante busca alcanzar dos objetivos principales:

- En primer lugar, la discriminación, que se refiere a la evaluación de hasta qué punto un conjunto de características observadas en los individuos permite extraer dimensiones que distinguen a los grupos, y cuáles de estas características exhiben la mayor capacidad de discriminación.
- En segundo lugar, el análisis discriminante busca la clasificación, que implica utilizar el clasificador (una combinación de las variables de entrada) para predecir la clase de una nueva observación sin etiqueta, utilizando como base un conjunto de muestras previas. Tanto la discriminación como la clasificación, se consideran técnicas de aprendizaje supervisado en el ámbito del aprendizaje automático.

Podemos tomar como ejemplo un caso médico en el que un paciente llega a la sala de emergencias con dolores de estómago y síntomas que indican tanto intoxicación alimentaria como apendicitis. En este caso, es importante determinar cuál de las dos enfermedades es más probable para poder tratar al paciente adecuadamente. Si se diagnostica incorrectamente, se podrían aplicar tratamientos no apropiados que podrían tener consecuencias graves, incluso mortales. Para ayudar en este proceso de diagnóstico, se puede utilizar un clasificador construido a partir de la experiencia previa de otros pacientes.

Sin embargo, para poder aplicar el análisis discriminante lineal (LDA) tenemos que verificar una serie de hipótesis. Estos supuestos representan condiciones que garantizan la existencia de una relación entre la situación real y la predicción. Los supuestos son los siguientes:

- Normalidad multivariante. Cada grupo representa una muestra aleatoria extraída de una población con distribución normal multivariante sobre las variables discriminantes. En particular, todas las variables discriminantes deben ser normales univariantes. Para comprobar la normalidad univariante utilizaremos una prueba gráfica basada en la distancia de Mahalanobis y en el test de Shapiro-Wilks.
- La homogeneidad en las matrices de covarianzas de los distintos grupos. Entre las diferentes pruebas a las que podemos recurrir para comprobar la hipótesis de varianzas iguales en los grupos se encuentra la prueba del test de Levene.
- Ausencia de multicolinealidad ya que esta implica que varias variables están asociadas linealmente de forma aproximada y por tanto existe información redundante. Es decir una variable discriminante no puede ser combinación lineal de otras consideradas en el análisis. Se puede detectar multicolinealidad por pares de variables a partir de la matriz de correlaciones y de los gráficos de dispersión por pares de variables. También pueden realizarse mapas de calor que permitan ver las asociaciones entre las variables.

De las n variables independientes del dataset, LDA extrae $p \leq n$ nuevas variables independientes que separan lo mejor posible las clases de la variable dependiente. Es decir, encuentra las direcciones de máxima separación de clases.

En nuestro caso, al estar trabajando con un conjunto de datos de dimensión moderada, este algoritmo es muy útil ya que descubrirá una proyección lineal que le permitirá

optimizar el espacio de datos en un subespacio de menor dimensión que maximiza la separación de clases. El algoritmo LDA se puede entender de manera general con el siguiente ejemplo que puede verse gráficamente en la figura 4.1: digamos que tenemos una fábrica de ruedas y tenemos dos eventos: la rueda aprobó el control de calidad (verde) y la rueda no aprobó el control de calidad (rojo) basándonos únicamente en la curvatura de la rueda.



Figura 4.1: Clasificación. Con una sola característica se produce superposición. Fuente: [31].

Se observa que la utilización de una única característica provoca la existencia de solapamientos entre diferentes categorías o clases, lo cual dificulta la toma de decisiones o la obtención de una perspectiva clara de la situación. Es necesario determinar el significado de esta superposición para poder abordar adecuadamente la problemática. La superposición significa que hay valores similares de la característica que se están superponiendo entre las diferentes categorías o clases. Este gráfico 4.1 nos da información de un falso negativo, que sería una rueda buena que no pasó el control de calidad (círculo verde), y dos falsos positivos, unas ruedas malas que pasaron el control de calidad. Para evitar eso, podemos agregar otra característica, por ejemplo, el diámetro de la rueda, de esta manera, tenemos un diagrama de dispersión bidimensional, figura 4.2, que ayudará a visualizar el algoritmo LDA.

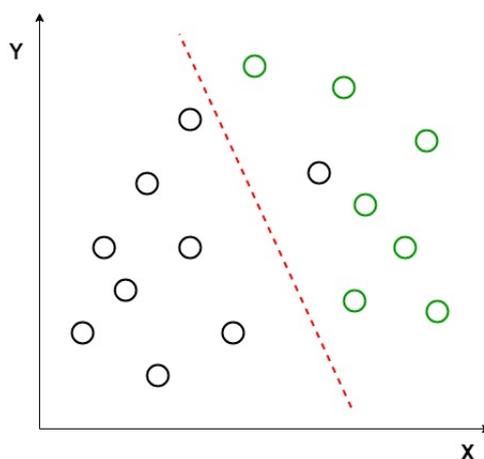


Figura 4.2: Diagrama de dispersión bidimensional. Fuente: [31].

A partir del diagrama de dispersión que aparece en la figura 4.2, resulta sumamente complicado trazar una línea recta que logre separar eficientemente los diferentes grupos. Sin embargo, es posible aplicar el análisis discriminante lineal (LDA) para reducir el plano bidimensional a uno unidimensional y así mejorar la capacidad de discriminación entre los distintos grupos.

El algoritmo LDA tiene como objetivo crear un nuevo eje que maximice la distancia entre las medias de los dos grupos y minimice la varianza dentro de cada grupo. Por último, proyectará todos los datos en este nuevo eje. En el caso de no haber errores de clasificación puede mostrar como resultado un gráfico como el que aparece en la figura 4.3.



Figura 4.3: Proyecciones de los puntos con clasificación perfecta. Fuente: [31].

De esta forma el algoritmo nos da mejor información sobre las ruedas en base a las dos características pero mostradas como una sola, dándonos una clasificación que nos ayudará en nuestra toma de decisiones sobre nuestro sistema de control de calidad de ruedas.

Entonces, en pocas palabras, a menudo el objetivo de un LDA es proyectar un espacio de características (un conjunto de datos de muestras n -dimensionales) en un subespacio más pequeño k (dónde $k \leq n - 1$) manteniendo la información discriminativa de clase. En este sentido, el algoritmo LDA puede ser una herramienta muy útil en el análisis y clasificación de datos multivariantes. Además, la reducción de la dimensionalidad también ayuda a evitar el sobreajuste que se produce cuando el modelo se ajusta demasiado bien a los datos de entrenamiento conduciendo a una predicción deficiente en los datos de prueba. La “maldición de la dimensionalidad” se refiere al hecho de que a medida que aumenta el número de variables o características en un conjunto de datos, el número de combinaciones posibles también aumenta exponencialmente. Esto puede hacer que sea muy difícil para el modelo encontrar patrones y relaciones significativas en los datos. Al reducir la dimensionalidad, se puede minimizar este problema y permitir que el modelo se enfoque en las características más importantes para la clasificación.

En resumen, de las variables independientes del conjunto de datos, LDA extrae nuevas variables independientes, combinaciones lineales de las originales, que separan lo mejor posible las clases de la variable dependiente.

Desde el punto de vista algorítmico estos son los pasos generales para realizar un análisis discriminante lineal:

1. Aplicar escalado de variables a la matriz de características X , compuesta por n variables independientes.
2. Sea C el número de clases. Se calculan C vectores n -dimensionales, de modo que cada uno contenga las medias de las características de las observaciones para cada clase. Si suponemos que $C = 2$ y llamamos 0 y 1 a las dos clases y x_j^i a la característica

j-ésima de la observación i-ésima, entonces

$$\mu_0 = \frac{1}{n_0} \left(\sum_{\substack{i=1, \dots, n \\ y^i \in \text{clase 0}}} x_1^i, \dots, \sum_{\substack{i=1, \dots, n \\ y^i \in \text{clase 0}}} x_n^i \right)$$

$$\mu_1 = \frac{1}{n_1} \left(\sum_{\substack{i=1, \dots, n \\ y^i \in \text{clase 1}}} x_1^i, \dots, \sum_{\substack{i=1, \dots, n \\ y^i \in \text{clase 1}}} x_n^i \right)$$

3. Se calculan las matrices de productos cruzados centrados en la media de cada clase, que mide la varianza dentro de cada clase. Con nuestro ejemplo de las clases 0 y 1, las dos matrices de productos cruzados S_0 Y S_1 para las respectivas clases 0 y 1 son:

$$S_0 = \sum_{\substack{i=1, \dots, n \\ y^i \in \text{clase 0}}} ((x_1^i, \dots, x_n^i) - \mu_0)((x_1^i, \dots, x_n^i) - \mu_0)^T$$

$$S_1 = \sum_{\substack{i=1, \dots, n \\ y^i \in \text{clase 1}}} ((x_1^i, \dots, x_n^i) - \mu_1)((x_1^i, \dots, x_n^i) - \mu_1)^T$$

4. Se calcula la covarianza normalizada de todas de todas las matrices anteriores, W . Con nuestro ejemplo de las clases 0 y 1, la covarianza normalizada W es simplemente

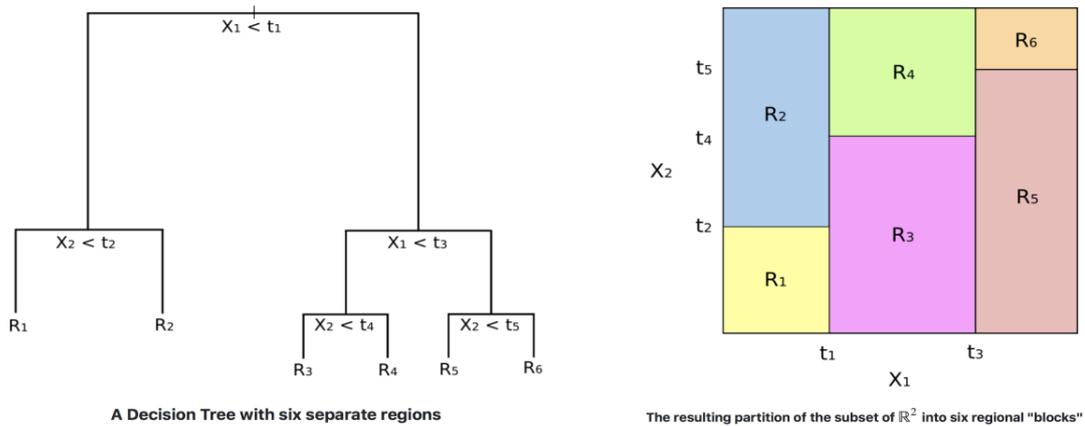
$$W = \frac{1}{n_0} S_0 + \frac{1}{n_1} S_1.$$

5. Calculamos la matriz de covarianza global entre clases, B . En nuestro ejemplo de las clases 0 y 1, la matriz de covarianza global entre las clases B es

$$B = n_0(\mu_0 - \mu)(\mu_0 - \mu)^T + n_1(\mu_1 - \mu)(\mu_1 - \mu)^T$$

donde $\mu = \frac{1}{n_0+n_1}(n_0\mu_0 + n_1\mu_1)$.

6. Se calculan los valores y vectores propios de la matriz $W^{-1}B$.
7. Se eligen los p valores propios más grandes como el número de dimensiones reducidas.
8. Los p vectores propios asociados a los p valores propios más grandes son los discriminantes lineales. El espacio n -dimensional del dataset original se proyecta al nuevo sub-espacio p -dimensional de características aplicando la matriz de proyecciones (que tiene los p vectores propios de las columnas).



$$RSS = \sum_{m=1}^M \sum_{i \in R_m} (y_i - \hat{y}_{R_m})^2$$

Figura 4.4: Árbol de decisión y valor del error de clasificación total. Fuente: [28]

4.1.2. Árboles de decisión

Un árbol de decisión es un modelo de predicción que tiene como propósito fundamental el proceso de aprendizaje inductivo a partir de observaciones y la creación de estructuras lógicas proporcionando un aporte visual a la toma de decisiones con un esquema que permite visualizar las soluciones de cada alternativa. Como puede verse en la figura 4.4, se representa gráficamente por un conjunto de nodos, hojas y ramas. En un árbol de decisión, el nodo principal o raíz se define como el atributo que da inicio al proceso de clasificación, mientras que los nodos internos se refieren a cada una de las preguntas relacionadas con el atributo específico del problema en cuestión. Los nodos hijos se encargan de representar cada posible respuesta a las preguntas planteadas, mientras que las ramas que se derivan de ellos se etiquetan con los distintos valores que puede tomar el atributo en cuestión. Los nodos hoja, a su vez, simbolizan una decisión final y representan una de las variables que se desean clasificar o resolver en el problema también conocida como la variable objetivo o variable dependiente.

El procedimiento para generar árboles de decisión se realiza de la siguiente forma. Primero en una fase de inducción se genera un árbol de decisión a partir del conjunto de datos de entrenamiento de manera que cada nodo interno del árbol está compuesto por un atributo de prueba y se divide la porción del conjunto de entrenamiento correspondiente a ese nodo de acuerdo con los valores que pueda tomar dicho atributo. Es decir, el algoritmo elige la variable más predictiva en la que se deben dividir los datos siguiendo un enfoque de división binaria recursiva. La "predictibilidad" se basa en la disminución de la impureza de los nodos en cada división. Un nodo en el árbol se considera "puro" si, en el 100 % de los casos, los nodos caen en una categoría específica del campo objetivo. En consecuencia, un

nodo menos impuro requiere menos información para ser descrito mientras que un nodo más impuro necesita más información. Esta impureza de los nodos se calcula mediante la entropía de los datos en el nodo. La entropía es la cantidad de desorden de la información, o la cantidad de aleatoriedad en los datos. En los árboles de decisión, buscamos a los árboles que tengan la entropía más pequeña en sus nodos. Una propiedad relevante de este algoritmo es el criterio de partición debido a que una elección inadecuada de la partición, especialmente en las secciones superiores del árbol, producirá un árbol de menor calidad. La idea básica de este criterio de partición es la siguiente. Por ejemplo, en la figura 4.5 queremos crear un modelo para predecir los estudiantes que jugarán al cricket. Tenemos una muestra de 30 estudiantes basada en todos los valores de las variables Género, Altura y Clase y sabemos que 15 juegan al cricket en su tiempo libre. Para tal fin tendremos que dividir los elementos en grupos de acuerdo a cada variable y así poder determinar cuál es la que tiene mayor poder de segregación en el grupo y mayor disparidad entre grupos. Las medidas de selección de atributos más utilizadas y que seleccionaremos como parámetros predeterminados en la búsqueda de hiperparámetros de nuestros análisis son el Índice de Gini y la Entropía.

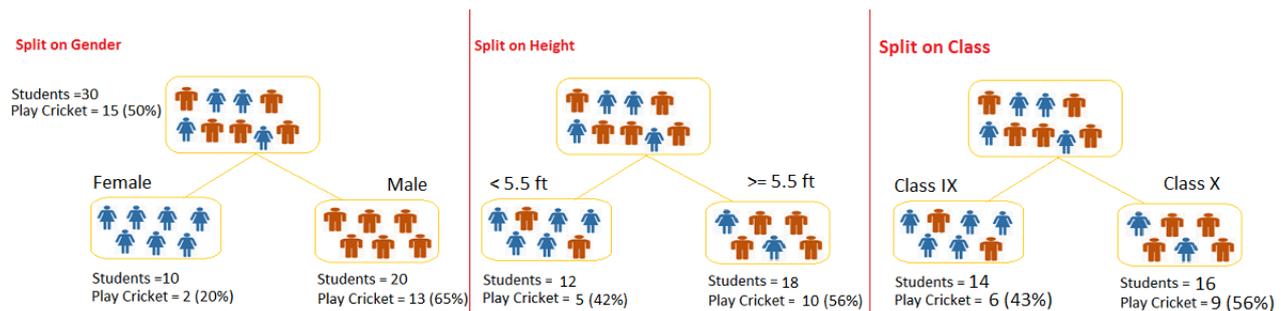


Figura 4.5: Ejemplo de construcción de árbol. Fuente: [28]

Una vez que hemos elegido dicho atributo como nodo decisión estamos segmentando el conjunto total de elementos en subconjuntos más pequeños por lo que el siguiente paso se hará de forma recursiva para cada nodo hijo. Durante la última fase el árbol de inducción clasifica cada nodo interno desde el raíz hasta las hojas e identificando la clase a la que pertenece de acuerdo a las elecciones realizadas.

4.2. Construcción y evaluación de modelos

Antes de llevar a cabo los análisis, es necesario tomar decisiones sobre los métodos que se utilizarán para llevar a cabo una selección apropiada de la división del conjunto de datos, los hiperparámetros correspondientes y las métricas de error para posteriormente en el análisis evaluar el rendimiento de los modelos.

4.2.1. Medida del error

A continuación definiremos unas métricas que nos ayudarán a evaluar el rendimiento de nuestros modelos. Nuestras métricas se construirán sobre la matriz de confusión, una representación en forma de matriz que confronta cada predicción con la clase real de cada instancia.

Como se muestra en la figura 4.6, esta matriz está compuesta por dos etiquetas de clase. La fila Positive son aquellos elementos que han sido clasificados como positivos, y la fila Negative los que cuya predicción en la clasificación ha sido negativa. Con respecto a las columnas, la Positive representa las instancias que en realidad son positivas, y la columna Negative las que son negativas.

		<u>Ground Truth</u>	
		Positive	<u>Negative</u>
<u>Prediction</u>	Positive	True positives	False positives
	<u>Negative</u>	False negatives	<u>True negatives</u>

Figura 4.6: Matriz de Confusión. Fuente: [4]

De esta forma, todas las observaciones deben pertenecer a uno de los grupos:

- Verdadero Positivo (TP): cantidad de casos cuya predicción es Verdadera y son Verdaderos en la clase real.
- Verdadero Negativo (TN): cantidad de casos cuya predicción es Falsa y son Falsos en la clase real.
- Falso Positivo (FP): cantidad de casos cuya predicción es Verdadera y son Falsos en la clase real.
- Falso Negativo (FN): cantidad de casos cuya predicción es Falso y son Verdaderos en la clase real. Es por ello que cuantos más Falsos positivos y Falsos negativos haya peor será la predicción de nuestro modelo.

Las métricas que consideraremos en nuestra clasificación serán:

- **Accuracy (*Exactitud*)**: mide el porcentaje de casos que el modelo ha acertado.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.1)$$

- **Error de clasificación**:

$$\text{Error} = 1 - \text{Accuracy} = \frac{FP + FN}{TP + TN + FP + FN} \quad (4.2)$$

- **Precision (*Precisión*)**: nos informa de la calidad de la predicción es decir, qué porcentaje de elementos que se han clasificado como positivos son realmente positivos.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (4.3)$$

- **Recall (*ratio de verdaderos positivos*)**: nos informa de la cantidad, es decir, cuantos valores positivos son correctamente clasificados.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4.4)$$

- **F1**: Combina el precision y el recall, y es útil cuando el conjunto de datos a analizar está desbalanceado

$$F1 = 2 \cdot \frac{\text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}} \quad (4.5)$$

De todas estas métricas, la fundamental es el error de clasificación (4.2) por lo que le dedicaremos especial atención a los resultados obtenidos con ella.

4.2.2. División del conjunto de datos

Los procedimientos de remuestreo se fundamentan en la extracción repetida de muestras de un conjunto de datos de entrenamiento para ajustar el modelo a cada muestra.

Cuando la cantidad de datos es limitada, la validación cruzada es uno de los métodos más empleados para estimar el error de prueba asociado y para evaluar el rendimiento del modelo, utilizando diversos fragmentos del conjunto de datos como conjuntos de validación. Este proceso permite aprovechar la mayor cantidad de registros para el entrenamiento, lo que se supone incrementa la precisión del clasificador.

La técnica de validación cruzada de k iteraciones o k -fold cross validation, implica la partición de los datos originales en k subconjuntos. Durante el proceso de entrenamiento, cada subconjunto es utilizado una vez como conjunto de prueba, mientras que los $k - 1$ restantes se emplean como conjunto de entrenamiento (véase la figura 4.7). Este procedimiento se repite k veces, con un conjunto de prueba diferente en cada iteración, utilizando el resto de los datos como conjunto de entrenamiento. Cuando se han realizado

todas las iteraciones, en nuestro caso haremos cinco, se realiza la media de los resultados de precisión y error obtenidos para cada conjunto de prueba.

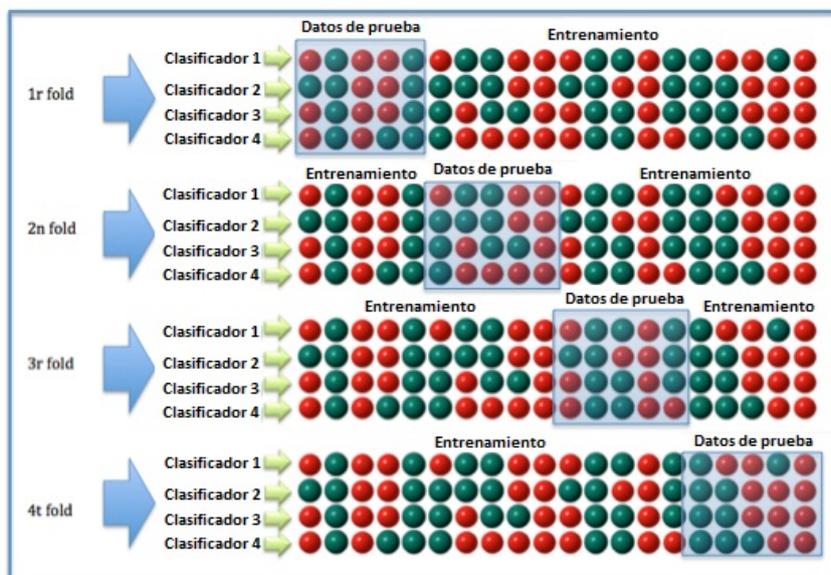


Figura 4.7: Validación cruzada de K iteraciones con K=4. Fuente: [36]

Es claro que la técnica de validación cruzada puede lograr una precisión muy alta, pero el coste computacional es notable dado que el tamaño de los datos es mayor.

4.2.3. Búsqueda de hiperparámetros

Los hiperparámetros son variables específicas para cada modelo que se definen previamente y que no se aprenden en el proceso de entrenamiento. Necesitamos la combinación más óptima de estos parámetros así que experimentaremos entrenando el modelo secuencialmente con diferentes conjuntos de ellos para ver cuáles nos permiten alcanzar el mejor rendimiento. Utilizaremos la técnica GridSearch, que es una búsqueda exhaustiva, con el fin de encontrar los valores de hiperparámetros óptimos para una lista concreta de ellos como, por ejemplo, la tasa de aprendizaje, que es la velocidad a la que el modelo aprende de los datos, o la profundidad de un árbol que determina la máxima cantidad de divisiones, que en nuestro caso fijaremos en 3 para encontrar un balance entre la insuficiente adaptación del modelo a los datos y el exceso de adaptación.

4.3. Análisis de componentes principales

La selección de un adecuado número de variables en relación con la cantidad de observaciones es fundamental en el análisis de datos. Teniendo en cuenta que se recomiendan aproximadamente 10 observaciones por variable, en nuestro estudio, como se comentará con más detalle en el capítulo 5.1, nos encontramos con un escenario en el que la cantidad

total de observaciones es limitada en comparación con la cantidad de variables presentes. Específicamente, contamos con 11 variables y no disponemos del número ideal de 110 observaciones que se recomendaría en este caso [30]. La discrepancia existente entre la cantidad de observaciones y variables puede suponer un reto significativo debido a que la inclusión de las 11 variables en el modelo puede dar lugar a una condición de sobreajuste, con lo que se obtendría un modelo demasiado ajustado a los datos disponibles pero que podría no funcionar bien para otras muestras posibles.

Por ello, hemos considerado la conveniencia de implementar el análisis de componentes principales (ACP) como solución eficaz para identificar las combinaciones lineales de las variables originales (llamadas componentes principales) que logren capturar la mayor parte de la variabilidad del conjunto de datos, ya que esta variabilidad se asocia a la cantidad de información recogida por dicha combinación lineal. De esta manera, se conserva la mayor parte posible de la información dada por el conjunto total de variables explicativas en un número menor de variables, disminuyendo de ese modo la posibilidad de sobreajuste.

El ACP actúa transformando los ejes del sistema de coordenadas de las variables hacia una nueva base de coordenadas ortogonales cuyos ejes principales coinciden con las direcciones de máxima varianza de los datos. Para determinar dicha rotación primero debemos hacer que las observaciones tengan media 0 centrándolas en el centro del eje de coordenadas.

A continuación, se calcula la recta que se adapta mejor a los datos que han sido centralizados y escalados, minimizando así el error residual. Esta recta define lo que identificaremos como la primera componente principal (Z_1). Una vez se ha calculado la primera componente principal, se procede a calcular la segunda componente principal (Z_2) de manera que también pase por el origen, sea ortogonal a la primera y recoja de la mayor cantidad posible de la información no recogida por la anterior componente principal, y así sucesivamente hasta completar el resto de componentes. La figura 4.8 muestra la construcción de las dos primeras componentes principales.

Cada componente principal generada por ACP está asociado a un eigenvector (autovector) y a un eigenvalue (autovalor) específicos de la matriz de datos originales estandarizados. Una vez obtenidos los autovalores y autovectores, se calcula la Proporción de Varianza Explicada Acumulada (PVEA) para cada componente principal dividiendo la suma acumulada de los autovalores por la suma total de los autovalores. A continuación graficamos la PVEA para ayudar a determinar el número óptimo de componentes principales y, finalmente, se transforman los datos originales en las nuevas coordenadas del espacio de los componentes principales. Resulta relevante recalcar que la varianza del componente Z_i está dada por el autovalor λ_i

$$Var(Z_i) = \lambda_i$$

Por lo tanto, esta medida cuantifica la contribución de cada componente principal determinando la cantidad total de información que se mantiene en el conjunto de datos o, lo que es lo mismo, cuánta información se pierde al proyectar los datos sobre las primeras componentes principales.

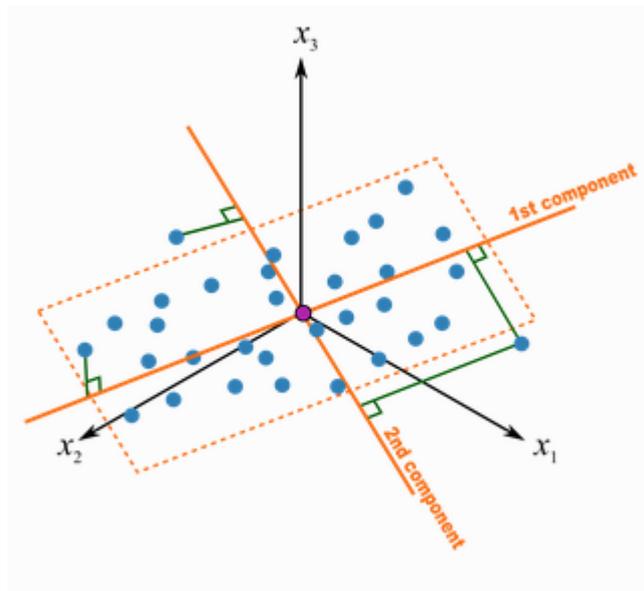


Figura 4.8: Determinación del plano formado por las dos primeras componentes principales. Fuente: [14]

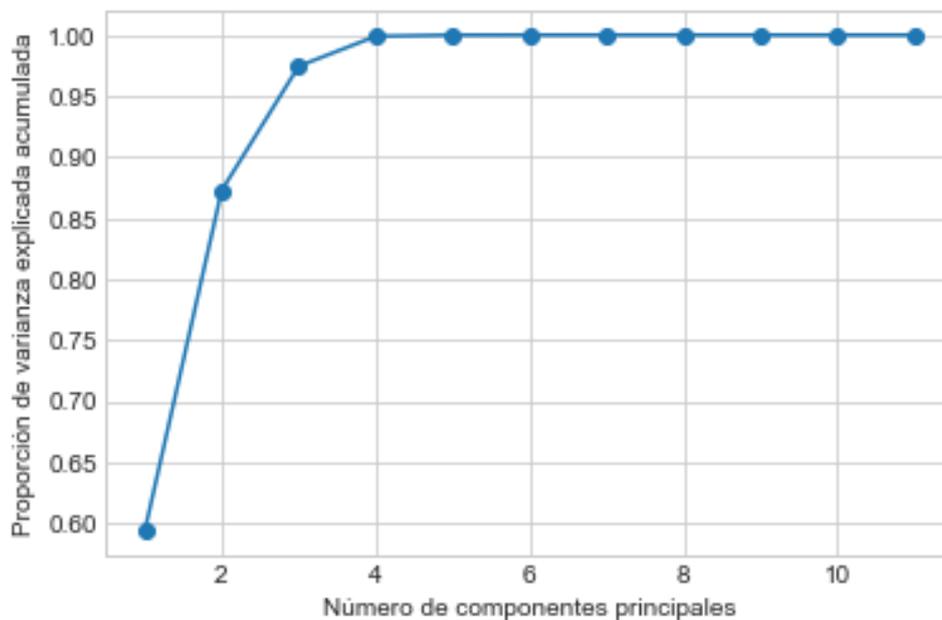


Figura 4.9: Gráfico de Proporción de Varianza Explicada Acumulada (PVEA) mediante ACP de Corriente en Carga Alta. Análisis 1 contra 2

Para determinar el número de componentes a extraer es habitual utilizar un gráfico como el que aparece en la figura 4.9. En el eje x del gráfico de la figura se encuentran los componentes principales, numerados de manera ascendente. Por otro lado, en el eje y se representa la proporción de varianza explicada acumulada, expresada como un porcentaje. En este caso, se observa que la primera componente principal explica una proporción muy alta de la variabilidad, superior al 95 %. Podemos observar que a partir de la tercera componente, la adición de más componentes principales proporciona poca ganancia adicional en la varianza explicada.

4.4. Análisis de la varianza

La técnica denominada análisis de la varianza (ANOVA), concebida por Sir Ronald Fisher en 1930, es una prueba utilizada en Estadística que permite contrastar si existe una diferencia estadísticamente significativa entre las medias dos o más poblaciones o, de forma equivalente, para estudiar el efecto que tiene uno o más factores sobre la media de una variable continua en diferentes niveles de los factores. Aunque en el trabajo se realizarán procedimientos ANOVA con varios factores se describe aquí fundamentalmente el ANOVA de un solo factor.

La forma en que se lleva a cabo el estudio es un contraste de hipótesis. Específicamente, si denominamos μ_i a la media de la población i (o a la media del nivel i del factor), las hipótesis que se contrastan sobre las medias en un ANOVA de un factor son:

$$\begin{cases} H_0 : \forall i, j \mu_i = \mu_j \\ H_1 : \exists i, j \mu_i \neq \mu_j \end{cases}$$

Para tomar esta decisión se efectúa lo que se denomina partición de la variabilidad. La variabilidad completa de los datos con respecto a la media global se particiona en la variabilidad de los datos de cada grupo (población) con respecto a la media del grupo y la variabilidad de las medias de cada grupo con respecto a la media global. La primera variabilidad se utiliza para estimar el error aleatorio y se compara con la segunda. Si la variabilidad de las medias de las poblaciones entre sí (numerador) es suficientemente grande frente al error aleatorio (denominador) se descartará la hipótesis de igualdad de medias y se concluirá que hay diferencias entre ellas. De manera formal el estadístico de contraste se escribe como:

$$F_{ratio} = \frac{n \frac{\sum_{j=1}^k (\bar{y}_j - \bar{y})^2}{k-1}}{\frac{\sum_{j=1}^k \sum_{i=1}^n (y_{ij} - \bar{y}_j)^2}{N-k}}$$

En caso de rechazarse la hipótesis es habitual utilizar pruebas post-hoc para determinar entre qué niveles del factor existen esas diferencias. Existen múltiples pruebas post-hoc. En este trabajo se utilizará la prueba HSD de Tukey [35] por ser la que aparece en las implementaciones del ANOVA en Python que se han utilizado en este trabajo.

Hay que señalar también que se pueden realizar análisis de la varianza con más de un factor en los que, además de valorar los efectos principales de cada uno de dichos factores,

pueden evaluarse las posibles interacciones entre ellos para determinar si los efectos de cada factor dependen de los niveles a los que se encuentran los otros.

Capítulo 5

Datos

5.1. Descripción del conjunto de datos

Los datos de este proyecto proceden de un experimento realizado por el departamento de Ingeniería Eléctrica de la Escuela de Ingenierías Industriales de la Universidad de Valladolid. En el experimento se recogen observaciones del funcionamiento de un motor de inducción con fallos en rodamientos debido a la introducción de carburo de silicio en los mismos de forma controlada y progresiva. Ello permite la clasificación del estado del motor dependiendo del grado de deterioro en las siguientes clases:

- Motor sano (1).
- Fallo incipiente, motor levemente dañado (2).
- Motor ligeramente dañado (3).
- Motor medianamente dañado (4).
- Motor severamente dañado (5).
- Fallo catastrófico, motor gravemente dañado (6).

Los datos se tomaron a dos niveles de carga del motor (alta y baja) lo que permite estudiar la influencia del nivel de carga en los resultados. Todos los datos se recogieron en estado estacionario de funcionamiento del motor (es decir, la señal se encuentra en un estado estabilizado) y se tomaron medidas de tres fuentes de información: corriente, sonido y vibración (véase sección 2.4) mediante las que se pretende diagnosticar el estado del motor. Las mediciones que se consideran en este trabajo están tomadas con un motor alimentado directamente de la red, es decir, que no se considera el uso de un variador de frecuencia conectado a la fuente de alimentación eléctrica que regule la velocidad de giro del motor.

Las observaciones tomadas están inicialmente balanceadas en torno a todas las 6 clases del estado de motor, los niveles de carga y la fuente de información, lo que hace que la

capacidad predictiva de los procedimientos sea más estable, ya que si un conjunto de datos tiene un importante desequilibrio entre las clases, los algoritmos tienden a favorecer la clase con mayor proporción de observaciones pudiendo obtener un modelo que no predice de forma correcta la clase minoritaria.

Para ser más precisos, en este proyecto se consideran un total de 120 observaciones, 60 en carga alta y 60 en carga baja. En cada uno de los estados de carga se dispone de 10 observaciones para cada uno de los 6 estados de deterioro del motor.

El estado del motor es la variable a predecir en nuestro problema. En cuanto a las variables explicativas o predictores, como ya se ha mencionado, para cada observación se han recogido datos de las vibraciones generadas por el motor en cada uno de los 3 ejes de coordenadas (x, y, z), el sonido del motor y la corriente en las tres fases $f1$, $f2$ y $f3$. Tras la recogida y procesado inicial de la información, para cada una de estas tres fuentes de información, ejes y fases, se calculan los estadísticos de orden superior que van a utilizarse para el diagnóstico definidos en [29, 15] y ya considerados en trabajos anteriores como [2], con lo que dispone inicialmente de datos de 11 variables descritas en la tabla 5.1, lo que da un total de 33 variables para vibraciones, 11 para sonido y 33 para corriente. De acuerdo a la opinión de los expertos del departamento de Ingeniería Eléctrica, se han promediado los valores de las tres fases en el caso de la corriente ya que las fases no aportan información diferencial entre ellas, lo que permite, para esta fuente de información, reducir el número de predictores a 11. Sin embargo, como parece lógico no ha podido hacerse una reducción similar para el caso de las vibraciones ya que los distintos ejes sí pueden proporcionar información diferente.

Nombre	Fórmula	Descripción
c_1, m_1	$\frac{1}{n} \sum_{i=1}^n x_i$	Cumulante de primer orden (media)
c_2	$m_2 - m_1^2$	Cumulante de segundo orden
c_3	$m_3 - 3m_2m_1 + 2m_1^3$	Cumulante de tercer orden
c_4	$m_4 - 4m_3m_1 - 3m_2^2 + 12m_2m_1^2 - 6m_1^4$	Cumulante de cuarto orden
m_2	$\frac{1}{n} \sum_{i=1}^n x_i^2$	Momento de segundo orden (varianza)
m_3	$\frac{1}{n} \sum_{i=1}^n x_i^3$	Momento de tercer orden
m_4	$\frac{1}{n} \sum_{i=1}^n x_i^4$	Momento de cuarto orden
$skew$	$\frac{c_3}{\sqrt{c_2^3}}$	Coefficiente de asimetría
$kurt$	$\frac{c_4 + 3c_2^2}{\sqrt{c_2^4}}$	Coefficiente de apuntamiento
x_p	$max x_i $	Máximo valor absoluto
x_r	$(\frac{1}{n} \sum_{i=1}^n \sqrt{ x_i })^2$	Valor cuadrático medio

Cuadro 5.1: Variables predictoras

5.2. Filtrado de los datos

Resulta fundamental llevar a cabo una exploración preliminar de los datos, enfocándonos especialmente en los casos extremos (outliers) debido a su potencial impacto sobre los resultados del análisis. El efecto de estos datos puede tener consecuencias negativas, no solo debido a los sesgos “ocultos”, sino también porque su eliminación debe evaluarse cuidadosamente en términos de su impacto en el tamaño de la muestra disponible para el análisis. Esta pérdida de información que genera su eliminación reduce el tamaño muestral, pudiendo comprometer la precisión de las estimaciones o reducir la potencia del estudio.

Con el propósito de lograr identificar estos outliers, se han llevado a cabo verificaciones mediante el uso de varias técnicas, examinando las observaciones desde distintas perspectivas, ya sea univariante, bivariante o multivariante [30].

En una perspectiva univariante, se examina una sola variable a la vez realizando un análisis de la distribución de observaciones de esa variable y señalando como datos atípicos los que se encuentran fuera de los intervalos definidos por la distribución. Con este propósito, se han utilizado los diagramas de Caja y Bigotes. La caja representa el rango intercuartílico, es decir, el rango de valores que se encuentran entre el percentil 25 y el percentil 75 de la distribución. Es importante notar si la caja es simétrica o asimétrica, lo que puede indicar una distribución normal o no normal de los datos. Para ello nos fijaremos en si la mediana se sitúa en el centro de la caja, entonces la distribución es simétrica. Los bigotes se extienden desde la caja hasta el valor mínimo y máximo. Dicha longitud no debe ser superior al 150 % del rango intercuartílico. Los puntos fuera de los bigotes se consideran posibles outliers y se representarán como puntos individuales. Estos diagramas de caja y bigotes se incluyen en el anexo A.1 permitiendo a los lectores examinar los datos de manera más detallada y evaluar la validez de los resultados presentados. El código Python utilizado para su obtención puede encontrarse en el anexo B.1.

En contraste con la perspectiva univariante, cuando se analizan datos en una perspectiva bivariante o multivariante, se lleva a cabo la exploración de dos o más variables simultáneamente. Para el caso bivariante, se utilizan gráficos de dispersión que representan cada observación bivariante como un punto en la gráfica y permiten detectar visualmente aquellas observaciones que se ubiquen claramente fuera de la nube de puntos. Con el propósito de hacer más comprensible la información, en lugar de ofrecer todos los gráficos bivariantes, en la tabla 5.2 se enumeran los posibles valores atípicos identificados en estos diagramas de dispersión.

En la evaluación multivariante, se usa una medida objetiva de la posición multidimensional de cada observación relativa a un punto común, tal como la llamada “distancia de Mahalanobis” [30], la cual registra la distancia de cada observación en un espacio multidimensional respecto del centro de las observaciones, utilizando como regla de medida la matriz de dispersión de dichas observaciones. El anexo B.2 contiene el código en Python utilizado para la obtención de los valores de la distancia de Mahalanobis para todas las observaciones consideradas en este trabajo. La tabla 5.3 indica las observaciones detectadas como posiblemente anómalas con esta metodología. Una de las ventajas de esta

Tipo	Carga	nº obs.	Variables “extrañas”	Valores
Red	CA	0	c1, c2	-0.007, 0.239
Red	CA	50	c1, c2	-0.005, 0.206
Red	CA	0	c4, m2	-0.084, 0.240
Red	CA	20	kurt, xp	1.542, 0.793
Red	CA	0	c3, m4	1.898e-05, 0.083
Red	CA	20	skew, kurt	-0.0003, 1.543
Red	CA	0	kurt, xp	1.528, 0.784
Red	CB	39	skew, kurt	0.002, 1.501
Red	CB	30	kurt, xp	1.510, 0.481
Sonido	CA	50	c3, m4	3.895e-08, 1.126e-07
Sonido	CA	50	c1, c2	0.298, 0.0001
Sonido	CA	46	skew, kurt	0.259, 12.963
Sonido	CA	46	kurt, xp	12.963, 0.401
Sonido	CB	57	c3, m4	-3.761e-07, 3.504e-07
Sonido	CB	58	c3, mc4	-3.928e-09, 3.944e-07
Sonido	CB	53	c4, m2	2.433e-09, 0.081
Sonido	CB	57	c4, m2	2.1308e-07, 0.080
Vibraciones	CA	50	c4_x, m2_x	-1.778e+17, 4.134e+08
Vibraciones	CA	50	m3_z, m4_z	-8.622e+09, 5.954e+16
Vibraciones	CB	57	c1_z, c2_z	-731.279, 9.995e+07
Vibraciones	CB	58	c1_z, c2_z	-755.374, 1.110e+07

Cuadro 5.2: Posibles puntos anómalos detectados mediante análisis bivariantes

metodología de detección de puntos anómalos es la posibilidad de efectuar contrastes de hipótesis para determinar si una observación concreta es anómala. Las observaciones que aparecen en la tabla 5.3 tienen un p-valor inferior a 0,001 con lo que hay una seguridad alta de que efectivamente se trata de outliers.

Dado el carácter global de esta metodología y la posibilidad de disponer de una valoración más fiable del riesgo que se corre al declarar un punto como anómalo, en los análisis que se efectuarán en este trabajo se mantendrán todas aquellas observaciones que no aparecen en la tabla 5.3.

Tipo	Carga	Eje (Sólo Vibraciones)	Nº Obs
Corriente	CA	#	20, 30
Sonido	CA	#	33, 36, 40, 41, 42, 43
Vibraciones	CA	x	18, 33, 40, 41, 42
Vibraciones	CA	y	33, 40, 41, 43
Vibraciones	CA	z	33, 40, 41, 42, 43
Corriente	CB	#	10, 12, 37
Sonido	CB	#	46, 47, 48, 49
Vibraciones	CB	x	15, 46, 48, 49
Vibraciones	CB	y	10, 41, 42, 45, 47, 48
Vibraciones	CB	z	43, 45, 47, 48

Cuadro 5.3: Puntos anómalos finales detectados mediante Distancia de Mahalanobis

5.3. Análisis exploratorio de los datos

Como hemos dicho anteriormente, con este análisis trataremos de contribuir a una descripción de las diferencias entre el estado sano del motor y el fallo incipiente, identificando las dimensiones en función de las cuales pueden establecerse esas diferencias y las variables que contribuyen en mayor medida a tales dimensiones. Además, pretendemos construir reglas que nos permitan diagnosticar el estado de nuevos motores. Los datos de partida de este estudio corresponden, como se describió en la sección 5.1, a motores que pueden tener un estado saludable o con fallos.

Uno de los métodos que va a utilizarse para construir las reglas de diagnóstico es el análisis discriminante lineal (LDA) descrito en la sección 4.1.1. Antes de realizar el análisis estadístico, debemos verificar las hipótesis de normalidad y homogeneidad de varianza. Esto según [30]) nos dará mayor confianza en que los métodos sean válidos y produzcan resultados confiables. Adicionalmente, verificaremos la posible existencia de multicolinealidad que puede influir también en la estabilidad de los resultados.

Comenzaremos con la primera suposición. La normalidad univariante contribuye, aunque no es condición suficiente, a la normalidad multivariante. Aquí realizaremos una comprobación de la normalidad de las variables consideradas separadamente, a fin de aportar evidencias que apoyen la posibilidad de una multinormalidad en las variables estudiadas.

Para comprobar el supuesto de normalidad hemos realizado la prueba de Shapiro-Wilks. El test de Shapiro-Wilks plantea la hipótesis nula que una muestra proviene de una distribución normal. Hemos elegido un nivel de significación $\alpha = 0,05$, y tenemos la hipótesis alternativa que sostiene que la distribución no es normal. Como los p-valores obtenidos son mayores que 0,05 no rechazamos la hipótesis nula afirmando que el comportamiento de las variables es acorde con la distribución gaussiana en las variables consideradas.

La valoración de estos resultados se ha hecho teniendo en cuenta que las separaciones ligeras del supuesto de multinormalidad no afectan excesivamente al análisis discriminante y no tienen consecuencias significativas en el análisis.

Para comprobar el supuesto de homogeneidad de varianzas se utilizó el test de Bartlett. La homocedasticidad se refiere a la suposición de que las variables dependientes tienen el mismo nivel de varianza en todo el rango de las variables predictoras, es decir, en cada grupo, según [18]. La homocedasticidad es deseable porque queremos que el término de error sea el mismo en todos los valores de las variables independientes. Si consideramos el nivel de significación $\alpha = 0,05$, en los casos en los que el valor p es superior a 0,05 se acepta nuestra hipótesis nula (los grupos tienen igual varianza) y podrían provenir de poblaciones similares. Sin embargo, p-valor menor que 0,05 indica que la evidencia en contra de la igualdad de varianzas es estadísticamente significativa. No se obtuvieron valores en estos tests que indicaran problemas con la homogeneidad de la varianza entre los grupos considerados.

Para valorar la posible existencia de multicolinealidad se construyen los diagramas de dispersión de las variables por pares e histogramas dispuestos en forma de matriz, como el que aparece en la figura 5.1, para el análisis de los datos de corriente en carga baja en el caso del grupo 1 contra el grupo 2+3+4+5+6, que han permitido observar la existencia de relaciones tanto lineales como no lineales entre varias parejas de variables explicativas.

Adicionalmente, con el objetivo de ver más claramente entre qué parejas de variables existe una correlación alta se construyen los denominados mapas de calor para las variables. Por ejemplo, la figura 5.2 es este mapa de calor para las parejas de variables para el análisis de los datos de corriente en carga baja en el caso del grupo 1 contra 2+3+4+5+6 mientras que, también como ejemplo, la tabla 5.4 acota los pares de variables con correlación mayor que 0,9 para el análisis de los datos de corriente en carga baja en el caso del grupo 1 contra el grupo 2+3+4+5+6.

Con estos resultados, se ha considerado la posible eliminación de una de las variables del par si su correlación es alta. No obstante, dado que se decidió utilizar un análisis en componentes principales previo en el caso del análisis discriminante lineal para evitar problemas de multicolinealidad en el modelo y que los árboles de decisión tienen su propio método de selección de variables, no fue necesaria la eliminación completa de ninguna de las variables originales.

Pares de Variables	Correlación
c2 - m2	0.999
c2 - m4	0.998
c3 - skew	0.980
m2 - m4	0.979

Cuadro 5.4: Parejas de variables con correlaciones superiores a 0,9 para las variables de corriente en carga baja del análisis 1 contra 2+3+4+5+6.

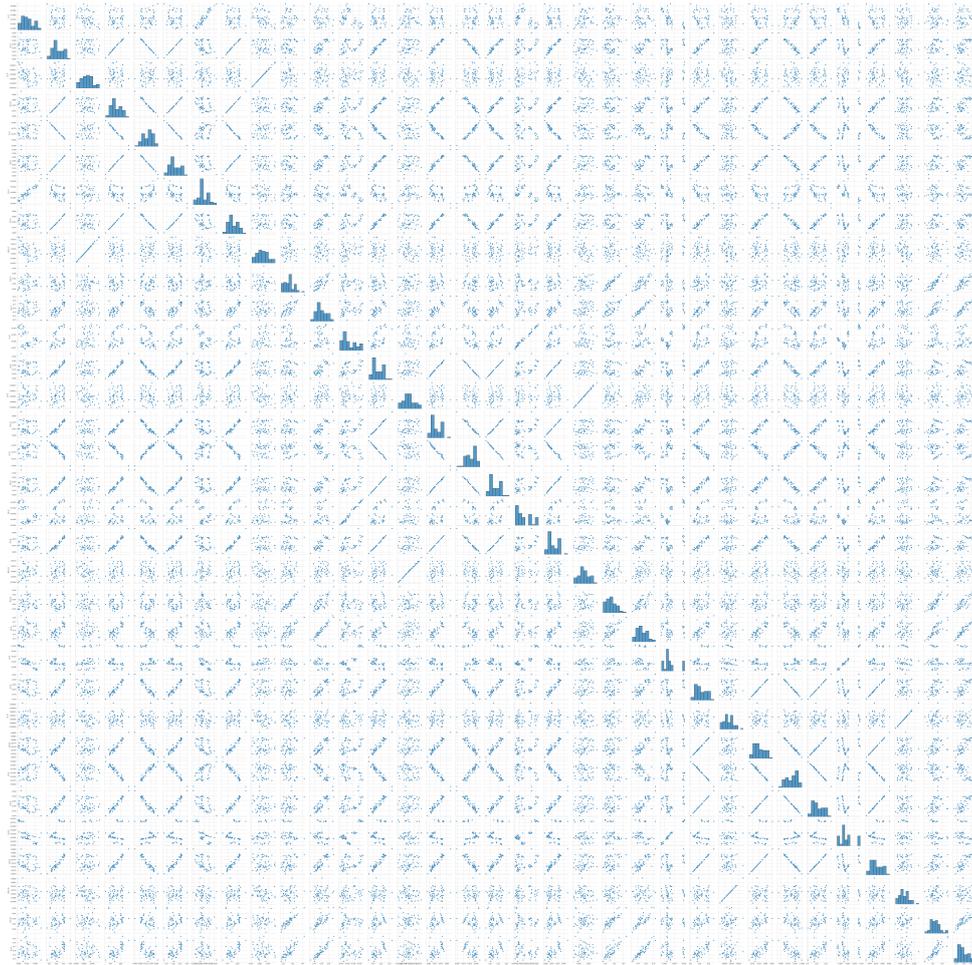


Figura 5.1: Matriz de dispersión de las variables preliminares evaluadas por pares de corriente en carga baja del análisis del grupo 1 contra el grupo 2+3+4+5+6.

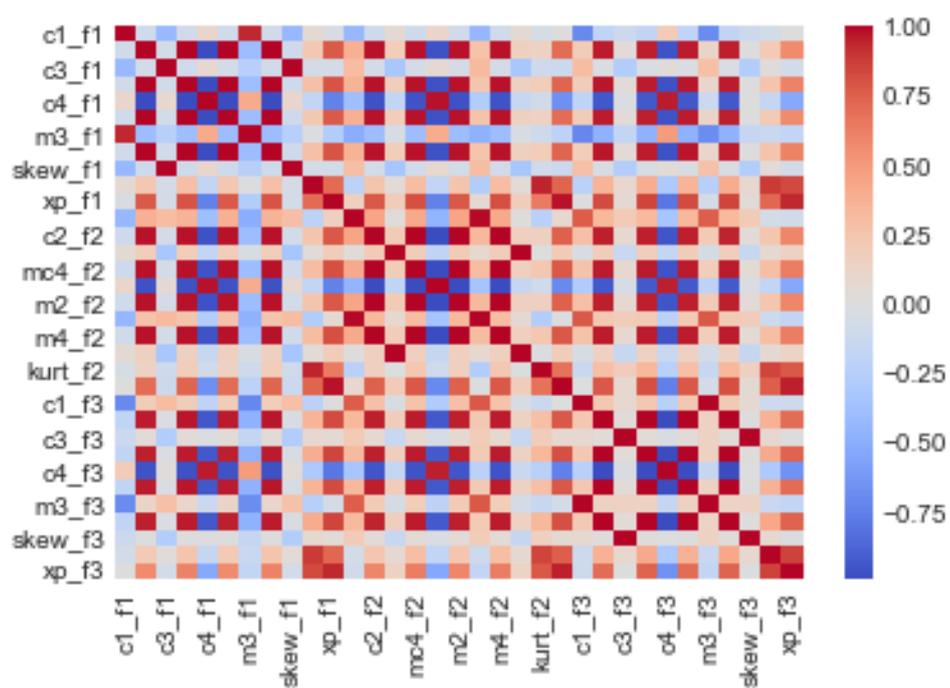


Figura 5.2: Mapa de calor de las variables preliminares de corriente en carga baja del análisis 1 contra 2+3+4+5+6 evaluadas por pares.

Capítulo 6

Análisis y resultados

Como se mencionó en la sección 1.2, entre los objetivos de este trabajo están:

- La detección del fallo incipiente y la detección de cualquier tipo de fallo por lo que fundamentalmente se abordará la clasificación entre los estados 1 y 2 del motor (los posibles estados del motor fueron descritos en la sección 5.1) y la clasificación del estado 1 contra el resto de los estados.
- El estudio de si técnicas estadísticas de aprendizaje supervisado como el análisis discriminante lineal (LDA) y los árboles de decisión (AD) son buenas para la detección de ese tipo de fallos.
- Determinar la influencia de la carga del motor en la calidad de la clasificación de los fallos.
- Conocer qué fuentes de información de entre las disponibles (corriente, sonido y vibraciones) proporcionan mayor capacidad de clasificación de los fallos.

Para poder lograr todos estos objetivos, se dividen los análisis en dos bloques separados para abordar las dos situaciones mencionadas en el primer punto. En el análisis 1 contra 2, que aparece en la sección 6.2, se abordará la detección de fallo incipiente, mientras que en la sección 6.3 se aborda la detección de cualquier nivel de fallo. Dentro de cada una de estas secciones se consideran análisis mediante las dos técnicas de clasificación utilizadas (LDA y AD), para cada uno de los dos niveles de carga (alta y baja) y con las diferentes fuentes de datos disponibles (corriente, sonido y los 3 ejes de vibraciones). Esto arroja un total de 20 análisis en cada una de las dos secciones.

Dado que el desarrollo exhaustivo de estas 40 posibilidades es a todas luces excesivo, se ha decidido ofrecer el análisis detallado de las 5 fuentes de datos disponibles para el caso de carga alta y resumir los resultados del resto de las situaciones para favorecer de este modo la lectura de este trabajo.

Antes de ofrecer los detalles de estos resultados, la sección 6.1 describe la experimentación con los datos disponibles, realizada para cada una de las 40 situaciones que se han mencionado.

6.1. Experimento

El enfoque adoptado en el tratamiento de los datos ha sido el siguiente (la figura 6.1 contiene un resumen de estos pasos).

1. Primera partición: dividimos el conjunto de datos en entrenamiento (train) y en prueba (test) bajo la proporción 0,7 y 0,3.
2. Validación cruzada: a continuación realizaremos una 5-fold Validation en el subconjunto train a su vez dividiéndolo en 5 particiones eludiendo así el posible sobreajuste. También conseguiremos esto en la selección de hiperparámetros GridSearch en el algoritmo de árboles de decisión donde fijaremos la profundidad máxima del árbol en 3. Ahora se seleccionará cada una de las particiones y en un proceso iterativo se utilizarán como set de prueba mientras que los sobrantes 4 subconjuntos serán los subconjuntos de entrenamiento.
3. Entrenamiento: durante el paso anterior se selecciona el mejor modelo y ahora se entrena con todos los datos de entrenamiento.
4. Medición del rendimiento: por último, evaluamos el modelo ya entrenado utilizando el conjunto test de la partición inicial y que no han sido empleados ni en el entrenamiento ni en la búsqueda de hiperparámetros.

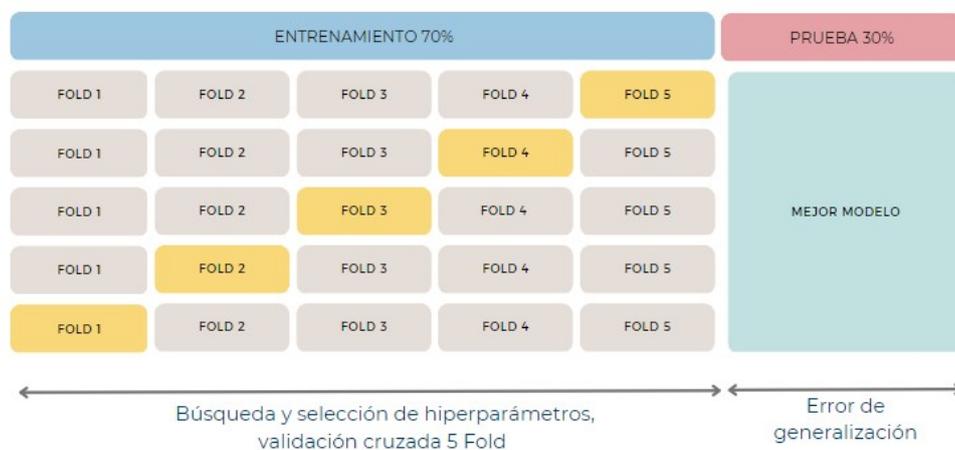


Figura 6.1: Resumen del procedimiento. Adaptado de [3]

6.2. Análisis 1. Grupo 1 contra grupo 2

6.2.1. Análisis discriminante lineal

En esta sección se utiliza el análisis discriminante lineal para generar reglas de clasificación para cada uno de los escenarios mencionados anteriormente. Como ya se comentó

en la sección 4.3, se dispone de un número limitado de observaciones en comparación con el número de variables por lo que se ha llevado a cabo un análisis en componentes principales con el objetivo de reducir el número de variables en el modelo manteniendo la mayor cantidad de información posible. El código Python utilizado en los análisis discriminantes lineales puede encontrarse en el anexo B.3

Como puede verse en las tablas A.1 a A.20 incluidas en el anexo A.2, la inclusión de tres componentes principales es suficiente para capturar una cantidad significativa de variabilidad en los datos. Específicamente, se ha comprobado que al considerar estas tres componentes principales, se logra abarcar al menos el 85 % de la variabilidad presente en los datos. En algunos casos, incluso se alcanza un nivel más alto, llegando hasta el 94 %.

Con base en este análisis, se ha adoptado el criterio de retener únicamente las tres primeras componentes principales en todos los casos. Esto se debe a que estas tres componentes son capaces de explicar una cantidad sustancial de información contenida en los datos originales.

Además, se ha establecido un umbral de relevancia para las variables en cada componente. Se considera que aquellas variables cuyos coeficientes (en valor absoluto) superan 0.3 están proporcionando información significativa en esa componente específica. Al analizar los resultados, se observa, de nuevo en las tablas A.1 a A.20, que todas las variables originales aparecen como relevantes en al menos una de las tres componentes, excepto la variable 10 en el caso de las vibraciones y carga baja, y la variable 8 en el caso de las vibraciones en el eje z y carga baja. Por lo tanto, no parece justificado eliminar ninguna de las variables originales, ya que todas ellas desempeñan un papel importante en alguna de las componentes principales.

En consecuencia, se lleva a cabo el análisis discriminante lineal mediante las tres primeras componentes principales para todos los casos. Esto permite mantener la mayor cantidad posible de información y reducir el riesgo de sobreajuste puesto que solamente se están considerando tres variables en el análisis discriminante. Merece la pena resaltar también que en la construcción de las componentes principales no interviene la variable respuesta por lo que no se genera ningún sesgo en la clasificación.

A continuación se detallan los resultados obtenidos para carga alta. Como ya se ha mencionado, el resto de resultados se ofrecen de forma resumida.

LDA en Carga Alta para Corriente

Como se ha comentado en la sección previa, se han utilizado como variables explicativas las tres primeras componentes principales generadas a través de los estadísticos 5.1 de los datos de corriente en carga alta.

Los resultados obtenidos para esta situación aparecen de forma gráfica en la figura 6.2. Esta figura está compuesta por 6 subfiguras que contienen información visual sobre la disposición tridimensional de los datos (recordar que estamos considerando tres variables explicativas, las tres primeras componentes principales) y las métricas de clasificación definidas en la sección 4.2.1. Concretamente, la subfigura 6.2a contiene la representación tridimensional de los datos con colores diferentes para cada una de las dos clases de

la variables respuesta, la subfigura 6.2b muestra los resultados para cada uno de los 5 folds de la métrica accuracy definida en (4.1) tanto en el conjunto de entrenamiento (train) como en el de test y las subfiguras 6.2c,6.2d,6.2e,6.2f muestran los resultados correspondientes a las métricas Error, F1, Precision y Recall definidas en (4.2),(4.5), (4.3) y (4.4) respectivamente.

Puede verse en la figura como los resultados son muy buenos para todas las métricas. Por ejemplo, si se considera la métrica Error, que es equivalente a Accuracy y es la que se utiliza más habitualmente en la práctica, y que consideraremos en la sección en la que compararemos los diferentes resultados obtenidos, puede verse que en 4 de los 5 folds el error obtenido en el conjunto test es 0 y que en uno de ellos el error es 0,333. En consecuencia la evaluación del error mediante 5-fold cross validation será $\frac{0,333}{5} = 0,067$. De forma equivalente puede decirse que la precisión (Accuracy) de la regla de clasificación obtenida en este caso mediante el análisis discriminante lineal utilizando como variables explicativas las tres primeras componentes principales de las variables originales es de $1 - 0,067 = 0,933$.

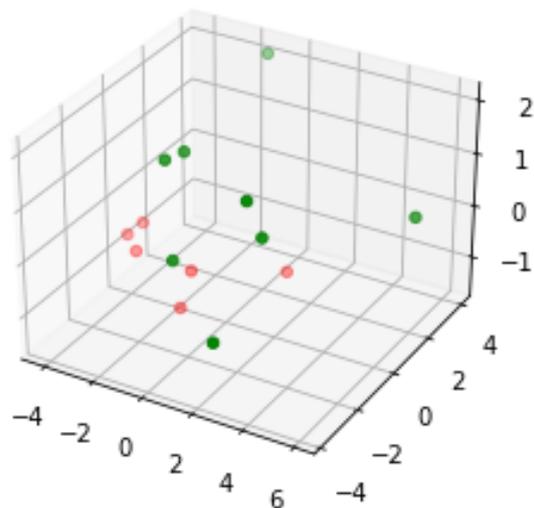
LDA en Carga Alta para Sonido

Los resultados correspondientes al análisis discriminante lineal en carga alta para los datos de sonido aparecen en la figura 6.3 que tiene la misma estructura que la figura del análisis de los datos de corriente (figura 6.2) descrita en la sección anterior.

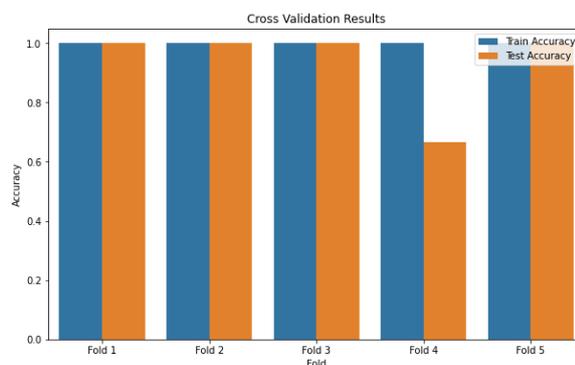
Como puede verse en la subfigura 6.3b correspondiente a la precisión, el error de todos los folds es 0 excepto en el quinto para el que el error es 0,5. En consecuencia, el error de clasificación de esta regla discriminante lineal es $\frac{0,5}{5} = 0,1$ o, de manera equivalente, el Accuracy de esta regla es 0,9.

LDA en Carga Alta para Vibraciones

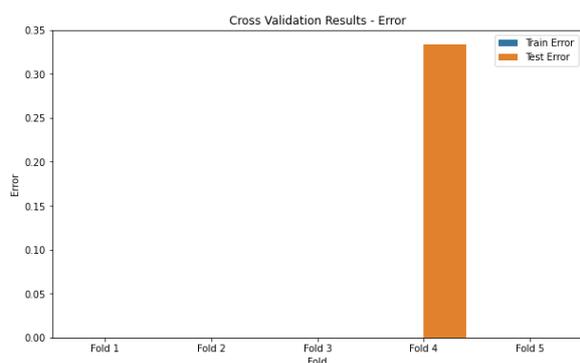
Para el caso de vibraciones, como ya se ha comentado previamente, se dispone de tres análisis, uno para cada uno de los ejes x, y, z. Los resultados correspondientes a estos análisis aparecen en las figuras 6.4 para el eje x, 6.5 para el eje y y 6.6 para el eje z. En las subfiguras 6.4c, 6.5c y 6.6c puede verse como para cada uno de los ejes se obtiene un error 0 en cada uno de los 5 folds por lo que el error estimado de cada una de estas tres reglas lineales de clasificación construidas a partir de las tres primeras componentes principales de cada uno de los ejes es 0.



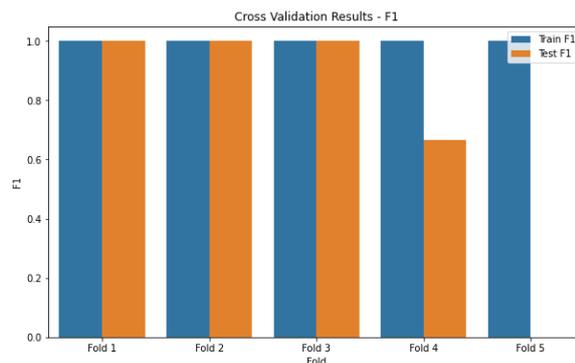
(a) Figura 3D



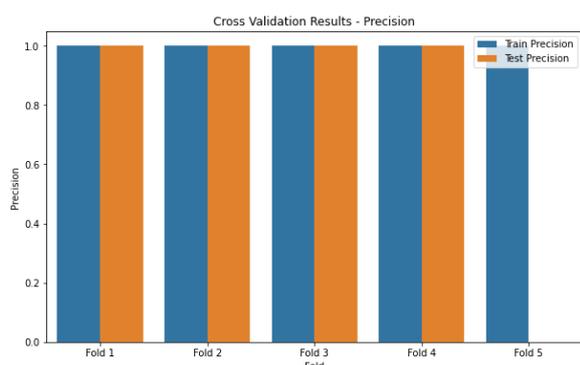
(b) Gráfico de Barras Accuracy



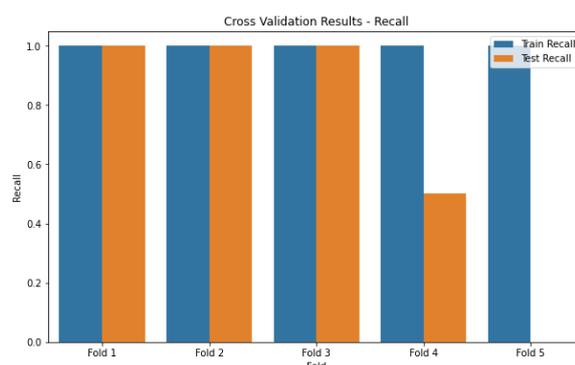
(c) Gráfico de Barras Error



(d) Gráfico de Barras F1 CA_LDA_Corriente

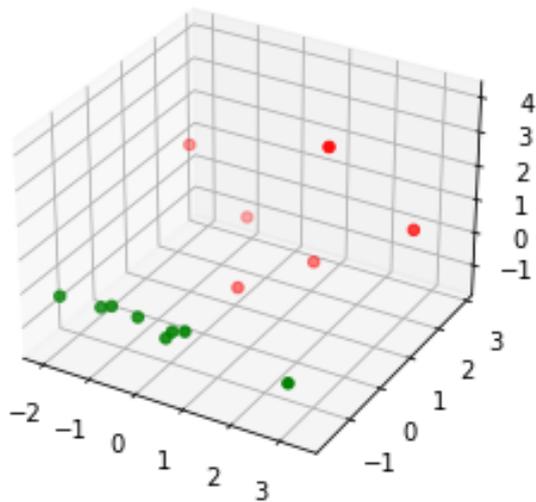


(e) Gráfico de Barras Precision

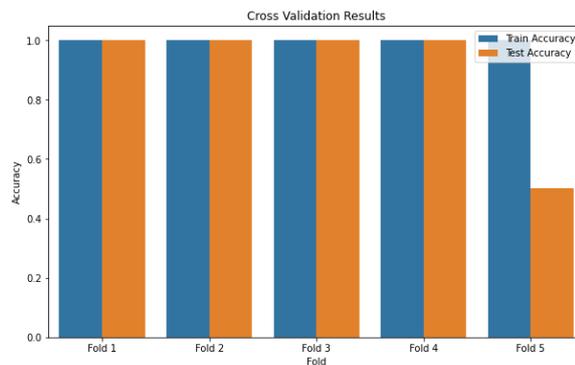


(f) Gráfico de Barras Recall

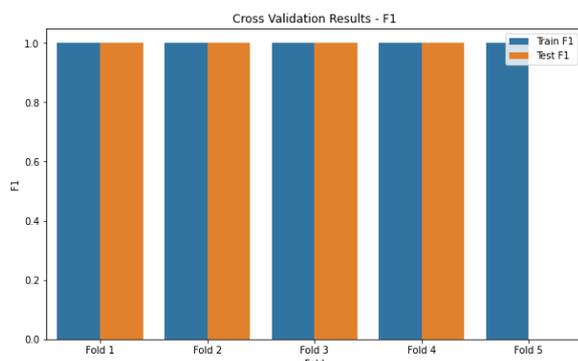
Figura 6.2: Métricas Análisis 1-CA_LDA_Corr



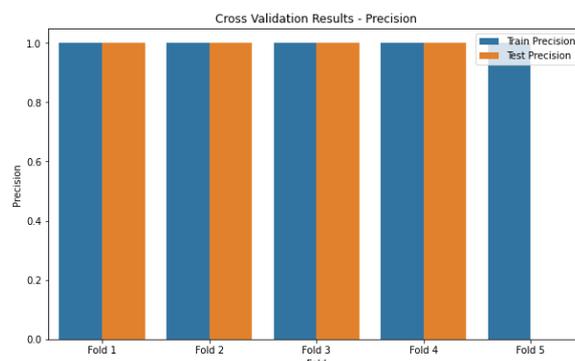
(a) Figura 3D



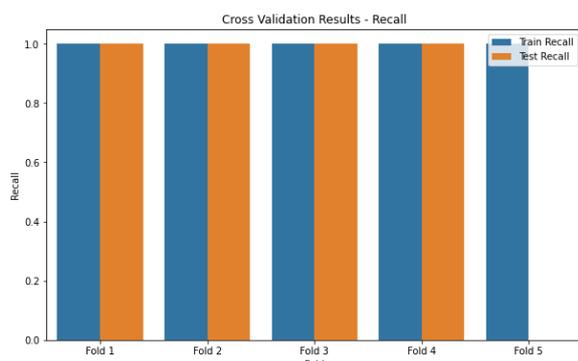
(b) Gráfico de Barras Accuracy



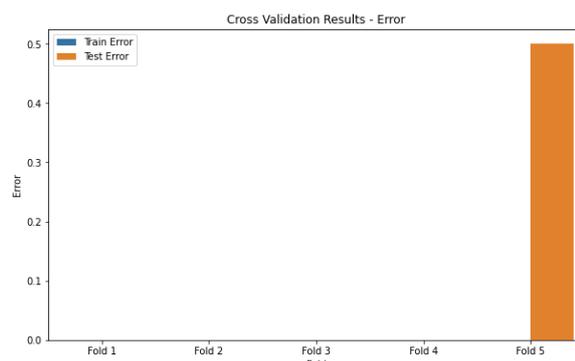
(c) Gráfico de Barras F1



(d) Gráfico de Barras Precision

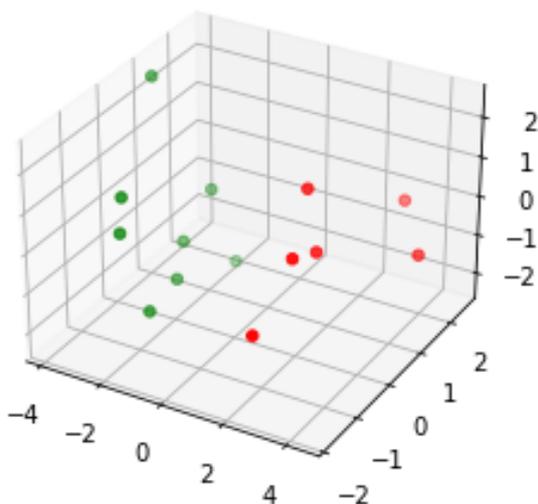


(e) Gráfico de Barras Recall

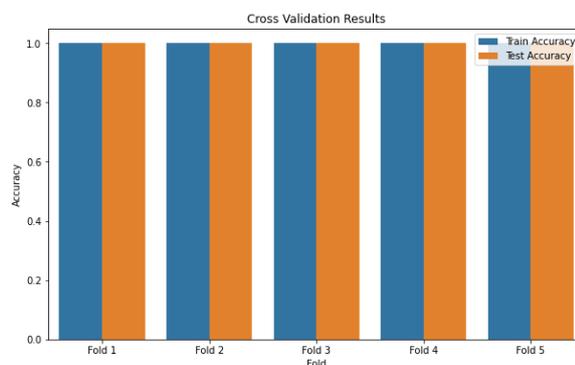


(f) Gráfico de Barras Error

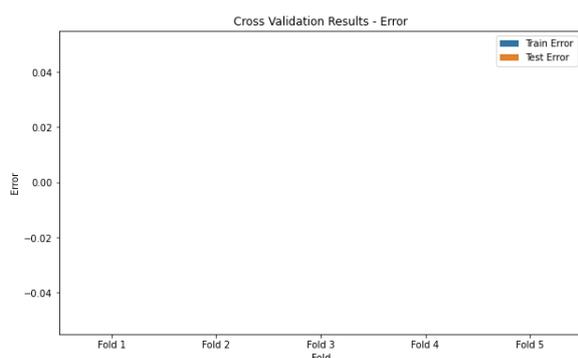
Figura 6.3: Métricas Análisis 1-CA_LDA_Sonido



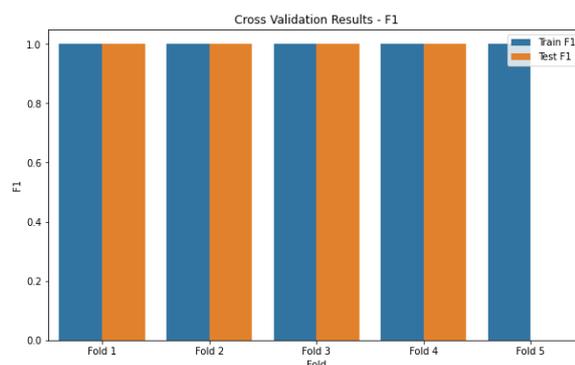
(a) Figura 3D



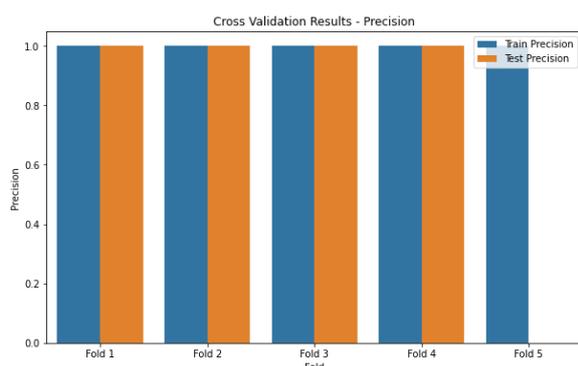
(b) Gráfico de Barras Accuracy



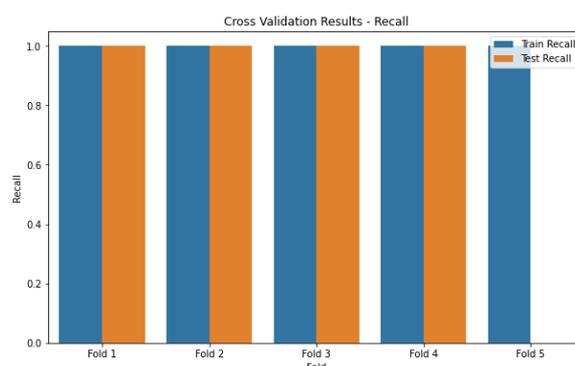
(c) Gráfico de Barras Error



(d) Gráfico de Barras F1

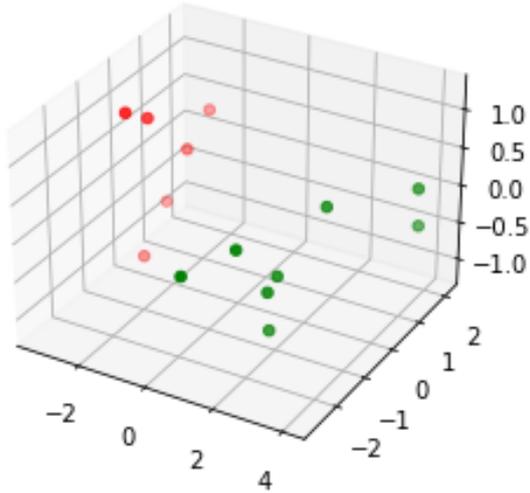


(e) Gráfico de Barras Precision

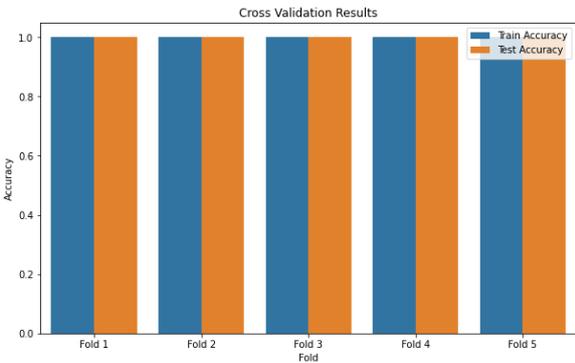


(f) Gráfico de Barras Recall

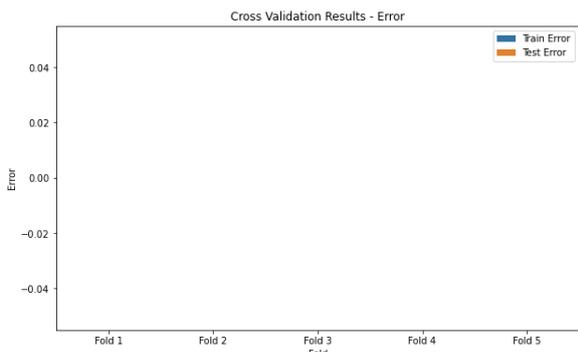
Figura 6.4: Métricas Análisis 1-CA_LDA_Vibración eje x



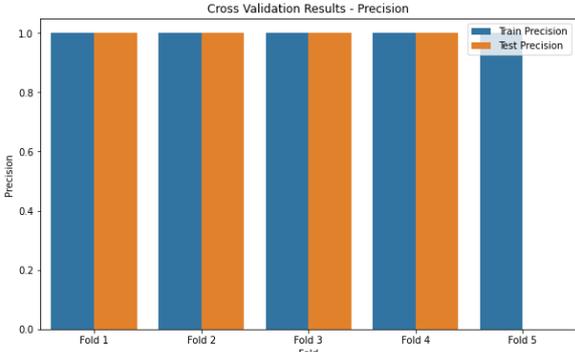
(a) Figura 3D



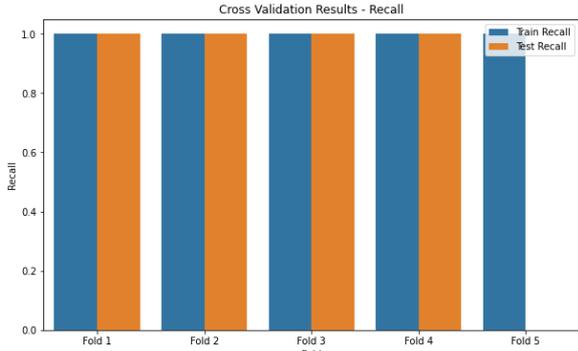
(b) Gráfico de Barras Accuracy



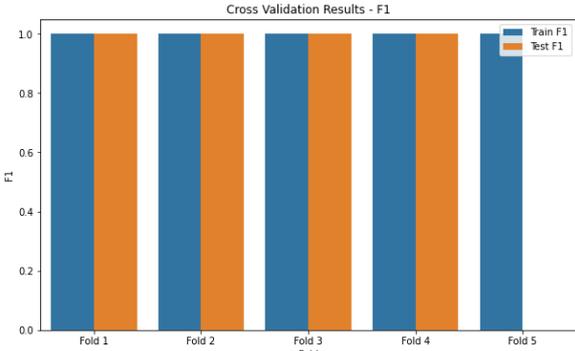
(c) Gráfico de Barras Error



(d) Gráfico de Barras Precision

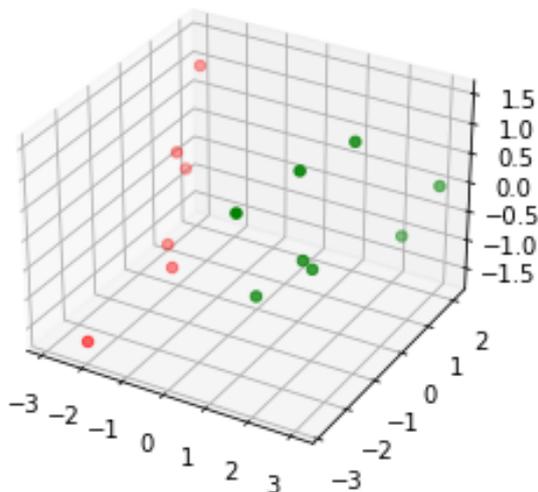


(e) Gráfico de Barras Recall

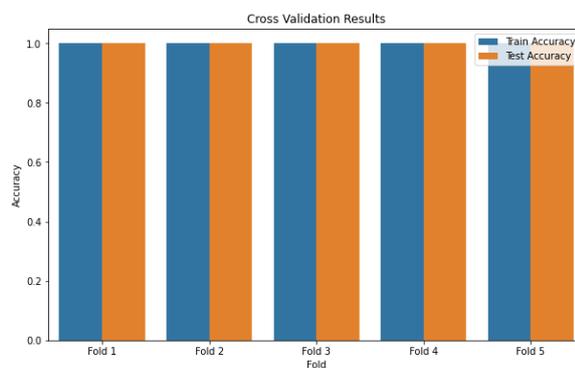


(f) Gráfico de Barras F1

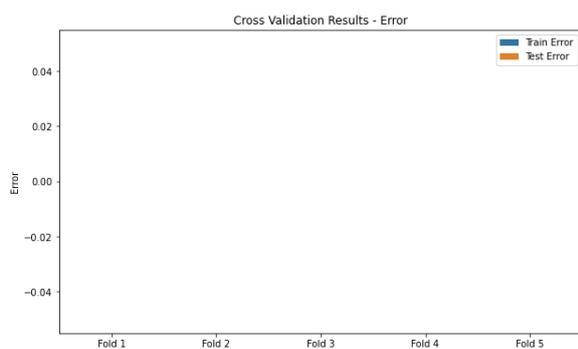
Figura 6.5: Métricas Análisis 1-CA_LDA_Vibración eje y



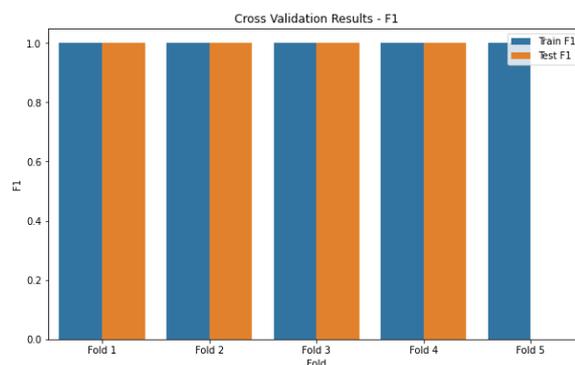
(a) Figura 3D



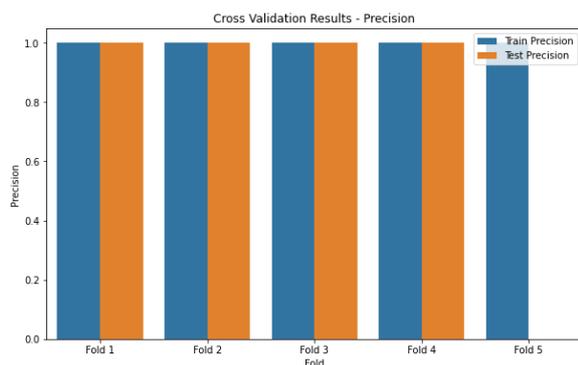
(b) Gráfico de Barras Accuracy



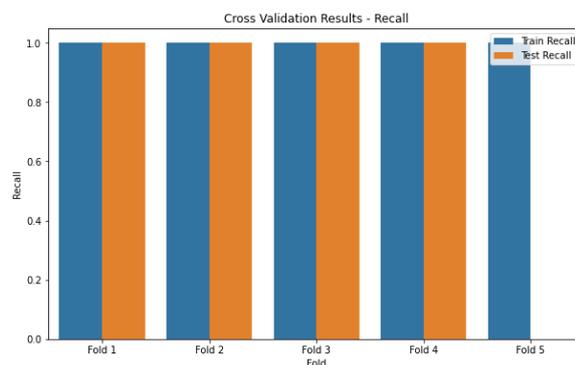
(c) Gráfico de Barras Error



(d) Gráfico de Barras F1



(e) Gráfico de Barras Precision



(f) Gráfico de Barras Recall

Figura 6.6: Métricas Análisis 1-CA_LDA_Vibración eje z

Resumen de resultados LDA para cargas alta y baja

Los resultados completos de este análisis con LDA para el caso de carga baja pueden encontrarse en el Apéndice A.3.2. Para este caso se mantiene el gráfico 3D que permite una valoración visual de la regla y el resto de los gráficos han sido reemplazados por tablas resumen de los valores de las distintas métricas de evaluación de las reglas. La figura A.21 y la tabla A.41 contienen los resultados para corriente, la figura A.22 y la tabla A.42 los de sonido y las figuras A.23, A.24 y A.25 y las tablas A.43, A.44 y A.45 los correspondientes a los ejes x, y y z de vibraciones, respectivamente.

La tabla 6.1 contiene los resultados para la métrica Error que es la de uso más común. Puede verse que el Error de las reglas es 0 para todos los casos en carga baja y en todos los ejes de vibraciones en carga alta, mientras que eso no ocurre para corriente y sonido en carga alta. Estos resultados son coherentes con los que se obtienen habitualmente que parecen indicar que, en general, es más fácil la detección de los fallos en carga baja. En cualquier caso, posteriormente se realizarán análisis más detallados para determinar cuáles de estas diferencias son estadísticamente significativas.

	Fold	1	2	3	4	5	Promedio
Carga alta	Corriente	0	0	0	0.333	0	0.067
	Sonido	0	0	0	0	0.5	0.1
	Vibraciones x	0	0	0	0	0	0
	Vibraciones y	0	0	0	0	0	0
	Vibraciones z	0	0	0	0	0	0
Carga baja	Corriente	0	0	0	0	0	0
	Sonido	0	0	0	0	0	0
	Vibraciones x	0	0	0	0	0	0
	Vibraciones y	0	0	0	0	0	0
	Vibraciones z	0	0	0	0	0	0

Cuadro 6.1: Error obtenido en el análisis 1 para cada fold y promedio para el Análisis Discriminante Lineal

6.2.2. Árboles de decisión

Como ya se ha comentado, se han considerado árboles de decisión con un máximo de tres variables de entre las descritas en la tabla 5.1 para, de alguna manera, obtener modelos de una complejidad equivalente al LDA considerado en la sección anterior que se llevaba a cabo con las tres primeras componentes principales obtenidas a partir de todas las variables originales. Además, el considerar este tipo de árboles de una profundidad limitada permite también intentar controlar el riesgo de sobreajuste en los modelos y el

hecho de que se construyan a partir de las variables originales facilita la interpretación de los resultados. El código Python de los análisis correspondientes a los árboles de decisión puede encontrarse en el anexo B.4

De nuevo se ha efectuado un análisis para cada una de las 5 posibilidades de que se dispone en cuanto a tipo de datos y con el objetivo de facilitar la lectura de este trabajo se ofrece un resumen de los resultados en lugar de detallarlos completamente.

Los gráficos y tablas que ilustran los resultados tienen la misma estructura que los que se han ofrecido para el LDA de la sección anterior. Para el gráfico 3D en los ejes de coordenadas están, en lugar de las tres primeras componentes principales, las variables que intervinieron en la construcción del árbol correspondiente a cada caso. Los gráficos para carga alta pueden verse en el Apéndice A.3.1. La figura A.16 contiene los resultados para corriente, la figura A.17 los de sonido y las figuras A.18, A.19 y A.20 los correspondientes a los ejes x, y y z de vibraciones, respectivamente.

Para el caso de carga baja, los gráficos y tablas se encuentran en el Apéndice A.3.2. La figura A.26 y la tabla A.46 contienen los resultados para corriente, la figura A.27 y la tabla A.47 los de sonido y las figuras A.28, A.29 y A.30 y las tablas A.48, A.49 y A.50 los correspondientes a los ejes x, y y z de vibraciones, respectivamente.

La tabla 6.2 contiene los resultados para la métrica Error que es la de uso más común. Puede verse que en carga baja el Error de las reglas es 0 para todos los casos y que en carga alta solo lo es para vibraciones en eje x y para vibraciones en eje y. De nuevo se observa que los resultados son mejores en carga baja que en carga alta.

	Fold	1	2	3	4	5	Promedio
Carga alta	Corriente	0	0	0.333	0	0	0.067
	Sonido	0	0.333	0	0	0.5	0.167
	Vibraciones x	0	0	0	0	0	0
	Vibraciones y	0	0	0	0	0	0
	Vibraciones z	0.333	0	0	0	0	0.067
Carga baja	Corriente	0	0	0	0	0	0
	Sonido	0	0	0	0	0	0
	Vibraciones x	0	0	0	0	0	0
	Vibraciones y	0	0	0	0	0	0
	Vibraciones z	0	0	0	0	0	0.067

Cuadro 6.2: Error obtenido en el análisis 1 para cada fold y promedio para los árboles de decisión

Adicionalmente, la tabla 6.3 contiene las 3 variables que definen los árboles de decisión en cada uno de los escenarios considerados. Hay que notar que esta tabla no puede hacerse para el caso del LDA puesto que en esa situación se han considerado las componentes principales y todas las variables originales participan en cada una de las componentes

principales. Esta tabla es útil porque permite conocer cuáles de las variables originales son relevantes en la construcción de los modelos. Puede verse como las variables kurt (coeficiente de apuntamiento) y xp (máximo valor absoluto) participan en todos los árboles construidos.

Tipo de dato	Carga alta	Carga baja
Corriente	c1	c2
	kurt	kurt
	xp	xp
Sonido	c2	skew
	kurt	kurt
	xp	xp
Vibración x	c4	m4
	kurt	kurt
	xp	xp
Vibración y	m4	kurt
	kurt	xp
	xp	skew
Vibración z	m2	m2
	kurt	kurt
	xp	xp

Cuadro 6.3: Tabla variables para los Árboles de decisión en el análisis 1

6.2.3. Análisis de la varianza de los resultados del grupo 1 contra el grupo 2

En esta sección se analizan con más profundidad los resultados obtenidos en las tablas 6.1 y 6.2 con el objetivo de determinar cuáles son los factores influyentes en el error que cometen las distintas reglas de clasificación obtenidas en este análisis. Para llegar a este propósito se ha efectuado un análisis de la varianza con tres factores, el método de clasificación (LDA o AD), la carga del motor (Alta o Baja) y el tipo de datos empleados en la clasificación (Corriente, Sonido, Vibraciones eje x, Vibraciones eje y o vibraciones eje z). Se ha considerado además la posibilidad de la existencia de interacciones entre los efectos de los factores. El resultado obtenido en este análisis se muestra en la tabla 6.4.

Source	Sum of Squares	Df	Mean Square	F-Ratio	P-Value
MAIN EFFECTS					
A:Método	0,00998001	1	0,00998001	1,03	0,3138
B:Carga	0,03996	1	0,03996	4,11	0,0458
C:TipoDatos	0,0621778	4	0,0155444	1,60	0,1823
INTERACTIONS					
AB	0,00110889	1	0,00110889	0,11	0,7363
AC	0,0177422	4	0,00443556	0,46	0,7672
BC	0,0710622	4	0,0177656	1,83	0,1313
ABC	0,00443556	4	0,00110889	0,11	0,9772
RESIDUAL	0,776956	80	0,00971195		
TOTAL (CORRECTED)	0,983423	99			

Cuadro 6.4: Tabla ANOVA con interacciones correspondiente al análisis 1

Puede observarse en la tabla 6.4 que las interacciones entre los factores no son significativas por lo que se han ido eliminando de forma secuencial. Tras la eliminación de las interacciones se obtienen los resultados que se muestran en la tabla 6.5.

Source	Sum of Squares	Df	Mean Square	F-Ratio	P-Value
MAIN EFFECTS					
A:Método	0,00998001	1	0,00998001	1,07	0,3047
B:Carga	0,03996	1	0,03996	4,27	0,0417
C:TipoDatos	0,0621778	4	0,0155444	1,66	0,1661
RESIDUAL	0,871305	93	0,00936887		
TOTAL (CORRECTED)	0,983423	99			

Cuadro 6.5: Tabla ANOVA sin interacciones correspondiente al análisis 1

En esta tabla se ve que los efectos principales correspondientes al método de análisis y al tipo de datos continúan siendo no significativos por lo que de nuevo se eliminan de forma secuencial obteniéndose la tabla final 6.6.

Source	Sum of Squares	Df	Mean Square	F-Ratio	P-Value
MAIN EFFECTS					
A:Carga	0,03996	1	0,03996	4,15	0,0443
RESIDUAL	0,943463	98	0,00962717		
TOTAL (CORRECTED)	0,983423	99			

Cuadro 6.6: Tabla ANOVA final correspondiente al análisis 1

En la tabla 6.6 puede verse que el factor carga es el único significativo en este análisis puesto que su p-valor es ligeramente inferior al valor usual de referencia 0,05. Esto quiere decir que, en este análisis, ni cambiar el método de análisis ni el tipo de datos supone un cambio significativo en el error de las reglas de clasificación. Es por tanto indiferente, desde este punto de vista el uso de LDA o de AD y tampoco es relevante el uso de un tipo de datos frente al resto. En cambio, si lo es la carga a la que está funcionando el motor. Si efectuamos un análisis post-hoc mediante el método de Tukey obtenemos los resultados que se muestran en la tabla 6.7. Esta tabla pone de manifiesto que para la detección de fallos incipiente que se está llevando a cabo en el análisis 1 el nivel de carga al que está trabajando el motor es relevante y que se obtiene una capacidad de detección de los fallos superior cuando el motor está trabajando en carga baja. Esto quiere decir que es más difícil detectar los fallos cuando el motor está trabajando de la forma habitual y que si se quiere tener más capacidad de detección deberían programarse ciclos de baja de carga en el motor en los que efectuar la lectura de los datos y establecer el diagnóstico.

Carga	Count	Mean	Homogeneous Groups
1 (Baja)	50	0,00666	X
2 (Alta)	50	0,04664	X

Cuadro 6.7: Análisis post-hoc mediante el método de Tukey de los resultados correspondientes al análisis 1

6.3. Análisis 2. Grupo 1 contra grupo 2+3+4+5+6

6.3.1. Análisis discriminante lineal

De igual forma que en el análisis 1 del grupo 1 contra el grupo 2 podemos observar en las tablas A.21 a A.40 del anexo A.2 que conservando las tres componentes principales se captura una cantidad significativa de la información contenida en los datos originales. Siendo fieles al umbral de relevancia de 0.3 y examinando de nuevo los autovalores y autovectores de las tablas A.21 a A.40 podemos reafirmar que todas las variables originales se presentan como significativas en al menos una de las tres componentes, excepto la variables 10 en vibraciones Carga Baja y Alta, sonido Carga Alta y corriente Carga

Alta y 11 en vibraciones Carga Baja. Acto seguido se presentan en detalle los resultados obtenidos para la carga alta. Como se mencionó previamente, los resultados restantes se presentan de manera resumida.

Resumen de resultados LDA para cargas alta y baja

Los resultados completos de este análisis con LDA para el caso de carga alta pueden encontrarse en el Apéndice A.4.1. Como ya hicimos en el análisis 1, mantendremos el gráfico 3D que proporciona una evaluación visual de la regla, mientras que el resto de los gráficos han sido reemplazados por tablas resumen que contienen los valores de las diferentes métricas de evaluación de las reglas.

La figura A.31 y la tabla A.51 contienen los resultados para corriente, la figura A.32 y la tabla A.52 los de sonido y las figuras A.33, A.34 y A.35 y las tablas A.53, A.54 y A.55 los correspondientes a los ejes x, y y z de vibraciones, respectivamente.

Asimismo, los resultados completos de este análisis con LDA para el caso de carga baja pueden encontrarse en el Apéndice A.4.2. La figura A.40 y la tabla A.60 contienen los resultados para corriente, la figura A.41 y la tabla A.61 los de sonido y las figuras A.42, A.43 y A.44 y las tablas A.62, A.63 y A.64 los correspondientes a los ejes x, y y z de vibraciones, respectivamente.

Los resultados en la tabla 6.8 muestran los errores asociados al conjunto de prueba, en cada uno de los 5 folds. Aunque el análisis detallado se llevará a cabo más adelante mediante un análisis de la varianza, puede observarse que, en general, se obtienen mejores resultados con los datos de corriente que con los de sonido o vibraciones.

	Fold	1	2	3	4	5	Promedio
Carga alta	Corriente	0	0	0	0	0.125	0.025
	Sonido	0	0.222	0.125	0.125	0.375	0.1694
	Vibraciones x	0	0.222	0.125	0.125	0.375	0.1694
	Vibraciones y	0	0.125	0.125	0.125	0.222	0.1194
	Vibraciones z	0	0.222	0.125	0.125	0.375	0.1694
Carga baja	Corriente	0	0.111	0.125	0.125	0.125	0.0972
	Sonido	0	0.222	0.125	0.25	0.375	0.1944
	Vibraciones x	0	0.222	0.125	0.125	0.375	0.1694
	Vibraciones y	0	0.222	0	0	0.25	0.0944
	Vibraciones z	0.222	0.222	0.125	0.125	0.375	0.2138

Cuadro 6.8: Error obtenido en el análisis 2 para cada fold y promedio para los análisis discriminantes lineales

6.3.2. Árboles de decisión

Resumen de resultados AD para cargas alta y baja

Una vez más, se ha llevado a cabo un análisis para cada una de las 5 posibilidades disponibles en cuanto al tipo de datos.

De igual manera, los resultados completos de este análisis con árboles de decisión para el caso de carga alta pueden encontrarse en el Apéndice A.4.1 y los de carga baja en el Apéndice A.4.2. La figura A.35 y la tabla A.55 contienen los resultados para corriente CA, la figura A.36 y la tabla A.56 los de sonido CA y las figuras A.37, A.38 y A.39 y las tablas A.57, A.58 y A.59 los correspondientes a los ejes x, y y z de vibraciones CA, respectivamente.

La figura A.45 y la tabla A.65 contienen los resultados para corriente CB, la figura A.46 y la tabla A.66 los de sonido CB y las figuras A.47, A.48 y A.49 y las tablas A.67, A.68 y A.69 los correspondientes a los ejes x, y y z de vibraciones CB, respectivamente.

Los resultados en la tabla 6.9 muestran los errores asociados al conjunto de prueba, en cada uno de los 5 folds. En este caso un análisis preliminar permite observar que los datos de corriente y sonido ofrecen mejores resultados que los de vibraciones.

	Fold	1	2	3	4	5	Promedio
Carga alta	Corriente	0	0	0	0	0	0
	Sonido	0	0	0	0	0	0
	Vibraciones x	0.111	0	0	0	0.125	0.0472
	Vibraciones y	0.111	0	0.125	0.125	0	0.0722
	Vibraciones z	0	0	0.25	0.125	0.125	0.100
Carga baja	Corriente	0	0	0	0	0	0
	Sonido	0	0	0	0	0	0
	Vibraciones x	0	0	0	0.125	0	0.025
	Vibraciones y	0	0.222	0	0	0.25	0.0944
	Vibraciones z	0.222	0	0	0.125	0	0.0694

Cuadro 6.9: Error obtenido en el análisis 2 para cada fold y promedio para los árboles de decisión

Además, en la tabla 6.10 se incluyen las tres variables que participan en los árboles de decisión. De igual manera que ocurría en el análisis 1 se puede observar que las variables kurt (coeficiente de apuntamiento) y xp (máximo valor absoluto) están presentes en todos los árboles construidos.

Tipo de dato	Carga alta	Carga baja
Corriente	c1	c1
	kurt	kurt
	xp	xp
Sonido	c2	c2
	kurt	kurt
	xp	xp
Vibración x	c3	c1
	kurt	kurt
	xp	xp
Vibración y	c2	c1
	kurt	kurt
	xp	xp
Vibración z	c2	m2
	kurt	kurt
	xp	xp

Cuadro 6.10: Tabla variables para los árboles de decisión en el análisis 2

6.3.3. Análisis de la varianza de los resultados del grupo 1 contra el grupo 2+3+4+5+6

Al igual que se hizo para el análisis 1, en esta sección se lleva a cabo un análisis más profundo mediante un análisis de la varianza de tres factores de los resultados de error de clasificación presentados en las tablas 6.8 y 6.9. Como en el caso anterior, los factores son el método de análisis (2 niveles), el nivel de carga del motor (2 niveles) y el tipo de datos empleado en la construcción de la regla de clasificación (5 niveles). Al igual que en el caso anterior se dispone de un total de 100 observaciones para este análisis puesto que para cada uno de los $2 \cdot 2 \cdot 5 = 20$ cruces de los tres factores se tienen 5 folds. Los resultados del análisis completo con las interacciones se presentan en la tabla 6.11.

Source	Sum of Squares	Df	Mean Square	F-Ratio	P-Value
MAIN EFFECTS					
A:Método	0,234837	1	0,234837	25,36	0,0000
B:Carga	0,00425104	1	0,00425104	0,46	0,5000
C:TipoDatos	0,120992	4	0,030248	3,27	0,0156
INTERACTIONS					
AB	0,00917764	1	0,00917764	0,99	0,3225
AC	0,0952592	4	0,0238148	2,57	0,0440
BC	0,00606356	4	0,00151589	0,16	0,9561
ABC	0,00577676	4	0,00144419	0,16	0,9598
RESIDUAL	0,740729	80	0,00925911		
TOTAL (CORRECTED)	121,709	99			

Cuadro 6.11: Tabla ANOVA con todas las interacciones correspondiente al análisis 2

En la tabla 6.11 se observa que la interacción de los tres factores no es significativa y que tampoco lo son la interacción del segundo factor con cada uno de los otros dos. Si eliminamos secuencialmente estas interacciones no significativas se obtienen los resultados que se muestran en la tabla 6.12.

Source	Sum of Squares	Df	Mean Square	F-Ratio	P-Value
MAIN EFFECTS					
A:Método	0,234837	1	0,234837	27,44	0,0000
B:Carga	0,00425104	1	0,00425104	0,50	0,4828
C:TipoDatos	0,120992	4	0,030248	3,53	0,0101
INTERACTIONS					
AC	0,0952592	4	0,0238148	2,78	0,0314
RESIDUAL	0,761747	89	0,00855895		
TOTAL (CORRECTED)	121,709	99			

Cuadro 6.12: Tabla ANOVA con solo las interacciones significativas correspondiente al análisis 2

Es claro en la tabla 6.12 que el efecto principal del factor carga sigue sin ser significativo. Dado que las interacciones de este factor con el resto tampoco lo son, se procede a eliminar este factor del modelo con lo que se obtiene el modelo definitivo puesto que, como puede verse en la tabla 6.13, una vez eliminado este efecto principal, tanto los efectos principales de los otros dos factores como su interacción sí son significativos.

6.3. Análisis 2. Grupo 1 contra grupo 2+3+4+5+6

Source	Sum of Squares	Df	Mean Square	F-Ratio	P-Value
MAIN EFFECTS					
A:Método	0,234837	1	0,234837	27,59	0,0000
B:TipoDatos	0,120992	4	0,030248	3,55	0,0097
INTERACTIONS					
AB	0,0952592	4	0,0238148	2,80	0,0306
RESIDUAL	0,765998	90	0,00851109		
TOTAL (CORRECTED)	121,709	99			

Cuadro 6.13: Tabla ANOVA final correspondiente al análisis 2

A diferencia de lo que ocurría en el análisis 1, en este segundo análisis en el que se pretende diferenciar el motor sano de un motor con cualquier tipo de fallo (con mayor o menor gravedad), el nivel de carga no es un factor significativo en la calidad de la clasificación mientras que sí lo son tanto el método de clasificación como el tipo de datos utilizados para construir la regla como la interacción entre ambos factores. El gráfico de interacción entre estos factores, gráfico 6.7, muestra cómo son las interacciones entre ambos efectos.

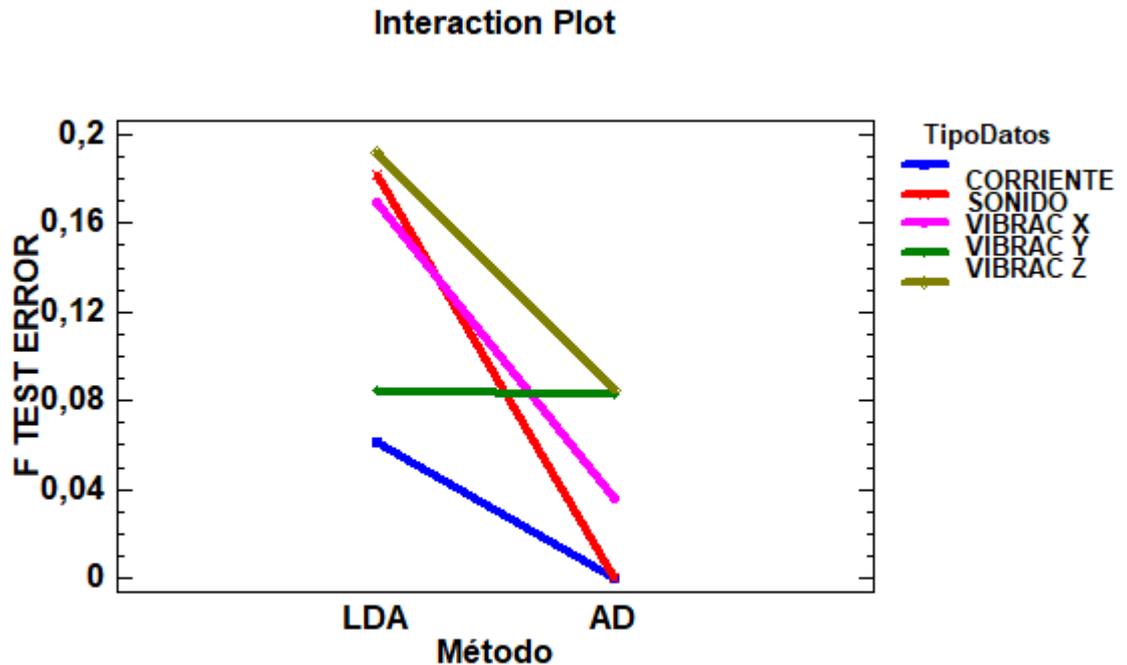


Figura 6.7: Gráfico de interacciones del modelo ANOVA final del análisis 2

Puede observarse en el gráfico 6.7 cómo, en general, para este segundo análisis el método de árboles de decisión da un mejor resultado (errores más bajos) que el análisis discriminante lineal y cómo el uso de los datos de corriente da unos resultados al menos tan buenos como el resto de tipos de datos. La mejora que se obtiene con el uso de los árboles de decisión no es siempre igual de importante puesto que, por ejemplo, la mejora es muy alta cuando se consideran los datos de sonido o de vibraciones en el eje x pero prácticamente inexistente cuando se consideran las vibraciones en el eje y.

Capítulo 7

Conclusiones y trabajo futuro

Como se comentó en la sección 1.2, el objetivo fundamental del proyecto es el desarrollo de un sistema de diagnóstico de fallos incipientes en rodamientos en motores de inducción mediante aprendizaje supervisado. Los resultados obtenidos en la sección 6 demuestran que se pueden diagnosticar tanto los fallos incipientes como fallos de cualquier gravedad mediante los métodos considerados en este trabajo (análisis discriminante lineal y árboles de decisión) con un alto porcentaje de acierto. Estos métodos, además, son métodos simples y computacionalmente no costosos con lo que, como se planteaba en el primer objetivo específico, no parece necesario recurrir a métodos más elaborados y opacos como se hacía en otros trabajos como [2] o [3]. La ventaja adicional de los métodos considerados en este trabajo es la facilidad de interpretación. Así, por ejemplo, se sabe cuáles son las variables que han sido consideradas en los árboles de decisión para conseguir la clasificación obtenida. Hay que señalar también que, aunque no hay diferencias significativas entre estos dos métodos en la detección del fallo incipiente, sí la hay en la detección de cualquier tipo de fallo, caso en el que es mejor el uso de los árboles de decisión. Dado que en este método es más directo el conocimiento de las variables influyentes en el diagnóstico, parece razonable recomendar los árboles de decisión como método de construcción de las reglas de clasificación.

El resto de objetivos específicos enumerados en la sección 1.2 también han sido alcanzados. Así, se ha concluido que la carga del motor es un factor relevante en la capacidad de clasificación cuando se aborda la detección de fallos incipientes (análisis 1) pero no cuando se aborda la detección de cualquier tipo de fallo. Esta conclusión implica, desde el punto de vista industrial, que, si el interés es la detección del fallo incipiente, es interesante efectuar un seguimiento lo más continuo posible del comportamiento del motor programando revisiones periódicas en las que se baje la carga puesto que los fallos incipientes sí pueden detectarse más fácilmente en carga baja (que no es la carga usual de funcionamiento del motor).

En cuanto a las fuentes de información, se ha visto que no hay influencia significativa del tipo de fuente en el caso de la detección de fallo incipiente. En este caso parece recomendable el uso de la corriente como fuente de información puesto que es una información que se obtiene de forma menos invasiva y más directa. Para el caso de la detección de

cualquier tipo de fallo se ha observado que el uso de la corriente sí parece arrojar resultados mejores que el resto de fuentes de información con lo que puede recomendarse el uso de la corriente como fuente de información para la construcción de las reglas en todos los casos.

Los análisis aquí efectuados dejan también abiertas una serie de preguntas que pueden constituir futuras líneas de trabajo. Por ejemplo, cabe preguntarse si, aunque ya se han obtenido resultados muy buenos con una sola fuente de información, los resultados podrían mejorarse combinando la información obtenida de diferentes fuentes. Para conseguir este objetivo sería importante conseguir un tamaño muestral superior puesto que esta limitación es lo que ha impedido que esta cuestión se aborde en el presente trabajo.

También sería interesante saber si los buenos resultados obtenidos mediante métodos no excesivamente complejos como los considerados aquí para el estado estacionario del motor (funcionamiento normal del motor) pueden obtenerse también en otras situaciones como en el caso del análisis del comportamiento del motor en el estado transitorio (cuando está arrancando) o en situaciones en las que se pretende clasificar el estado del motor en un número superior de clases, situación que, de nuevo, no se ha abordado aquí por no disponer de suficientes observaciones.

Otra situación de interés tiene que ver con la alimentación del motor. Como se ha comentado en 5.1, en este trabajo se ha abordado la situación en la que el motor está alimentado directamente de la red, sin la presencia de variadores que regulen esta alimentación. El estudio de la influencia del variador en la detección de los fallos es otro problema interesante a abordar.

El estudio de fallos en otras partes del motor como, por ejemplo, el estator también puede ser un futuro campo de trabajo.

Parte III

Apéndices

Apéndice A

Gráficos y tablas complementarios

En el presente apéndice se exponen las gráficas y diagramas correspondientes al análisis estadístico detallado proporcionando una perspectiva completa de los resultados obtenidos a través de técnicas de análisis estadístico de los datos. Las gráficas incluyen representaciones como gráficas de dispersión, gráficas de caja y bigotes, y tablas con estadística descriptiva, resultados de pruebas y otros tests. Su objetivo persigue agregar detalle y complementar las conclusiones presentadas en el cuerpo principal de análisis del estudio.

A.1. Análisis exploratorio. Diagramas de caja

En la siguiente sección se muestran las representaciones gráficas de caja y bigotes de las variables que presentan signos de contener valores atípicos o extremos. Estas representaciones visuales permiten observar la distribución de los datos y facilitan la identificación de las observaciones que no se ajustan a la distribución normal de las variables analizadas.

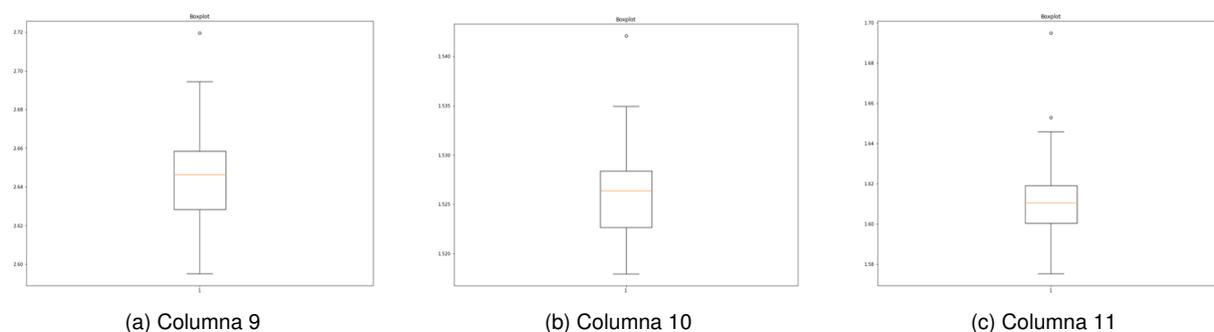


Figura A.1: Gráficas de Caja y Bigotes de Corriente a Carga Alta en fase 1

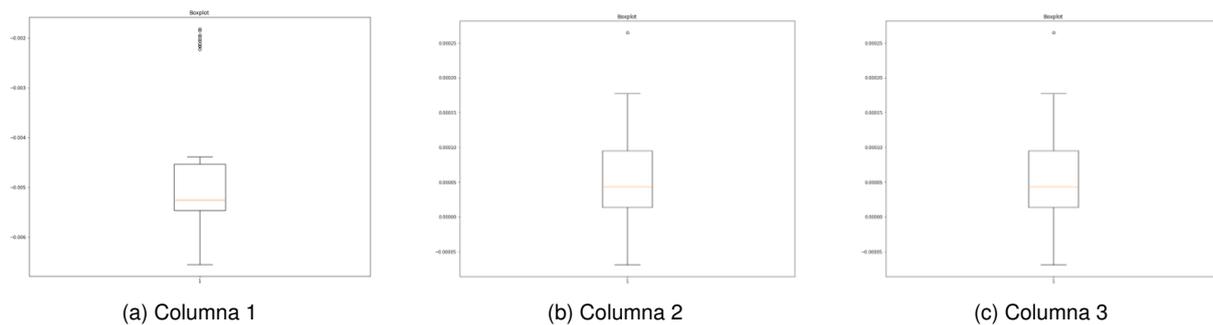


Figura A.2: Gráficas de Caja y Bigotes de Corriente a Carga Alta en fase 3

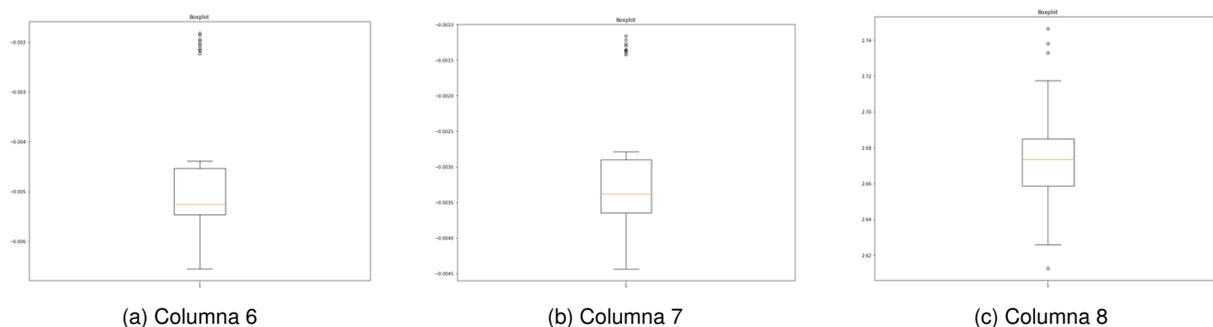


Figura A.3: Gráficas de Caja y Bigotes de Corriente a Carga Alta en fase 3

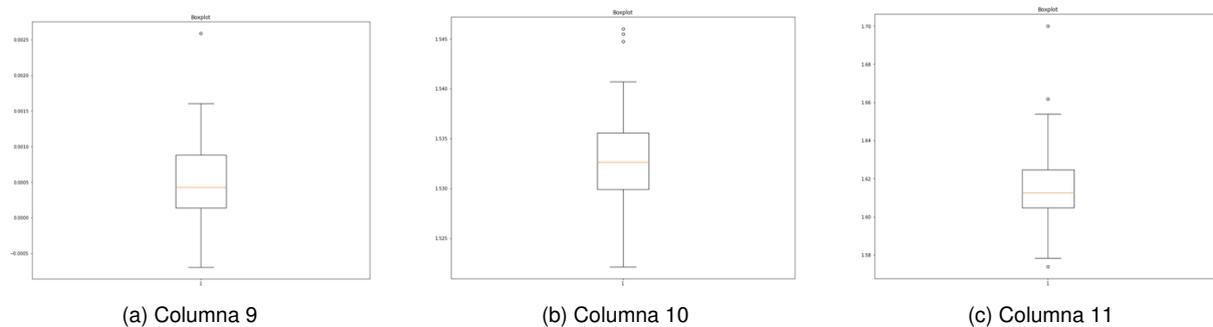


Figura A.4: Gráficas de Caja y Bigotes de Corriente a Carga Alta en fase 3

A.1. Análisis exploratorio. Diagramas de caja

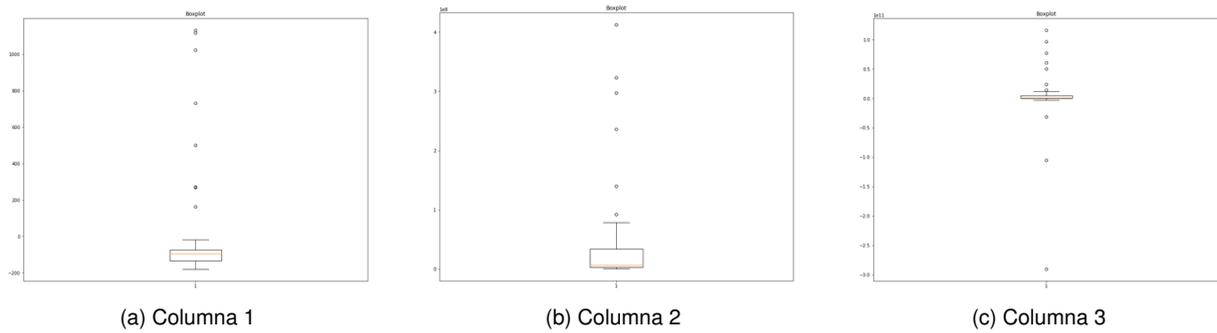


Figura A.5: Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje x

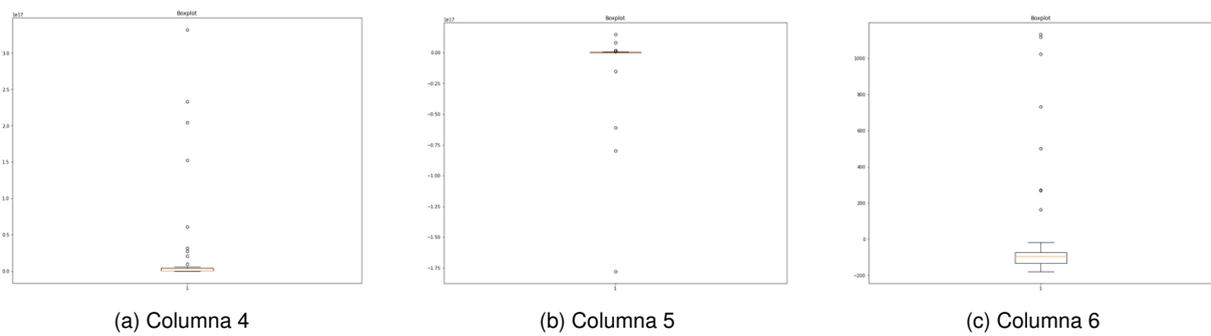


Figura A.6: Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje x

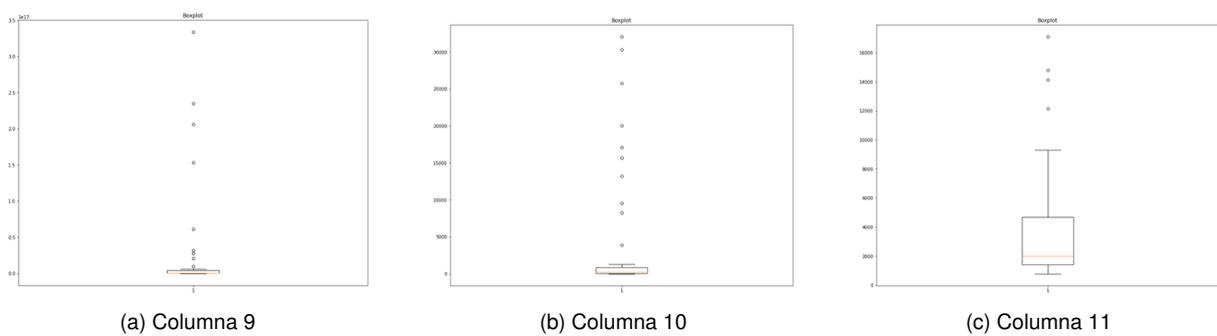


Figura A.7: Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje x

Apéndice A. Gráficos y tablas complementarios

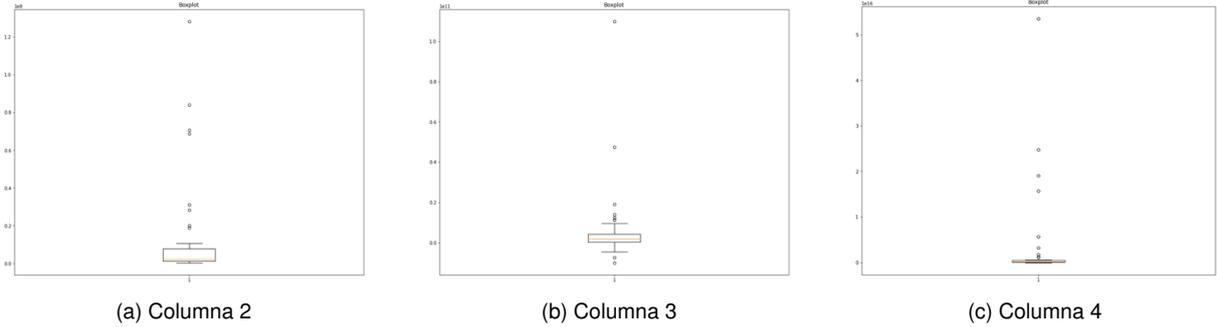


Figura A.8: Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje y

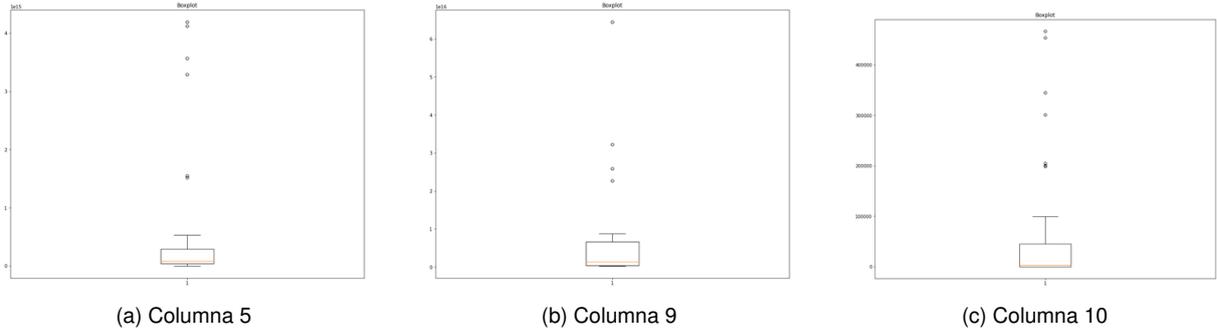


Figura A.9: Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje y

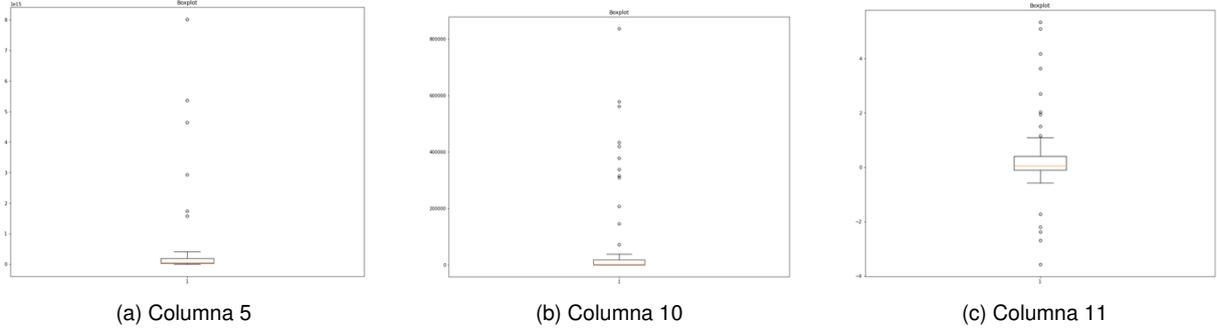


Figura A.10: Gráficas de Caja y Bigotes de Vibraciones a Carga Alta en eje z

A.1. Análisis exploratorio. Diagramas de caja

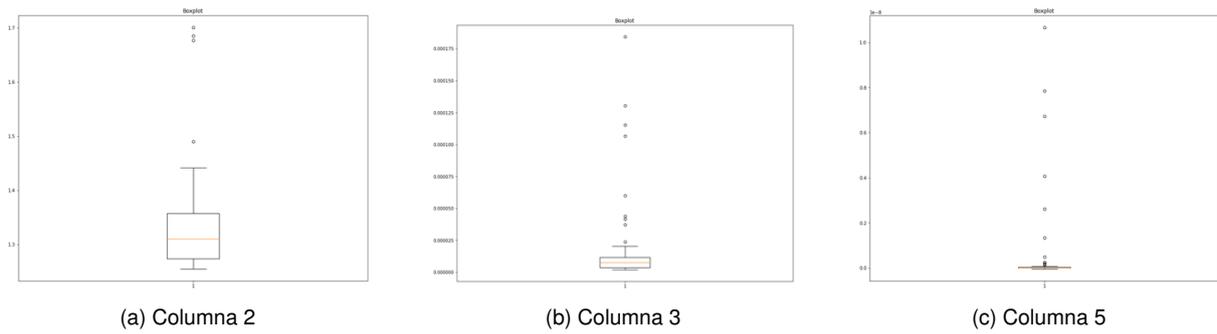


Figura A.11: Gráficas de Caja y Bigotes de Sonido a Carga Alta

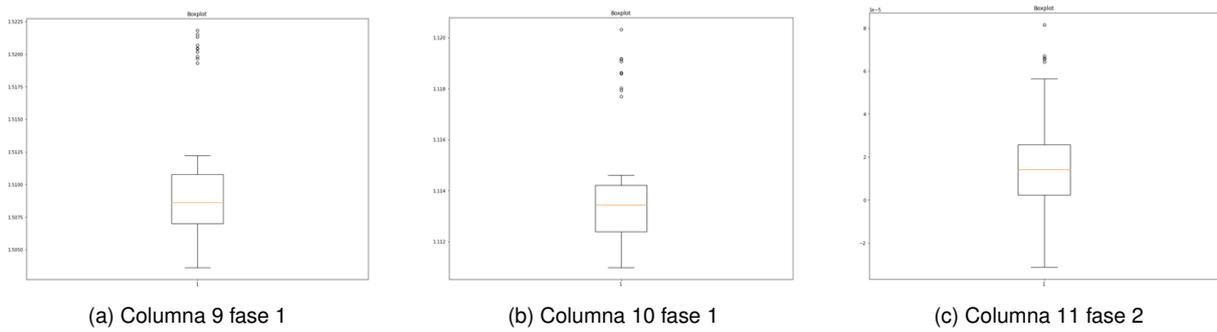


Figura A.12: Gráficas de Caja y Bigotes de Corriente a Carga Baja

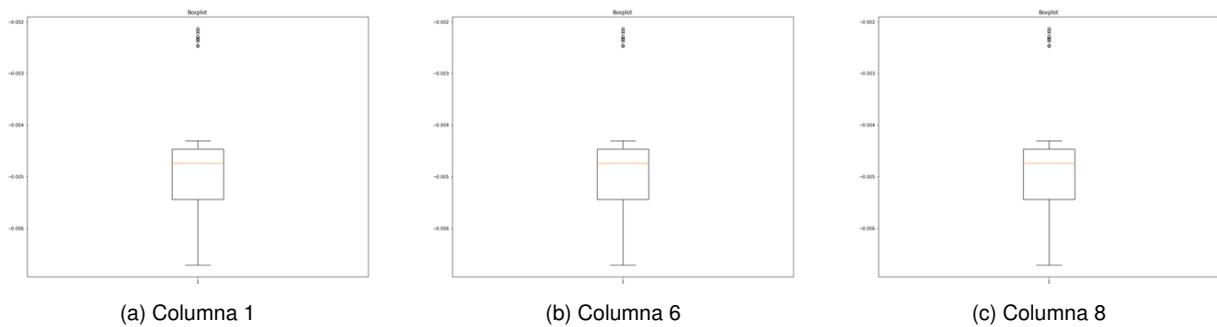


Figura A.13: Gráficas de Caja y Bigotes de Corriente a Carga Baja en fase 3

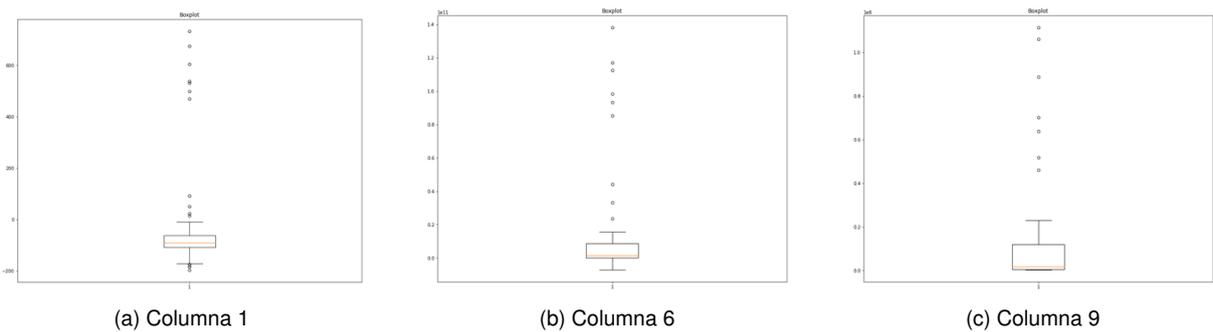


Figura A.14: Gráficas de Caja y Bigotes de Vibraciones a Carga Baja en eje x

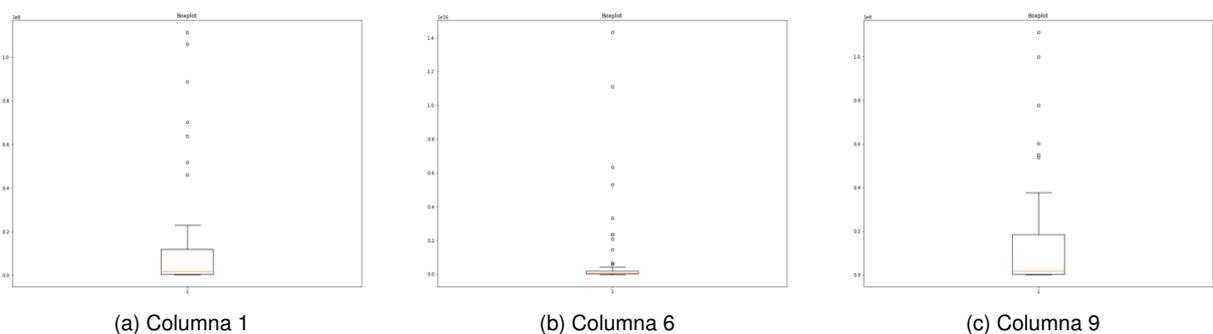


Figura A.15: Gráficas de Caja y Bigotes de Vibraciones a Carga Baja en eje y

A.2. Resultados de los ACP

En esta sección se ofrecen los resultados completos para los diferentes análisis en componentes principales que se han realizado previamente al ajuste de los modelos mediante LDA. Hay que notar que se han efectuado análisis para las dos situaciones consideradas, grupo 1 contra grupo 2, y grupo 1 contra grupo 2+3+4+5+6 para dos niveles de carga y para 5 diferentes tipos de datos lo que da un total de 20 análisis. Para cada análisis se dan las tablas de autovectores y de autovalores correspondientes.

A.2.1. Análisis 1. Grupo 1 contra grupo 2. Resultados ACP.

Autovalores	% de variabilidad
6,529	0,594
3,070	0,279
1,119	0,102
0,270	0,025
0,011	0,001
0,001	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.1: Tabla de Autovalores Corriente en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 2	-0,384	0,078	0,108	0,146	-0,175	-0,194	0,489	0,059	-0,391	-0,591	0,004
Coef 3	-0,107	-0,544	-0,103	0,111	-0,075	0,386	0,120	-0,004	-8,975	0,000	0,707
Coef 4	-0,385	0,079	0,087	0,113	-0,088	0,110	-0,413	0,340	-0,607	0,389	-0,002
Coef 5	0,382	-0,082	-0,129	-0,177	0,083	-0,053	0,273	0,845	-0,037	0,002	0,000
Coef 6	-0,384	0,079	0,108	0,147	-0,176	-0,193	0,489	0,091	0,384	0,592	-0,004
Coef 7	-0,107	-0,544	-0,103	0,111	-0,075	0,386	0,120	-0,004	-8,973	0,000	-0,707
Coef 8	-0,385	0,080	0,086	0,114	-0,088	0,111	-0,412	0,394	0,574	-0,387	0,002
Coef 9	-0,087	-0,550	-0,128	0,113	0,126	-0,765	-0,244	0,009	0,000	0,000	0,000
Coef 10	-0,295	0,010	-0,503	-0,739	-0,328	-0,045	-0,026	-0,059	0,003	0,000	0,000

Cuadro A.2: Tabla de Autovectores Corriente en Carga Alta

Autovalores	% de variabilidad
4,890	0,445
2,718	0,247
1,979	0,180
1,327	0,121
0,055	0,005
0,023	0,002
0,006	0,001
0,002	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.3: Tabla de Autovalores Sonido en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	-0.449	0.072	-0.003	-0.035	-0.004	-0.007	0.080	0.183	-0.672	0.501	-0.223
Coef 2	-0.040	-0.590	0.063	0.164	-0.042	0.067	-0.712	-0.256	-0.201	-0.001	-0.665
Coef 3	0.008	-0.018	-0.536	-0.564	0.013	0.620	-0.091	-0.036	-0.003	0.000	0.004
Coef 4	-0.047	-0.585	0.041	0.173	-0.457	0.269	0.584	-0.021	0.024	0.002	0.156
Coef 5	0.023	0.091	0.482	-0.611	-0.584	-0.148	-0.149	-0.021	0.006	-0.001	0.077
Coef 6	-0.449	0.068	-0.001	-0.027	0.007	-0.013	0.064	-0.054	-0.293	-0.501	0.670
Coef 7	-0.449	0.065	0.001	-0.020	0.018	-0.019	0.045	-0.290	0.086	-0.498	-0.672
Coef 8	-0.450	0.061	0.002	-0.012	0.030	-0.027	0.024	-0.523	0.466	0.500	0.225
Coef 9	-0.002	-0.392	-0.453	-0.351	0.077	-0.704	0.127	0.012	0.005	0.000	-0.036
Coef 10	0.024	-0.318	0.518	-0.355	0.664	0.131	0.213	-0.024	-0.020	0.001	-0.051
Coef 11	-0.433	-0.169	0.037	0.002	0.007	0.048	-0.216	0.733	0.442	-0.002	1.399

Cuadro A.4: Tabla de Autovectores Sonido en Carga Alta

Autovalores	% de variabilidad
6,686	0,608
1,522	0,138
1,486	0,135
0,572	0,052
0,407	0,037
0,211	0,019
0,102	0,009
0,013	0,001
0,001	0,000
0,000	0,000
0,000	0,000

Cuadro A.5: Tabla de Autovalores Vibraciones eje x en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	-0.326	-0.217	-0.049	0.473	-0.339	0.113	-0.588	-0.260	0.280	0.000	0.000
Coef 2	-0.376	-0.044	-0.174	-0.082	0.051	0.054	0.000	-0.211	-0.519	0.705	0.059
Coef 3	-0.326	-0.320	0.135	-0.172	0.424	-0.141	0.312	-0.479	0.467	0.000	-0.001
Coef 4	-0.373	-0.096	-0.168	0.036	0.076	0.164	0.122	0.502	0.150	-0.059	0.705
Coef 5	0.255	-0.415	-0.275	-0.384	0.415	-0.027	-0.589	0.140	-0.016	-0.447	4.495
Coef 6	-0.376	-0.044	-0.174	-0.082	0.051	0.054	0.001	-0.211	-0.520	-0.704	-0.058
Coef 7	0.104	-0.511	0.492	0.501	0.241	-0.123	0.080	0.195	-0.344	0.000	0.000
Coef 8	-0.373	-0.096	-0.168	0.036	0.076	0.164	0.122	0.502	0.149	0.059	-0.705
Coef 9	-0.149	-0.278	0.550	-0.570	-0.473	0.182	-0.071	0.099	-0.031	0.936	-0.071
Coef 10	0.322	-0.263	-0.173	0.083	-0.011	0.826	0.256	-0.206	0.002	-0.612	2.131
Coef 11	0.156	-0.499	-0.461	-0.013	-0.485	-0.420	0.321	0.001	-0.005	1.022	-4.449

Cuadro A.6: Tabla de Autovectores Vibraciones eje x en Carga Alta

Autovalores	% de variabilidad
7,190	0,654
2,705	0,246
0,545	0,050
0,502	0,046
0,044	0,004
0,008	0,001
0,005	0,000
0,001	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.7: Tabla de Autovalores Vibraciones eje y en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.088	-0.446	0.596	-0.647	0.035	-0.077	0.0655	-0.065	-0.002	0.0322352	0.026
Coef 2	-0.335	-0.260	-0.109	0.017	0.125	0.078	0.153	-0.157	-0.422	-0.483	-0.570
Coef 3	0.309	-0.320	-0.007	0.196	-0.516	0.622	0.125	-0.294	0.077	0.000	0.001
Coef 4	0.053	-0.591	-0.216	0.108	-0.075	-0.057	-0.519	0.552	-0.059	-0.021	0.029
Coef 5	0.321	-0.279	-0.217	0.171	-0.195	-0.731	0.213	-0.351	-0.004	-8.198	9.225
Coef 6	-0.341	-0.234	-0.144	0.054	0.123	0.083	0.149	-0.15	-0.456	0.696	0.206
Coef 7	0.344	0.225	0.126	-0.028	-0.145	-0.048	-0.149	0.135	-0.174	0.463	-0.710
Coef 8	-0.336	-0.253	-0.134	0.024	0.156	0.011	0.101	-0.068	0.756	0.255	-0.355
Coef 9	0.355	-0.136	0.067	0.231	0.395	0.100	0.641	0.466	-0.010	-6.492	-0.000
Coef 10	0.360	-0.099	0.053	0.190	0.665	0.137	-0.409	-0.437	0.006	3.379	6.713
Coef 11	0.271	0.051	-0.689	-0.637	0.109	0.154	0.078	-0.010	-0.000	-3.914	-1.411

Cuadro A.8: Tabla de Autovectores Vibraciones eje y en Carga Alta

Autovalores	% de variabilidad
5,310	0,483
3,341	0,304
1,299	0,118
0,655	0,060
0,276	0,025
0,069	0,006
0,036	0,003
0,008	0,001
0,005	0,000
0,000	0,000
0,000	0,000

Cuadro A.9: Tabla de Autovalores Vibraciones eje z en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.300	0.012	0.433	0.456	0.234	0.461	0.066	0.438	0.226	8.691	-5.209
Coef 2	0.370	0.113	0.055	-0.261	0.032	-0.121	-0.094	0.296	-0.410	0.696	-0.124
Coef 3	0.138	-0.635	0.093	-0.011	-0.563	-0.297	0.062	0.326	0.233	0.000	0.001
Coef 4	0.364	0.121	0.095	-0.335	0.025	-0.007	0.019	-0.225	0.423	-0.125	-0.696
Coef 5	-0.306	0.127	0.427	-0.314	-0.053	-0.002	0.767	0.079	-0.106	1.809	0.000
Coef 6	0.370	0.113	0.055	-0.260	0.032	-0.120	-0.094	0.295	-0.410	-0.696	0.126
Coef 7	0.307	-0.293	0.353	0.212	-0.166	0.160	0.057	-0.641	-0.427	0.000	-0.002
Coef 8	0.364	0.121	0.096	-0.334	0.024	-0.006	0.019	-0.228	0.424	0.125	0.696
Coef 9	-0.066	-0.647	-0.018	-0.278	0.706	0.004	0.024	-0.005	-0.019	5.089	0.000
Coef 10	-0.324	-0.083	0.255	-0.449	-0.264	0.572	-0.469	0.064	-0.030	1.245	4.305
Coef 11	-0.235	0.121	0.638	0.102	0.164	-0.563	-0.401	-0.052	0.076	-2.106	-6.239

Cuadro A.10: Tabla de Autovectores Vibraciones eje z en Carga Alta

Autovalores	% de variabilidad
7,196	0,654
2,886	0,262
0,770	0,070
0,092	0,008
0,053	0,005
0,002	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.11: Tabla de Autovalores Corriente eje x en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.341	0.048	-0.344	-0.782	-0.388	0.019	-0.017	-0.002	0.002	-6.690	0.000
Coef 2	0.362	0.111	0.146	0.139	-0.103	-0.006	0.552	-0.003	-0.341	-0.620	0.003
Coef 3	-0.125	0.550	-0.135	0.037	-0.017	-0.402	-0.010	-0.002	0.000	-1.140	0.707
Coef 4	0.362	0.108	0.157	0.156	-0.108	0.014	-0.372	-0.376	-0.633	0.340	-0.001
Coef 5	-0.363	-0.106	-0.146	-0.160	0.108	-0.007	0.338	-0.827	-0.028	0.000	0.000
Coef 6	0.362	0.111	0.147	0.140	-0.102	-0.006	0.552	-0.025	0.338	0.620	0.000
Coef 7	-0.125	0.550	-0.135	0.037	-0.017	-0.402	-0.010	-0.002	0.000	-1.139	-0.707
Coef 8	0.362	0.108	0.159	0.159	-0.107	0.013	-0.371	-0.418	0.606	-0.340	0.001
Coef 9	-0.141	0.542	-0.102	0.031	0.011	0.821	0.013	0.004	0.000	-2.911	0.000
Coef 10	-0.219	0.169	0.847	-0.452	-0.014	-0.026	-0.003	0.010	0.000	-0.342	0.000
Coef 11	0.358	0.102	-0.051	-0.257	0.890	-0.014	-0.012	0.001	0.013	3.123	0.000

Cuadro A.12: Tabla de Autovectores Corriente en Carga Baja

Autovalores	% de variabilidad
7,880	0,716
2,054	0,187
0,783	0,071
0,224	0,020
0,035	0,003
0,014	0,001
0,009	0,001
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.13: Tabla de Autovalores Sonido en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	-0.343	0.158	0.158	0.111	0.072	-0.091	-0.238	0.062	-0.670	0.497	0.223
Coef 2	0.296	0.378	0.132	-0.054	-0.143	0.235	-0.289	-0.766	-0.055	0.000	-0.008
Coef 3	0.205	-0.427	0.555	0.460	0.357	0.355	-0.048	-0.008	0.008	-0.115	0.004
Coef 4	0.284	0.408	0.145	-0.079	-0.200	0.362	-0.387	0.629	0.097	0.001	0.110
Coef 5	-0.223	-0.394	0.501	-0.627	-0.374	0.049	-0.080	-0.036	0.013	-0.047	0.009
Coef 6	-0.343	0.154	0.154	0.116	0.078	-0.111	-0.230	0.016	-0.221	-0.502	-0.670
Coef 7	-0.344	0.151	0.149	0.120	0.085	-0.130	-0.221	-0.030	0.223	-0.498	0.672
Coef 8	-0.344	0.147	0.145	0.125	0.091	-0.150	-0.211	-0.075	0.663	0.503	-0.224
Coef 9	0.333	0.019	0.318	0.367	-0.506	-0.624	0.060	0.048	-0.016	7.735	0.017
Coef 10	0.292	0.296	0.344	-0.425	0.596	-0.372	0.176	0.057	-0.012	-4.471	-0.004
Coef 11	-0.268	0.415	0.302	0.118	-0.184	0.311	0.721	0.002	0.007	0.000	-0.126

Cuadro A.14: Tabla de Autovectores Sonido en Carga Baja

Autovalores	% de variabilidad
6,301	0,573
1,670	0,152
1,362	0,124
0,712	0,065
0,521	0,047
0,229	0,021
0,197	0,018
0,007	0,001
0,001	0,000
0,000	0,000
0,000	0,000

Cuadro A.15: Tabla de Autovalores Vibraciones eje x en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	-0.353	-0.081	0.154	0.378	0.096	0.270	-0.481	-0.623	-0.006	0.001	0.000
Coef 2	-0.393	0.043	-0.010	0.102	-0.138	-0.166	-0.002	0.179	0.504	-0.611	-0.356
Coef 3	-0.228	-0.528	-0.185	-0.362	0.052	-0.466	0.261	-0.463	-0.036	5.924	0.001
Coef 4	-0.393	0.024	0.008	0.099	-0.150	-0.131	-0.025	0.225	-0.493	0.356	-0.611
Coef 5	0.213	-0.211	0.434	0.667	0.153	-0.336	0.374	0.008	-0.012	0.120	-1.558
Coef 6	-0.393	0.043	-0.010	0.102	-0.138	-0.166	-0.002	0.181	0.505	0.610	0.356
Coef 7	-0.059	-0.729	0.071	-0.053	0.306	0.313	-0.219	0.465	0.049	0.000	0.000
Coef 8	-0.393	0.024	0.008	0.099	-0.150	-0.131	-0.025	0.226	-0.494	-0.356	0.611
Coef 9	0.187	-0.338	-0.421	0.328	-0.699	0.245	0.114	-0.057	0.009	-3.986	5.827
Coef 10	0.331	-0.118	0.235	-0.105	-0.336	-0.508	-0.660	0.060	0.011	0.420	1.434
Coef 11	-0.107	-0.078	0.718	-0.344	-0.431	0.302	0.250	-0.089	0.022	-1.607	1.232

Cuadro A.16: Tabla de Autovectores Vibraciones eje x en Carga Baja

Autovalores	% de variabilidad
7,911	0,719
1,837	0,167
0,860	0,078
0,204	0,019
0,143	0,013
0,041	0,004
0,003	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.17: Tabla de Autovalores Vibraciones eje y en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.201	0.172	-0.824	-0.316	-0.386	0.031	0.019	0.019	0.001	0.017	-0.008
Coef 2	0.343	0.188	0.002	0.072	0.170	-0.079	0.014	0.241	0.439	-0.598	0.445
Coef 3	-0.312	0.258	-0.188	0.520	-0.133	-0.713	-0.070	-0.005	-0.023	4.684	0.000
Coef 4	0.314	0.339	-0.001	0.163	0.146	0.088	-0.059	-0.852	0.028	-0.014	-0.035
Coef 5	-0.257	0.476	-0.003	0.440	-0.216	0.648	-0.045	0.212	0.009	0.000	0.201
Coef 6	0.343	0.187	0.025	0.082	0.183	-0.081	0.013	0.244	0.455	0.729	-0.041
Coef 7	-0.343	-0.182	-0.017	-0.069	-0.178	0.069	-0.015	-0.265	0.181	0.289	0.788
Coef 8	0.342	0.191	0.016	0.069	0.175	-0.046	0.011	0.195	-0.752	0.162	0.422
Coef 9	-0.329	0.185	-0.220	-0.101	0.500	0.029	0.740	-0.030	0.000	-1.645	2.787
Coef 10	-0.325	0.196	-0.204	-0.271	0.549	0.031	-0.660	0.050	0.001	3.033	-1.959
Coef 11	-0.119	0.592	0.441	-0.552	-0.300	-0.204	0.066	0.008	0.000	0.083	0.681

Cuadro A.18: Tabla de Autovectores Vibraciones eje y en Carga Baja

Autovalores	% de variabilidad
6,880	0,625
1,818	0,165
1,271	0,116
0,428	0,039
0,351	0,032
0,158	0,014
0,087	0,008
0,004	0,000
0,002	0,000
0,000	0,000
0,000	0,000

Cuadro A.19: Tabla de Autovalores Vibraciones eje z en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.300	0.012	0.433	0.456	0.234	0.461	0.066	0.438	0.226	8.691	-5.209
Coef 2	0.370	0.113	0.055	-0.261	0.032	-0.121	-0.094	0.296	-0.410	0.696	-0.124
Coef 3	0.138	-0.635	0.093	-0.011	-0.563	-0.297	0.062	0.326	0.233	0.000	0.001
Coef 4	0.364	0.121	0.095	-0.335	0.025	-0.007	0.019	-0.225	0.423	-0.125	-0.696
Coef 5	-0.306	0.127	0.427	-0.314	-0.053	-0.002	0.767	0.079	-0.106	1.809	0.000
Coef 6	0.370	0.113	0.055	-0.260	0.032	-0.120	-0.094	0.295	-0.410	-0.696	0.126
Coef 7	0.307	-0.293	0.353	0.212	-0.166	0.160	0.057	-0.641	-0.427	0.000	-0.002
Coef 8	0.364	0.121	0.096	-0.334	0.024	-0.006	0.019	-0.228	0.424	0.125	0.696
Coef 9	-0.066	-0.647	-0.018	-0.278	0.706	0.004	0.024	-0.005	-0.019	5.089	0.000
Coef 10	-0.324	-0.083	0.255	-0.449	-0.264	0.572	-0.469	0.064	-0.030	1.245	4.305
Coef 11	-0.235	0.121	0.638	0.102	0.164	-0.563	-0.401	-0.052	0.076	-2.106	-6.239

Cuadro A.20: Tabla de Autovectores Vibraciones eje z en Carga Baja

A.2.2. Análisis 2. Grupo 1 contra grupo 2+3+4+5+6. Resultados ACP.

Autovalores	% de variabilidad
5,906	0,537
3,019	0,274
1,488	0,135
0,535	0,049
0,049	0,004
0,001	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.21: Tabla de Autovalores Corriente en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.147	0.195	-0.518	0.819	0.042	0.016	-0.003	0.003	0.001	-5.303	0.000
Coef 2	0.403	-0.073	-0.107	-0.120	-0.102	0.052	-0.538	0.061	-0.358	-0.610	-0.014
Coef 3	0.104	0.553	0.073	-0.096	-0.004	-0.410	-0.029	-0.006	0.000	-3.895	0.707
Coef 4	0.405	-0.077	-0.066	-0.088	-0.117	-0.025	0.407	0.378	-0.606	0.358	0.008
Coef 5	-0.399	0.073	0.148	0.141	0.051	-0.001	-0.290	0.841	0.008	0.001	0.000
Coef 6	0.403	-0.073	-0.106	-0.121	-0.102	0.051	-0.539	0.053	0.357	0.610	0.014
Coef 7	0.104	0.553	0.073	-0.096	-0.004	-0.410	-0.029	-0.006	0.000	-3.927	-0.707
Coef 8	0.405	-0.077	-0.065	-0.090	-0.117	-0.025	0.404	0.367	0.615	-0.357	-0.008
Coef 9	0.079	0.559	0.091	-0.106	0.024	0.810	0.064	0.012	0.001	3.356	0.000
Coef 10	0.160	-0.041	0.696	0.449	-0.526	0.012	-0.021	-0.094	-0.001	0.000	0.000
Coef 11	0.335	-0.093	0.418	0.184	0.819	-0.015	-0.017	0.008	8.524	-1.219	0.000

Cuadro A.22: Tabla de Autovectores Corriente en Carga Alta

Autovalores	% de variabilidad
5,633	0,512
2,252	0,205
1,597	0,145
1,056	0,096
0,300	0,027
0,126	0,011
0,031	0,003
0,004	0,000
0,002	0,000
0,000	0,000
0,000	0,000

Cuadro A.23: Tabla de Autovalores Sonido en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	-0.408	0.156	0.013	0.074	-0.012	-0.118	-0.200	-0.131	-0.652	0.507	0.226
Coef 2	0.337	0.307	-0.043	0.323	0.207	0.417	-0.047	0.620	-0.288	0.003	3.546
Coef 3	-0.154	-0.248	-0.542	0.353	-0.633	0.185	0.233	0.072	-0.040	0.000	0.584
Coef 4	0.301	0.338	-0.105	0.434	-0.123	0.056	-0.528	-0.510	0.195	-0.001	0.164
Coef 5	0.269	0.471	-0.039	0.015	-0.293	-0.713	0.269	0.190	-0.040	0.000	0.860
Coef 6	-0.406	0.166	0.013	0.087	-0.003	-0.089	-0.182	0.028	-0.247	-0.495	-0.672
Coef 7	-0.404	0.175	0.012	0.100	0.007	-0.060	-0.165	0.188	0.169	-0.504	0.669
Coef 8	-0.402	0.184	0.012	0.111	0.016	-0.030	-0.147	0.348	0.595	0.495	-0.222
Coef 9	0.005	-0.121	-0.727	0.061	0.601	-0.301	-0.029	-0.018	0.016	5.920	0.008
Coef 10	0.072	0.247	-0.384	-0.734	-0.256	0.195	-0.362	0.111	-0.020	0.000	0.179
Coef 11	-0.199	0.562	-0.124	-0.072	0.166	0.354	0.582	-0.356	0.073	0.000	-1.416

Cuadro A.24: Tabla de Autovectores Sonido en Carga Alta

Autovalores	% de variabilidad
7,230	0,657
1,209	0,110
1,132	0,103
0,747	0,068
0,617	0,056
0,035	0,003
0,018	0,002
0,009	0,001
0,001	0,000
0,000	0,000
0,000	0,000

Cuadro A.25: Tabla de Autovalores Vibraciones eje x en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.353	-0.140	-0.103	0.078	-0.245	-0.667	-0.505	-0.224	0.171	-0.001	0.001
Coef 2	0.365	-0.089	-0.017	-0.007	-0.167	0.416	-0.311	0.245	-0.042	-0.019	-0.706
Coef 3	-0.232	-0.548	-0.134	0.141	-0.583	0.205	0.174	-0.408	-0.171	-0.002	0.001
Coef 4	0.370	0.032	-0.023	0.000	-0.072	0.065	0.400	-0.120	0.427	0.703	-0.018
Coef 5	-0.344	-0.277	-0.046	0.082	-0.246	-0.186	0.059	0.708	0.442	0.004	-0.001
Coef 6	0.365	-0.088	-0.017	-0.007	-0.167	0.413	-0.308	0.249	-0.047	0.020	0.707
Coef 7	0.366	-0.030	-0.049	0.031	-0.161	-0.347	0.444	0.363	-0.622	0.009	-0.005
Coef 8	0.370	0.032	-0.023	0.000	-0.073	0.061	0.402	-0.109	0.417	-0.710	0.021
Coef 9	-0.039	-0.015	-0.796	-0.601	0.061	0.006	0.003	0.014	0.002	1.389	-0.015
Coef 10	-0.080	0.500	-0.520	0.667	-0.150	0.077	-0.024	0.020	-0.006	0.142	-0.719
Coef 11	0.141	-0.579	-0.249	0.401	0.649	0.020	0.024	0.011	-0.005	-3.247	0.607

Cuadro A.26: Tabla de Autovectores Vibraciones eje x en Carga Alta

Autovalores	% de variabilidad
5,813	0,528
1,917	0,174
1,901	0,173
0,815	0,074
0,301	0,027
0,167	0,015
0,071	0,006
0,012	0,001
0,003	0,000
0,000	0,000
0,000	0,000

Cuadro A.27: Tabla de Autovalores Vibraciones eje y en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.046	-0.607	-0.369	0.076	0.148	-0.206	-0.185	0.260	0.188	0.523	0.107
Coef 2	-0.390	-0.149	-0.105	0.062	0.292	-0.301	-0.247	0.357	-0.391	-0.540	-0.041
Coef 3	-0.342	-0.191	0.086	0.252	-0.755	0.249	0.075	0.370	-0.038	0.003	-0.007
Coef 4	-0.389	-0.206	-0.046	0.171	-0.086	-0.190	-0.006	-0.649	-0.199	0.180	-0.488
Coef 5	-0.352	-0.236	-0.056	-0.125	0.428	0.753	0.223	-0.026	0.013	0.000	0.171
Coef 6	-0.338	0.360	0.208	-0.012	0.110	-0.071	-0.045	0.072	-0.454	0.567	0.401
Coef 7	0.345	-0.353	-0.164	-0.002	-0.189	0.145	-0.007	-0.349	-0.535	-0.180	0.482
Coef 8	-0.409	-0.029	0.014	0.144	-0.007	-0.207	0.053	-0.315	0.512	-0.219	0.597
Coef 9	0.087	-0.225	0.652	0.113	0.077	0.187	-0.668	-0.087	0.096	0.000	0.003
Coef 10	0.127	-0.349	0.563	0.160	0.181	-0.265	0.631	0.110	-0.079	8.613	-0.002
Coef 11	-0.170	-0.218	0.165	-0.907	-0.214	-0.164	-0.006	-0.011	0.026	-4.817	0.000

Cuadro A.28: Tabla de Autovectores Vibraciones eje y en Carga Alta

Autovalores	% de variabilidad
7,501	0,682
1,436	0,131
0,951	0,086
0,675	0,061
0,345	0,031
0,076	0,007
0,013	0,001
0,003	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.29: Tabla de Autovalores Vibraciones eje z en Carga Alta

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.329	0.036	0.157	-0.366	0.444	0.017	0.718	-0.138	0.007	-0.007	0.000
Coef 2	-0.348	0.069	0.026	-0.198	0.369	0.373	-0.212	-0.119	0.045	-0.096	-0.699
Coef 3	0.360	0.025	0.069	-0.147	0.160	-0.031	-0.347	-0.008	-0.830	-0.088	0.001
Coef 4	-0.365	-0.019	-0.026	0.009	0.029	-0.020	0.256	0.519	-0.195	-0.694	0.098
Coef 5	-0.343	-0.043	0.083	-0.207	0.264	-0.837	-0.145	-0.194	0.038	0.002	-1.397
Coef 6	-0.348	0.069	0.025	-0.196	0.366	0.372	-0.208	-0.128	0.024	0.101	0.701
Coef 7	0.352	0.030	0.096	-0.210	0.287	-0.077	-0.338	0.682	0.380	0.101	-0.002
Coef 8	-0.365	-0.019	-0.030	0.020	0.010	-0.014	0.270	0.421	-0.352	0.693	-0.101
Coef 9	0.045	-0.438	-0.779	-0.440	-0.077	0.014	-0.007	-0.003	0.007	0.000	-0.187
Coef 10	0.059	-0.696	0.038	0.554	0.450	0.025	0.000	-0.018	-0.004	0.000	0.232
Coef 11	-0.087	-0.556	0.586	-0.423	-0.383	0.112	-0.041	0.005	0.008	0.001	0.190

Cuadro A.30: Tabla de Autovectores Vibraciones eje z en Carga Alta

Autovalores	% de variabilidad
7,175	0,652
2,550	0,232
0,685	0,062
0,515	0,047
0,072	0,007
0,003	0,000
0,001	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000
0,000	0,000

Cuadro A.31: Tabla de Autovalores Corriente en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.253	0.153	0.487	-0.782	0.252	-0.010	0.007	-0.001	0.002	-2.328	0.000
Coef 2	0.350	-0.209	-0.080	-0.034	-0.161	-0.087	-0.543	0.023	-0.324	-0.626	-0.002
Coef 3	-0.227	-0.480	0.245	-0.014	-0.017	-0.390	0.070	0.003	0.128	0.000	-0.707
Coef 4	0.348	-0.219	-0.096	-0.037	-0.149	0.073	0.339	-0.406	-0.635	0.330	0.001
Coef 5	-0.352	0.206	0.063	0.016	0.160	-0.093	-0.381	-0.806	-0.010	0.003	0.000
Coef 6	0.350	-0.210	-0.081	-0.033	-0.161	-0.085	-0.538	0.017	0.333	0.626	0.000
Coef 7	-0.227	-0.480	0.245	-0.014	-0.017	-0.390	0.070	0.003	0.128	0.000	0.707
Coef 8	0.348	-0.220	-0.098	-0.035	-0.149	0.075	0.349	-0.428	0.617	-0.327	-0.001
Coef 9	-0.252	-0.446	0.228	-0.037	0.004	0.813	-0.153	-0.008	2.067	0.000	0.000
Coef 10	-0.218	-0.254	-0.739	-0.467	0.349	-0.023	-0.002	0.035	0.000	0.000	0.000
Coef 11	0.334	-0.156	0.093	0.403	0.833	0.004	-0.010	-0.001	0.000	2.908	0.000

Cuadro A.32: Tabla de Autovectores Corriente en Carga Baja

Autovalores	% de variabilidad
5,898	0,536
3,104	0,282
1,064	0,097
0,520	0,047
0,320	0,029
0,078	0,007
0,011	0,001
0,004	0,000
0,001	0,000
0,000	0,000
0,000	0,000

Cuadro A.33: Tabla de Autovalores Sonido en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	-0.337	0.321	-0.023	0.061	-0.079	-0.095	-0.216	0.646	-0.021	0.499	-0.224
Coef 2	0.367	0.148	-0.140	-0.197	0.477	-0.510	0.412	0.248	-0.257	-0.004	4.932
Coef 3	-0.246	-0.217	-0.314	-0.836	-0.294	-0.083	0.020	0.006	-0.011	0.000	0.273
Coef 4	0.347	0.273	-0.135	0.008	-0.301	-0.356	-0.044	-0.047	0.751	0.009	0.000
Coef 5	0.327	0.291	-0.118	0.071	-0.513	-0.187	-0.302	-0.191	-0.603	-0.006	8.291
Coef 6	-0.333	0.331	-0.031	0.054	-0.060	-0.077	0.003	0.256	-0.011	-0.501	0.670
Coef 7	-0.328	0.340	-0.039	0.048	-0.041	-0.058	0.224	-0.133	-0.013	-0.499	-0.671
Coef 8	-0.323	0.349	-0.046	0.041	-0.022	-0.038	0.445	-0.517	-0.026	0.501	0.225
Coef 9	-0.008	-0.154	-0.898	0.355	0.025	0.181	0.083	0.065	-0.001	0.000	0.831
Coef 10	0.355	0.242	0.081	-0.145	-0.287	0.624	0.477	0.298	-0.007	-0.002	4.047
Coef 11	0.120	0.488	-0.180	-0.315	0.487	0.359	-0.455	-0.196	0.064	0.002	-2.734

Cuadro A.34: Tabla de Autovectores Sonido en Carga Baja

Autovalores	% de variabilidad
7,320	0,665
1,393	0,127
0,881	0,080
0,731	0,066
0,527	0,048
0,084	0,008
0,036	0,003
0,027	0,002
0,001	0,000
0,000	0,000
0,000	0,000

Cuadro A.35: Tabla de Autovalores Vibraciones x en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.358	0.009	0.130	-0.043	0.218	0.022	-0.230	0.859	0.119	0.001	0.001
Coef 2	0.365	-0.025	0.031	0.044	-0.018	0.395	-0.357	-0.233	-0.159	0.009	-0.707
Coef 3	0.309	-0.030	0.434	-0.065	0.449	-0.519	-0.134	-0.373	0.289	0.002	0.001
Coef 4	0.363	0.012	-0.172	-0.021	-0.012	0.180	0.437	-0.057	0.343	-0.703	-0.009
Coef 5	-0.277	-0.015	0.619	-0.014	0.373	0.516	0.352	0.021	-0.100	-1.753	0.000
Coef 6	0.365	-0.025	0.031	0.044	-0.017	0.393	-0.356	-0.233	-0.164	-0.011	0.707
Coef 7	0.365	0.012	-0.053	-0.055	0.141	-0.265	0.392	0.038	-0.785	-0.009	-0.004
Coef 8	0.363	0.012	-0.172	-0.021	-0.012	0.177	0.436	-0.057	0.327	0.711	0.011
Coef 9	0.009	-0.646	0.082	-0.730	-0.207	0.017	-0.001	0.007	-0.003	1.325	0.119
Coef 10	-0.114	0.572	-0.300	-0.642	0.360	0.112	-0.108	-0.067	-0.009	-9.779	-0.969
Coef 11	0.154	0.503	0.501	-0.205	-0.647	-0.084	0.061	0.026	0.005	5.638	0.049

Cuadro A.36: Tabla de Autovectores Vibraciones x en Carga Baja

Autovalores	% de variabilidad
6,256	0,569
2,264	0,206
1,574	0,143
0,686	0,062
0,146	0,013
0,044	0,004
0,020	0,002
0,008	0,001
0,002	0,000
0,000	0,000
0,000	0,000

Cuadro A.37: Tabla de Autovalores Vibraciones y en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	-0.010	-0.507	0.499	-0.062	-0.395	-0.096	0.173	-0.197	0.001	0.446	0.240
Coef 2	0.384	-0.111	0.083	-0.027	-0.500	-0.201	0.270	0.365	0.098	-0.532	-0.209
Coef 3	0.362	-0.234	0.046	0.138	0.491	-0.109	0.379	0.049	-0.627	0.006	-0.016
Coef 4	0.376	-0.190	0.049	0.172	-0.066	0.148	-0.620	0.240	-0.089	0.328	-0.455
Coef 5	0.366	-0.221	0.053	0.163	0.440	0.035	0.141	-0.128	0.747	0.003	0.005
Coef 6	0.325	0.289	-0.306	0.024	-0.117	-0.096	0.093	0.458	0.072	0.440	0.527
Coef 7	-0.315	-0.324	0.280	0.010	0.277	-0.011	-0.305	0.573	0.029	-0.277	0.377
Coef 8	0.394	-0.039	-0.046	0.138	-0.147	0.194	-0.377	-0.427	-0.158	-0.376	0.522
Coef 9	-0.158	-0.390	-0.534	0.207	-0.049	-0.668	-0.170	-0.131	0.024	0.000	0.797
Coef 10	-0.180	-0.424	-0.462	0.212	-0.169	0.648	0.262	0.116	-0.011	8.425	0.000
Coef 11	0.170	-0.265	-0.254	-0.901	0.111	0.056	-0.091	-0.016	0.000	0.000	0.000

Cuadro A.38: Tabla de Autovectores Vibraciones y en Carga Baja

Autovalores	% de variabilidad
7,351	0,668
1,399	0,127
0,903	0,090
0,995	0,082
0,300	0,027
0,033	0,003
0,013	0,001
0,007	0,001
0,001	0,000
0,000	0,000
0,000	0,000

Cuadro A.39: Tabla de Autovalores Vibraciones z en Carga Baja

	v (v1)	v (v2)	v (v3)	v (v4)	v (v5)	v (v6)	v (v7)	v (v8)	v (v9)	v (v10)	v (v11)
Coef 1	0.312	0.149	0.482	-0.099	-0.296	0.341	0.381	0.517	-0.147	-0.009	-0.001
Coef 2	-0.337	0.152	0.313	-0.056	-0.357	-0.216	0.109	-0.277	0.038	-0.093	-0.700
Coef 3	0.360	0.053	0.196	-0.045	-0.117	-0.016	-0.523	-0.287	-0.673	-0.074	-0.001
Coef 4	-0.367	-0.004	0.041	0.011	-0.099	0.014	-0.430	0.441	0.052	-0.680	0.091
Coef 5	-0.363	-0.047	-0.004	0.029	0.038	0.865	-0.004	-0.329	-0.084	-0.008	0.000
Coef 6	-0.338	0.151	0.311	-0.055	-0.355	-0.218	0.112	-0.276	0.013	0.092	0.702
Coef 7	0.354	0.064	0.245	-0.059	-0.174	0.176	-0.464	-0.143	0.706	0.105	0.001
Coef 8	-0.368	-0.007	0.030	0.014	-0.087	0.000	-0.393	0.417	-0.125	0.709	-0.095
Coef 9	0.037	-0.009	0.188	0.981	-0.009	-0.018	0.009	-0.005	0.005	0.000	1.132
Coef 10	0.086	-0.746	-0.243	0.031	-0.611	0.018	0.047	-0.005	-0.013	-0.001	-2.577
Coef 11	-0.095	-0.605	0.614	-0.117	0.473	-0.096	0.003	-0.028	0.017	0.001	0.258

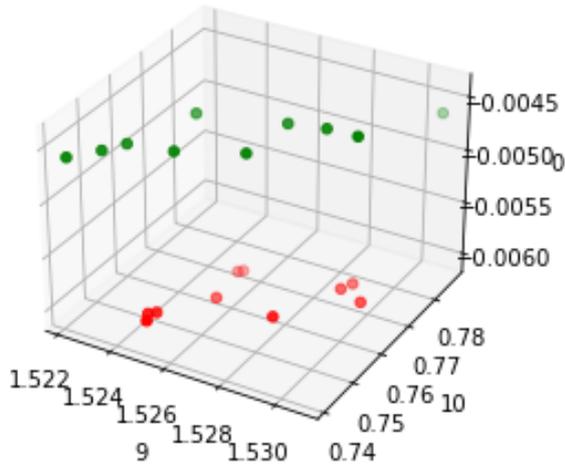
Cuadro A.40: Tabla de Autovectores Vibraciones z en Carga Baja

A.3. Resultados gráficos adicionales del Análisis 1

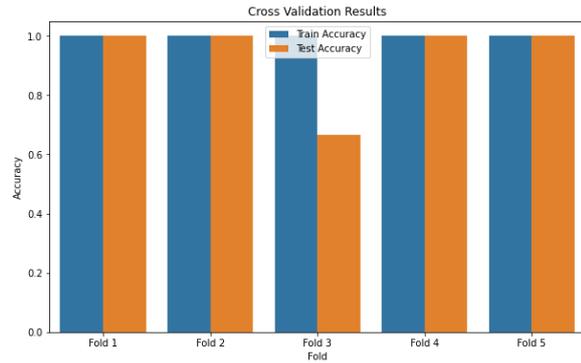
Como ya se ha comentado en el texto principal, debido a la gran cantidad de gráficos generados por cada análisis realizado, se ha decidido adoptar una metodología selectiva para la presentación de resultados. En este anexo se ofrecen los resultados gráficos del análisis 1 correspondientes a los árboles de decisión en carga alta y los resultados en carga baja tanto para el LDA como para los árboles de decisión.

A.3.1. Resultados en Carga Alta

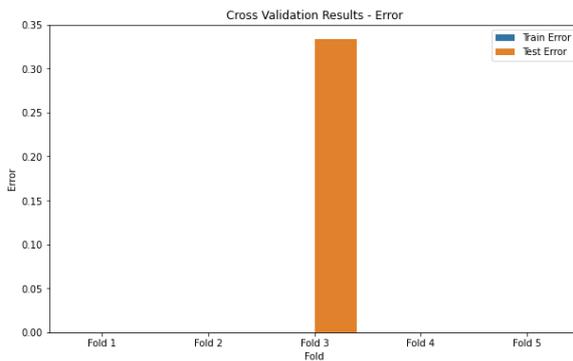
AD. Datos de Corriente



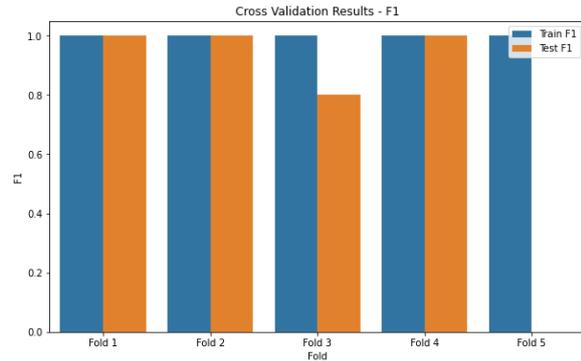
(a) Figura 3D



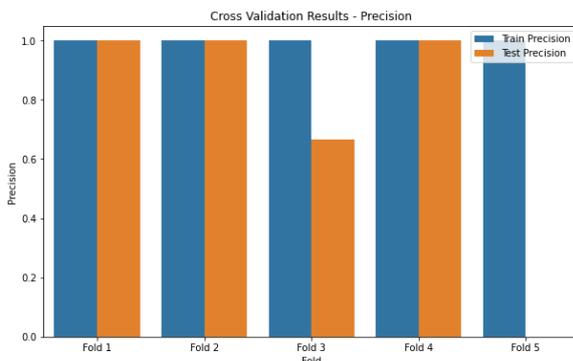
(b) Gráfico de Barras Accuracy



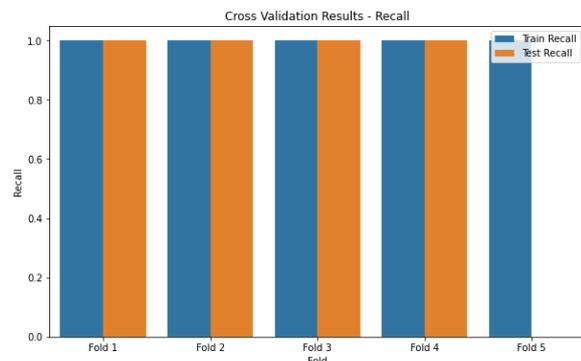
(c) Gráfico de Barras Error



(d) Gráfico de Barras F1



(e) Gráfico de Barras Precision



(f) Gráfico de Barras Recall

Figura A.16: Métricas Análisis 1-CA_AD_Corriente

AD. Datos de Sonido

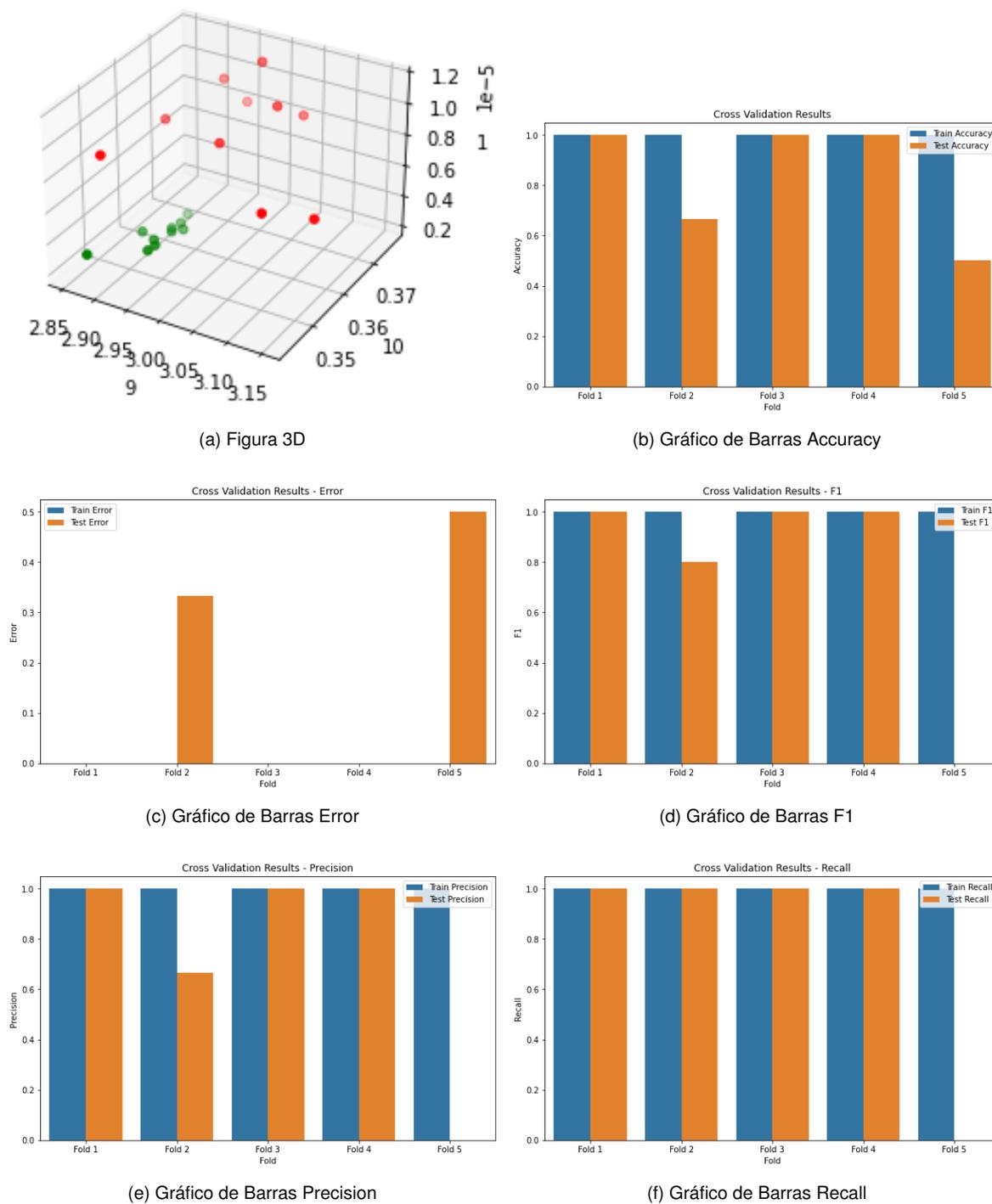
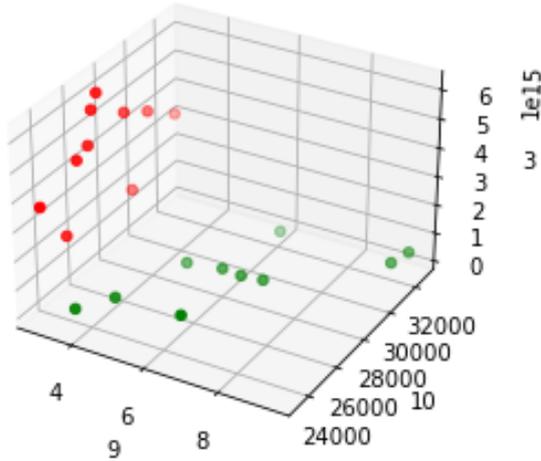
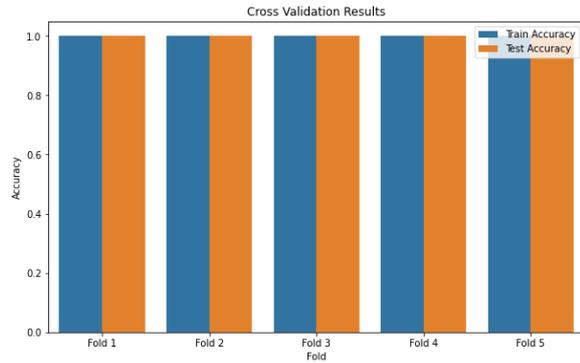


Figura A.17: Métricas Análisis 1-CA_AD_Sonido

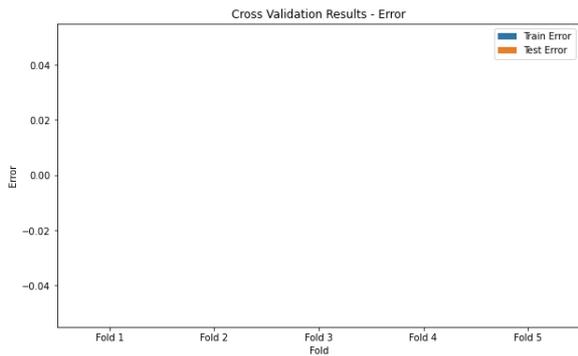
AD. Datos de Vibraciones



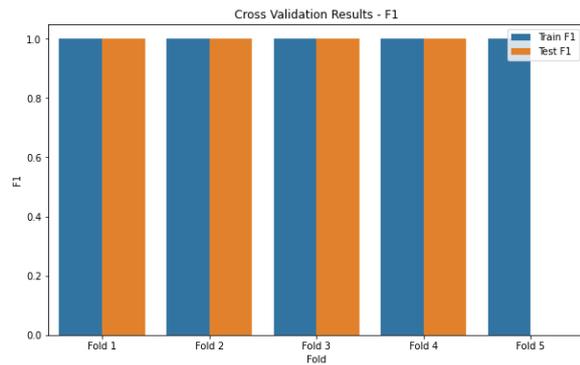
(a) Figura 3D



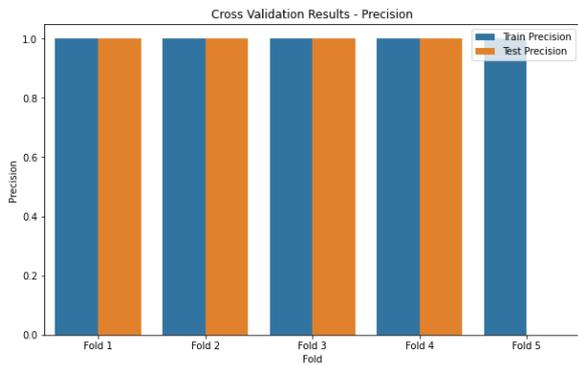
(b) Gráfico de Barras Accuracy



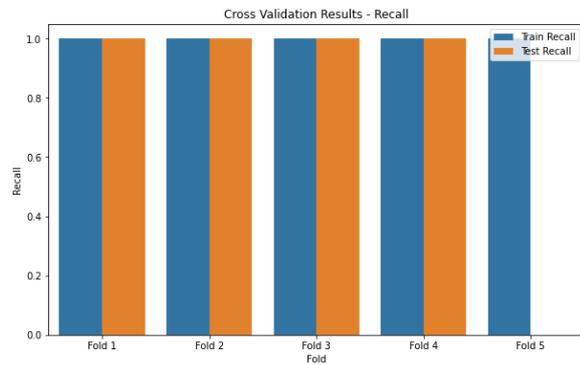
(c) Gráfico de Barras Error



(d) Gráfico de Barras F1

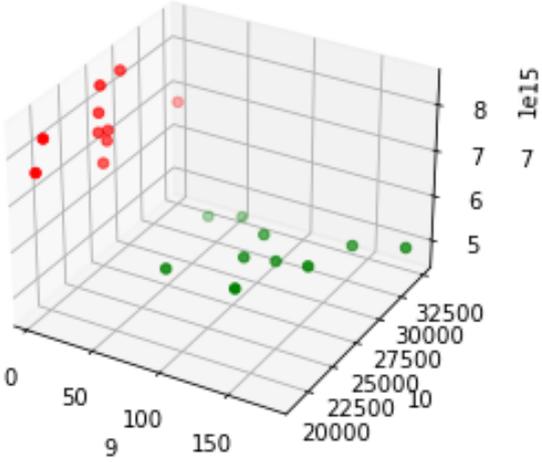


(e) Gráfico de Barras Precision

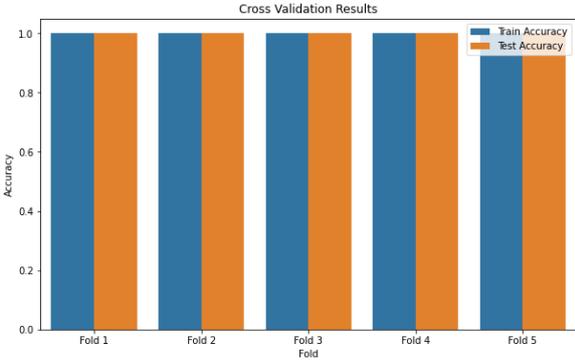


(f) Gráfico de Barras Recall

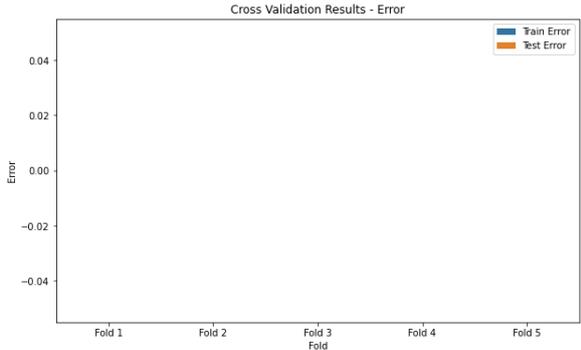
Figura A.18: Métricas Análisis 1-CA_AD_Vibración eje x



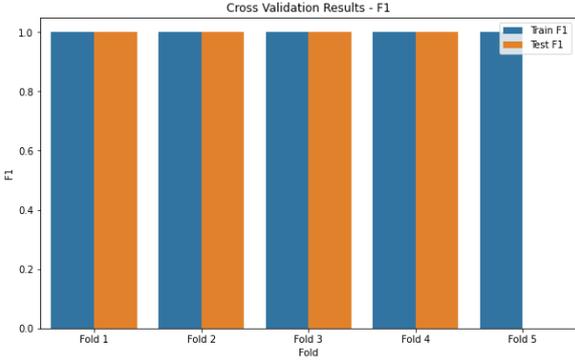
(a) Figura 3D



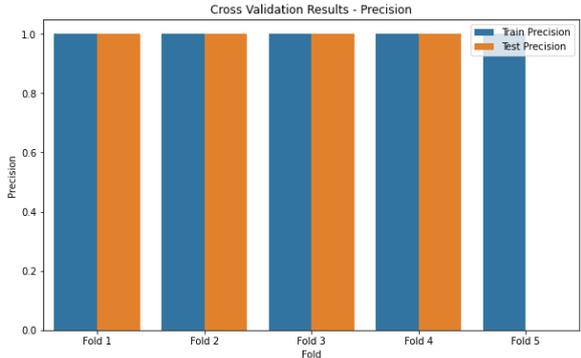
(b) Gráfico de Barras Accuracy



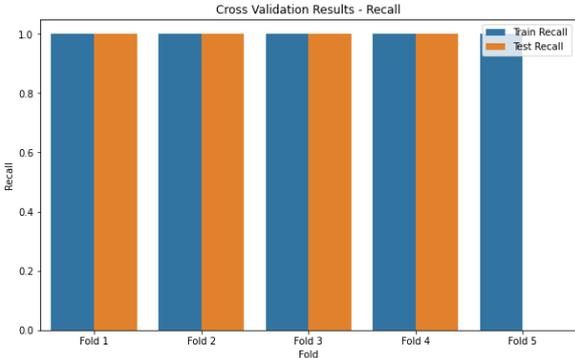
(c) Gráfico de Barras Error



(d) Gráfico de Barras F1



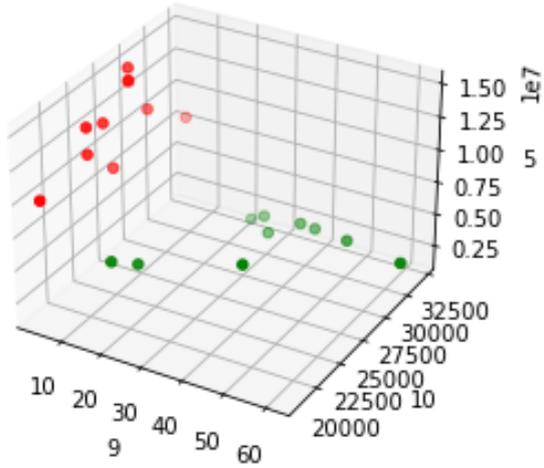
(e) Gráfico de Barras Precision



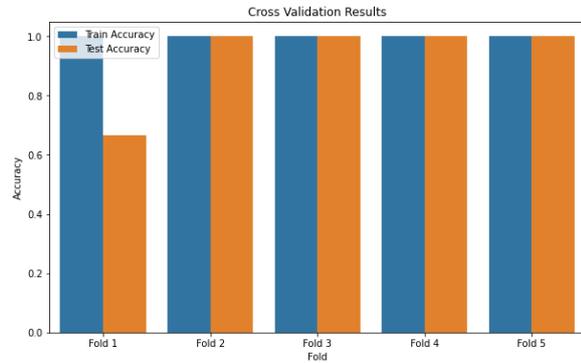
(f) Gráfico de Barras Recall

Figura A.19: Métricas Análisis6-CA_AD_Vibración eje y

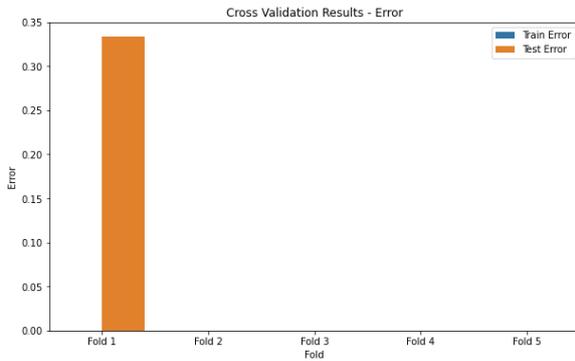
A.3. Resultados gráficos adicionales del Análisis 1



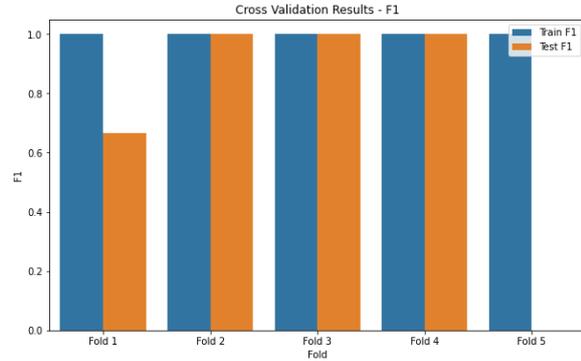
(a) Figura 3D



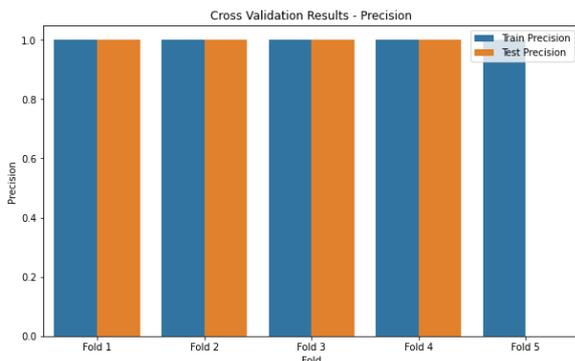
(b) Gráfico de Barras Accuracy



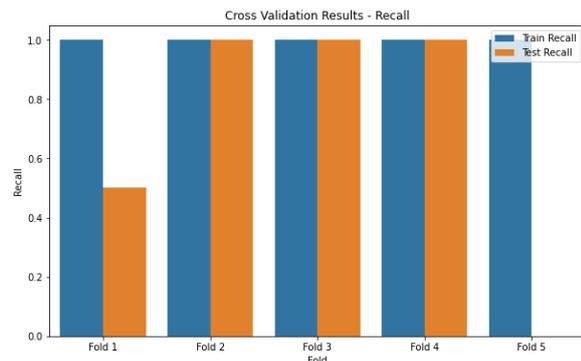
(c) Gráfico de Barras Error



(d) Gráfico de Barras F1



(e) Gráfico de Barras Precisión



(f) Gráfico de Barras Recall

Figura A.20: Métricas Análisis 1-CA_AD_Vibración eje z

A.3.2. Resultados en Carga Baja

LDA. Datos de Corriente

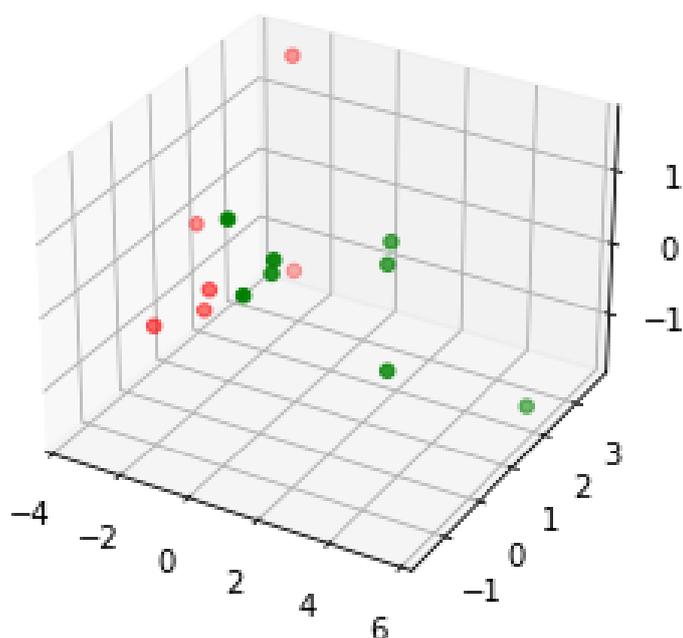


Figura A.21: Gráfico 3D. Análisis 1-CB_LDA_Corriente

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1º Fold	1	1	1	1	1	1	1	1	0	0
2º Fold	1	1	1	1	1	1	1	1	0	0
3º Fold	1	1	1	1	1	1	1	1	0	0
4º Fold	1	1	1	1	1	1	1	1	0	0
5º Fold	1	1	1	0	1	0	1	0	0	0

Cuadro A.41: Métricas. Análisis 1-CB_LDA_Corriente

LDA. Datos de Sonido

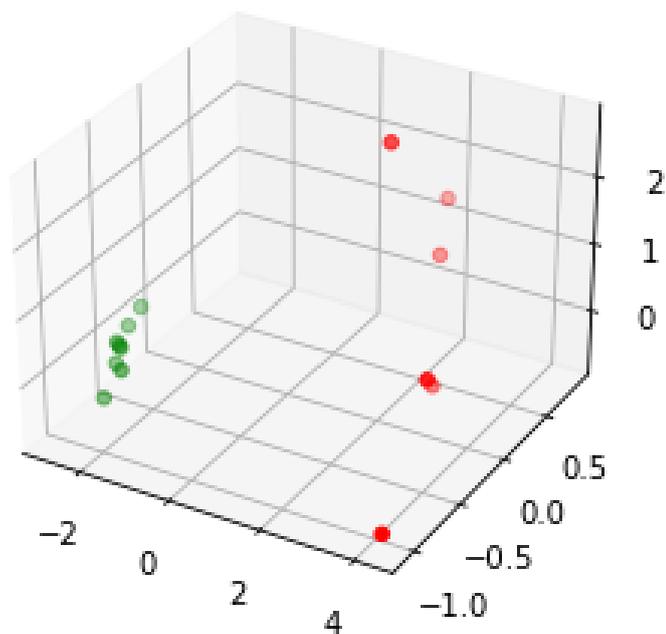


Figura A.22: Gráfico 3D. Análisis 1-CB_LDA_Sonido

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1º Fold	1	1	1	1	1	1	1	1	0	0
2º Fold	1	1	1	1	1	1	1	1	0	0
3º Fold	1	1	1	1	1	1	1	1	0	0
4º Fold	1	1	1	1	1	1	1	1	0	0
5º Fold	1	1	1	0	1	0	1	0	0	0

Cuadro A.42: Métricas. Análisis 1-CB_LDA_Sonido

LDA. Datos de Vibraciones

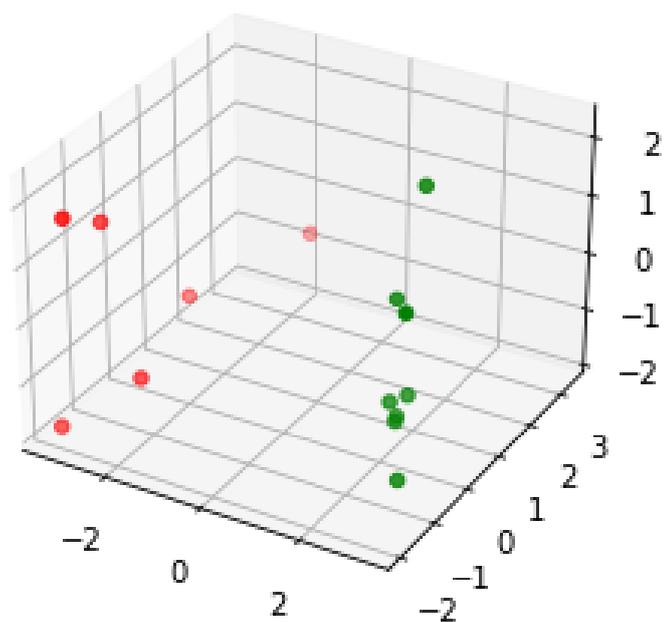


Figura A.23: Gráfico 3D. Análisis 1-CB_LDA_Vibraciones eje x

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1º Fold	1	1	1	1	1	1	1	1	0	0
2º Fold	1	1	1	1	1	1	1	1	0	0
3º Fold	1	1	1	1	1	1	1	1	0	0
4º Fold	1	1	1	1	1	1	1	1	0	0
5º Fold	1	1	1	0	1	0	1	0	0	0

Cuadro A.43: Métricas. Análisis 1-CB_LDA_Vibraciones eje x

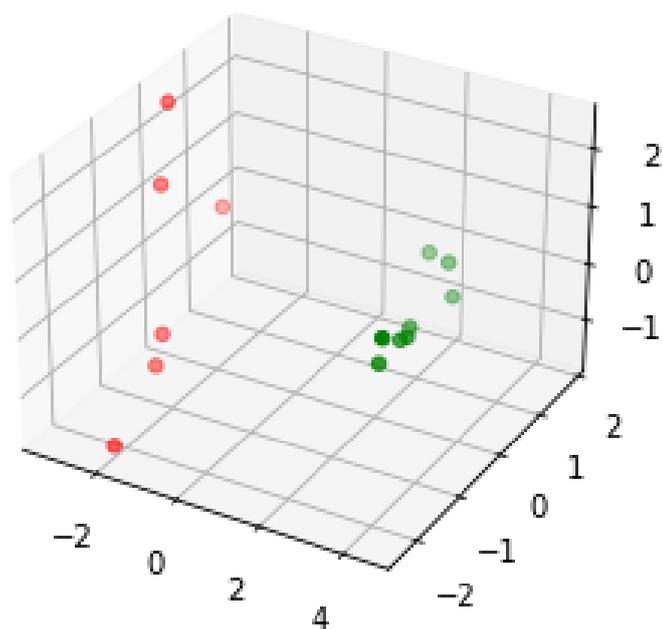


Figura A.24: Gráfico 3D. Análisis 1-CB_LDA_Vibraciones eje y

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1º Fold	1	1	1	1	1	1	1	1	0	0
2º Fold	1	1	1	1	1	1	1	1	0	0
3º Fold	1	1	1	1	1	1	1	1	0	0
4º Fold	1	1	1	1	1	1	1	1	0	0
5º Fold	1	1	1	0	1	0	1	0	0	0

Cuadro A.44: Métricas. Análisis 1-CB_LDA_Vibraciones eje y

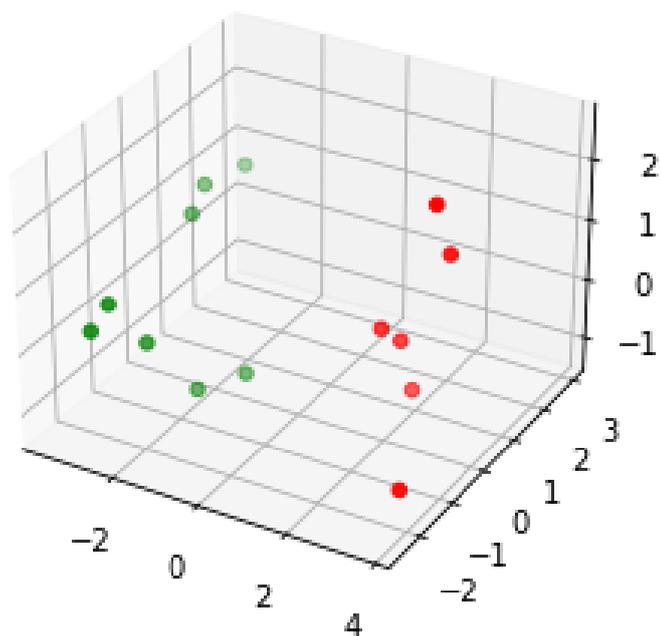


Figura A.25: Gráfico 3D. Análisis 1-CB_LDA_Vibraciones eje z

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1º Fold	1	1	1	1	1	1	1	1	0	0
2º Fold	1	1	1	1	1	1	1	1	0	0
3º Fold	1	1	1	1	1	1	1	1	0	0
4º Fold	1	1	1	1	1	1	1	1	0	0
5º Fold	1	1	1	0	1	0	1	0	0	0

Cuadro A.45: Métricas. Análisis 1-CB_LDA_Vibraciones eje z

Árboles de decisión. Datos de Corriente

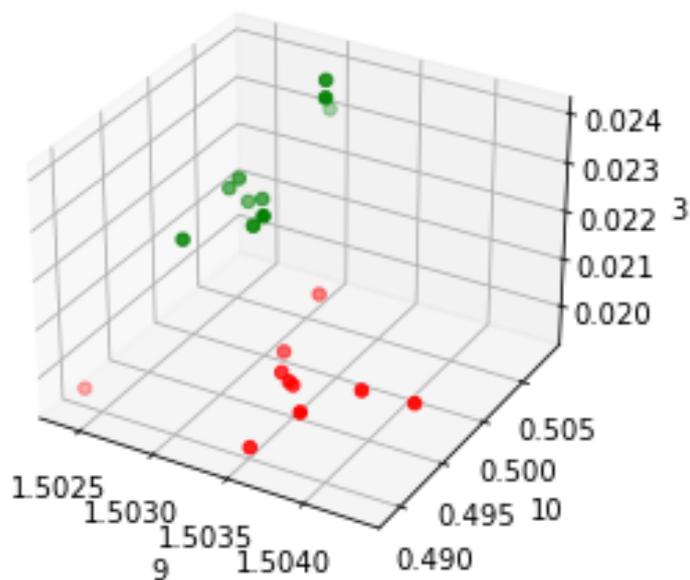


Figura A.26: Gráfico 3D. Análisis 1-CB_AD_Corriente

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1º Fold	1	0.667	1	0.667	1	1	1	0.8	0	0.333
2º Fold	1	1	1	1	1	1	1	1	0	0
3º Fold	1	1	1	1	1	1	1	1	0	0
4º Fold	1	1	1	1	1	1	1	1	0	0
5º Fold	1	1	1	0	1	0	1	0	0	0

Cuadro A.46: Métricas. Análisis 1-CB_AD_Corriente

Árboles de decisión. Datos de Sonido

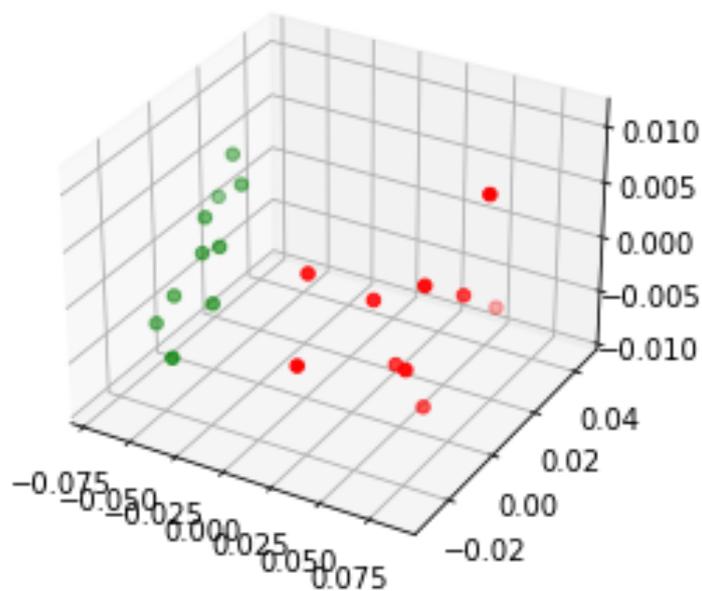


Figura A.27: Gráfico 3D. Análisis 1-CB_AD_Sonido

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1º Fold	1	1	1	1	1	1	1	1	0	0
2º Fold	1	1	1	1	1	1	1	1	0	0
3º Fold	1	1	1	1	1	1	1	1	0	0
4º Fold	1	1	1	1	1	1	1	1	0	0
5º Fold	1	1	1	0	1	0	1	0	0	0

Cuadro A.47: Métricas. Análisis 1-CB_AD_Sonido

Árboles de decisión. Datos de Vibraciones

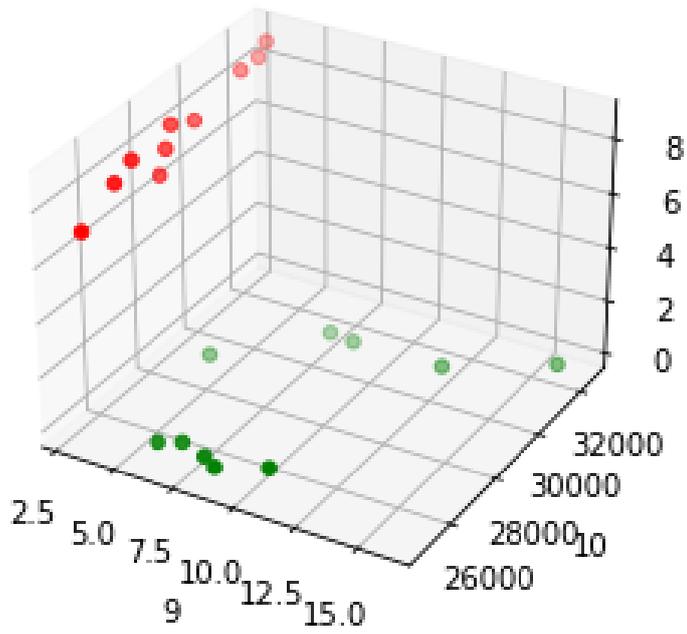


Figura A.28: Gráfico 3D. Análisis 1-CB_AD_Vibraciones eje x

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	1	1	1	1	1	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	0.667	1	1	1	1	0.5	1	0.677	0.333
4ºFold	1	1	1	1	1	1	1	1	0	0
Fold	1	1	1	0	1	0	1	0	0	0

Cuadro A.48: Métricas. Análisis 1-CB_AD_Vibraciones eje x

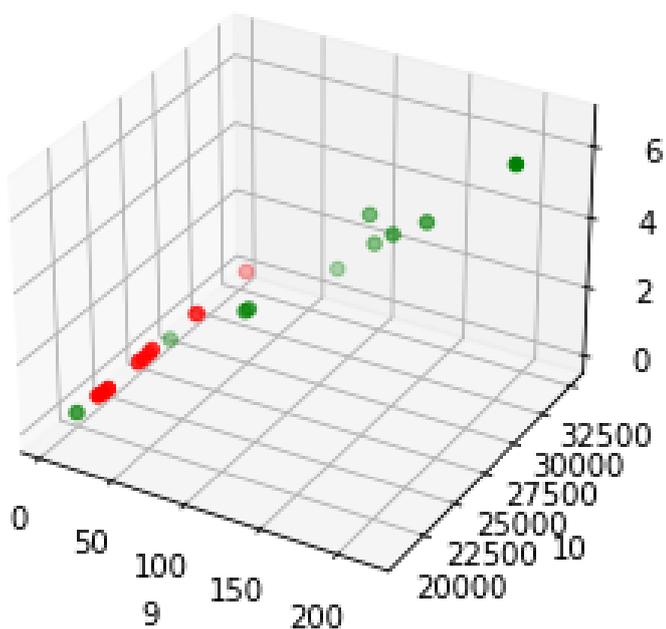


Figura A.29: Gráfico 3D. Análisis 1-CB_AD_Vibraciones eje y

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	1	1	1	1	1	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	1	1	1	1	1	1	1	0	0
5ºFold	1	1	1	0	1	0	1	0	0	0

Cuadro A.49: Métricas. Análisis 1-CB_AD_Vibraciones eje y

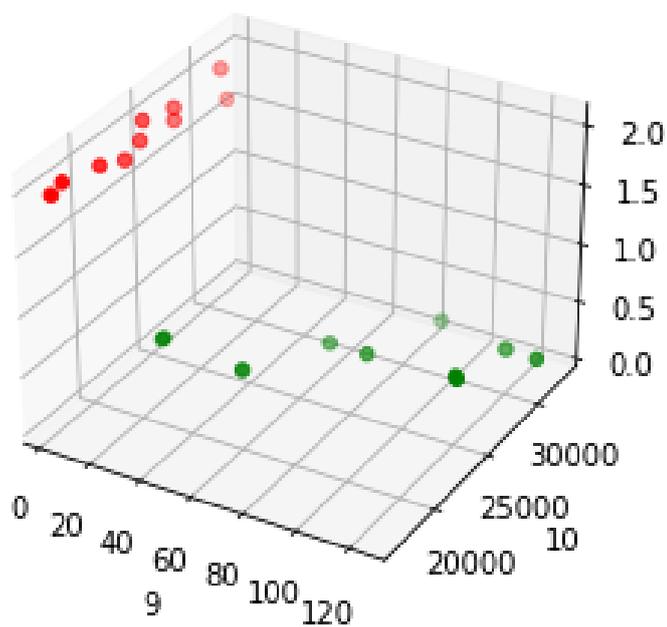


Figura A.30: Gráfico 3D. Análisis 1-CB_AD_Vibraciones eje z

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	1	1	1	1	1	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	0	0	0
4ºFold	1	1	1	1	1	1	1	1	0	0
5ºFold	1	1	1	0	1	0	1	0	0	0

Cuadro A.50: Métricas. Análisis 1-CB_AD_Vibraciones eje z

A.4. Resultados gráficos y numéricos del Análisis 2

En este anexo se ofrecen los resultados completos para el análisis 2 tanto para el LDA como para los árboles de decisión en carga alta y en carga baja. Para este análisis se han sustituido los gráficos correspondientes a las métricas por tablas que contienen los resultados numéricos correspondientes a dichas métricas.

A.4.1. Resultados en Carga Alta

LDA. Datos de Corriente

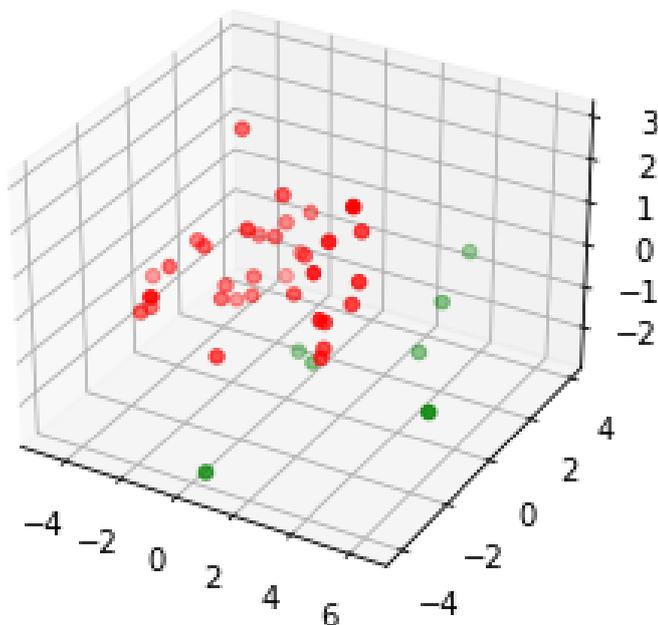


Figura A.31: Gráfico 3D. Análisis 2-CA_LDA_Corriente

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	0	1	0	1	0	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	1	1	1	1	1	1	1	0	0
5ºFold	1	0.875	1	1	1	0.667	1	0.8	0	0.125

Cuadro A.51: Métricas. Análisis 2-CA_LDA_Corriente

LDA. Datos de Sonido

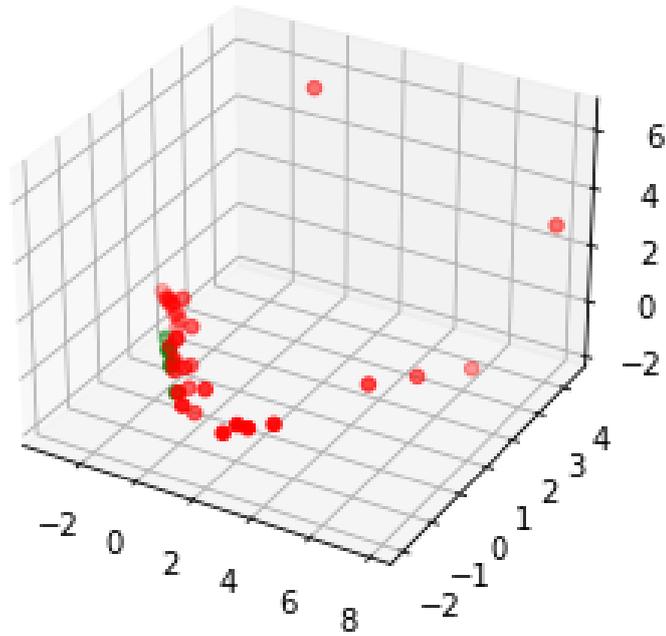


Figura A.32: Gráfico 3D. Análisis 2-CA_LDA_Sonido

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.788	1	0	0	0	0	0	0	0.212	0
2ºFold	0.848	0.778	0	0	0	0	0	0	0.152	0.222
3ºFold	0.824	0.875	0	0	0	0	0	0	0.176	0.125
4ºFold	0.824	0.875	0	0	0	0	0	0	0.176	0.125
5ºFold	0.882	0.625	0	0	0	0	0	0	0.118	0.375

Cuadro A.52: Métricas. Análisis 2-CA_LDA_Sonido

LDA. Datos de Vibraciones

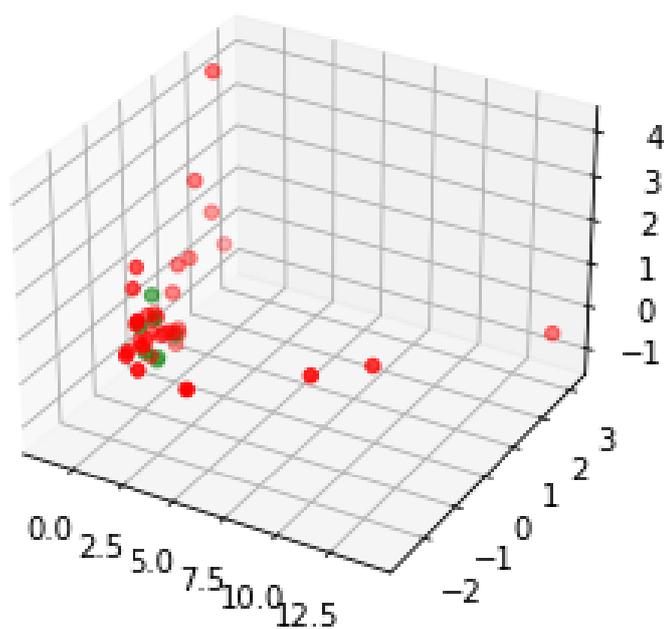


Figura A.33: Gráfico 3D. Análisis 2-CA_LDA_Vibración eje x

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.788	1	0	0	0	0	0	0	0.212	0
2ºFold	0.848	0.778	0	0	0	0	0	0	0.152	0.222
3ºFold	0.824	0.875	0	0	0	0	0	0	0.176	0.125
4ºFold	0.824	0.875	0	0	0	0	0	0	0.176	0.125
5ºFold	0.882	0.625	0	0	0	0	0	0	0.118	0.375

Cuadro A.53: Métricas. Análisis 2-CA_LDA_Vibración eje x

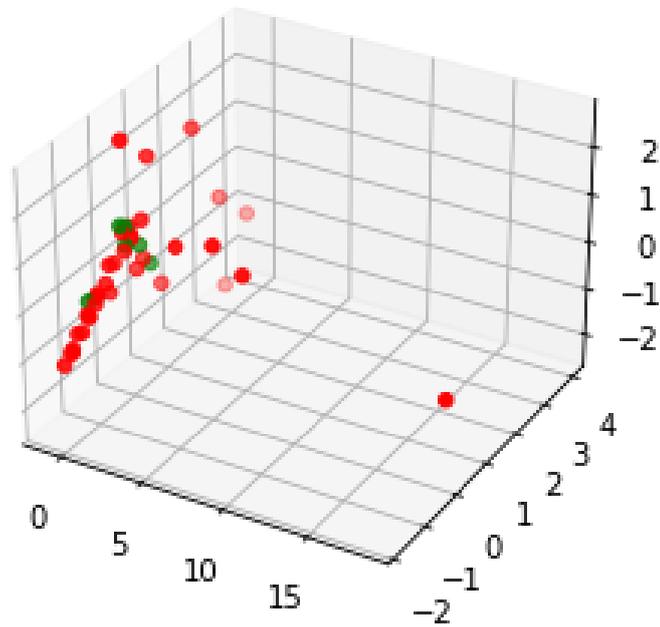


Figura A.34: Gráfico 3D. Análisis 2-CA_LDA_Vibración eje z

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.788	1	0	0	0	0	0	0	0.212	0
2ºFold	0.848	0.778	0	0	0	0	0	0	0.152	0.222
3ºFold	0.824	0.875	0	0	0	0	0	0	0.176	0.125
4ºFold	0.824	0.875	0	0	0	0	0	0	0.176	0.125
5ºFold	0.882	0.625	0	0	0	0	0	0	0.118	0.375

Cuadro A.54: Métricas. Análisis 2-CA_LDA_Vibración eje z

AD. Datos de Corriente

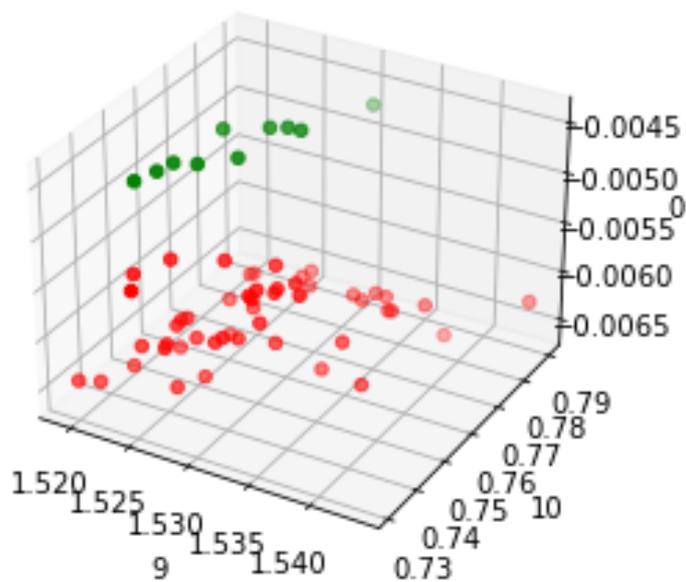


Figura A.35: Gráfico 3D. Análisis 2-CA_AD_Corriente

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	0	1	0	1	0	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	1	1	1	1	1	1	1	0	0
5ºFold	1	1	1	1	1	1	1	1	0	0

Cuadro A.55: Métricas. Análisis 2-CA_AD_Corriente

AD. Datos de Sonido

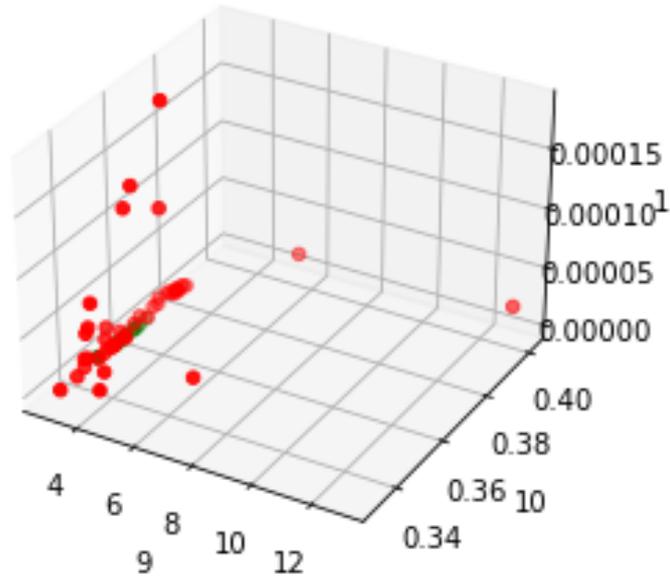


Figura A.36: Gráfico 3D. Análisis 2-CA_AD_Sonido

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	0	1	0	1	0	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	1	1	1	1	1	1	1	0	0
5ºFold	1	1	1	1	1	1	1	1	0	0

Cuadro A.56: Métricas. Análisis 2-CA_AD_Sonido

AD. Datos de Vibraciones

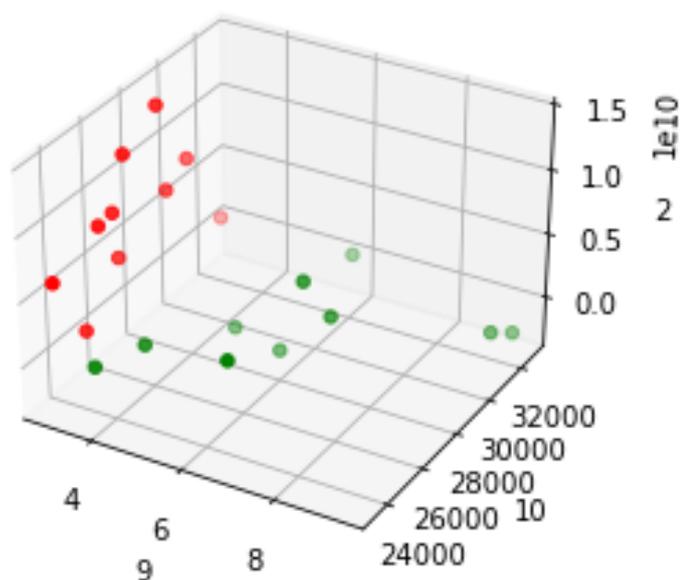


Figura A.37: Gráfico 3D. Análisis 2-CA_AD_Vibraciones eje x

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	0.889	1	0	1	0	1	0	0	0.111
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	1	1	1	1	1	1	1	0	0
5ºFold	1	0.875	1	1	1	0.667	1	0.8	0	0.125

Cuadro A.57: Métricas. Análisis 2-CA_AD_Vibraciones eje x

A.4. Resultados gráficos y numéricos del Análisis 2

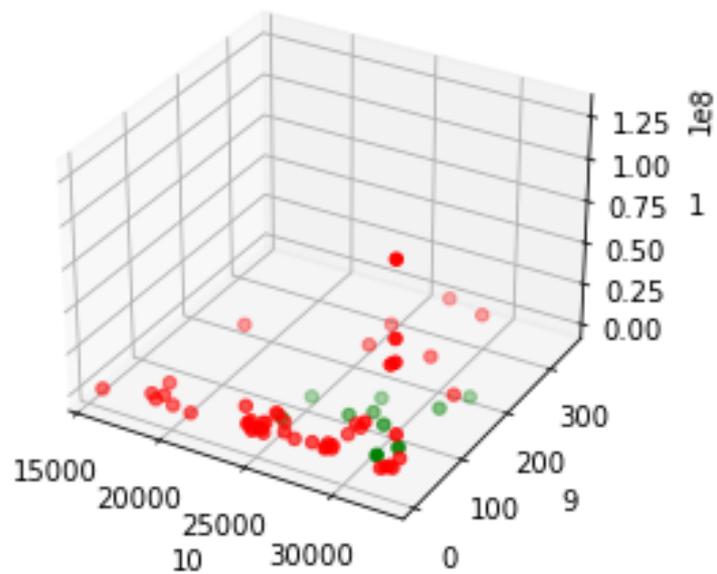


Figura A.38: Gráfico 3D. Análisis 2-CA_AD_Vibraciones eje y

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	0.889	1	0	1	0	1	0	0	0.111
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	0.875	1	0	1	0	1	0	0	0.125
4ºFold	1	0.875	1	0	1	0	1	0	0	0.125
5ºFold	1	1	1	1	1	1	1	1	0	0

Cuadro A.58: Métricas. Análisis 2-CA_AD_Vibraciones eje y

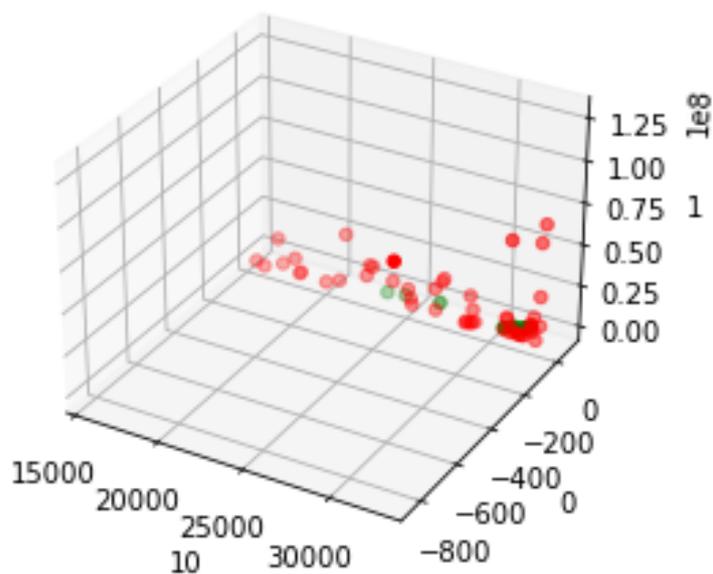


Figura A.39: Gráfico 3D. Análisis 2-CA_AD_Vibraciones eje z

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.97	1	1	0	0.857	0	0.923	0	0.03	0
2ºFold	0.939	1	0.714	1	1	1	0.833	1	0.061	0
3ºFold	1	0.75	1	0.333	1	1	1	0.5	0	0.25
4ºFold	0.971	0.875	0.857	0.5	1	1	0.923	0.667	0.029	0.125
5ºFold	0.941	0.875	0.667	0.75	1	1	0.8	0.857	0.059	0.125

Cuadro A.59: Métricas. Análisis 2-CA_AD_Vibraciones eje z

A.4.2. Resultados en Carga Baja

LDA. Datos de Corriente

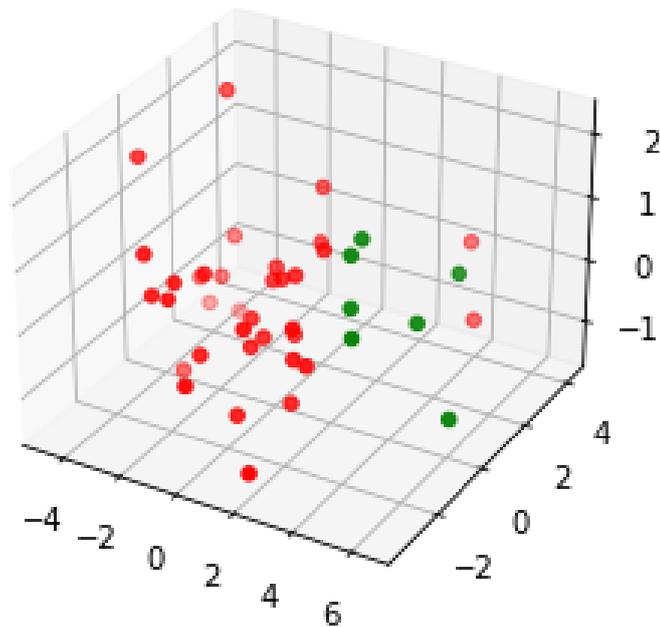


Figura A.40: Gráfico 3D. Análisis 2-CB_LDA_Corriente

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.97	1	0.875	0	1	0	0.933	0	0.03	0
2ºFold	0.97	0.889	0.833	0.667	1	1	0.909	0.8	0.03	0.111
3ºFold	0.941	0.875	0.75	0	1	0	0.857	0	0.059	0.125
4ºFold	0.971	0.875	0.857	0.5	1	1	0.923	0.667	0.029	0.125
5ºFold	0.912	0.875	1	1	0.25	0.667	0.4	0.8	0.088	0.125

Cuadro A.60: Métricas. Análisis 2-CB_LDA_Corriente

LDA. Datos de Sonido

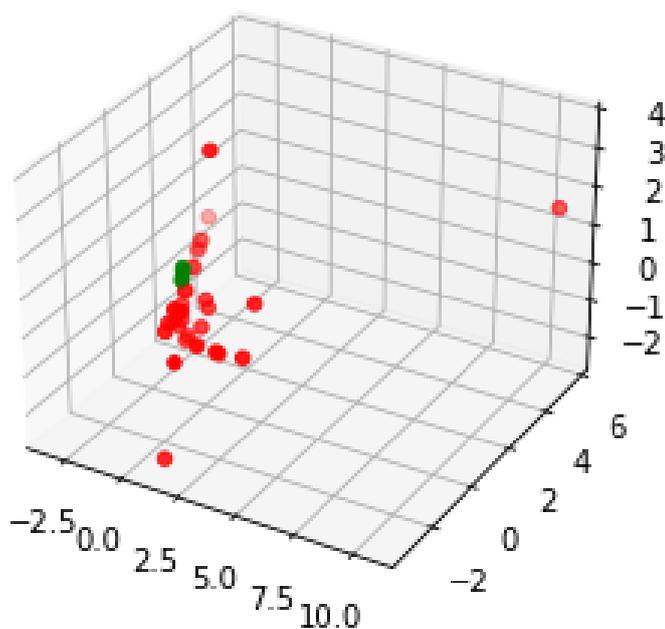


Figura A.41: Gráfico 3D. Análisis 2-CB_LDA_Sonido

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.758	1	0	0	0	0	0	0	0.242	0
2ºFold	0.818	0.778	0	0	0	0	0	0	0.182	0.222
3ºFold	0.794	0.875	0	0	0	0	0	0	0.206	0.125
4ºFold	0.824	0.75	0	0	0	0	0	0	0.176	0.25
5ºFold	0.853	0.625	0	0	0	0	0	0	0.147	0.375

Cuadro A.61: Métricas. Análisis 2-CB_LDA_Sonido

LDA. Datos de Vibraciones

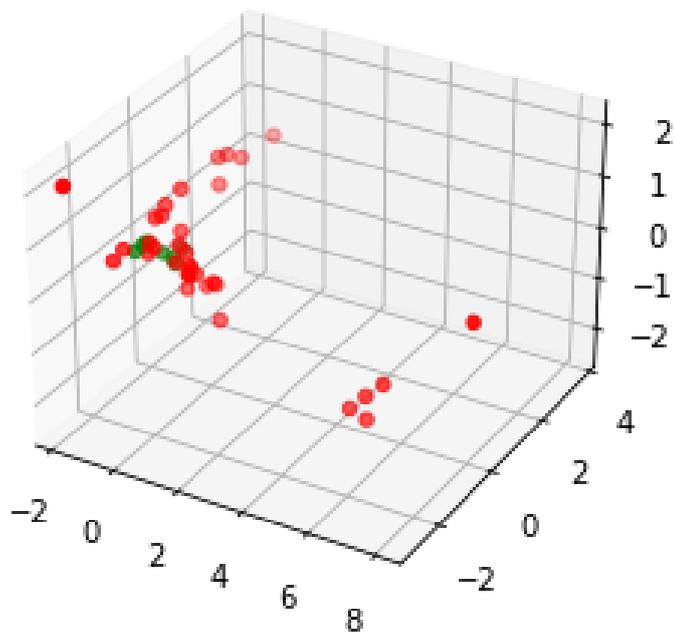


Figura A.42: Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje x

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.788	1	0	0	0	0	0	0	0.212	0
2ºFold	0.848	0.778	0	0	0	0	0	0	0.152	0.222
3ºFold	0.824	0.875	0	0	0	0	0	0	0.176	0.125
4ºFold	0.824	0.875	0	0	0	0	0	0	0.176	0.125
5ºFold	0.882	0.625	0	0	0	0	0	0	0.118	0.375

Cuadro A.62: Métricas. Análisis 2-CB_LDA_Vibraciones eje x

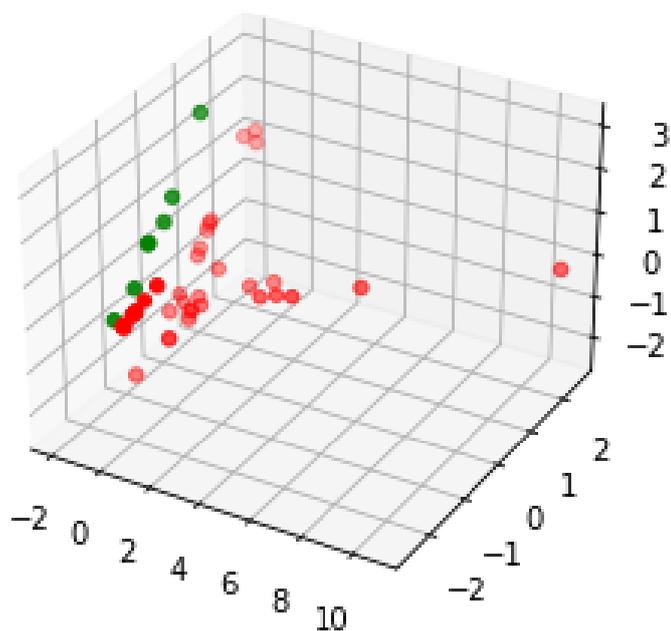


Figura A.43: Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje y

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.909	1	0.833	0	0.714	0	0.769	0	0.091	0
2ºFold	0.97	0.778	1	0.5	0.8	0.5	0.889	0.5	0.03	0.222
3ºFold	0.941	1	1	1	0.667	1	0.8	1	0.059	0
4ºFold	0.941	1	1	1	0.667	1	0.8	1	0.059	0
5ºFold	0.912	0.75	1	1	0.25	0.333	0.4	0.5	0.088	0.25

Cuadro A.63: Métricas. Análisis 2-CB_LDA_Vibraciones eje y

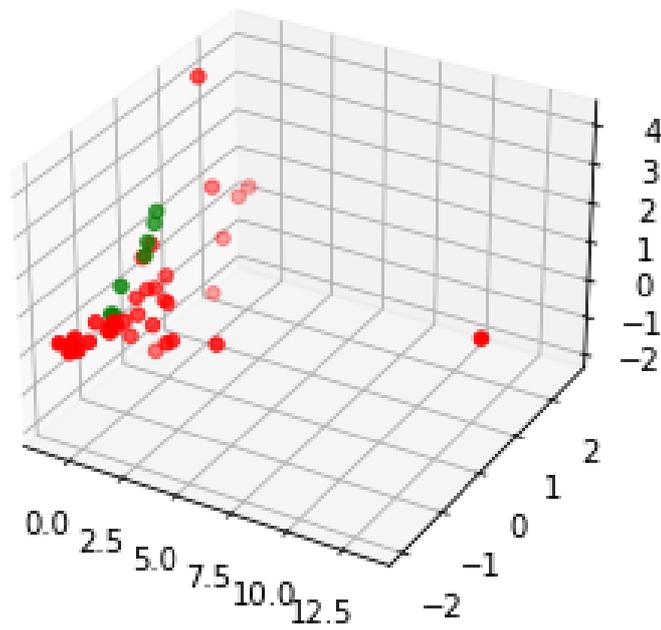


Figura A.44: Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje z

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.848	0.778	0.667	0	0.571	0	0.615	0	0.152	0.222
2ºFold	0.818	0.778	0	0	0	0	0	0	0.182	0.222
3ºFold	0.794	0.875	0	0	0	0	0	0	0.206	0.125
4ºFold	0.794	0.875	0	0	0	0	0	0	0.206	0.125
5ºFold	0.853	0.625	0	0	0	0	0	0	0.147	0.375

Cuadro A.64: Métricas. Análisis 2-CB_LDA_Vibraciones eje z

AD. Datos de Corriente

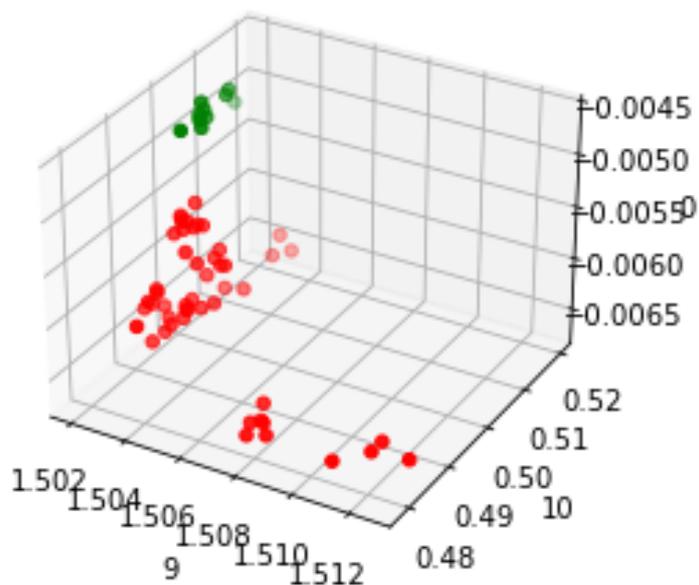


Figura A.45: Gráfico 3D. Análisis 2-CB_AD_Corriente

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	0	1	0	1	0	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	1	1	1	1	1	1	1	0	0
5ºFold	1	1	1	1	1	1	1	1	0	0

Cuadro A.65: Métricas. Análisis 2-CB_AD_Corriente

AD. Datos de Sonido

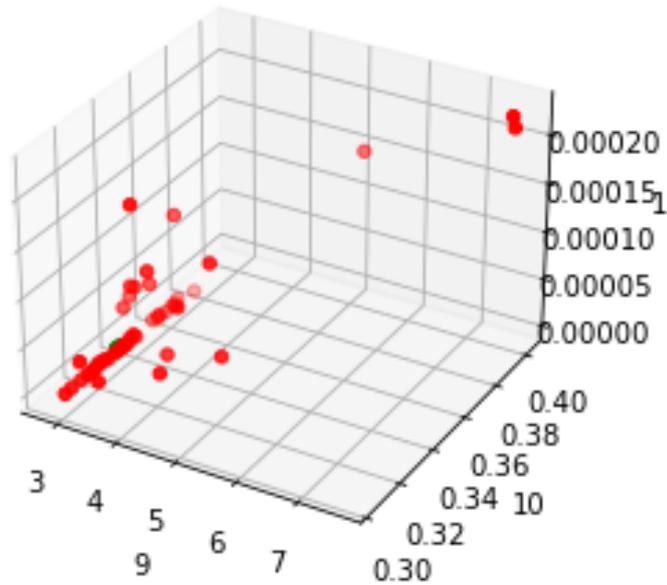


Figura A.46: Gráfico 3D. Análisis 2-CB_LDA_Sonido

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	0	1	0	1	0	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	1	1	1	1	1	1	1	0	0
5ºFold	1	1	1	1	1	1	1	1	0	0

Cuadro A.66: Métricas. Análisis 2-CB_LDA_Sonido

AD. Datos de Vibraciones

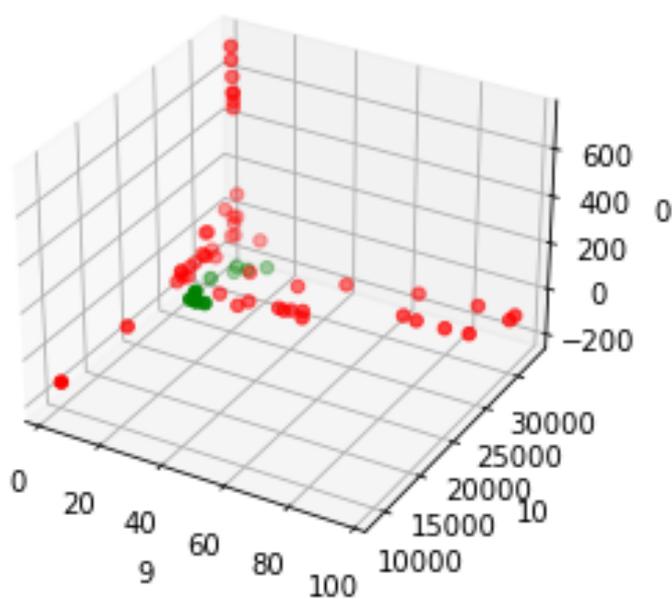


Figura A.47: Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje x

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	1	1	0	1	0	1	0	0	0
2ºFold	1	1	1	1	1	1	1	1	0	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	0.875	1	0	1	0	1	0	0	0.125
5ºFold	1	1	1	1	1	1	1	1	0	0

Cuadro A.67: Métricas. Análisis 2-CB_LDA_Vibraciones eje x

A.4. Resultados gráficos y numéricos del Análisis 2

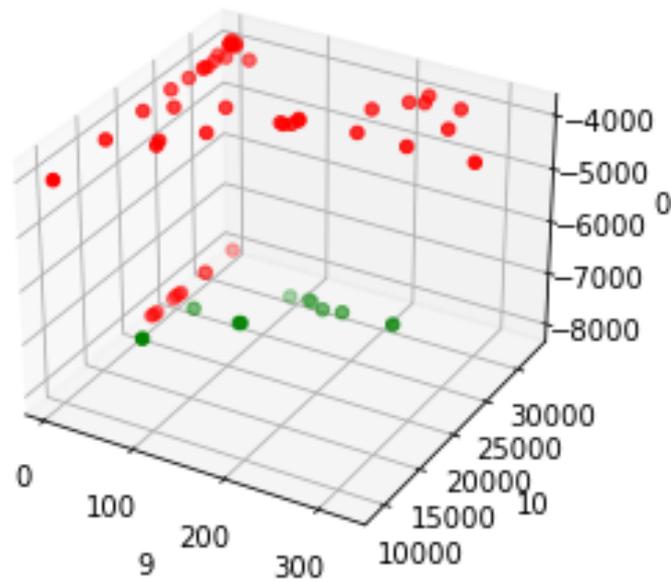


Figura A.48: Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje y

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	0.97	1	1	0	0.857	0	0.923	0	0.03	0
2ºFold	1	0.778	1	0.5	1	1	1	0.667	0	0.222
3ºFold	0.971	1	1	1	0.833	1	0.909	1	0.029	0
4ºFold	0.971	1	1	1	0.833	1	0.909	1	0.029	0
5ºFold	1	0.75	1	1	1	0.333	1	0.5	0	0.25

Cuadro A.68: Métricas. Análisis 2-CB_LDA_Vibraciones eje y

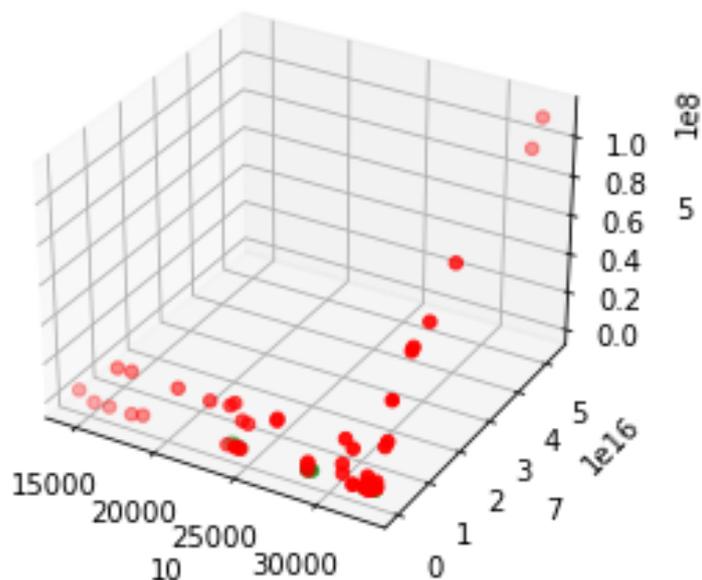


Figura A.49: Gráfico 3D. Análisis 2-CB_LDA_Vibraciones eje z

Fold/Métricas	Train Accuracy	Test Accuracy	Train Precision	Test Precision	Train Recall	Test Recall	Train F1	Test F1	Train Error	Test Error
1ºFold	1	0.778	1	0	1	0	1	0	0	0.222
2ºFold	0.97	1	0.833	1	1	1	0.909	1	0.03	0
3ºFold	1	1	1	1	1	1	1	1	0	0
4ºFold	1	0.875	1	0.5	1	1	1	0.667	0	0.125
5ºFold	1	1	1	1	1	1	1	1	0	0

Cuadro A.69: Métricas. Análisis 2-CB_LDA_Vibraciones eje z

Apéndice B

Código

A continuación se ofrece el código Python para los diferentes procedimientos desarrollados en el trabajo.

B.1. Diagramas de caja

```
1 #Diagrama de cajas Corriente
2 #Importación de librerías
3 import numpy as np
4 import matplotlib.pyplot as plt
5 import pandas as pd
6 #colf1
7 df = pd.DataFrame(DatosCACorr_f1_f2_f3, columns= ['col1f1','col2f1',
8     ↪ ', 'col3f1','col4f1','col5f1','col6f1','col7f1','col8f1',''
9     ↪ 'col9f1','col10f1','col11f1','col12f1','col13f1','col14f1',''
10    ↪ 'col15f1','col16f1','col17f1' ] )
11
12 columnsf1 = ['col1f1','col2f1','col3f1','col4f1','col5f1','col6f1',
13     ↪ 'col7f1','col8f1','col9f1','col10f1','col11f1','col12f1',''
14     ↪ 'col13f1','col14f1','col15f1','col16f1','col17f1' ]
15 for col in columnsf1:
16     plt.figure(figsize=(14,11))    # plots figure for each iteration
17     plt.boxplot(df[col])
18     plt.title("Boxplot")
19
20 #colf2
21 df2 = pd.DataFrame(DatosCACorr_f1_f2_f3, columns= ['col1f2','col2f2
22     ↪ ', 'col3f2','col4f2','col5f2','col6f2','col7f2','col8f2',''
23     ↪ 'col9f2','col10f2','col11f2','col12f2','col13f2','col14f2',''
24     ↪ 'col15f2','col16f2','col17f2' ] )
25
```

```

18 columnsf2 = ['col1f2','col2f2','col3f2','col4f2','col5f2','col6f2',
    ↪ 'col7f2','col8f2','col9f2','col10f2','col11f2','col12f2','
    ↪ col13f2','col14f2','col15f2','col16f2','col17f2' ]
19 for col in columnsf2:
20     plt.figure(figsize=(14,11))    # plots figure for each iteration
21     plt.boxplot(df2[col])
22     plt.title("Boxplot")
23
24 #colf3
25 df3 = pd.DataFrame(DatosCACorr_f1_f2_f3, columns= ['col1f3','col2f3
    ↪ ','col3f3', 'col4f3', 'col5f3', 'col6f3', 'col7f3', 'col8f3',
    ↪ 'col9f3', 'col10f3', 'col11f3', 'col12f3', 'col13f3', '
    ↪ col14f3', 'col15f3', 'col16f3', 'col17f3'])
26
27 columnsf3 = ['col1f3','col2f3','col3f3', 'col4f3', 'col5f3', '
    ↪ col6f3', 'col7f3', 'col8f3', 'col9f3', 'col10f3', 'col11f3',
    ↪ 'col12f3', 'col13f3', 'col14f3', 'col15f3', 'col16f3', '
    ↪ col17f3']
28 for col in columnsf3:
29     plt.figure(figsize=(14,11))    # plots figure for each iteration
30     plt.boxplot(df3[col])
31     plt.title("Boxplot")
32
33 #Diagrama de cajas Sonido
34
35 #Importación de librerías
36 import numpy as np
37 import matplotlib.pyplot as plt
38 import pandas as pd
39 #col
40 df = pd.DataFrame(DatosCASonido, columns= ['col1','col2','col3','
    ↪ col4','col5','col6','col7','col8','col9','col10','col11','
    ↪ col12','col13','col14','col15','col16','col17' ] )
41
42 columns = ['col1','col2','col3','col4','col5','col6','col7','col8',
    ↪ 'col9','col10','col11','col12','col13','col14','col15','col16
    ↪ ','col17' ]
43 for col in columns:
44     plt.figure(figsize=(14,11))    # plots figure for each iteration
45     plt.boxplot(df[col])
46     plt.title("Boxplot")
47
48 #Diagrama de cajas Vibraciones
49 #colx
50 dfx = pd.DataFrame(DatosCAVibraciones_x_y_z, columns= ['col1x',
    ↪ col2x','col3x','col4x','col5x','col6x','col7x','col8x','col9x

```

```

    ↪ ', 'col10x', 'col11x', 'col12x', 'col13x', 'col14x', 'col15x', '
    ↪ col16x', 'col17x' ] )
51
52 columnsx = ['col1x', 'col2x', 'col3x', 'col4x', 'col5x', 'col6x', 'col7x'
    ↪ ', 'col8x', 'col9x', 'col10x', 'col11x', 'col12x', 'col13x', 'col14x'
    ↪ ', 'col15x', 'col16x', 'col17x' ]
53 for col in columnsx:
54     plt.figure(figsize=(14,11)) # plots figure for each iteration
55     plt.boxplot(dfx[col])
56     plt.title("Boxplot")
57
58 #coly
59 dfy = pd.DataFrame(DatosCAVibraciones_x_y_z, columns= ['col1y', '
    ↪ col2y', 'col3y', 'col4y', 'col5y', 'col6y', 'col7y', 'col8y', 'col9y
    ↪ ', 'col10y', 'col11y', 'col12y', 'col13y', 'col14y', 'col15y', '
    ↪ col16y', 'col17y' ] )
60
61 columnsy = ['col1y', 'col2y', 'col3y', 'col4y', 'col5y', 'col6y', 'col7y'
    ↪ ', 'col8y', 'col9y', 'col10y', 'col11y', 'col12y', 'col13y', 'col14y'
    ↪ ', 'col15y', 'col16y', 'col17y' ]
62 for col in columnsy:
63     plt.figure(figsize=(14,11)) # plots figure for each iteration
64     plt.boxplot(dfy[col])
65     plt.title("Boxplot")
66
67 #colfz
68 dfz = pd.DataFrame(DatosCAVibraciones_x_y_z, columns= ['col1z', '
    ↪ col2z', 'col3z', 'col4z', 'col5z', 'col6z', 'col7z', 'col8z',
    ↪ 'col9z', 'col10z', 'col11z', 'col12z', 'col13z', 'col14z', '
    ↪ col15z', 'col16z', 'col17z'])
69
70 columnsz = ['col1z', 'col2z', 'col3z', 'col4z', 'col5z', 'col6z', '
    ↪ col7z', 'col8z', 'col9z', 'col10z', 'col11z', 'col12z', '
    ↪ col13z', 'col14z', 'col15z', 'col16z', 'col17z']
71 for col in columnsz:
72     plt.figure(figsize=(14,11)) # plots figure for each iteration
73     plt.boxplot(dfz[col])
74     plt.title("Boxplot")
75
76 #

```

B.2. Distancia de Mahalanobis por Grupos de Respuesta

```

1 #Importamos las librerías
2 import numpy as np
3 import matplotlib.pyplot as plt
4 plt.style.use('seaborn-whitegrid')
5 import pandas as pd
6 import scipy as stats
7
8 #Importamos los datos
9 DatosCACorriente= pd.read_excel('
    ↪ Corr_f1_f2_f3_CA1contra2MahalanobisGrupo1.xlsx', dtype=float
    ↪ ), index_col=None)
10 dfx = pd.DataFrame(DatosCACorriente, columns= ['c1_f1', 'c2_f1', '
    ↪ c3_f1', 'c4_f1', 'xp_f1', 'c1_f2', 'c2_f2', 'c3_f2', 'c4_f2', 'xp_f2
    ↪ ', 'c1_f3', 'c2_f3', 'c3_f3', 'c4_f3', 'xp_f3'] )
11 #Cambiamos el orden de las columnas
12 nuevas_columnas = ['c1_f1', 'c1_f2', 'c1_f3', 'c2_f1', 'c2_f2', '
    ↪ c2_f3', 'c3_f1', 'c3_f2', 'c3_f3', 'c4_f1', 'c4_f2', 'c4_f3',
    ↪ 'xp_f1', 'xp_f2', 'xp_f3']
13 df = dfx.reindex(columns=nuevas_columnas)
14
15 # Calculamos la media de cada grupo de columnas y formamos nuevas
    ↪ columnas con los resultados
16 c1_mean = df.groupby(lambda x: str(x).split('_')[0])['c1_f1', '
    ↪ c1_f2', 'c1_f3'].mean()
17 c2_mean = df.groupby(lambda x: str(x).split('_')[0])['c2_f1', '
    ↪ c2_f2', 'c2_f3'].mean()
18 c3_mean = df.groupby(lambda x: str(x).split('_')[0])['c3_f1', '
    ↪ c3_f2', 'c3_f3'].mean()
19 c4_mean = df.groupby(lambda x: str(x).split('_')[0])['c4_f1', '
    ↪ c4_f2', 'c4_f3'].mean()
20 xp_mean = df.groupby(lambda x: str(x).split('_')[0])['xp_f1', '
    ↪ xp_f2', 'xp_f3'].mean()
21
22 df_new = df.join(c1_mean, rsuffix='_c1_mean')
23 df_new = df_new.join(c2_mean, rsuffix='_c2_mean')
24 df_new = df_new.join(c3_mean, rsuffix='_c3_mean')
25 df_new = df_new.join(c4_mean, rsuffix='_c4_mean')
26 df_new = df_new.join(xp_mean, rsuffix='_xp_mean')
27
28 df_transformado = pd.DataFrame()
29 df_transformado['c1'] = (df['c1_f1'] + df['c1_f2'] + df['c1_f3']) /
    ↪ 3

```

```

30 df_transformado['c2'] = (df['c2_f1'] + df['c2_f2'] + df['c2_f3']) /
    ↪ 3
31 df_transformado['c3'] = (df['c3_f1'] + df['c3_f2'] + df['c3_f3']) /
    ↪ 3
32 df_transformado['c4'] = (df['c4_f1'] + df['c4_f2'] + df['c4_f3']) /
    ↪ 3
33 df_transformado['xp'] = (df['xp_f1'] + df['xp_f2'] + df['xp_f3']) /
    ↪ 3
34
35 #Función que calcula
36 def distanciaMahalanobis(y=None, data=None, cov=None):
37
38     y_mu = y - np.mean(data)
39     if not cov:
40         cov = np.cov(data.values.T)
41         inv_covmat = np.linalg.inv(cov)
42         left = np.dot(y_mu, inv_covmat)
43         mahal = np.dot(left, y_mu.T)
44         return mahal.diagonal()
45
46 #Creamos una nueva columna en dfx con la distancia Mahalanobis para
    ↪ cada fila
47 df_transformado['distanciaMahalanobis'] = distanciaMahalanobis(y=
    ↪ df_transformado, data=df_transformado[['c1','c2','c3','c4','
    ↪ xp']])
48 print(df_transformado)

```

B.3. Análisis Algoritmo LDA

```

1 # Importacion de librerias
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import pandas as pd
5 import seaborn as sns
6
7 import time
8 from sklearn import metrics, preprocessing, tree
9 from sklearn.metrics import f1_score, make_scorer
10 from sklearn.model_selection import cross_val_score
11 from sklearn.model_selection import GridSearchCV,
    ↪ RandomizedSearchCV
12
13 # Primero importamos el dataset

```

```

14 dataset = pd.read_excel('
    ↪ Corr_f1_f2_f3_CA1Contra2componentes_principales.xlsx', dtype
    ↪ =float(), index_col=None)
15
16 #Formamos dos grupos
17 df1=dataset[dataset['Respuesta']==1]
18 df2=dataset[dataset['Respuesta']==2]
19
20 #Comprobamos el Supuesto Normalidad multivariante Prueba de Shapiro
    ↪ -wilks
21 #Por grupos
22 #Grupo 1
23 from scipy.stats import shapiro
24 estadistico, p_value = shapiro(df1)
25 print('Estadistico=%.3f,p_value=%.3f' %(estadistico,p_value))
26
27 #Grupo 2
28 from scipy.stats import shapiro
29 estadistico, p_value = shapiro(df2)
30 print('Estadistico=%.3f,p_value=%.3f' %(estadistico,p_value))
31
32
33 #Por variables
34 from scipy.stats import shapiro
35 for col in dataset:
36     estadistico, p_value = shapiro(dataset[col])
37     print('Estadistico=%.3f,p_value=%.3f' %(estadistico,p_value))
38
39 import seaborn as sns
40
41 #Excluimos la columna 'Respuesta'
42 columns_to_plot = [col for col in dataset.columns if col != '
    ↪ Respuesta']
43
44 # Creamos el gráfico de pares
45 sns.pairplot(dataset[columns_to_plot])
46
47
48 #Comprobamos el Supuesto de homocedasticidad # Levene test
49 import pandas as pd
50 import scipy.stats as stats
51 for col in dataset:
52     estadistico, p_value = stats.bartlett(df1[col], df2[col] )
53     print('Estadistico=%.3f,p_value=%.3f' %(estadistico,p_value))
54
55 #Comprobamos el Supuesto de multicolinealidad

```

```
56 import seaborn as sns
57 import matplotlib.pyplot as plt
58
59 #Excluimos la columna 'Respuesta'
60 dataset_without_response = dataset.drop('Respuesta', axis=1)
61
62 # Calculamos la matriz de correlación
63 dataset1 = dataset_without_response.corr()
64
65 # Generamos el mapa de calor
66 f, ax = plt.subplots(figsize=(12, 9))
67 sns.heatmap(dataset1, vmax=0.8, square=True)
68 plt.show()
69
70 #LDA CLASIFICACIÓN
71 #Formado por todas las columnas excepto por la última
72 X = dataset.iloc[:, 0:3].values
73 #y es la última
74 y = dataset.iloc[:, -1].values
75
76 # el tamaño de x e y
77 X.shape, y.shape
78
79 # Dividimos el data set en conjunto de entrenamiento y conjunto de
80 ↪ testing
81 from sklearn.model_selection import train_test_split
82 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size
83 ↪ = 0.3, random_state = 0)
84
85 #Ajustamos el modelo al conjunto de entrenamiento
86 from sklearn.discriminant_analysis import
87 ↪ LinearDiscriminantAnalysis as LDAClassifier
88 classifier = LDAClassifier()
89 classifier.fit(X_train, y_train)
90
91 #Predecimos en el Conjunto de Testing
92 y_pred = classifier.predict(X_test)
93
94 #Creamos la matriz de confusión
95 from sklearn.metrics import confusion_matrix
96 cm = confusion_matrix(y_test, y_pred)
97
98 #Validación cruzada k fold con 5 matrices de confusión y errores
99 import numpy as np
100 from sklearn.model_selection import KFold
```

```

98 from sklearn.metrics import accuracy_score, precision_score,
    ↪ recall_score, f1_score
99
100 def cross_validation(model, X, y, cv=5):
101     kf = KFold(n_splits=cv)
102     fold_results = []
103     test_error_sum = 0
104
105     for train_index, test_index in kf.split(X):
106         X_train, X_test = X[train_index], X[test_index]
107         y_train, y_test = y[train_index], y[test_index]
108
109         model.fit(X_train, y_train)
110         y_train_pred = model.predict(X_train)
111         y_test_pred = model.predict(X_test)
112
113         fold_result = {
114             "Train_Accuracy": accuracy_score(y_train, y_train_pred)
115             ↪ ,
116             "Test_Accuracy": accuracy_score(y_test, y_test_pred),
117             "Train_Precision": precision_score(y_train,
118             ↪ y_train_pred),
119             "Test_Precision": precision_score(y_test, y_test_pred),
120             "Train_Recall": recall_score(y_train, y_train_pred),
121             "Test_Recall": recall_score(y_test, y_test_pred),
122             "Train_F1": f1_score(y_train, y_train_pred),
123             "Test_F1": f1_score(y_test, y_test_pred),
124             "Train_Error": 1 - accuracy_score(y_train, y_train_pred)
125             ↪ ),
126             "Test_Error": 1 - accuracy_score(y_test, y_test_pred)
127         }
128
129         test_error_sum += fold_result['Test_Error']
130         fold_results.append(fold_result)
131
132     test_error_average = test_error_sum / cv
133
134     return fold_results, test_error_average
135
136 resultados, promedio_test_error = cross_validation(classifier,
137     ↪ X_train, y_train, 5)
138 print("Resultados_por_Folds:", resultados)
139 print("Promedio_de_Test_Error:", promedio_test_error)
140
141 def save_results_to_excel(results, filename):

```

```

139     df = pd.DataFrame(results)
140     df = df.round(3)
141     df.to_excel(filename, index=False)
142
143 resultados, promedio_test_error = cross_validation(classifier,
144     ↪ X_train, y_train, 5)
144 save_results_to_excel(resultados, 'resultados2.xlsx')
145 print("Resultados por Folds:", resultados)
146 print("Promedio de Test Error:", promedio_test_error)
147
148 #Gráfico accuracy
149 import matplotlib.pyplot as plt
150
151 def plot_cross_validation_results(resultados):
152     train_acc = [fold["Train Accuracy"] for fold in resultados]
153     test_acc = [fold["Test Accuracy"] for fold in resultados]
154
155     plt.figure(figsize=(10, 6))
156     plt.plot(range(1, len(train_acc) + 1), train_acc, 'ro-', label=
157     ↪ 'Train Accuracy')
157     plt.plot(range(1, len(test_acc) + 1), test_acc, 'bo-', label=
158     ↪ 'Test Accuracy')
158     plt.xlabel('Fold')
159     plt.ylabel('Accuracy')
160     plt.title('Cross Validation Results')
161     plt.legend()
162     plt.show()
163
164 plot_cross_validation_results(resultados)
165
166 #Gráfico de barras accuracy
167 import seaborn as sns
168 import matplotlib.pyplot as plt
169
170 def plot_cross_validation_results(results):
171     train_acc = [fold["Train Accuracy"] for fold in results]
172     test_acc = [fold["Test Accuracy"] for fold in results]
173     fold_labels = [f"Fold_{i+1}" for i in range(len(train_acc))]
174
175     data = pd.DataFrame({'Fold': fold_labels, 'Train Accuracy':
176     ↪ train_acc, 'Test Accuracy': test_acc})
176     data = data.melt('Fold', var_name='Dataset', value_name='
177     ↪ Accuracy')
177
178     plt.figure(figsize=(10, 6))
179     sns.barplot(x='Fold', y='Accuracy', hue='Dataset', data=data)

```

```

180     plt.xlabel('Fold')
181     plt.ylabel('Accuracy')
182     plt.title('Cross_Validation_Results')
183     plt.legend()
184     plt.show()
185
186 plot_cross_validation_results(resultados)
187
188 #Gráfico precision
189 import matplotlib.pyplot as plt
190
191 def plot_cross_validation_results(resultados):
192     train_precision = [fold["Train_Precision"] for fold in
193         ↪ resultados]
194     test_precision = [fold["Test_Precision"] for fold in resultados
195         ↪ ]
196
197     plt.figure(figsize=(10, 6))
198     plt.plot(range(1, len(train_precision) + 1), train_precision, '
199         ↪ ro-', label='Train_Accuracy')
200     plt.plot(range(1, len(test_precision) + 1), test_precision, 'bo
201         ↪ -', label='Test_Accuracy')
202     plt.xlabel('Fold')
203     plt.ylabel('Precision')
204     plt.title('Cross_Validation_Results')
205     plt.legend()
206     plt.show()
207
208 plot_cross_validation_results(resultados)
209
210 #Gráfico barras precision
211 import seaborn as sns
212 import matplotlib.pyplot as plt
213
214 def plot_cross_validation_results(results):
215     train_precision = [fold["Train_Precision"] for fold in results]
216     test_precision = [fold["Test_Precision"] for fold in results]
217     fold_labels = [f"Fold_{i+1}" for i in range(len(train_precision)
218         ↪ ))]
219
220     data = pd.DataFrame({'Fold': fold_labels, 'Train_Precision':
221         ↪ train_precision, 'Test_Precision': test_precision})
222     data = data.melt('Fold', var_name='Dataset', value_name='
223         ↪ Precision')
224
225     plt.figure(figsize=(10, 6))

```

```

219     sns.barplot(x='Fold', y='Precision', hue='Dataset', data=data)
220     plt.xlabel('Fold')
221     plt.ylabel('Precision')
222     plt.title('Cross_Validation_Results_-_Precision')
223     plt.legend()
224     plt.show()
225
226 plot_cross_validation_results(resultados)
227
228
229 #Gráfico recall
230 import matplotlib.pyplot as plt
231
232 def plot_cross_validation_results(resultados):
233     train_recall = [fold["Train_Recall"] for fold in resultados]
234     test_recall = [fold["Test_Recall"] for fold in resultados]
235
236     plt.figure(figsize=(10, 6))
237     plt.plot(range(1, len(train_recall) + 1), train_recall, 'ro-',
238             ↪ label='Train_Recall')
239     plt.plot(range(1, len(test_recall) + 1), test_recall, 'bo-',
240             ↪ label='Test_Recall')
241     plt.xlabel('Fold')
242     plt.ylabel('Recall')
243     plt.title('Cross_Validation_Results_-_Recall')
244     plt.legend()
245     plt.show()
246
247 plot_cross_validation_results(resultados)
248
249 #Gráfico de barras recall
250 import seaborn as sns
251 import matplotlib.pyplot as plt
252
253 def plot_cross_validation_results(results):
254     train_recall = [fold["Train_Recall"] for fold in results]
255     test_recall = [fold["Test_Recall"] for fold in results]
256     fold_labels = [f"Fold_{i+1}" for i in range(len(train_recall))]
257
258     data = pd.DataFrame({'Fold': fold_labels, 'Train_Recall':
259             ↪ train_recall, 'Test_Recall': test_recall})
260     data = data.melt('Fold', var_name='Dataset', value_name='Recall'
261             ↪ ')
262
263     plt.figure(figsize=(10, 6))
264     sns.barplot(x='Fold', y='Recall', hue='Dataset', data=data)

```

```

261     plt.xlabel('Fold')
262     plt.ylabel('Recall')
263     plt.title('Cross_Validation_Results_-_Recall')
264     plt.legend()
265     plt.show()
266
267 plot_cross_validation_results(resultados)
268
269 #Grafico F1
270 import matplotlib.pyplot as plt
271
272 def plot_cross_validation_results(resultados):
273     train_f1 = [fold["Train_F1"] for fold in resultados]
274     test_f1 = [fold["Test_F1"] for fold in resultados]
275
276     plt.figure(figsize=(10, 6))
277     plt.plot(range(1, len(train_f1) + 1), train_f1, 'ro-', label='
↳ Train_F1')
278     plt.plot(range(1, len(test_f1) + 1), test_f1, 'bo-', label='
↳ Test_F1')
279     plt.xlabel('Fold')
280     plt.ylabel('F1')
281     plt.title('Cross_Validation_Results_-_F1')
282     plt.legend()
283     plt.show()
284
285 plot_cross_validation_results(resultados)
286
287 #Grafico de barras F1
288 import seaborn as sns
289 import matplotlib.pyplot as plt
290
291 def plot_cross_validation_results(results):
292     train_f1 = [fold["Train_F1"] for fold in results]
293     test_f1 = [fold["Test_F1"] for fold in results]
294     fold_labels = [f"Fold_{i+1}" for i in range(len(train_f1))]
295
296     data = pd.DataFrame({'Fold': fold_labels, 'Train_F1': train_f1,
↳ 'Test_F1': test_f1})
297     data = data.melt('Fold', var_name='Dataset', value_name='F1')
298
299     plt.figure(figsize=(10, 6))
300     sns.barplot(x='Fold', y='F1', hue='Dataset', data=data)
301     plt.xlabel('Fold')
302     plt.ylabel('F1')
303     plt.title('Cross_Validation_Results_-_F1')

```

```

304     plt.legend()
305     plt.show()
306
307 plot_cross_validation_results(resultados)
308
309 #Grafico error de clasificación
310 import matplotlib.pyplot as plt
311
312 def plot_cross_validation_results(resultados):
313     train_error = [1 - fold["Train_Accuracy"] for fold in
314         ↪ resultados]
315     test_error = [1 - fold["Test_Accuracy"] for fold in resultados]
316
317     plt.figure(figsize=(10, 6))
318     plt.plot(range(1, len(train_error) + 1), train_error, 'ro-',
319         ↪ label='Train_Error')
320     plt.plot(range(1, len(test_error) + 1), test_error, 'bo-',
321         ↪ label='Test_Error')
322     plt.xlabel('Fold')
323     plt.ylabel('Error')
324     plt.title('Cross_Validation_Results_-_Error')
325     plt.legend()
326     plt.show()
327
328 plot_cross_validation_results(resultados)
329
330 #Grafico de barras error de clasificacion
331 import seaborn as sns
332 import matplotlib.pyplot as plt
333
334 def plot_cross_validation_results(resultados):
335     train_error = [1 - fold["Train_Accuracy"] for fold in
336         ↪ resultados]
337     test_error = [1 - fold["Test_Accuracy"] for fold in resultados]
338     fold_labels = [f"Fold_{i+1}" for i in range(len(train_error))]
339
340     data = pd.DataFrame({'Fold': fold_labels, 'Train_Error':
341         ↪ train_error, 'Test_Error': test_error})
342     data = data.melt('Fold', var_name='Dataset', value_name='Error'
343         ↪ )
344
345     plt.figure(figsize=(10, 6))
346     sns.barplot(x='Fold', y='Error', hue='Dataset', data=data)
347     plt.xlabel('Fold')
348     plt.ylabel('Error')
349     plt.title('Cross_Validation_Results_-_Error')

```

```

344     plt.legend()
345     plt.show()
346
347 plot_cross_validation_results(resultados)
348
349 #Grafico Error
350 import matplotlib.pyplot as plt
351
352 def plot_cross_validation_results(resultados):
353     train_error = [fold["Train_Error"] for fold in resultados]
354     test_error = [fold["Test_Error"] for fold in resultados]
355
356     plt.figure(figsize=(10, 6))
357     plt.plot(range(1, len(train_error) + 1), train_error, 'ro-',
358             ↪ label='Train_Error')
359     plt.plot(range(1, len(test_error) + 1), test_error, 'bo-',
360             ↪ label='Test_Error')
361     plt.xlabel('Fold')
362     plt.ylabel('Error')
363     plt.title('Cross_Validation_Results_-_Error')
364     plt.legend()
365     plt.show()
366
367 plot_cross_validation_results(resultados)
368
369 #Grafico de barras Error
370 import seaborn as sns
371 import matplotlib.pyplot as plt
372
373 def plot_cross_validation_results(results):
374     train_error = [fold["Train_Error"] for fold in results]
375     test_error = [fold["Test_Error"] for fold in results]
376     fold_labels = [f"Fold_{i+1}" for i in range(len(train_error))]
377
378     data = pd.DataFrame({'Fold': fold_labels, 'Train_Error':
379             ↪ train_error, 'Test_Error': test_error})
380     data = data.melt('Fold', var_name='Dataset', value_name='Error'
381             ↪ )
382
383     plt.figure(figsize=(10, 6))
384     sns.barplot(x='Fold', y='Error', hue='Dataset', data=data)
385     plt.xlabel('Fold')
386     plt.ylabel('Error')
387     plt.title('Cross_Validation_Results_-_Error')
388     plt.legend()
389     plt.show()

```

```

386
387 plot_cross_validation_results(resultados)
388
389 #GRAFICO Z1, Z2 Y Z3
390 from matplotlib.colors import ListedColormap
391
392 X_set, y_set = X_train[:, :3], y_train # Utilizar las tres
    ↪ primeras componentes principales
393 X1, X2, X3 = np.meshgrid(np.arange(start=X_set[:, 0].min() - 1,
    ↪ stop=X_set[:, 0].max() + 1, step=0.1),
394                        np.arange(start=X_set[:, 1].min() - 1,
    ↪ stop=X_set[:, 1].max() + 1, step
    ↪ =0.1),
395                        np.arange(start=X_set[:, 2].min() - 1,
    ↪ stop=X_set[:, 2].max() + 1, step
    ↪ =0.1))
396
397 # Realizar las predicciones para cada punto en la malla
398 Z = classifier.predict(np.array([X1.ravel(), X2.ravel(), X3.ravel()
    ↪ ]).T)
399
400 # Crear una malla 3D para la representación gráfica
401 Z = Z.reshape(X1.shape)
402 fig = plt.figure()
403 ax = fig.add_subplot(111, projection='3d')
404 ax.scatter(X_set[:, 0], X_set[:, 1], X_set[:, 2], c=y_set, cmap=
    ↪ ListedColormap(('green', 'red')))
405 ax.contourf(X1, X2, X3, Z, alpha=0.75, cmap=ListedColormap(('red',
    ↪ 'green')))
406 ax.set_xlim(X1.min(), X1.max())
407 ax.set_ylim(X2.min(), X2.max())

```

B.4. Árboles de decisión

```

1 # Importacion de librerias
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import pandas as pd
5 import seaborn as sns
6
7 import time
8 from sklearn import metrics, preprocessing, tree
9 from sklearn.metrics import f1_score, make_scorer
10 from sklearn.model_selection import cross_val_score

```

```

11 from sklearn.model_selection import GridSearchCV,
    ↪ RandomizedSearchCV
12
13 # Primero importamos el dataset
14 dataset = pd.read_excel('Corr_f1_f2_f3_CA1Contra211VarOriginalesAD.
    ↪ xlsx', dtype=float(), index_col=None)
15
16 #Formamos dos grupos
17 df1=dataset[dataset['Respuesta']==1]
18 df2=dataset[dataset['Respuesta']==2]
19
20 #Comprobamos el Supuesto Normalidad multivariante Prueba de Shapiro
    ↪ -wilks
21 #Por grupos
22 #Grupo 1
23 from scipy.stats import shapiro
24 estadistico, p_value = shapiro(df1)
25 print('Estadistico=%.3f,p_value=%.3f' %(estadistico,p_value))
26
27
28 #Grupo 2
29 from scipy.stats import shapiro
30 estadistico, p_value = shapiro(df2)
31 print('Estadistico=%.3f,p_value=%.3f' %(estadistico,p_value))
32
33
34 #Por variables
35 from scipy.stats import shapiro
36 for col in dataset:
37     estadistico, p_value = shapiro(dataset[col])
38     print('Estadistico=%.3f,p_value=%.3f' %(estadistico,p_value))
39
40
41 #Excluimos la columna 'Respuesta'
42 columns_to_plot = [col for col in dataset.columns if col != '
    ↪ Respuesta']
43
44 # Creamos el gráfico de pares
45 sns.pairplot(dataset[columns_to_plot])
46
47 #Comprobamos el Supuesto de homocedasticidad # Levene test
48 import pandas as pd
49 import scipy.stats as stats
50 for col in dataset:
51     estadistico, p_value = stats.bartlett(df1[col], df2[col] )
52     print('Estadistico=%.3f,p_value=%.3f' %(estadistico,p_value))

```

```

53
54 #Comprobamos el Supuesto de multicolinealidad
55 import seaborn as sns
56 import matplotlib.pyplot as plt
57
58 #Excluimos la columna 'Respuesta'
59 dataset_without_response = dataset.drop('Respuesta', axis=1)
60
61 # Calculamos la matriz de correlación
62 dataset1 = dataset_without_response.corr()
63
64 # Generamos el mapa de calor
65 f, ax = plt.subplots(figsize=(12, 9))
66 sns.heatmap(dataset1, vmax=0.8, square=True)
67 plt.show()
68
69 #LDA CLASIFICACIÓN
70 #Formado por todas las columnas excepto por la última
71 X = dataset.iloc[:, 0:11].values
72
73 #y es la última
74 y = dataset.iloc[:, -1].values
75
76 # el tamaño de x e y
77 X.shape, y.shape
78
79 # Dividimos el data set en conjunto de entrenamiento y conjunto de
80 ↪ testing
81 from sklearn.model_selection import train_test_split
82 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size
83 ↪ = 0.3, random_state = 0)
84
85 #Ajustamos el clasificador de Árbol de Decisión en el Conjunto de
86 ↪ Entrenamiento
87 from sklearn.tree import DecisionTreeClassifier
88 classifier = DecisionTreeClassifier(criterion = "entropy",
89 ↪ random_state = 0)
90 classifier.fit(X_train, y_train)
91
92 #Predecimos en el Conjunto de Testing
93 y_pred = classifier.predict(X_test)
94
95 #Creamos la matriz de confusión
96 from sklearn.metrics import confusion_matrix
97 cm = confusion_matrix(y_test, y_pred)
98

```

```

95 #Validación cruzada k fold con 5 matrices de confusion y errores
96 import numpy as np
97 from sklearn.model_selection import KFold, GridSearchCV
98 from sklearn.metrics import accuracy_score, precision_score,
    ↪ recall_score, f1_score
99 from sklearn.tree import DecisionTreeClassifier
100 import numpy as np
101 import matplotlib.pyplot as plt
102 from matplotlib.colors import ListedColormap
103
104 def cross_validation_with_hyperparameters(X, y, cv=5):
105
106     param_grid = {
107         'criterion': ['gini', 'entropy'],
108         'max_depth': [3],
109         'min_samples_split': [2, 5, 10],
110         'min_samples_leaf': [1, 2, 4]
111     }
112
113     classifier = DecisionTreeClassifier()
114
115     kf = KFold(n_splits=cv)
116     fold_results = []
117     test_error_sum = 0 # Variable inicializada
118
119     for train_index, test_index in kf.split(X):
120         X_train, X_test = X[train_index], X[test_index]
121         y_train, y_test = y[train_index], y[test_index]
122
123         grid_search = GridSearchCV(classifier, param_grid, cv=cv)
124         grid_search.fit(X_train, y_train)
125
126         best_classifier = grid_search.best_estimator_
127
128         y_train_pred = best_classifier.predict(X_train)
129         y_test_pred = best_classifier.predict(X_test)
130
131         fold_result = {
132             "Train_Accuracy": accuracy_score(y_train, y_train_pred)
133             ↪ ,
134             "Test_Accuracy": accuracy_score(y_test, y_test_pred),
135             "Train_Precision": precision_score(y_train,
136             ↪ y_train_pred),
137             "Test_Precision": precision_score(y_test, y_test_pred),
138             "Train_Recall": recall_score(y_train, y_train_pred),
139             "Test_Recall": recall_score(y_test, y_test_pred),

```

```

138     "Train_F1": f1_score(y_train, y_train_pred),
139     "Test_F1": f1_score(y_test, y_test_pred),
140     "Train_Error": 1 - accuracy_score(y_train, y_train_pred
141     ↪ ),
142     "Test_Error": 1 - accuracy_score(y_test, y_test_pred)
143 }
144
145     test_error_sum += fold_result['Test_Error']
146     fold_results.append(fold_result)
147
148     test_error_average = test_error_sum / cv
149
150     return fold_results, test_error_average, best_classifier
151
152
153 resultados, promedio_test_error, best_classifier =
154     ↪ cross_validation_with_hyperparameters(X_train, y_train, 5)
155 print("Resultados por Folds:", resultados)
156 print("Promedio de Test Error:", promedio_test_error)
157
158 # Creamos el gráfico del árbol de decisión
159 from sklearn.tree import DecisionTreeClassifier
160 from sklearn import tree
161 import matplotlib.pyplot as plt
162
163 feature_names = ['c1', 'c2', 'c3', 'mc4', 'c4', 'm2', 'm3', 'm4', '
164     ↪ skew', 'kurt', 'xp']
165
166 # Graficamos el mejor modelo de árbol de decisión
167 fig = plt.figure(figsize=(12, 10))
168 _ = tree.plot_tree(best_classifier,
169     feature_names=feature_names,
170     class_names=['1', '2'],
171     filled=True,
172     rounded=True,
173     impurity=True,
174     proportion=True,
175     precision=2)

```

B.5. Métricas

```

1 # Gráfico accuracy
2 import matplotlib.pyplot as plt

```

```

3
4 def plot_cross_validation_results(resultados):
5     train_acc = [fold["Train_Accuracy"] for fold in resultados]
6     test_acc = [fold["Test_Accuracy"] for fold in resultados]
7
8     plt.figure(figsize=(10, 6))
9     plt.plot(range(1, len(train_acc) + 1), train_acc, 'ro-', label=
10         ↪ 'Train_Accuracy')
11     plt.plot(range(1, len(test_acc) + 1), test_acc, 'bo-', label='
12         ↪ Test_Accuracy')
13     plt.xlabel('Fold')
14     plt.ylabel('Accuracy')
15     plt.title('Cross_Validation_Results')
16     plt.legend()
17     plt.show()
18
19 plot_cross_validation_results(resultados)
20
21 #Gráfico de barras accuracy
22 import seaborn as sns
23 import matplotlib.pyplot as plt
24
25 def plot_cross_validation_results(results):
26     train_acc = [fold["Train_Accuracy"] for fold in results]
27     test_acc = [fold["Test_Accuracy"] for fold in results]
28     fold_labels = [f"Fold_{i+1}" for i in range(len(train_acc))]
29
30     data = pd.DataFrame({'Fold': fold_labels, 'Train_Accuracy':
31         ↪ train_acc, 'Test_Accuracy': test_acc})
32     data = data.melt('Fold', var_name='Dataset', value_name='
33         ↪ Accuracy')
34
35     plt.figure(figsize=(10, 6))
36     sns.barplot(x='Fold', y='Accuracy', hue='Dataset', data=data)
37     plt.xlabel('Fold')
38     plt.ylabel('Accuracy')
39     plt.title('Cross_Validation_Results')
40     plt.legend()
41     plt.show()
42
43 plot_cross_validation_results(resultados)
44
45 #Gráfico precision
46 import matplotlib.pyplot as plt
47
48 def plot_cross_validation_results(resultados):

```

```

45     train_precision = [fold["Train_Precision"] for fold in
    ↪ resultados]
46     test_precision = [fold["Test_Precision"] for fold in resultados
    ↪ ]
47
48     plt.figure(figsize=(10, 6))
49     plt.plot(range(1, len(train_precision) + 1), train_precision, '
    ↪ ro-', label='Train_Accuracy')
50     plt.plot(range(1, len(test_precision) + 1), test_precision, 'bo
    ↪ -', label='Test_Accuracy')
51     plt.xlabel('Fold')
52     plt.ylabel('Precision')
53     plt.title('Cross_Validation_Results')
54     plt.legend()
55     plt.show()
56
57 plot_cross_validation_results(resultados)
58
59 #Grafico barras precision
60 import seaborn as sns
61 import matplotlib.pyplot as plt
62
63 def plot_cross_validation_results(results):
64     train_precision = [fold["Train_Precision"] for fold in results]
65     test_precision = [fold["Test_Precision"] for fold in results]
66     fold_labels = [f"Fold_{i+1}" for i in range(len(train_precision
    ↪ ))]
67
68     data = pd.DataFrame({'Fold': fold_labels, 'Train_Precision':
    ↪ train_precision, 'Test_Precision': test_precision})
69     data = data.melt('Fold', var_name='Dataset', value_name='
    ↪ Precision')
70
71     plt.figure(figsize=(10, 6))
72     sns.barplot(x='Fold', y='Precision', hue='Dataset', data=data)
73     plt.xlabel('Fold')
74     plt.ylabel('Precision')
75     plt.title('Cross_Validation_Results_-_Precision')
76     plt.legend()
77     plt.show()
78
79 plot_cross_validation_results(resultados)
80
81
82 #Gráfico recall
83 import matplotlib.pyplot as plt

```

```

84
85 def plot_cross_validation_results(resultados):
86     train_recall = [fold["Train_Recall"] for fold in resultados]
87     test_recall = [fold["Test_Recall"] for fold in resultados]
88
89     plt.figure(figsize=(10, 6))
90     plt.plot(range(1, len(train_recall) + 1), train_recall, 'ro-',
91             ↪ label='Train_Recall')
92     plt.plot(range(1, len(test_recall) + 1), test_recall, 'bo-',
93             ↪ label='Test_Recall')
94     plt.xlabel('Fold')
95     plt.ylabel('Recall')
96     plt.title('Cross_Validation_Results_-_Recall')
97     plt.legend()
98     plt.show()
99
100 plot_cross_validation_results(resultados)
101
102 # Grafico de barras recall
103 import seaborn as sns
104 import matplotlib.pyplot as plt
105
106 def plot_cross_validation_results(results):
107     train_recall = [fold["Train_Recall"] for fold in results]
108     test_recall = [fold["Test_Recall"] for fold in results]
109     fold_labels = [f"Fold_{i+1}" for i in range(len(train_recall))]
110
111     data = pd.DataFrame({'Fold': fold_labels, 'Train_Recall':
112             ↪ train_recall, 'Test_Recall': test_recall})
113     data = data.melt('Fold', var_name='Dataset', value_name='Recall
114             ↪ ')
115
116     plt.figure(figsize=(10, 6))
117     sns.barplot(x='Fold', y='Recall', hue='Dataset', data=data)
118     plt.xlabel('Fold')
119     plt.ylabel('Recall')
120     plt.title('Cross_Validation_Results_-_Recall')
121     plt.legend()
122     plt.show()
123
124 plot_cross_validation_results(resultados)
125
126 # Grafico F1
127 import matplotlib.pyplot as plt
128
129 def plot_cross_validation_results(resultados):

```

```

126     train_f1 = [fold["Train_F1"] for fold in resultados]
127     test_f1 = [fold["Test_F1"] for fold in resultados]
128
129     plt.figure(figsize=(10, 6))
130     plt.plot(range(1, len(train_f1) + 1), train_f1, 'ro-', label='
    ↪ Train_F1')
131     plt.plot(range(1, len(test_f1) + 1), test_f1, 'bo-', label='
    ↪ Test_F1')
132     plt.xlabel('Fold')
133     plt.ylabel('F1')
134     plt.title('Cross_Validation_Results_-_F1')
135     plt.legend()
136     plt.show()
137
138 plot_cross_validation_results(resultados)
139
140 #Grafico de barras F1
141 import seaborn as sns
142 import matplotlib.pyplot as plt
143
144 def plot_cross_validation_results(results):
145     train_f1 = [fold["Train_F1"] for fold in results]
146     test_f1 = [fold["Test_F1"] for fold in results]
147     fold_labels = [f"Fold_{i+1}" for i in range(len(train_f1))]
148
149     data = pd.DataFrame({'Fold': fold_labels, 'Train_F1': train_f1,
    ↪ 'Test_F1': test_f1})
150     data = data.melt('Fold', var_name='Dataset', value_name='F1')
151
152     plt.figure(figsize=(10, 6))
153     sns.barplot(x='Fold', y='F1', hue='Dataset', data=data)
154     plt.xlabel('Fold')
155     plt.ylabel('F1')
156     plt.title('Cross_Validation_Results_-_F1')
157     plt.legend()
158     plt.show()
159
160 plot_cross_validation_results(resultados)
161
162 #Grafico error de clasificación
163 import matplotlib.pyplot as plt
164
165 def plot_cross_validation_results(resultados):
166     train_error = [1 - fold["Train_Accuracy"] for fold in
    ↪ resultados]
167     test_error = [1 - fold["Test_Accuracy"] for fold in resultados]

```

```

168
169     plt.figure(figsize=(10, 6))
170     plt.plot(range(1, len(train_error) + 1), train_error, 'ro-',
171             ↪ label='Train_Error')
172     plt.plot(range(1, len(test_error) + 1), test_error, 'bo-',
173             ↪ label='Test_Error')
174     plt.xlabel('Fold')
175     plt.ylabel('Error')
176     plt.title('Cross_Validation_Results_-_Error')
177     plt.legend()
178     plt.show()
179
180 plot_cross_validation_results(resultados)
181
182 #Grafico de barras error de clasificacion
183 import seaborn as sns
184 import matplotlib.pyplot as plt
185
186 def plot_cross_validation_results(resultados):
187     train_error = [1 - fold["Train_Accuracy"] for fold in
188                 ↪ resultados]
189     test_error = [1 - fold["Test_Accuracy"] for fold in resultados]
190     fold_labels = [f"Fold_{i+1}" for i in range(len(train_error))]
191
192     data = pd.DataFrame({'Fold': fold_labels, 'Train_Error':
193                 ↪ train_error, 'Test_Error': test_error})
194     data = data.melt('Fold', var_name='Dataset', value_name='Error'
195                 ↪ )
196
197     plt.figure(figsize=(10, 6))
198     sns.barplot(x='Fold', y='Error', hue='Dataset', data=data)
199     plt.xlabel('Fold')
200     plt.ylabel('Error')
201     plt.title('Cross_Validation_Results_-_Error')
202     plt.legend()
203     plt.show()

```

Bibliografía

- [1] H. O. Alwan y Q. S-Al Sabbagh. “Various Types of Faults and Their Detection Techniques in Three Phase Induction Motors Fault”. En: *International Journal of Engineering Research and Applications* 07 (2017), págs. 24-33.
- [2] M. Astorgano. *Diagnóstico de fallos de rodamientos en motores de inducción en estado estacionario mediante técnicas boosting y redes neuronales*. Trabajo Fin de Grado. Universidad de Valladolid, 2022.
- [3] A. Barón. *Detección y Clasificación de Fallos en Motores mediante Procedimientos Boosting*. Trabajo Fin de Grado. Universidad de Valladolid, 2020.
- [4] D. Burrueco. *Machine Learning*. Recuperado de <https://interactivechaos.com/es/manual/tutorial-de-machine-learning/disposicion-de-la-matriz-de-confusion>. [Accedido el 13 de junio de 2023].
- [5] CCOO. *La Digitalización y la Industria 4.0*. Disponible en: <https://industria.ccoo.es/4290fc51a3697f785ba14fce86528e10000060.pdf>. [Accedido el 13 de junio de 2023].
- [6] S. J. Chapman. *Máquinas Eléctricas*. 6ta ed. McGraw-Hill Interamericana, 2008.
- [7] A. Chiba, T. Fukao y M. A. Rahman. “Vibration Suppression of a Flexible Shaft with a Simplified Bearingless Induction Motor Drive”. En: *Conference Record of the 2006 IEEE Industry Applications Conference Forty-First IAS Annual Meeting*. Vol. 2. 2006, págs. 836-842. DOI: 10.1109/IAS.2006.256622.
- [8] S. Daley, J. Hätönen y K. Tammi. “Instantaneous harmonic vibration control of a flexible rotor”. En: *Proceedings of the 2006 International Symposium on Active Control of Sound and Vibration*. 2006.
- [9] F. Filippetti, G. Franceschini y C. Tassoni. “Neural networks aided on-line diagnostics of induction motor rotor faults”. En: *Conference Record of the 1993 IEEE Industry Applications Conference Twenty-Eighth IAS Annual Meeting*. 1993, 316-323 vol.1. DOI: 10.1109/IAS.1993.298942.
- [10] L. Frosini et al. “A novel approach to detect short circuits in low voltage induction motor by stray flux measurement”. En: *2012 XXth International Conference on Electrical Machines*. 2012, págs. 1538-1544. DOI: 10.1109/ICE1Mach.2012.6350083.

- [11] Z. Gao, T.G. Habetler y R.G. Harley. “An Online Adaptive Stator Winding Temperature Estimator Based on a Hybrid Thermal Model for Induction Machines”. En: *IEEE International Conference on Electric Machines and Drives*. 2005, págs. 754-761. DOI: 10.1109/IEMDC.2005.195807.
- [12] L. A. García-Escudero et al. “Robust Detection of Incipient Faults in VSI-Fed Induction Motors Using Quality Control Charts”. En: *IEEE Transactions on Industry Applications* 53.3 (2017), págs. 3076-3085. DOI: 10.1109/TIA.2016.2617300.
- [13] J. Gil Flores. *Análisis Discriminante*. La Muralla, 2001.
- [14] C. Gil Martín. *Análisis de Componentes Principales (PCA)*. Recuperado de https://rpubs.com/Cristina_Gil/PCA. [Accedido el 13 de junio de 2023].
- [15] Q. He, F. Kong y R. Yan. “Subspace-based gearbox condition monitoring by kernel principal component analysis”. En: *Mechanical Systems and Signal Processing* 21.4 (2007), págs. 1755-1772.
- [16] Hyundai. *Historia del Motor Eléctrico*. Recuperado de <https://www.hyundai.com/canarias/es/blog/historia-del-motor-electrico/>. [Accedido el 13 de junio de 2023].
- [17] Mordor Intelligence. *Motor de inducción Tamaño del mercado, crecimiento (2022 - 27) | Análisis de la Industria*. Disponible en: <https://www.mordorintelligence.com/industry-reports/induction-motor-market>. [Accedido el 13 de junio de 2023].
- [18] A. J. Izenman. *Modern multivariate statistical techniques*. Vol. 1. Springer, 2008.
- [19] G. B. Kliman y J. Stein. “Methods of Motor Current Signature Analysis”. En: *Electric Machines & Power Systems* 20.5 (1992), págs. 463-474. DOI: 10.1080/07313569208909609.
- [20] R C Kryter y H D Haynes. “Condition monitoring of machinery using motor current signature analysis”. En: *Sound Vibrations* 23.9 (1989), págs. 14-21.
- [21] S. Kumar et al. “A Comprehensive Review of Condition Based Prognostic Maintenance (CBPM) for Induction Motor”. En: *IEEE Access* 7 (2019), págs. 90690-90704. DOI: 10.1109/ACCESS.2019.2926527.
- [22] P.C.M. Lamim Filho, R. Pederiva y J.N. Brito. “Detection of stator winding faults in induction machines using flux and vibration analysis”. En: *Mechanical Systems and Signal Processing* 42.1 (2014), págs. 377-387. ISSN: 0888-3270. DOI: <https://doi.org/10.1016/j.ymssp.2013.08.033>.
- [23] Y.-S. Lee et al. “An acoustic diagnostic technique for use with electric machine insulation”. En: *IEEE Transactions on Dielectrics and Electrical Insulation* 1.6 (1994), págs. 1186-1193. DOI: 10.1109/94.368645.
- [24] M. A. Martínez-Prieto, J. Silvestre y A. Bregón. *Marcos de Trabajo Ágiles*. Disponible en: https://uvadoc.uva.es/bitstream/handle/10324/37210/1_agile.pdf?sequence=7&isAllowed=y. [Accedido el 13 de junio de 2023].

-
- [25] N. Mehla y R. Dahiya. “An Approach of Condition Monitoring of Induction Motor Using MCSA”. En: *International Journal of Systems Applications, Engineering & Development* 1 (2007), págs. 13-17.
- [26] J.F. Moreno, F.P. Hidalgo y M.D. Martínez. “Realisation of tests to determine the parameters of the thermal model of an induction machine”. En: *IEE Proceedings: Electric Power Applications* 148.5 (2001), 393 – 397.
- [27] O.I. Okoro. “Steady and transient states thermal analysis of a 7.5-kW squirrel-cage induction machine at rated-load operation”. En: *IEEE Transactions on Energy Conversion* 20.4 (2005), págs. 730-736. DOI: 10.1109/TEC.2005.852965.
- [28] J. Orellana Alvear. *Árboles de Decisión y Random Forest*. Recuperado de <https://bookdown.org/content/2031/arboles-de-decision-parte-i.html>. [Accedido el 13 de junio de 2023].
- [29] B. S Payne, A Ball y F Gu. “Detection and Diagnosis on Induction Motor Faults using Statistical Measures”. En: *International Journal of Condition Monitoring and Diagnostics Engineering Management* 5.2 (2002), págs. 5-19.
- [30] D. Peña. *Análisis de Datos Multivariantes*. McGraw Hill, 2002.
- [31] S. Raman. *Linear Discriminant Analysis*/. Recuperado de <https://www.geeksforgeeks.org/ml-linear-discriminant-analysis/>. [Accedido el 13 de junio de 2023].
- [32] R.K. Singal, K. Williams y S.P. Verma. “Vibration behaviour of stators of electrical machines, part II: Experimental study”. En: *Journal of Sound and Vibration* 115.1 (1987), págs. 13-23. DOI: [https://doi.org/10.1016/0022-460X\(87\)90489-5](https://doi.org/10.1016/0022-460X(87)90489-5).
- [33] D.J.T. Siyambalapitiya y Peter G. McLaren. “Reliability improvement and economic benefits of online monitoring systems for large induction machines”. En: *IEEE Transactions on Industry Applications* 26.6 (1990), págs. 1018-1025. DOI: 10.1109/28.62371.
- [34] P.J. Tavner, B.G. Gaydon y D.M. Ward. “Monitoring generators and large motors”. En: *IEE Proceedings B (Electric Power Applications)* 133 (3 1986), 169-180(11).
- [35] J. W. Tukey. “Comparing Individual Means in the Analysis of Variance”. En: *Biometrics* 5.2 (1949), págs. 99-114.
- [36] Wikipedia. *Validación Cruzada*. Recuperado de https://es.wikipedia.org/wiki/Validaci3n_cruzada. [Accedido el 13 de junio de 2023].
- [37] Z.-H. Zhou. *Machine Learning*. 2nd. Cambridge, UK: Cambridge University Press, 2016.