



INSTITUTO TECNOLÓGICO SUPERIOR DE MISANTLA

MAESTRÍA EN SISTEMAS COMPUTACIONALES

IDENTIFICACIÓN DE MICROCALCIFICACIONES EN
MASTOGRAFÍAS DIGITALES EMPLEANDO DEEP
LEARNING E HIPERPARAMETRIZACIÓN

T E S I S

QUE PARA OPTAR POR EL GRADO DE:
Maestro en Sistemas Computacionales

PRESENTA:

José Aurelio Carrera Melchor

DIRECTOR:

Dr. Eddy Sánchez de la Cruz

CO-DIRECTORA:

Dra. María Victoria Carreras Cruz (externa)
Universidad Panamericana



Misantla, Veracruz, 2020



**INSTITUTO TECNOLÓGICO SUPERIOR DE MISANTLA
DIVISIÓN DE ESTUDIOS PROFESIONALES
AUTORIZACIÓN DE IMPRESIÓN DE TRABAJO DE TITULACIÓN MAESTRÍA**

FECHA: 03 de Julio de 2020.

ASUNTO: **AUTORIZACIÓN DE IMPRESIÓN
DE TESIS.**

A QUIEN CORRESPONDA:

Por medio de la presente se hace constar que el (la) C:

JOSÉ AURELIO CARRERA MELCHOR

estudiante de la maestría en SISTEMAS COMPUTACIONALES con No. de Control 162T0076 ha cumplido satisfactoriamente con lo estipulado por el **Lineamiento de Posgrado para la obtención del grado de Maestría** mediante **Tesis.**

Por tal motivo se **Autoriza** la impresión del **Tema** titulado:

**IDENTIFICACIÓN DE MICROCALCIFICACIONES EN MASTOGRAFÍAS
DIGITALES EMPLEANDO DEEP LEARNING E HIPERPARAMETRIZACIÓN**

Dándose un plazo no mayor de un mes de la expedición de la presente a la solicitud del examen para la obtención del grado de maestría.

ATENTAMENTE


Dr. Eddy Sánchez de la Cruz
Presidente




Dr. Luis Carlos Sandoval Herazo
Secretario


MSC. Galdino Martínez Flores
Vocal

Archivo.

VER. 01/03/09

F-SA-39

A Mi padre:

El cual ha derramado gotas de sudor para ganarse el pan y apoyarme en mi carrera profesional y aunque muchas veces los hice correr por algunos descuidos, sé que esta es una forma de responder el grandioso apoyo, base y fortaleza que han sido para mí.

A Mis hermanos:

Que han sido mi motivación en los tiempos más difíciles que hemos tenido como familia, su ternura, sus consejos, su apoyo incondicional, han sido armas enfrentar los problemas, aceptar los retos y tomar riesgos con el objetivo de salir adelante.

A Mis amigos:

Más que mis amigos, se han convertido en hermanos, compañeros de desvelos, risas, y presión para entregar algún trabajo, que han hecho ameno mi paso por mi maestría y mi estancia profesional.

A Mis Docentes:

Que se esforzaron en labrar en mí el conocimiento necesario para desempeñar y explotar los recursos a mi alcance para lograr prototipos, proyectos y desarrollos de calidad para la sociedad.

Agradecimientos

Mi gratitud, principalmente está dirigida al Dios todopoderoso por haberme dado la existencia y permitido llegar al final de mi maestría.

Mis más sinceras gracias a mi padre por todo el apoyo brindado desde el inicio de mi educación hasta el día de hoy, ya que juntos afrontamos grandes retos y levantamos éxitos, muchas gracias Aurelio Carrera y Oróstico.

Mis hermanos, Gabriela Itzel Carrera Melchor, José Eduardo Carrera Melchor motores e inspiración, por su apoyo incondicional y sincero.

Mis entrañables y mejores amigos Hugo Alberto Bravo compañero de aventuras, proyectos y grandes historias y Gabriela Cantellano Jarillo por su incondicional apoyo en los momentos más difíciles.

Mis amigos Héctor Miguel Quintero Carrillo, Rodolfo Zenil Cruz y Leonel Alejandro Velazquez que siempre fueron un refuerzo enorme en mi vida personal y profesional.

A mi asesor Dr. Eddy Sanchez de la Cruz, a mi co-asesora Dr. María Victoria Carreras Cruz, a mis docentes los doctores Rajesh Roshan Biswal, Roberto Ángel Armenta, Jorge Mario Figueroa, Luis Alberto Morales, y a mi Jefe de Maestría M.S.C. Galdino Matínez Flores.

Al Centro de Innovación & Transferencia de Tecnología donde encontré a excelentes personas que me brindaron su amistad y conocimientos Erik Balderas, Jorge Gerón, Erika Bandala, Guillermo Carmona y Sindy Mota.

Al Instituto Tecnológico Superior de Misantla y a la Universidad Panamericana por haber permitido desarrollar e investigar en sus instalaciones y brindarme su apoyo.

Al Consejo Nacional de Ciencia y Tecnología por el apoyo económico brindado para poder realizar mis estudios dentro del programa Nacional de Posgrados de Calidad.

Resumen

En México dentro de la gama de enfermedades crónicas se encuentra el cáncer, dentro de las variantes de esta enfermedad el cáncer de mama se presenta de manera frecuente en las mujeres alcanzando el primer lugar a nivel nacional, asimismo éste es la segunda causa de muerte por cáncer a nivel mundial. Esta tasa de mortandad se ha reducido gracias a la detección temprana derivada de un análisis correcto de una mastografía. Sin embargo existen pocos especialistas en materia capaces de dar un análisis correcto a las mastografías elevando el porcentaje de error para su detección oportuna y correcta.

Actualmente las mastografías digitales pueden ser asistidas por computadora y en esta investigación se toma como referencia la aplicación de preprocesamientos de imágenes en conjunto con Aprendizaje Profundo e Hiperparametrización para mejorar la eficiencia de detección.

En el presente documento se realiza un análisis del uso del datasetMIAS en bruto hasta su preprocesado máximo dividiéndose en 6 niveles, a su vez estos tienen 4 sub divisiones (Multiclase, Normal-Maligno, Normal-Benigno, y Benigno-Maligno) contando con un total de 24 Datasets sometidos a análisis cada uno con más de 30 algoritmos obteniendo resultados competitivos del 94.187% y con la integración de la Hiperparametrización de un 95.2512% .

Índice general

Resumen	XI
Índice de figuras	XIII
Índice de tablas	XV
1. Generalidades	1
1.1. Introducción	1
1.2. Planteamiento del problema	2
1.3. Justificación	4
1.4. Objetivo	4
1.4.1. Objetivo General	4
1.4.2. Objetivos Específicos	5
1.5. Hipótesis	5
1.6. Propuesta de solución	5
1.7. Alcances y Limitaciones	6
1.7.1. Alcances	6
1.7.2. Limitaciones	6
1.8. Estructura de la tesis	7
2. Marco Teórico	9
2.1. Patología	9
2.2. Imágenes médicas	9
2.2.1. Mastografía digital	9
2.3. Clasificación	10
2.3.1. Aprendizaje Profundo	10
2.3.1.1. Herramientas para implementar Aprendizaje Profundo	11
2.3.2. Hiperparametrizaciones	11
2.4. Microcalcificaciones	13
2.5. Métricas de Evaluación	14
2.5.1. Matriz de confusión	14
2.5.2. Espacio ROC: Sensibilidad y especificidad	14
2.5.3. Comparación con el Estado del Arte	14

3. Estado del arte	17
4. Metodología	19
4.1. Selección del Dataset	19
4.2. Clasificaciones	22
4.2.1. Clasificación 1 - Datos en bruto	22
4.2.1.1. Preprocesamiento	22
4.2.1.2. Procesamiento	25
4.2.1.3. Creación de Datasets	26
4.2.2. Clasificación 2 - Datos en bruto Zoom 250px	26
4.2.2.1. Preprocesamiento	27
4.2.2.2. Procesamiento	27
4.2.2.3. Creación de Datasets	28
4.2.3. Clasificación 3 - Histograma de imágenes	28
4.2.3.1. Preprocesamiento	28
4.2.3.2. Procesamiento	28
4.2.3.3. Creación de datasets	28
4.2.4. Clasificación 4 - Mejoramiento de imagen y músculo pectoral	29
4.2.4.1. Preprocesamiento	29
4.2.4.2. Procesamiento	34
4.2.4.3. Creación de datasets	36
4.3. Descripción de la Metodología del proceso general	36
5. Experimentos y resultados	41
5.1. Requerimientos	41
5.2. Resultados	42
5.2.1. Difusión de resultados	42
5.2.2. Análisis General	42
5.2.3. Análisis específico	43
5.2.4. Dataset Multiclase	44
5.2.5. Dataset Binario Normal Maligno	45
5.2.6. Dataset Binario Normal Benigno	45
5.2.7. Dataset Binario Benigno Maligno	46
6. Conclusiones y trabajos futuros	47
6.1. Conclusiones	47
6.2. Trabajos Futuros	48
Bibliografía	49
7. Anexos	53
7.1. Código Python	53
7.1.1. Generación de CSV con ID por fila	53
7.1.2. Generación de Histograma	55

7.2. Código Matlab	55
7.2.1. Filtros de imagen	55
7.2.2. Eliminación de objetos pequeños y ruido	56
7.2.3. Recorte en región seleccionada	56
7.3. Tablas de Resultados	57

Índice de figuras

2.1. Mastografía digital aplicada sobre una paciente.	10
2.2. Abstracción	11
2.3. Microcalcificación en el seno.	13
4.1. Cambio de dirección de las imágenes impares.	23
4.2. Patrón Alfa su corte se aprecia en color azul.	23
4.3. Patrón Beta su corte se aprecia en color amarillo.	23
4.4. Patrón Gamma su corte se aprecia en color verde.	24
4.5. Patrón Delta su corte se aprecia en color lila.	24
4.6. No existen cortes posibles sin alterar la estructura o homologar con sus afines.	25
4.7. Homologado de imágenes disminuyendo ruido y registros.	25
4.8. Homologado de imágenes disminuyendo ruido y registros.	26
4.9. Zoom 250px aplicado a una mastografía	27
4.10. Proceso de homolgado de dirección en elementos de numeración Impar.	29
4.11. Binarización de la mastografía y detección de elementos innecesarios en- marcados.	30
4.12. Eliminación de elementos pequeños en la mastografía.	30
4.13. Multiplicación de matrices y resultado	31
4.14. Equalización de mastografía.	31
4.15. Detección del espacio negativo.	32
4.16. Discriminación de espacio negativo y consolidación de imagen.	32
4.17. Proceso de corte largo e imagen resultante.	33
4.18. Aplicación de filtro de mediana.	33
4.19. Pasos de mejora de contraste para detectar el músculo pectoral.	34
4.20. Resta de matrices no favorable.	34
4.21. Resta de matrices favorable.	35
4.22. Resultado listo para el análisis final.	35
4.23. Metodología propuesta basada en (22).	36
4.24. Resultado listo para el análisis final.	37
5.1. Resultados obtenidos en la presente investigación.	44

Índice de tablas

2.1. Herramientas para la utilizacion de aprendizaje profundo	12
4.1. Equivalencia de tonos por canal (profundidad)	21
5.1. Relación de avances de efectividad de datasets	44
5.2. Algoritmos ensamblados aplicados al dataset multiclase	45
5.3. Algoritmos ensamblados aplicados al dataset binario Normal-Maligno .	45
5.4. Algoritmos ensamblados aplicados al dataset binario Normal-Benigno .	46
5.5. Algoritmos ensamblados aplicados al dataset binario Normal-Benigno .	46
7.1. Dataset 1-Multiclase	58
7.2. Dataset 1-Binario: Normal-Maligno	59
7.3. Dataset 1-Binario: Normal-Benigno	60
7.4. Dataset 1-Binario: Benigno-Maligno	61
7.5. Dataset 2-Multiclase	62
7.6. Dataset 2-Binario: Normal-Maligno	63
7.7. Dataset 2-Binario: Normal-Benigno	64
7.8. Dataset 2-Binario: Benigno-Maligno	65
7.9. Dataset 3-Multiclase	66
7.10. Dataset 3-Binario: Normal-Maligno	67
7.11. Dataset 3-Binario: Normal-Benigno	68
7.12. Dataset 3-Binario: Benigno-Maligno	69
7.13. Dataset 4-Multiclase	70
7.14. Dataset 4- Binario:Normal-Maligno	71
7.15. Dataset 4-Binario: Normal-Benigno	72
7.16. Dataset 4-Binario:Benigno-Maligno	73
7.17. Dataset 5-Multiclase	74
7.18. Dataset 5-Binario:Normal-Maligno	75
7.19. Dataset 5-Binario:Normal-Benigno	76
7.20. Dataset 5-Binario:Benigno-Maligno	77
7.21. Dataset 6-Multiclase	78
7.22. Dataset 6-Binario: Normal-Maligno	79
7.23. Dataset 6-Binario: Normal-Benigno	80

ÍNDICE DE TABLAS

7.24. Dataset 6-Binario: Benigno-Maligno 81

Generalidades

1.1. Introducción

El ser humano está constituido por aproximadamente 30 o 40 billones de células según estudios realizados tiempos modernos, todas ellas imprescindibles para que el cuerpo humano funcione a la perfección, el no poseer las suficientes o tener algún defecto en ellas puede derivar en consecuencias a la estabilidad del organismo (9). Consideradas como unidades de vida las células son capaces de reproducirse por si mismas, sin embargo este proceso de crecimiento puede alterarse, debido a que existen patologías que corrompen el estado normal de una célula, el cáncer es una de ellas pues se trata del crecimiento y diseminación células de manera incontrolada(32), en un ciclo normal de una célula esta se divide de una célula madre, una vez que esta se daña o envejece es remplazada y el ciclo se reinicia. Sin embargo en las células cancerígenas crecen anormalmente y sobreviven para volver a dividirse.

A nivel mundial el cáncer es la segunda causa de muerte y en 2015, ocasionó 8.8 millones de muertes lo que en parámetros mas breves significa que una de cada seis muertes en el mundo es debido a este padecimiento (32), asimismo cerca del 70% de los decesos se han registrado en países de ingresos medios y bajos.

En México, desde mediados del decenio comprendido entre 2000 y 2010 el cáncer, específicamente el cáncer de mama es el más frecuente en el segmento de población femenina(17), inicialmente este lugar era ocupado por el cáncer cervicouterino, sin embargo este ha sido contrarrestado a partir de una vacuna desarrollada en el año 2006 la cual lucha contra varios tipos de papiloma (virus generador del padecimiento), mientras que aún no se han podido determinar las causas del incremento en la aparición del cancer de mama. Actualmente la incidencia de un tumor maligno de mama entre la población de 20 años ha incrementado y se presentan actualmente 14.80 casos nuevos por cada 100 mil habitantes, y de acuerdo con cifras del INEGI el cáncer de mama representa una tasa de mortalidad de 15.8 fallecimientos por cada 100 mil mujeres en

este rango de edad, las estadísticas refieren que hasta 2015; seis mil 273 mujeres murieron, la mayoría en la Ciudad de México con 924 casos, seguidas del Estado de México con 602 y Veracruz con 361 a estas cifras se suman los registros de Guadalajara con 503 decesos, 405 en Monterrey, 263 en Chihuahua, 225 en Puebla, 221 en Tamaulipas, Michoacán 209 y 49 en Quintana Roo (3).

Para la detección existen diferentes estrategias las cuales podemos dividir en: físico y antecedentes, pruebas de laboratorio, pruebas genéticas y procedimientos con imágenes, siendo este último el más utilizado en este tipo de problemática, la mastografía es uno de ellos, la cual apoya al diagnóstico y tratamiento de la enfermedad, terapias y planificación de procedimientos, no conforme con ello este procedimiento ha demostrado ser el más viable en relación costo-beneficio (12), no obstante aún cuando las imágenes resultan ser efectivas para detectar una anomalía existe la posibilidad que una anomalía cancerígena no se pueda apreciar a simple vista y el médico especialista pueda dar un fallo erróneo.

Para disminuir la posibilidad de fallo se han instaurado conforme al desarrollo de la tecnología diseños asistidos por computadora por sus siglas en inglés CAD, los cuales proveen de herramientas que se pueden emplear en tareas como detección y segmentación de anomalías cancerígenas, así como su seguimiento y control, por lo cual se puede deducir que en un futuro no muy lejano esta será frecuentemente usada en esta problemática, lo cual nos da como resultado la unión de los métodos clásicos de detección de cáncer de mama como la mastografía sumado a la aplicación de metodologías de procesamiento de imágenes y su posterior análisis con el empleo de algoritmos de aprendizaje profundo brindar mejores resultados para la detección de estados normales, benignos y malignos que pueda presentar en paciente atendido.

El aprendizaje profundo, en inglés Deep Learning, es una técnica frecuentemente recurrida en el área médica, que de acuerdo a la literatura consultada en la presente tesis tiene un comportamiento aceptable y en muchos casos competitivos comparandose incluso con el de un humano(29), aclarando que éstas herramientas no lo sustituyen, sino que ofrecen una alternativa o segunda opinión al experto para la futura toma de decisiones sobre las medidas a tomar, esta investigación aborda herramientas de decisión para detectar anomalías cancerígenas clasificándolas de acuerdo a su potencial estado.

1.2. Planteamiento del problema

El cáncer es una enfermedad considerada grave desde hace siglos y a nivel global es uno de los padecimientos que mayor incidencia el cual se ha reforzado a lo largo de los últimos años, afectando severamente a la población, según la Organización Mundial de la Salud (OMS) este ocupa el segundo lugar en causa de muerte provocando 8,8

millones de defunciones en 2015.

En México, desde el año 2006 el cáncer de mama es el más frecuente en el segmento de población femenina, para ser precisos en este año se superó al cáncer cervicouterino ya que para este último surgió una vacuna contra varios tipos de papiloma (virus generador del padecimiento), mientras que aún no se han podido determinar las causas del incremento en la aparición del cancer de mama.

Según cifras del Instituto Nacional de Estadística y Geografía (INEGI)(17), para el año 2012, se reportó cifra de 26.64 casos por cada 100,000 mujeres mayores de 20 años y para el año 2016 se registraron 16 defunciones por cada 100 000 mujeres de en el mismo grupo de edad.

Aunado a ello, de acuerdo a la Organización Panamericana de la Salud (OPS), estima que en los próximos años el número de personas que presentarán la enfermedad aumente en 46 % puesto que además que el cáncer de mama es uno de los más frecuentes, también que no existe manera de prevenirlo ya que la única forma de enfrentarlo es la detección temprana para lo que la mastografía es la mejor herramienta costo beneficio y es el único estudio que ha demostrado que su utilización disminuye la mortalidad por cáncer de mama hasta en un 30 %.(13)

Por lo anterior, las mastografías funcionan como una clave preventiva para disminuir la mortalidad por este padecimiento ya que, al realizar las mastografías se pueden detectar los signos de posible cáncer mamario y en caso de localizarlo aplicar tratamientos correctivos como la quimioterapia, radioterapia, cirugía y hormonoterapia para evitar que el cáncer de mama avance a otros segmentos del cuerpo y el paciente muera por esta causa.

La detección temprana permite disminuir la mortalidad por este padecimiento, de acuerdo a la OMS la probabilidad de cura en esta etapa es mayor. No conforme con ello es indispensable que las evaluaciones de las mamografías sean evaluadas por expertos certificados para interpretar estos exámenes de los cuales de acuerdo a la Dra. Carreras(11) existen poco más de 40 en México.

Es debido a la escasez de expertos en México que es necesario que las lecturas de las imágenes de forma, textura, tamaño y localización de microcalcificaciones sean detectadas con precisión apoyando a los expertos a mejorar la velocidad de interpretación.

Es por ello que la implementación de nuevas tecnologías como la digitalización de imágenes en conjunto con su análisis se convierte en un factor indispensable para abordar la problemática presentada.

1.3. Justificación

Actualmente no existe una entidad exacta que permita saber el origen del cáncer de mama, sumado a la baja cantidad de especialistas capaces de leer adecuadamente los análisis es imprescindible que la detección temprana juegue un rol puntero disminuir la cantidad de decesos en la población mexicana, ya que el beneficio principal para el paciente con un diagnóstico oportuno en las primeras etapas de desarrollo de la patología lo que aumenta las probabilidades de erradicar la enfermedad y disminuir los fallecimientos.

Asimismo la implementación de Aprendizaje profundo ha sido una herramienta de ayuda para la detección de enfermedades aportando información útil para el médico especialista ya que con un apoyo de inteligencia artificial ésta ayuda al experto en la toma final de desiciones para el diagnóstico final y tratamiento oportuno con beneficios que se presentan a continuación:

Impacto tecnológico. Sumando una nueva estrategia a la medicina para apoyar de forma positiva al diagnóstico oportuno de acuerdo al posible estado del paciente, aprovechando las capacidades de la inteligencia atificial para la categorización y estado final.

Impacto económico. Tomando como referencia las instancias de salud en México hasta el paciente; con una disminución la cantidad de pruebas o repetición de las mismas para asegurar el resultado final, decrementando el traslado de equipo para realizar esta prueba disminuyendo el costo.

Impacto social. Al categorizar anormalidades y su estado correctamente se apoya al especialista disminuyendo la cantidad de tiempo y material necesario para realizar un dignóstico certero, dando como resultado la detección en primeras etapas y por consecuencia aumentando la esperanza de vida aportando información para el tratamiento de la enfermedad.

1.4. Objetivo

1.4.1. Objetivo General

Identificar correctamente microcalcificaciones en mastografías digitales aplicando diversas técnicas de preprocesamiento en suma con algoritmos de aprendizaje profundo y empleando el método de hiperparametrización

1.4.2. Objetivos Específicos

1. Crear u obtener una base de datos de imágenes de mastografías digitales.
2. Aplicar el modelo de eliminación de ruido que resalte la imagen de las microcalcificaciones.
3. Aplicar segmentación de la imagen de microcalcificaciones ubicadas.
4. Definir los datasets para entrenamiento y pruebas.
5. Implementar sobre los datasets definidos algoritmos de aprendizaje profundo e hiperparametrización.
6. Analizar y validar los resultados empleando métricas de evaluación estadísticas.

1.5. Hipótesis

Es posible, a través de una base de mastografías digitales, aplicar una metodología de preprocesamiento para resaltar las posibles microcalcificaciones, posterior a ello implementar algoritmos de aprendizaje profundo para mejorar el diagnóstico asimilando un posible estado y con ello apoyar a la toma de medidas correctivas en pacientes propensas a padecer cancer de mama en el menor tiempo posible.

1.6. Propuesta de solución

El proyecto que a continuación se describe contará con 5 etapas:

En la primera etapa, se da la búsqueda de un dataset de mastografías digitales, el cual presente información detallada sobre el contenido detallando:

- El tipo, si es normal o presenta alguna anormalidad.
- Si la anormalidad (en caso de detectarse) es de características malignas o benignas.
- La posición (coordenadas x & y) de la anormalidad en caso de presentarse ya que ayudará al diagnóstico final.

En la segunda etapa, se realiza un pre-procesamiento de las imágenes para mejorar la detección de posibles microcalcificaciones, las cuales consisten desde recorte de áreas no reelevantes hasta la aplicación de filtros en zonas especiales de la imagen.

En la tercera etapa se realiza un proceso con el cual se obtengan los datasets necesarios a través de Python por cada una de las pruebas pertinentes abarcando las distintas imágenes generadas, asimismo se aplicará algoritmos ensamblados y sus respectivas pruebas con cada uno de ellos que ofrece la herramienta Weka con ello se pretende obtener los mejores resultados de precisión y basados en la sensibilidad y especificidad.

En la cuarta etapa, a través del convenio realizado con la Universidad Panamericana y colaboración de la Dra. Carreras, se realiza una estancia en dicha universidad con el fin de mejorar la calidad de mejora de la imagen en procesamiento para su posterior análisis, se le aplicará un proceso similar a la tercera etapa con algoritmos ensamblados y obteniendo los mejores resultados.

En la quinta etapa se realizó un análisis de ambas pruebas con los distintos datasets, para obtener lo mejores resultados apoyados de las pruebas de validación, con lo cual se pretende alcanzar los objetivos planteados en el presente trabajo.

1.7. Alcances y Limitaciones

1.7.1. Alcances

A través del análisis de mamografías digitalizadas procesadas en imágenes resultado del proceso que resalte la imagen de la posible masa tumoral o las microcalcificaciones y su posterior conversión en datos para su procesamiento e implementación de algoritmos de aprendizaje profundo que tienen como resultado parámetros que apoyen al experto en una rápida identificación de cáncer en pacientes femeninas con el fin de apoyar y facilitar el trabajo de radiólogos encargados de este rubro disminuyendo los tiempos de diagnósticos y disminuyendo el porcentaje posible de error en ellos.

1.7.2. Limitaciones

En la presente investigación podemos definir 2 tipos de instancias que pueden derivar en acotar los resultados las cuales se mencionan a continuación:

- **Tiempo:** Debido que la investigación trabaja sobre una base de imágenes el análisis fue tardado debido a la dimensión que presentan la mayoría de las mastografías digitales y el poder computacional requerido, ya que a pesar de poder solicitarse a instancias que cuenten con este material el proceso de petición y uso puede retrasó la investigación.
- **Recursos:** Se tomaron únicamente bancos de imágenes públicas que regularmente son de un espectro menos detallado ya que las bases actuales son difíciles de conseguir con las instancias correspondientes ya que al contener datos sensibles existen barreras de permiso de uso y datos privados.

1.8. Estructura de la tesis

Este trabajo está dividido en 8 capítulos, los cuales se describen como:

Capítulo 1. Generalidades, el cual abarca los conceptos básicos de la investigación desde la problemática hasta las limitaciones para la realización de la presente investigación.

Capítulo 2. Marco teórico, donde se toma la información que sustenta el presente trabajo recopilando terminología y elementos informativos.

Capítulo 3. Estado del arte, donde se referencian y analizan los trabajos previos de autores con metas afines a las de la presente investigación así como sus resultados.

Capítulo 4. Metodología, donde se realiza un recorrido a través de los distintos pasos para obtener los diferentes resultados en los diferentes datasets obtenidos en la presente investigación.

Capítulo 5. Experimentos y resultados, donde se hace un recuento del software y hardware que se empleó para la obtención de resultados asimismo se dan a detalle los resultados de las métricas obtenidas, así como también los diferentes comparaciones posibles entre los diferentes procesos..

Capítulo 6. Conclusiones y trabajos futuros, donde se analiza y balanceran los propósitos al inicio de la investigación y los resultados finales, si fueron alcanzados los propósitos iniciales y en que medida para sostener el propósito de la presente investigación asimismo se abordan los trabajos futuros donde se declaran las modificaciones o adiciones al proceso para mejorarlo en investigaciones posteriores para obtener una mejora.

Capítulo 7. Anexos, donde se integran las diferentes tablas, código o registros que por su dimensión no pueden estar presentes dentro del contexto general, sin embargo son vitales para el desarrollo de la investigación.

Capítulo 2

Marco Teórico

2.1. Patología

El término “patología”, sinónimo de anatomía patológica(21), tiene dos raíces, la palabra deriva del griego pathos y logos, que significan “enfermedad” y “estudio de”, respectivamente. La enfermedad es un estado anormal de la vida y la anatomía patológica, es el campo de la medicina que trata de conocer y explicar, con base racional, las condiciones bajo las que se da, sustentándose en el estudio de todos sus aspectos: 1) las causas, 2) los mecanismos de producción o patogénesis, 3) los cambios estructurales en células, tejidos y órganos y 4) las consecuencias funcionales de dichos cambios, expresados como síntomas y signos.

2.2. Imágenes médicas

Una imagen médica es aquella obtenida a través de técnicas y procesos usados para crear imágenes del cuerpo humano o secciones que lo conforman, con el propósito clínico de revelar, diagnosticar u observar el proceso de enfermedades o patologías.

Las imágenes médicas se han convertido en un instrumento indispensable ya que en la actualidad son capaces de identificar patologías con alto grado de antelación, aunque su utilización se encuentra en mayor parte dentro de la radiología estas se han extendido añadiendo elementos computacionales basados en un proceso indispensable antes de una cirugía (26).

2.2.1. Mastografía digital

La Mastografía de tamizaje es el método que ha demostrado que puede disminuir la mortalidad por cáncer de mama mediante la detección temprana (1). Por este método se detectan lesiones no palpables de mama, se valora anomalías palpables, dolor, secreción



Figura 2.1: Mastografía digital aplicada sobre una paciente.

por el pezón o un cambio en el tamaño o forma de la mama aún en presencia de implantes.

Consiste en tomar 2 radiografías de cada mama: de los lados y de arriba hacia abajo. Tarda aproximadamente 15 minutos y causa molestias que son tolerables ya que se requiere comprimir el pecho, cabe mencionar que no todos los resultados anormales significan cáncer, de cada 10 mujeres con mastografía normal, sólo una tiene cáncer (ver Figura 2.1). Por lo tanto, si tienes un resultado anormal, será necesario realizar más estudios para tener un diagnóstico definitivo. Si el estudio no es adecuado por errores en la toma se te pedirá que se lo repita (7).

Es un estudio radiológico de la mama con baja dosis de radiación que permite obtener imágenes de alta resolución las cuales son valoradas por el medico radiólogo en monitores especiales, donde se puede ajustar el brillo, el contraste y realizar ampliación de la imagen para valorar áreas de interés específico.

2.3. Clasificación

2.3.1. Aprendizaje Profundo

Deep Learning es una disciplina que consiste en la búsqueda de patrones a través de abstracciones profundas, estas se logran con el análisis de múltiples capas ocultas en una Red Neuronal Artificial (5). Podemos mencionar que podemos obtener una abstracción en una capa oculta, por ejemplo el borde de un área de interés en una imagen seleccionada.

Para conseguir abstracción, en una capa oculta, por ejemplo, el borde del área de interés se selecciona una imagen (ver Figura 2.2). La profundidad se alcanza repitiendo la abstracción tanto en las capas ocultas como en el número de neuronas que puede ir de 1 a n , una vez estableciéndolas se intenta resolver la problemática de acuerdo a las dimensiones y las propiedades de los datos en función de la experiencia de la Red Neuronal Artificial.

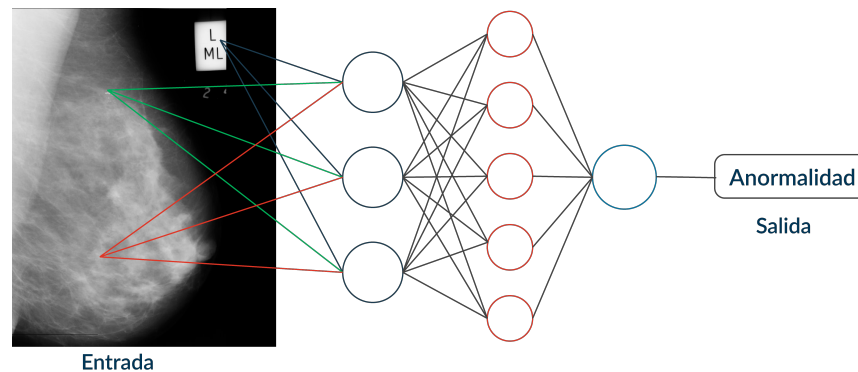


Figura 2.2: Abstracción

El resultado de ésta será lo que buscamos, en este caso microcalcificaciones, el cual corresponderá a una clasificación de acuerdo a las propiedades de la imagen que en este caso es un grupo de puntos.

2.3.1.1. Herramientas para implementar Aprendizaje Profundo

Existe una amplia variedad de herramientas en las que se puede implementar el aprendizaje profundo. (ver Tabla 2.1), Estas son los más representativos y se pueden encontrar en la web como código abierto, están orientadas a cualquier persona con poca experiencia en programación o personas autodidactas la cuales con un poco de paciencia, puedan explorar y utilizar de manera práctica el Aprendizaje Profundo

2.3.2. Hiperparametrizaciones

El software de aprendizaje automático WEKA (Hall et al., 2009) nos ayuda a poner técnicas de aprendizaje automático a disposición de muchos usuarios, resolviendo la incógnita de que surge al elegir entre las docenas de procedimientos de aprendizaje automático implementados en el software WEKA y la configuración de cada procedimiento para lograr un buen rendimiento.

AutoWeka aborda este problema al tratar todos los algoritmos como uno solo, aumentando el marco métrico de aprendizaje automático, (18) y el uso de la optimización bayesiana para encontrar un fuerte frecuencia para un conjunto de datos dado. Específicamente, considera el espacio combinado del aprendizaje de WEKA los algoritmos:

$$A = \{A^1, \dots, A^k\}$$

Tabla 2.1: Herramientas para la utilizacion de aprendizaje profundo

Software	Developer	website
Computational Network Toolkit CNTK	Microsoft Research	CNTK.ai
Deeplearning4j	Adam Gibson	deeplearning4j.org
Caffe	Berkeley Vision and Learning Center	caffe.berkeleyvision.org
Theano	University of Montreal's	deeplearning.net
Thorch	Ronan Collobert, Thorch Kuray Kavukcuoglu and Clement Farabet	torch.ch
Tensor Flow	Google Brain Team	tensorflow.org
WEKA	Machine Learning group at the University of Waikato	weka

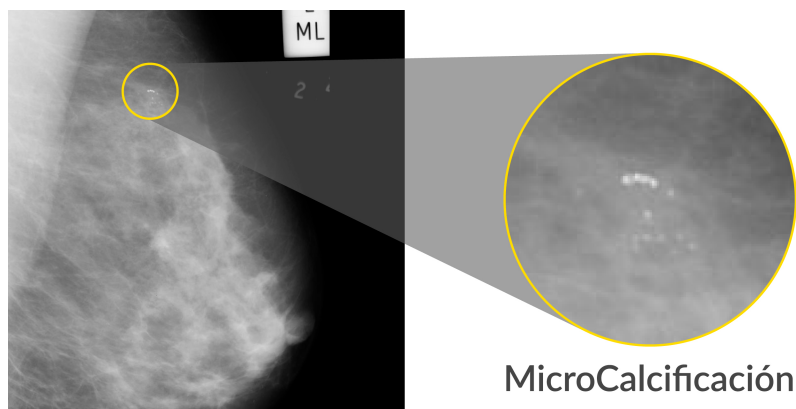


Figura 2.3: Microcalcificación en el seno.

y sus espacios de hiperparámetros asociados

$$A^1, \dots, A^k$$

y tiene como objetivo identificar la combinación de algoritmo

$$A^j \in \mathcal{A}$$

y los hiperparámetros

$$\lambda \in \Lambda^j$$

que minimiza la pérdida en validación cruzada.

$$A_{\lambda^*}^* \in \underset{A^{(j)} \in \mathcal{A}, \lambda \in \Lambda^j}{\operatorname{argmin}} \frac{1}{k} \sum_{i=1}^k \ell \left(A_{\lambda}^{(j)}, D_{train}^{(i)}, D_{test}^{(i)} \right)$$

2.4. Microcalcificaciones

Las micro-calcificaciones en los senos (ver Fig. 1.) son hallazgos frecuentes en la mastografía digital, la mayoría de ellas originadas por patologías benignas, las cuales pueden llegar a ser malignas, especialmente en carcinomas, cuya detección es difícil debido a su pequeño tamaño y a la falta de pericia para tomar y considerar lecturas precisas de la forma, textura, tamaño y ubicación. Sin embargo, la tasa de detección puede mejorar con programas asistidos por computadora que implementen algoritmos de clasificación de Aprendizaje profundo e hiperparametrización mejoran en gran medida su hallazgo (ver Figura 2.3).

2.5. Métricas de Evaluación

2.5.1. Matriz de confusión

Se le denomina a aquella que clasifica todos los casos del modelo en categorías, determinando si el valor predicho coincide con el valor real. Todos los casos en cada categoría se cuentan y los totales se muestran en la matriz la cual podemos delimitar como es una herramienta estándar para la evaluación de modelos estadísticos.

Una matriz de confusión es una herramienta importante para evaluar los resultados de la predicción porque facilita la comprensión y la explicación de los efectos de las predicciones incorrectas ya que al ver la cantidad y porcentajes en cada celda de esta matriz, puede ver rápidamente con qué frecuencia el modelo predijo con precisión.

2.5.2. Espacio ROC: Sensibilidad y especificidad

Dados los trabajos relacionados citados se integran en la presente investigación variables de evaluación importantes las cuales se describen a continuación:

Sensibilidad: La capacidad de prever los casos positivos cuando realmente son enfermos o con presencia de microcalcificaciones, es decir la capacidad para detectar enfermedad en mastografías con signos de microcalcificaciones.

$$sensibilidad = \frac{VP}{VP + FN}$$

Especificidad: La cual brindará información de casos negativos de los que son realmente sanos y la proporción de sanos correctamente identificados, es decir la capacidad de detectar la enfermedad en mastografías de características sanas.

$$especificidad = \frac{VN}{VN + FP}$$

La sensibilidad es la fracción de verdaderos positivos y la especificidad la fracción de verdaderos negativos (FVN).

No obstante, cada prueba tiene mayor o menos porcentaje tanto en especificidad (SP) como en sensibilidad (SE) dependiendo de su punto de corte. Lo ideal sería que no se crucen y al hacer un test solamente obtuviésemos verdaderos positivos y verdaderos negativos pero no es así. Por ello lo importante conocer la especificidad y la sensibilidad de cada estimador que usemos.

2.5.3. Comparación con el Estado del Arte

Comparando estos resultados con los trabajos previos, los clasificadores propuestos en el presente trabajo superan los resultados obtenidos en Arevalo et al. (6) con 71.62 % de efectividad y a Dalmi et al. (14) con una tasa de efectividad de 85.52 %. Sin embargo,

Arafi et al. (4) obtuvo un 94.74%, Carreras et al. (12) obtuvo 95%. Neto et al. (23) y Pedraza et al. (24) obtuvieron un porcentaje de 95.2 y 95.62% respectivamente. Por último, cabe mencionar a Moradkhani et al. (22) que obtuvieron una efectividad del 97%.

Capítulo 3

Estado del arte

A continuación, se enlista algunos trabajos de investigación en el ámbito médico y social sobre el cáncer de mama, estas investigaciones se apoyaron en técnicas de inteligencia artificial para dar solución a la problemática a resolver, cabe mencionar que se encuentran en el rubro de predicción, clasificación, detección entre otros. La presente sección se encuentra ordenada de menor a mayor de acuerdo al porcentaje de efectividad:

- Moradkhanian et al., basado en la extracción de imágenes del miniMIAS, cortó y removió la información adicional para luego usar un filtro en la imagen, obteniendo datos para después ser clasificados y obtener un método que brinda el 97 % (22).
- Pedraza et al., se entrenó una red neuronal convolucional basada en la arquitectura GoogLeNet, para desarrollar el modelo, después se llevó a cabo un proceso de validación cruzada. Así, el algoritmo proporciona una precisión del 95.62 % para un conjunto de 5750 instancias (24).
- Neto et al., para automatizar la segmentación de masas en mamografías, se utilizó la optimización de enjambre de partículas (PSO) y graph clusters, logrando un 95.2 % de efectividad (23).
- Carreras-Cruz et al., para atender la problemática de clasificar anomalías cancerígenas, utilizaron el dataset (MIAS, 1994), en el cual implementaron un algoritmo de agrupamiento parcial k-means y como resultado un único falso positivo, de la imagen mdb026. El resultado fue 95 % de confianza en la clasificación de tipos de cáncer en imágenes mastográficas (12).
- Arafí et al., implementaron un método para la detección de cáncer basado en SVM ¹, como técnica de aprendizaje supervisado para clasificar datos empíricos. Así se optimiza el rendimiento del clasificador resultante, de esta forma se tiene

¹SVM en inglés: Support Vector Machine

3. ESTADO DEL ARTE

un resultado de 94.74 %.(4)

- Gerazov et al., aplicaron métodos de aprendizaje profundo a un conjunto de datos de dominio de tiempo en mama homogénea del tejido adiposo. Aquí emplean redes neuronales convolucionales, así como el clasificador de entrada SVM para una precisión de 93.44 % (16).
- Lévy et al., implementaron un modelo integral de aprendizaje profundo para clasificar las masas mamarias pre-detectadas de mamografías, se utilizó la arquitectura AlexNet y GoogLeNet, obteniendo con la última el mayor porcentaje exactitud de 92.9 % (20).
- Al-Masni et al., el sistema CAD¹ que se realizó para la detección de masas de seno y clasificación de cáncer, implementó una CNN², para una eficacia de 93.20 % al clasificar imágenes benignas, mientras que las malignas un 78 % de efectividad, y su porcentaje global de 85.52 % para clasificar anomalías (2).
- Dalmi et al., Utilizaron el algoritmo ensamblado Random forests, combinándolo con una CNN, para la clasificación de lesiones, obteniendo el 85 % de efectividad al distinguir los distintos tipos de anomalías (14).
- Arevalo et al., se utilizó un enfoque híbrido donde las redes neuronales convolucionales se utilizan para aprender la representación de forma supervisada, obteniendo un porcentaje de 82 % (6).
- Cruz-Roa et al., el enfoque utilizado fue evaluar la exactitud y robustez de un método basado en el aprendizaje profundo para identificar automáticamente la extensión del tumor invasivo en las imágenes digitalizadas, este arrojó un de 75.86 % de instancias detectadas y un valor predictivo positivo de 71.62 % (27).
- Camacho et al., se implementó el método heurístico basado en minería de datos para extraer información esencial de las imágenes mamográficas y transformarlas en patrones (10).

En las investigaciones antes mencionadas podemos percatarnos de una elevación del porcentaje global con el uso de Deep Learning dentro del desarrollo del proyecto, la diferencia de porcentajes radica en el pre-procesamiento y procesamiento de sus datos base, es por ello que en la presente investigación pretende como un nicho de oportunidad aprovechar este puntero modificando variables tangibles que podemos mencionar como; un dataset diferente, pruebas con diferentes clasificaciones, la integración procesos híbridos en software con el fin de obtener una respuesta alterna a lo ya presentado pudiendo igualar o superar los porcentajes obtenidos.

¹Computer Aided Design (CAD) en español: Diseño Asistido por Computadora

²CNN en inglés: Convolutional Neural Network

Metodología

Para el desarrollo de la presente tesis se hace el uso de 4 pilares básicos los cuales se desempeñan de acuerdo al siguiente orden: La selección del Dataset, las clasificaciones, estas se pueden dividir en sub segmentos de acuerdo al avance de la misma, el análisis de los datos arrojados con Aprendizaje profundo con las respectivas variables y juego de algoritmos correspondientes para finalizar con una comparativa de resultados que se presentan a lo largo de las siguientes páginas.

4.1. Selección del Dataset

En paralelo con el objetivo de la presente investigación, el cual se define como la comparación de algoritmos de aprendizaje profundo e hiperparametrización se requiere de utilizar una base de datos estandarizada para que en un periodo determinado se puedan comparar los resultados. Aunque la mayoría de estas bases de datos no se encuentran disponibles al público existen bases de datos de proyectos antiguos reutilizables ya que en la actualidad hay pocos proyectos que desarrollen nuevas bases de imágenes de mastografías digitales.

De acuerdo a la las características solicitadas en la propuesta de solución para la presente investigación se requiere de una base de datos de mastografías digitales que especifiquen el tipo de imagen, es decir si está dentro los parámetros normales o si presenta alguna anormalidad, asimismo en caso de presentar alguna anormalidad si tiene características malignas y benignas, posterior a ello si cuenta con la posición de la anormalidad en coordenadas x y y ya que esto ayuda al diagnostico final. Dentro de las posibles candidatas para los fines establecidos se encuentran:

- **B-SCREEN.** Por sus siglas en inglés Bayesian Decision Support in Medical Screening, proyecto iniciado en 2006 es una digitalización de cribados holandeses del cáncer de mama almacenados en un archivo nacional, lo que facilita su uso en estudios posteriores, a la fecha aún no se encuentra disponible al 100% sumado a que para su uso se requiere poder de procesamiento debido a la gran cantidad

4. METODOLOGÍA

de datos ¹.

- DDSM. Por sus siglas en inglés Digital Database for Screening Mammography, esta base de datos fue desarrollada por el hospital General de Massachusetts, los Laboratorios Nacionales Sandia y el Departamento de Ingeniería e Informática de la Universidad del Sur de la Florida. Este dataset contiene 2.500 estudios, cada uno de ellos con dos imágenes de cada seno y anexa la información asociada con el paciente (edad, clasificación de densidad del seno, clasificación de anormalidades, descripción breve de las anormalidades e información de la imagen). Estas imágenes contienen áreas sospechosas ².
- AMDI. Atlas indexado de mamografías digitales AMDI por sus siglas en inglés, es una herramienta que permite al usuario descargar casos de la base de datos de mamografías, cada examen incluye cuatro vistas (dos vistas de cada seno: CC³ y OML⁴) los casos incluye; el límite del músculo pectoral (solo vistas OML), los contornos de las masas (si existen), las regiones de grupos de micro calcificaciones y el número (si existen), sumado a las ubicaciones y detalles de cualquier otra característica de interés⁵.
- NMD. Es el resultado de un análisis de 90 instalaciones que realizan mastografías, donde se recabó y analizó con éxito los datos de 3,181,437 mamografías de detección realizadas entre enero de 2008 y diciembre de 2012 la cual incluye información demográfica, hallazgos clínicos, la interpretación y los resultados de la biopsia⁶.
- IRMA. Es un proyecto relizado por la División de Procesamiento de Imágenes Médicas, la Cátedra de Informática VI de la Universidad Tecnológica de Aquisgrán, Departamento de Radiología Diagnóstica y el Departamento de Informática Médica, donde el objetivo se centra en la recuperación de imágenes basadas en contenido con aplicación prototípica a tareas médico-diagnósticas en un archivo de imágenes radiológicas ⁷.
- miniMIAS. Por sus siglas en inglés Mammographic Image Analysis Society, es una colección de 322 imágenes tomadas dentro del Programa Nacional de Evaluación del Seno del Reino Unido, las cuales fueron digitalizadas a un borde de 50 micrones⁸ de píxel con un microdensitómetro de barrido Joyce-Loebl, esta dispo-

¹http://www.diagnijmegen.nl/index.php/NW0_-_Bayesian_Decision_Support_in_Medical_Screening_%28B-SCREEN%29

²<http://marathon.csee.usf.edu/Mammography/Database.html>

³Cráneo-caudal

⁴Oblicuo mediolateral

⁵<http://www.lcc.ufu.br/amdi/>

⁶<https://www.acr.org/Practice-Management-Quality-Informatics/Registries/National-Mammography-Database>

⁷www.irma-project.org/index_en.php

⁸Unidad de longitud equivalente a una milésima parte de un milímetro

Tabla 4.1: Equivalencia de tonos por canal (profundidad)

Cantidad de bits	Tonos de canal(Por pixel)
8	256
10	1,024
12	4,095
14	16,383
16	65,532

nible al público a través del Pilot European Image Processing Archive (PEIPA) de la Universidad de Essex, cuentan con un tamaño de 1024 x 1024 px, incluyen una evaluación las marcas de "verdad" del radiólogo en las localizaciones de cualquier anomalía presente, una nomenclatura por cada imagen del tipo de microcalcificación(si se encuentra) ¹.

La selección del dataset se derivó directamente de las premisas anteriormente mencionadas, sin embargo aunque los datasets enlistados presentaban en su mayoría con características requeridas, algunos no estaban disponibles como el caso de B-SCREEN o IRMA, en el caso de NDM o AMDI se desarrollaron utilizando PostgreSQL con Image-Handling Extension de (PostgreSQL-IE), que es un sistema de gestión de bases de datos relacionales eXtension (XRDBMS), sin embargo la herramienta que ofrece la página oficial no se encuentra disponible, en el caso de DDSM la cantidad de estudios es 2620 divididos en 43 volúmenes que la convierte en una opción viable sin embargo la resolución varía entre 42, 43.5 y 50 micrones por haberse capturado en distintos escáneres sumado a que manejan valores de 12 a 16 bits de profundidad lo que provoca una diferencia significativa la cantidad total de posibles tonos (ver Tabla 4.1) y ya que se tienen mas capas de información en algunas imágenes puede provocar alguna variación en los resultados finales, en consecuencia y por descarte se eligió miniMIAS ya que están todas las imágenes digitalizadas están en el borde de 50 micrones de píxeles, cuentan con el mismo tamaño de 1024 px x 1024 px, tiene los requerimientos necesarios para el estudio y como agregado las 322 imágenes facilitarán el procesamiento del volumen completo.

¹<http://peipa.essex.ac.uk/info/mias.html>

4.2. Clasificaciones

4.2.1. Clasificación 1 - Datos en bruto

La clasificación inicial de la presente investigación se inició tomando el dataset miniMIAS justificado anteriormente por disponibilidad y su estandarización.

4.2.1.1. Preprocesamiento

Para poder clasificarlo se procedió a realizar un estudio profundo y búsqueda de patrones en las 322 imágenes contenidas en el dataset con el fin de normalizar los estudios posteriores y como resultado se arrojaron las características que a continuación se mencionan:

- Dirección de las imágenes: Se detectó que las imágenes presentan un patrón de dirección de acuerdo a la numeración original del dataset, para correcta interpretación podemos especificar que las mastografías digitales con numeración impar tenían una dirección de derecha a izquierda mientras que sus opuestos con numeración par tenían una dirección contraria.
- Ruido y registros: Del dataset completo existen elementos que contienen registros, los cuales pueden presentarse como cuadros blancos con simbología propia de una mastografía o escritura en color blanco, sumado a ello algunas cuentan con diversas líneas o polígonos en color blanco que podemos describir como ruido.
- Posición y espacio negativo: Las mastografías se encuentran a distinta distancia del borde de la imagen (sea izquierdo o derecho) asimismo esta distancia provoca un espacio negativo innecesario para el análisis de la imagen.

Homologado.

Es debido a las características y patrones detectados que se procede a realizar un homologado de todas las imágenes para contribuir a que el análisis posterior y los resultados sean lo más uniforme posible realizando los siguientes procedimientos:

- Dirección: Se realiza a través de la herramienta xview un proceso por lote de las imágenes de numeración impar reflejando la imagen en vertical consiguiendo una dirección igual a las imágenes con numeración par (véase imagen 4.1).
- Posición y espacio negativo: Una vez que las imágenes se homologadas en dirección se detectaron 5 patrones en cuanto a este rubro, las cuales clasificaremos Alfa, Beta, Gamma, Delta y XL las cuales presentan las siguientes particularidades;
 - -Alfa: Este agrupamiento cuenta con 151 imágenes las cuales presentan información a partir del pixel 188 en dirección X, con la herramienta xview

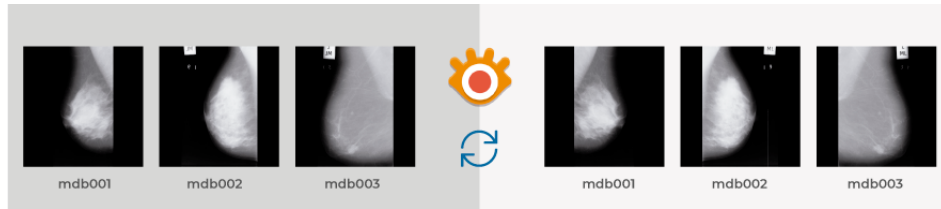


Figura 4.1: Cambio de dirección de las imágenes impares.

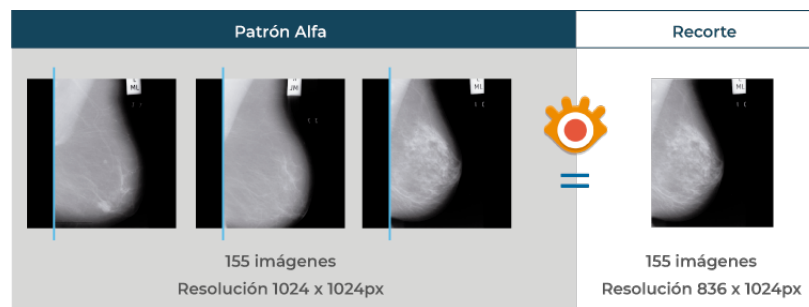


Figura 4.2: Patrón Alfa su corte se aprecia en color azul.

se realizó un corte del espacio negativo del lado izquierdo dando como resultado una imagen de 836 píxeles de ancho por 1024 de alto (véase imagen 4.2).

- -Beta: Este agrupamiento cuenta con 101 imágenes las cuales presentan información a partir del píxel 258 en dirección X, se realizó un corte en el espacio negativo del lado izquierdo dando como resultado una imagen de 766px de ancho por 1024 de alto (véase imagen 4.3).
- -Gamma: Este agrupamiento consta de 48 imágenes las cuales presentan información a partir del píxel 314 en dirección X, al igual que las anteriores se realizó un corte en el espacio negativo izquierdo resultando una imagen de 710px de ancho por 1024 de alto (véase imagen 4.4).



Figura 4.3: Patrón Beta su corte se aprecia en color amarillo.

4. METODOLOGÍA

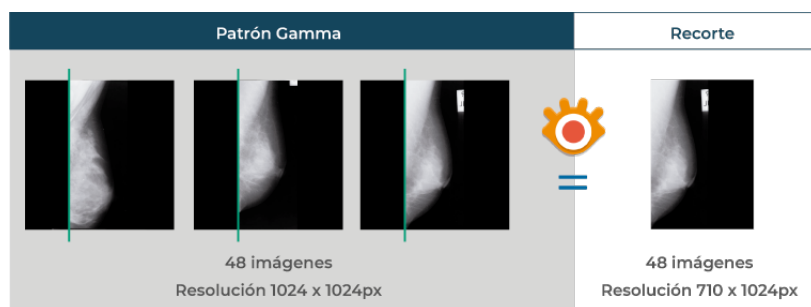


Figura 4.4: Patrón Gamma su corte se aprecia en color verde.

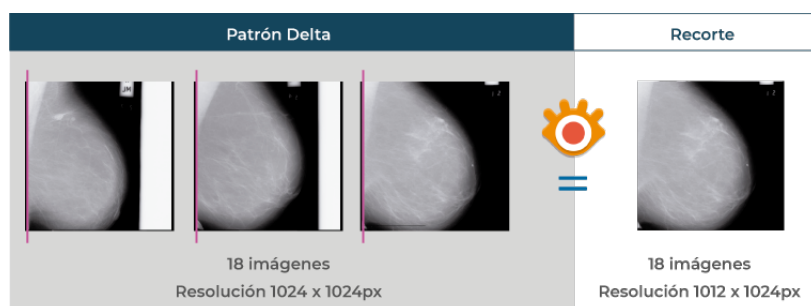


Figura 4.5: Patrón Delta su corte se aprecia en color lila.

- -Delta: Este agrupamiento consta únicamente de 18 imágenes las cuales manejan un espacio negativo pequeño pero de relevancia, presentan información a partir del pixel 12 en X, se procedió a realizar un corte en el espacio negativo izquierdo, resultando una imagen de 1012px de ancho por 1024 de alto (véase imagen 4.5).
- -XL: Costa únicamente de 4 imágenes (133,134, 151 y 152) las cuales no se apegan a ninguna de las clasificaciones anteriormente mencionadas e incluso presentan información incompleta debido a que la mama registrada en la mastografía digital supera el tamaño de la misma, asimismo para poder homologar con las anteriores clasificaciones Alfa, Beta, Gamma y Delta perderían información relevante es por ello que no se tomarán en cuenta para esta clasificación dejando una colección de 318 imágenes (véase imagen 4.6).
- Ruido y registros: Como medida para poder solventar esta problemática y retomando el trabajo anterior eliminando el espacio negativo izquierdo, se procede a eliminar el espacio negativo derecho por consecuente con ello los registros y ruido existente, debido a un nuevo análisis de las agrupaciones hechas se procede a recortar con la herramienta Xnview Alfa y Delta 36 y 212 pixeles respectivamente y verificando si no sufrieron daño de información, acto seguido se procede a anexar espacio negativo a las clasificaciones Beta y Gamma 34 y 90 pixeles

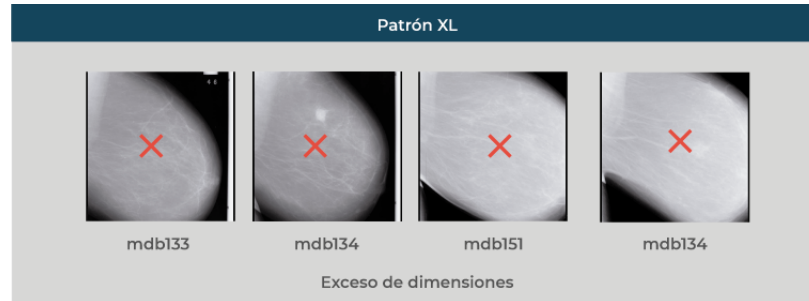


Figura 4.6: No existen cortes posibles sin alterar la estructura o homologar con sus afines.

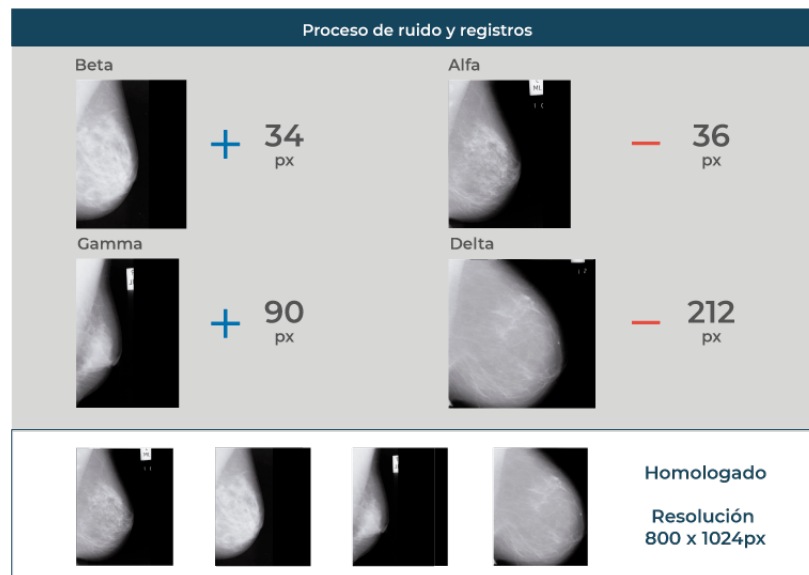


Figura 4.7: Homologado de imágenes disminuyendo ruido y registros.

respectivamente, para finalizar con un homologado (véase imagen 4.7).

Terminado este análisis, detección y tratamiento obtenemos un dataset resultante con 318 imágenes con medidas de 800 píxeles de ancho y 1024 píxeles de alto derivado de este pre-procesamiento atenderemos nuevos análisis para poder aplicar el aprendizaje profundo.

4.2.1.2. Procesamiento

Una vez obtenido teniendo una colección de mastografías digitales homologadas se procede al análisis con aprendizaje profundo, se opta por realizar 2 sub-procesos que se describen a continuación:

4. METODOLOGÍA

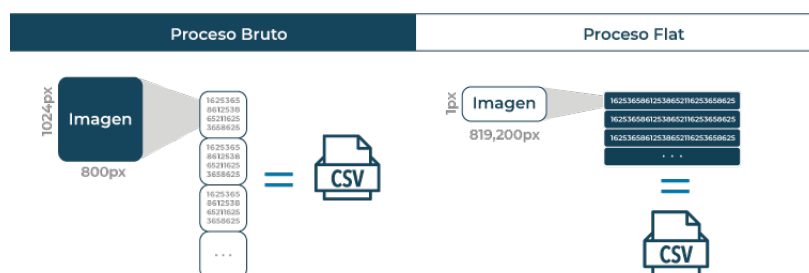


Figura 4.8: Homologado de imágenes disminuyendo ruido y registros.

- A) Proceso bruto: el cual consiste en la conversión de 318 imágenes a valores, es decir obtener la cuadrícula de la imagen de 800px por 1024px y a cada pixel asignarle un valor de 0 a 255, obteniendo 819,200 valores por cada imagen, anejando una cabecera y agregando una columna final en la cual se incluye el grado de severidad de la anomalía de acuerdo a los datos arrojados de los registros del dataset miniMIAS, para ello se utilizó un algoritmo desarrollado en Python (véase anexo 7.1) el cual nos ayudó a obtener el archivo .csv de cada imagen para posteriormente unirlos en un solo archivo que incluirá a las 318 cuadrículas de las imágenes convertidas.
- B) Proceso flat: Este se desarrolló con las mismas herramientas que el proceso anterior, pero interponiendo un método el cual convierte la cuadrícula en una sola línea, es decir aplanando la cuadrícula y con ello obteniendo un archivo más “pequeño” en cuanto a las filas que lo integran (véase imagen 4.8).

4.2.1.3. Creación de Datasets

Una vez que se finalizaron los procesos denominados bruto y flat se obtuvieron dos dataset multiclase los cuales denominaremos DT1 y DT2 respectivamente. Para ambos dataset y para una obtención de resultados más específicos se opta por dividirlo en dataset binarios los cuales tienen las clases de Normal-Benigno, Normal-Maligno y Benigno y Maligno.

4.2.2. Clasificación 2 - Datos en bruto Zoom 250px

La clasificación consiste en retomar el dataset denominado miniMIAS y realizar ajustes en su tamaño para su posterior análisis, inicialmente el portal de información del propio dataset muestra información relevante, dentro de la que podemos mencionar; el carácter de tejido de fondo, la clase de anomalía presente (Véase siguiente párrafo), severidad de la anomalía, las coordenadas del centro de la anomalía y el radio aproximado en píxeles de la circunferencia de la anomalía.

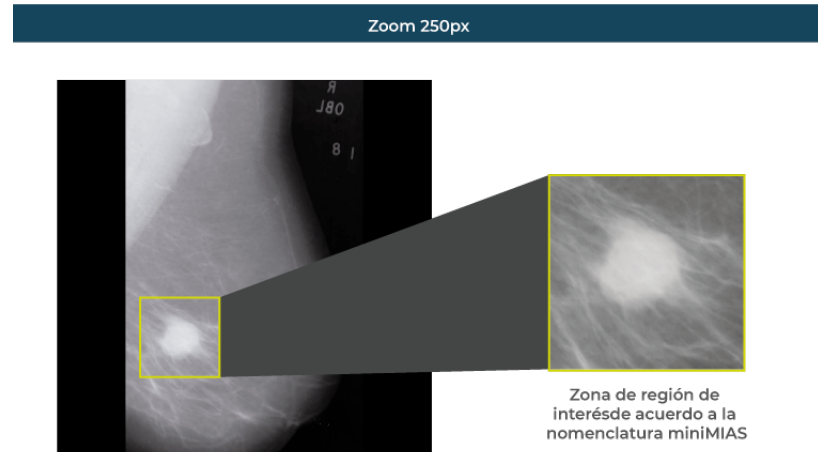


Figura 4.9: Zoom 250px aplicado a una mastografía

4.2.2.1. Preprocesamiento

Para la presente clasificación se decide basado en la metodología de (23) realizar un acercamiento a la anomalía, en lugar de analizar la mastografía digital por completo, para alcanzar el objetivo se opta por segmentar parte de ella que, conforme a los datos arrojados por la base de datos se obtiene una aproximación de recorte de 250px por 250px que permite aislar el total de las anomalías presentes en las mastografías (véase imagen 4.9).

Con ello tomando una perspectiva de la clasificación anterior solo se toma en cuenta la dirección de las imágenes únicamente ya que la eliminación de ruido y registros, así como la posición y espacio negativo no son impedimentos, para ello se procede a homologar los recortes de 250px en la dirección de izquierda a derecha con ayuda de la herramienta xview.

4.2.2.2. Procesamiento

Retomando los procesos brutos y flat se aplican a esta nueva colección de imágenes con la diferencia estas tienen 250px por 250px de medida lo que arroja una cuadrícula de 62,500 valores por imagen con un rango de 0 a 255, asimismo recordemos que se anexa una cabecera y una columna con el grado de severidad, reduciendo en gran medida el peso particular de los archivos .csv particulares y por ende el archivo .csv de que compila todas las imágenes.

4.2.2.3. Creación de Datasets

Finalizando los procesos anteriores resultan nuevos dataset multiclase que denominaremos DT3 y DT4 los cuales también se subdividen en dataset binarios los cuales tendrán las clases de Normal-Benigno, Normal-Maligno y Benigno y Maligno, para este grado de avance se tienen como resultado 4 datasets multiclase y 12 datasets binarios.

4.2.3. Clasificación 3 - Histograma de imágenes

4.2.3.1. Preprocesamiento

Para esta clasificación se hace uso de la colección de imágenes “Datos en bruto Zoom 250px” sin embargo se realizan cambios en el desarrollo para su análisis, inicialmente una vez que se tienen las 322 imágenes homologadas, se realizaron modificaciones al algoritmo realizado en Python par que realice un barrido de la imagen y obtenga su histograma.

4.2.3.2. Procesamiento

Histograma: Consiste en una gráfica donde se muestra el número de píxeles de cada nivel de gris que aparecen en la imagen, para este caso dado una imagen en niveles de grises se toma el rango de 256 tonos de gris (de 0 a 255) se obtiene el histograma de cada imagen es decir 256 barras verticales representadas en un vector.

El proceso para obtener el dataset resultante inicia con la conversión de cada imagen a su histograma con ayuda del algoritmo desarrollado y modificado en Python, insertando la numeración propia de la imagen y anexando la columna del grado de severidad, esto con las 322 imágenes teniendo como resultado un archivo .csv donde se concatenan todas las imágenes y tiene una dimensión reducida comparada con los anteriores dataset e integrando un nuevo valor de información convertida de la colección de imágenes inicial.

4.2.3.3. Creación de datasets

Una vez obtenido el dataset se integra una cabecera donde se permitirá su uso en weka, al finalizar este paso obtendremos un nuevo dataset que denominaremos DT5 el cual también dividiremos en los datasets binarios correspondientes de clases de Normal-Benigno, Normal-Maligno y Benigno y Maligno.



Figura 4.10: Proceso de homología de dirección en elementos de numeración Impar.

4.2.4. Clasificación 4 - Mejoramiento de imagen y músculo pectoral

4.2.4.1. Preprocesamiento

Filtros de control e histograma de imágenes: Esta clasificación se realizó bajo el seguimiento de la Dra. Carreras, consistió en utilizar el Dataset mini MIAS con sus valores originales, es decir las imágenes se toman con una medida de 1024 por 1024 píxeles, para luego ser procesadas a través de la herramienta Matlab para efecto de visualización del proceso en el presente manuscrito se elige la imagen de nomenclatura mdb013.pgm para aplicar las siguientes métricas.

Lectura de las imágenes: La lectura de las imágenes se realiza a través de un algoritmo modificado en este proceso es interceptado por una selección de imagen con nomenclatura par o impar (véase imagen 4.10), que recordando las clasificaciones anteriores se homologa con las imágenes hacia una dirección definida, en este caso de izquierda a derecha.

Corte Automático: Para procesar las imágenes se procede a la aplicación de una serie de filtros que ayudan a definir de mejor manera las secciones de interés que ayudan a detectar una probable anomalía dentro de una mastografía digital para ello haremos uso las sentencias que a continuación describiremos:

- Binarización: Con ayuda de la función `im2bw` perteneciente a Image Processing Toolbox en adelante (IPT) así como `graythresh` que podemos definir como umbral de imagen global utilizando el método de Otsu logramos nuestro primer cometido que se resume a la conversión de escala de grises con valores que pueden ir de 0 a 255 a valores de 1 o 0 (véase imagen 4.11). Cabe mencionar que esta técnica ayuda en gran medida a identificar los elementos que no requerimos dentro de la mastografía como los números y letras propios que pueden contener el nombre del paciente o algún identificador así como elementos que generan ruido a nuestra imagen como líneas aleatorias, rayones en la mastografía entre otros elementos que podemos visualizar en la siguiente imagen enmarcados en color amarillo.
- Selección de mayor área: Este proceso consiste en la selección de la masa mamaria

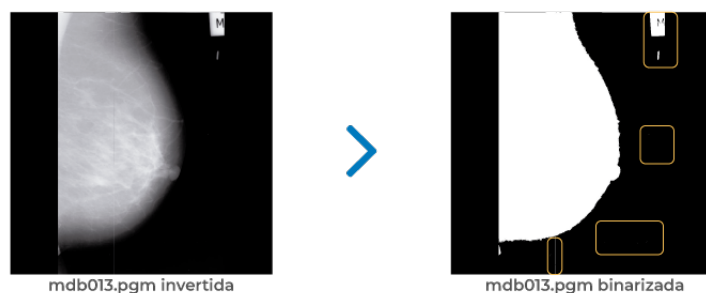


Figura 4.11: Binarización de la mastografía y detección de elementos innecesarios enmarcados.

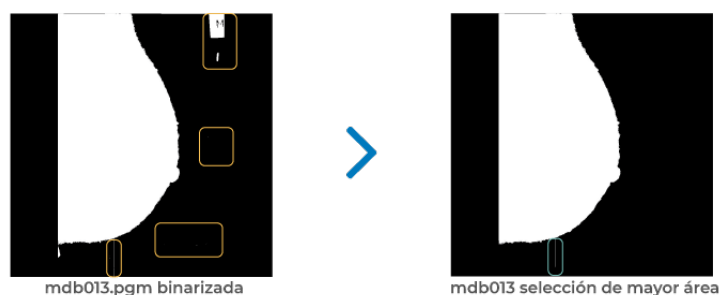


Figura 4.12: Eliminación de elementos pequeños en la mastografía.

este es guiado por la proporción, ello con el fin de discriminar el ruido presente en la mastografía u objetos pequeños que de derivados del paso anterior se visualizan a partir de la binarización de la imagen, para alcanzar este objetivo utilizamos la sentencia `bwareaopen` en paralelo con el número de píxeles que contiene el elemento a eliminar nos ofrece un resultado donde limpia la mayoría el ruido contenido en la mastografía sin embargo aunque es muy eficaz aún pueden quedar registros que se pueden eliminar más adelante como el que se enmarca en color verde (véase imagen 4.12).

- **Multiplicación:** Este proceso consiste en multiplicar la mastografía ya tratada con la original y es debido a que ambas son de una matriz cuadrada de 1024px obtenemos un producto final (véase imagen 4.13) que devolverá la mastografía original con la eliminación de ruido anteriormente hecha la cual disminuye sustancialmente los elementos blancos que pueden interferir en los análisis posteriores.
- **Equalización:** este proceso consiste en mejorar el contraste para destacar los elementos que son de posible interés para su posterior análisis, para ello se requiere de leer la imagen en escala de grises que tiene bajo contraste para mejorarlo saturando las intensidades bajas y altas, para ello retomamos la imagen hasta el



Figura 4.13: Multiplicación de matrices y resultado

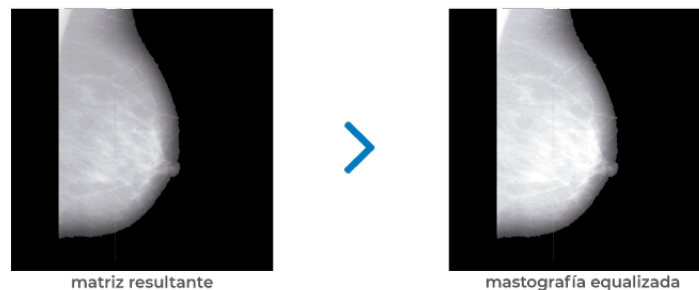


Figura 4.14: Equalización de mastografía.

punto tratada y utilizando una nueva sentencia denominada `imadjust`, asimismo este proceso da soporte a resaltar el músculo pectoral del cual hablaremos en siguientes procesos (véase imagen 4.14), cabe mencionar que este proceso es aplicable en la mayoría de las mastografías, sin embargo en masas mamarias que tienen una densidad tipo 3 o 4 se requiere de realizar análisis extras para poder dar un veredicto final.

- -Corte ancho: Este corte se realiza a través de seccionado de la imagen en dos partes iguales, es decir separar la imagen original de 1024px en dos imágenes de 512px, este tamaño se elige porque es el punto intermedio exacto donde existe información del seno ya sea a la izquierda o derecha sin importar tamaño, forma o posición, partir de este momento se analizan estas partes realizando un barrido de izquierda a derecha en la primera sección que denominaremos A y de derecha a izquierda de la sección B (véase imagen 4.15).

Asimismo una vez que son detectados los espacios negativos de los diferentes segmentos estos se eliminan gracias a que al realizar el barrido se lleva a cabo un contador de píxeles los cuales son recuperados y dan pie al espacio a eliminar, terminado este proceso los segmentos se unen en una sola imagen la cual será de ancho variable de acuerdo a cada mastografía y sus

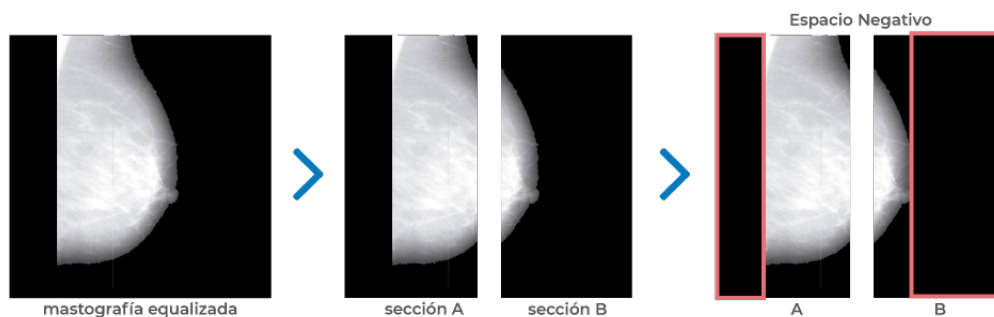


Figura 4.15: Detección del espacio negativo.

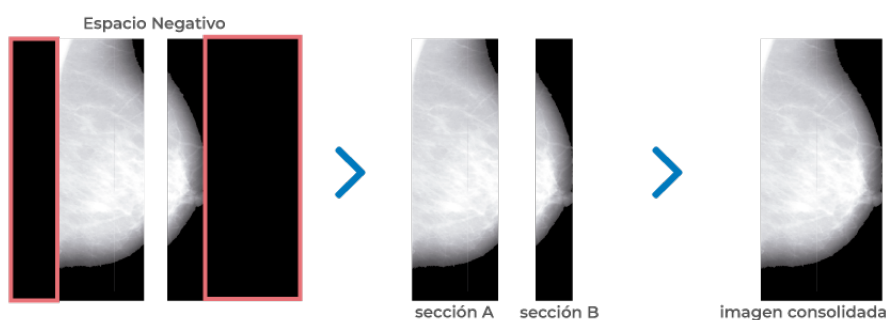


Figura 4.16: Discriminación de espacio negativo y consolidación de imagen.

características (véase imagen 4.16).

- -Corte largo: Este proceso lleva la misma secuencia que el anterior con la diferencia que la imagen consolidada se gira 90 grados antes de iniciar, nuevamente se realizan los cortes y se obtienen las secciones C y D, para continuar con un barrido para eliminar la zona obsoleta, para finalizar con un consolidado de imagen y un giro a su posición original (véase imagen 4.17).

Filtros: Una vez realizado la limpieza de ruido y los cortes necesarios para poder analizar exclusivamente la zona de interés se suma un nuevo proceso donde aplicaremos diversos filtros para mejorar aún más la calidad de la imagen brindando la posibilidad de obtener mejores resultados en procesos futuros (28).

- Filtro de mediana: Dentro de las imágenes contenidas en el dataset existen algunas las cuales tienen líneas blancas, grises o de tonos oscuros las cuales atraviesan la zona del seno y los anteriores procedimientos no pueden eliminar, por ello se ocupa un filtro de mediana que consiste en un análisis de cada píxel de la imagen y se reemplaza por la mediana de los píxeles vecinos, la base para realizar este filtro se toma de (25) que menciona que en la mayor parte del procesamiento de imágenes médicas este filtro es de gran utilidad para mejorar la calidad de la

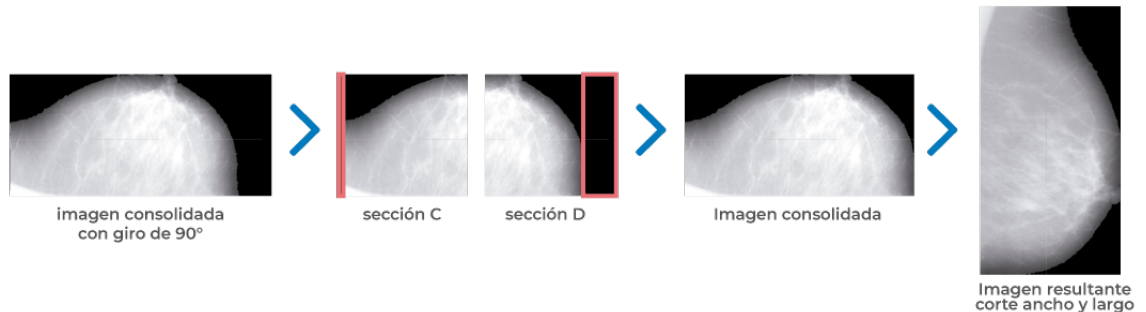


Figura 4.17: Proceso de corte largo e imagen resultante.

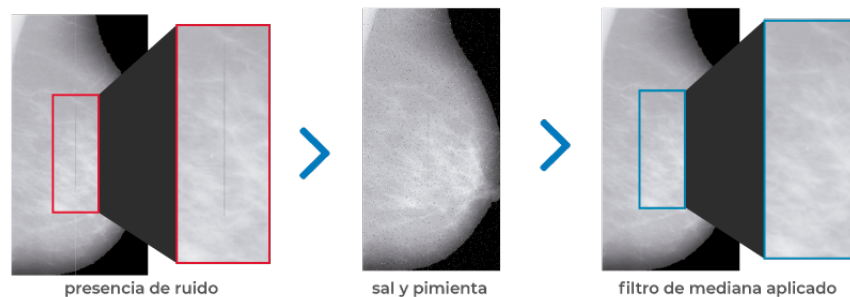


Figura 4.18: Aplicación de filtro de mediana.

imágenes (véase imagen 4.18).

Remoción del musculo pectoral: El musculo pectoral ubicado en la región anterosuperior del tórax es el causante de generar ruido blanco que dificulta detección de microcalcificaciones ya que ambos tanto la anomalía como el músculo son de un color blanco intenso el cual con seguridad genera confusión en un análisis con inteligencia artificial, por ello es de alta relevancia eliminarlo, para este proceso se realiza una serie de pasos para su eliminación.

- Mejora de contraste: Como paso 1 Se aumenta el contraste de la imagen de forma que se pueda identificar si la instrucción `imajust` genera el contraste suficiente para poder identificar el músculo pectoral enmarcado en blanco, como paso 2 como contraste no fue suficiente se vuelve a aplicar la instrucción `imajust` mejorando la imagen y obteniendo una imagen más suavizada, aunque en el paso 3 realizaremos una binarización para poder identificar con claridad los segmentos de color blanco (véase imagen (véase imagen 4.19)).
- Resta de matrices: Una vez que obtengamos la imagen ajustada de contraste esta se cruza con la imagen original para realizar una resta de segmentos y obtener una matriz donde se elimina la parte blanca sin embargo como podemos apreciar

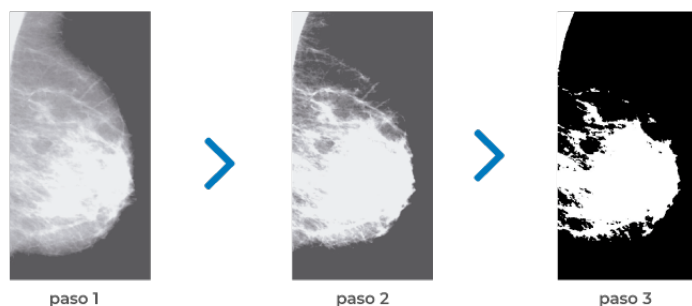


Figura 4.19: Pasos de mejora de contraste para detectar el músculo pectoral.



Figura 4.20: Resta de matrices no favorable.

en la imagen (véase imagen 4.20). Este proceso no es fructífero del todo ya que en áreas que son demasiado densas dentro del seno se puede confundir y cortar áreas y por ende perder información valiosa.

- Optimización de corte: Para solucionar el problema antes mencionado se realiza un análisis de las imágenes resultantes y se decide interceptar el proceso antes de realizar el cruce de matrices para restringir el área de corte con un polígono semitriangular que sigue la forma del músculo pectoral (véase imagen 4.21), ya que todas las imágenes se encuentran en la misma dirección y el músculo está situado en la misma zona podemos realizar este proceso el cual beneficia en gran medida ya que las áreas densas del seno no se eliminan y la pérdida de información se reduce en gran medida, sumado a que este proceso tiene un 90 % de efectividad en las imágenes finales como podemos apreciar en la imagen (véase imagen 4.22).

4.2.4.2. Procesamiento

Histograma: Para este clasificador usamos el histograma de imágenes como anteriores con la diferencia que la imagen de base se encuentra en mejores bases y mayor probabilidad de encuentra de muestra, asimismo se retomamos las 256 barras verticales



Figura 4.21: Resta de matrices favorable.

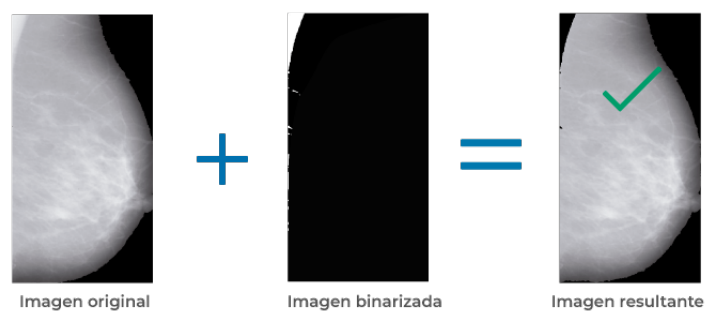


Figura 4.22: Resultado listo para el análisis final.

representadas en una matriz vector de información que aprovecharemos mas adelante.

4.2.4.3. Creación de datasets

Para esta clasificación tambien se integran las respectivas cabeceras a través de python y las respectivas nomenclaturas del estado en el cual se encuentra para poder realizar el análisis posterior y obtener resultados a travez de la utilización de algoritmo ensamblados.

4.3. Descripción de la Metodología del proceso general

Un modelo clasificador está asociado con el reconocimiento de patrones. Una breve descripción de la metodología se da a continuación(ver Figura 4.23):

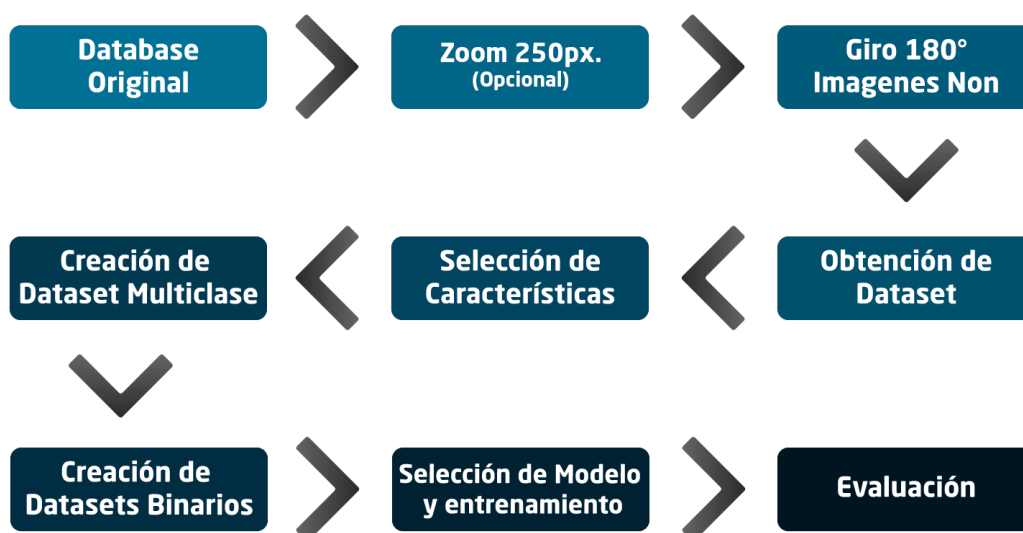


Figura 4.23: Metodología propuesta basada en (22).

- Database Original. Consiste en el uso de un database con 322 imágenes, con dimensiones de 1024x1024 pixeles, de las cuales 207 son mastografías normales, 60 tienen microcalcificaciones detectadas como benignas y 51 malignas que presentan una concentración atípica de células en el seno.
- Zoom 250 px (Opcional). Se realiza un recorte de las imágenes seleccionadas con una altura y anchura de 250 pixeles ello debido a que la imagen 226 y 239 cuentan con más de un grupo microcalcificaciones y el tamaño del recorte abarca directamente las presentes, o bien este proceso es pasado por alto cuando se trabajan

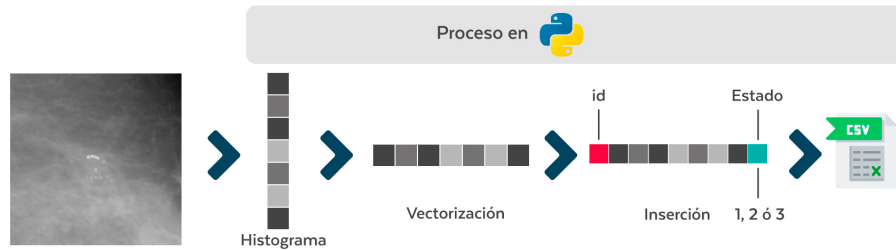


Figura 4.24: Resultado listo para el análisis final.

con imágenes en bruto o con una clasificación donde se incluyen un preprocesado de mayor peso.

- Giro de 180 grados Non. Las imágenes de numeración impar se realiza el procedimiento para que tengan la misma dirección que las imágenes pares, esto para tener una serie de imágenes con características similares, aunado a ello se descartan las imágenes 133,134, 151 y 152 por el exceso de dimensiones en la clasificación de zoom mientras que en la sotras se mantienen, ello para tener una serie de elementos con características similares, ello con el fin de analizar y comparar con otros sistemas de recorte que puedan existir en el futuro donde se maneje un pixelaje mayor.
- Obtención del dataset. A través de un algoritmo desarrollado en el lenguaje Python, las imágenes se someten a una obtención de su histograma y posterior vectorización del mismo, dicho algoritmo agrega un identificador (id) al inicio del vector recorriendo la información del histograma, asimismo al finalizar los datos obtenidos de la imagen se agrega al final del vector tres identificadores los cuales son: 1 (normal), 2 (benigno) y 3 (maligno), esto de acuerdo a los datos brindados por la página oficial del dataset MIAS. Es decir 318 archivos de extensión .csv los cuales son tratados posteriormente (ver Figura ??):
- Selección de características. En la obtención del dataset se integra un id, 250 elementos propios del histograma, más el estado Normal, Benigno o Maligno, es decir 252 características fungiendo como principal la última.
- Creación del dataset multiclase. A partir de los 318 o 322 archivos generados según la clasificación usada en la obtención del dataset estos a través de un enclaustramiento en una carpeta y utilizando funciones nativas de Python se compilan en un solo dataset, el cual contiene 318 o 322 líneas, sin embargo, para poder utilizarlo en posterior análisis se requiere insertar las cabeceras correspondientes de texto

4. METODOLOGÍA

asimismo como la conversión de los últimos valores que contiene los estados, es decir; 1 (sustituir por n), 2 (sustituir por b) y 3 (sustituir por m).

- Creación de datasets binarios. Tomando en cuenta que se tiene tres tipos de clases estos se ponen a prueba colocando versus entre ellos, para ello se divide el dataset multiclase en subdatasets con combinaciones de; Maligno-Benigno el cual contiene 111 registros de histograma vectorizadas, Normal-Benigno, el cual contiene 267 líneas y Normal-Maligno contando con 257 líneas.
- Para ello se utilizan los criterios de muestreo: 1/3-2/3, Validación cruzada con 10 iteraciones y muestra representativa. Los algoritmos mejor posicionados, es decir, los que mejores resultados generaron, fueron *LogitBoost+Dlj4Mlp*, *AttributeSelectedClassifier+Dlj4Mlp*, *FilteredClassifier+Dlj4Mlp* y *Staking+Dlj4Mlp* los cuales se describen a continuación:
 - LogitBoost (19). También conocido como regresión logística aditiva, optimiza la probabilidad directamente. Desde un punto de vista práctico, LogitBoost utiliza un esquema de regresión base. Este algoritmo puede ser visto como una optimización convexa, específicamente, dado que se busca un modelo aditivo de la forma:

$$f = \sum_t atht$$

Donde el algoritmo LogitBoost minimiza las pérdidas logísticas mediante;

$$\sum_i \log(1 + e^{-y_i f(x_i)})$$

- AttributeSelectedClassifier(30). Este algoritmo utiliza el ranking con InfoGainAttributeEval y la búsqueda de Ranker y puede eliminar atributos menos útiles. Este algoritmo debe ser usado para transformar los datos antes de pasarlos a su proceso.
- FilteredClassifier (30). Esta es una clase que ejecuta un clasificador arbitrario en datos que han pasado por un filtro. Al igual que un clasificador, la estructura del filtro se basa exclusivamente en los datos de formación y las instancias de prueba pueden ser procesadas por el filtro sin cambiar su estructura. Si existen pesos de instancia o pesos de atributo desiguales y el filtro o el clasificador no son capaces de tratarlos, las instancias y/o atributos se vuelven a muestrear con un reemplazo basado en los pesos antes de

pasarlos al filtro o al clasificador (según corresponda).

- Stacking (31). Algoritmo donde existe un conjunto de n miembros. Cada uno de estos miembros está entrenado en un conjunto dado de datos de entrenamiento. Los miembros este conjunto pueden compartir el mismo tipo de clasificador (homogéneo) o utilizar diferentes clasificadores (heterogéneos). La diversidad de datos fomenta entre los miembros para que cada miembro genere diferentes estimaciones.

- DljMlp4 (15). Es una librería de código abierto basada en Java disponible bajo la licencia Apache 2.0. Utiliza una clase de array multidimensional y proporciona un soporte de álgebra lineal y manipulación de matrices similar al proporcionado por Numpy.

El algoritmo permite configurar las siguientes capas para construir arquitecturas más sofisticadas: Capa de submuestreo, la cual subdivide grupos de unidades de la capa madre por diferentes estrategias (media, máxima, etc.); BatchNormalization, que aplica la estrategia común de normalización de lotes en las activaciones de la capa madre; OutputLayer, la cual genera salidas de clasificación/regresión, entre algunas otras para mejorar el aprendizaje.

- Evaluación. Habiendo ejecutado uno a uno los algoritmos de clasificación en conjunto con el algoritmo DLj4Mlp, se procedió a validar los mejores resultados mediante las siguiente métricas: i), matriz de confusión, ii) sensibilidad, que es la capacidad de prever los casos positivos cuando realmente son enfermos o con presencia de microcalcificaciones, es decir la capacidad para detectar enfermedad en mastografías con signos de micro-calcificaciones, iii) especificidad, la cual brinda información de casos negativos de los que son realmente sanos y la proporción de sanos correctamente identificados, es decir la capacidad de detectar la enfermedad en mastografías de características sanas.

Experimentos y resultados

A continuación se presenta el Hardware y Software utilizado para realizar el pre-procesamiento de las imágenes en cuanto a dimensiones, filtros, procesos, compactación de información y análisis con algoritmos, asimismo se hace referencia a los resultados obtenidos a través del desarrollo de la presente tesis de manera general y por cada clasificación hecha mostrando la evolución de los porcentajes obtenidos.

5.1. Requerimientos

- Los experimentos fueron realizados en una computadora con las siguientes características: Intel(R) Core i7-6500U CPU 2.50 GHz, Ram 8.00 GB, HDD Estado Sólido de 480Gb.
- Windows 10 Home Single Language. Sistema operativo de 64bits, procesador x64, anexado a ello la manipulación de imágenes fue realizado con el software XnView, los algoritmos fueron programados en Spyder(Python 3.6), Matlab versión 2018a y analizados en Weka 3.8.2.
- Xnview. Es un software gratuito que permite ver, cambiar el tamaño y editar imágenes y es compatible con más de 500 formatos diferentes incluido el formato de archivo de escala de grises .pgm utilizado en la presente investigación.
- Spyder Es un entorno de desarrollo de plataforma abierta (IDE) multiplataforma para programación científica. Spyder integra con una colección de paquetes científicos de Python, incluidos NumPy, SciPy, Matplotlib, pandas, entre otros, que son utilizados para el desarrollo de los algoritmos usados en el desarrollo de la presente tesis.

- Matlab Es un software matemático que ofrece un entorno de desarrollo integrado con un lenguaje de programación propio en el cual es posible la manipulación de imágenes y la integración de diferentes filtros aplicables a ellas.
- Weka Es una plataforma de software para el aprendizaje automático y la minería de datos escrito en Java y desarrollado en la Universidad de Waikato. Weka es software libre distribuido bajo la licencia GNU-GPL.

5.2. Resultados

Los experimentos fueron realizados en una computadora con las siguientes características: Windows 10 Home Single Language, Intel(R) Core i7-6500U CPU 2.50 GHz, Ram 8.00 GB, HDD Estado Sólido de 480Gb, Sistema operativo de 64bits, procesador x64, El tratamiento de imágenes fue realizado con el software XnView, los algoritmos fueron programados en Spyder (Python 3.6) y la clasificación en Weka 3.8.2.

Para realizar las pruebas se aplicaron algoritmos ensamblados a los datasets generados, es decir Multiclase, Binario Normal-Maligno, Binario Normal-Benigno, y Binario Benigno-Maligno, tomando los criterios de 1/3 2/3, validación cruzada de 10 iteraciones y Muestra representativa.

5.2.1. Difusión de resultados

Un segmento de la presente investigación para ser específicos hasta la clasificación numero 3 por histograma de imágenes fue mostrado en el Congreso Mexicano de Inteligencia Artificial (COMIA) en su edición 2018 (8), derivado de este movimiento se recomendaron seguir líneas de mejora del procesamiento de la imagen antes de realizar el procesado con inteligencia artificial bajo el proceso con el cual ya se contaba.

Este fue un punto clave para reforzar la estancia solicitada ante CONACYT en la Universidad Panamericana con el fin de mejorar los procesos apoyado de la Dr. Victoria Carreras cuya especialidad es el procesamiento de imágenes y amplio conocimiento del tema reforzaría los avances obtenidos hasta ese momento.

5.2.2. Análisis General

Retomando la experiencia obtenida a través de los experimentos realizados podemos concluir de manera general que:

- Entre menor cantidad de datos a analizar los algoritmos responden de manera más eficiente sin embargo no significa el hecho de discriminar información relevante.

- No es conveniente trabajar con matrices de datos con cuadradas sumadas en forma vertical ya que eleva la cantidad de requerimientos la máquina que los esté procesando.
- No es conveniente usar un rango de intensidad por cada pixel ya que eleva el peso a los archivos analizados como la disminución del rendimiento para la obtención del resultado.
- La forma más efectiva de elevar los niveles de efectividad es a través del uso del histograma de la imagen.
- Las imágenes en bruto no son las más convenientes para realizar un análisis con aprendizaje profundo con total seguridad se deben de usar un pre procesamiento de alta efectividad de detección en las imágenes.
- La hiperparametrización puede llegar a elevar el mejor resultado obtenido ya que al integrar todas las variables posibles en su uso esta mejora el rendimiento de los algoritmos en la mayoría de las veces.
- El filtro de mediana, ecualización de la imagen, ecualización del histograma, binarización, contraste de imagen, son de uso obligatorio para el mejoramiento de las imágenes médicas para su posterior procesado con Deep learning.
- Las mayores entidades de efectividad son proporcionadas cuando se compara el dataset binario Normal-Maligno.

5.2.3. Análisis específico

Clasificación 1 y 2: Los resultados no son óptimos para comparar con los trabajos similares ya que su tasa de detección y eficiencia solamente supera a Cruz y Camacho aunado a que la prueba de los dataset sometidos a los algoritmos excede el tiempo predispuesto para la prueba llegando incluso a ocupar días para ofrecer un resultado de baja gamma.

Clasificación: 3 Los resultados son competitivos aunque no son los óptimos tomando en cuenta el estado del arte, estos tienen probabilidad de mejora bajo las recomendaciones realizadas en el COMIA obteniendo un máximo de efectividad.

Clasificación 4: Los resultados están a la altura de los primeros 5 lugares de efectividad obtenidos comparado con el estado del arte, incluso en el uso de la hiperparametrización eleva este porcentaje mostrándolo competitivo ello a raíz de la estancia en la Universidad Panamericana e integrando un preprocesamiento mayor a antes de someterlo a los diferentes algoritmos los anteriores argumentos se sustentan en la siguiente tabla (ver Tabla 5.1).

A continuación se muestra de manera gráfica el incremento de los porcentajes en proporcional medida del cambio de datasets. Como se puede observar el máximo porcentaje de efectividad es de 95.25 utilizando el método de hiperparametrización y en

5. EXPERIMENTOS Y RESULTADOS

Tabla 5.1: Relación de avances de efectividad de datasets

Clasificaciones porcentuales según el dataset					
Dataset	Multiclase	Bin N-M	Bin N-B	Bin B-M	Hiperparametrización BNM
Dataset 1	59.1253	82.5756	76.3322	51.134	82.7862
Dataset 2	62.0943	82.6721	76.5145	51.383	82.8754
Dataset 3	61.1429	83.7116	77.0128	52.5192	83.9681
Dataset 4	63.0943	84.2741	77.1844	52.5455	85.1286
Dataset 5	65.1429	88.3721	81.0127	55.814	89.8522
Dataset 6	69.2166	94.187	84.7409	58.8607	95.2512

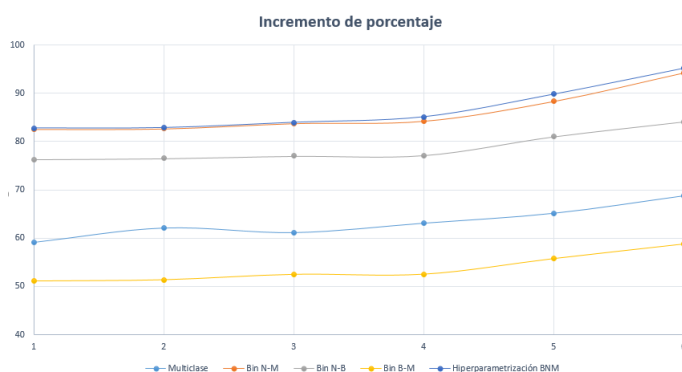


Figura 5.1: Resultados obtenidos en la presente investigación.

un método manual es obtiene un máximo de 94.28 convirtiendo la presente investigación en un puntero comparable con los mejores resultados obtenidos en investigaciones similares (véase imagen 5.1).

5.2.4. Dataset Multiclase

Para esta prueba se tomó un dataset que contiene 318 elementos y tres diferentes estados, para lo cual observamos que, de acuerdo a (ver Tabla 5.2). Los mejores algoritmos son LogitBoost + Dlj4Mlp y Staking + Dlj4Mlp en los criterios 2/3 1/3 y CV - 10 Folds con porcentajes iguales de 69.2166% y 69.1316% respectivamente, mientras que para el criterio de muestra representativa existe un algoritmo con una tasa de efectividad del 68.1852% a pesar de no ser beneficiado en los primeros dos criterios.

Tabla 5.2: Algoritmos ensamblados aplicados al dataset multiclase

Ensamblados(Meta)	DeepLearnig	2/3 – 1/3	CV-10	MR(45)
	Dlj4Mlp	50.4963	57.4146	51.8979
LogitBoost	Dlj4Mlp	68.2732	69.2166	67.5509
Staking	Dlj4Mlp	68.2882	69.1316	67.5659
FilteredClassifier	Dlj4Mlp	21.1668	54.5002	68.1852
25 algoritmos más ...	Dlj4Mlp

Tabla 5.3: Algoritmos ensamblados aplicados al dataset binario Normal-Maligno

Ensamblados(Meta)	DeepLearnig	2/3 – 1/3	CV-10	MR(39)
	Dlj4Mlp	71.1553	76.8112	64.7229
AttributeSelectedClassifier	Dlj4Mlp	94.187	86.1992	85.3572
LogitBoost	Dlj4Mlp	86.6815	85.1974	79.1039
FilteredClassifier	Dlj4Mlp	90.1699	79.1384	36.3815
25 algoritmos más ...	Dlj4Mlp

5.2.5. Dataset Binario Normal Maligno

Para esta prueba el dataset cuenta con 257 elementos, con dos estados posibles Normal y Maligno. El mejor algoritmo es AttributeSelectedClassifier + Dlj4Mlp ofreciendo un resultado del 94.187% de efectividad (ver Tabla 5.3), sin embargo existe la presencia de los algoritmos LogitBoost + Dlj4Mlp y FilteredClassifier + Dlj4Mlp que figuran también en la Tabla 1.

5.2.6. Dataset Binario Normal Benigno

En esta prueba el dataset cuenta 267 elementos, con los posibles estados de Normal y Benigno, de acuerdo a (ver Tabla 5.4), el algoritmo más competitivo es AttributeSelectedClassifier + Dlj4Mlp con un rendimiento de 84.7409%, y haciendo una retrospectiva a la Tabla 2., podemos definir que el algoritmo señalado es bueno para identificar y realizar una segmentación efectiva de una mastografía normal a una que puede presentar un grado de lesión o presencia de microcalcificaciones. Es importante señalar que los algoritmos Staking + Dlj4Mlp y FilteredClassifier + Dlj4Mlp en el criterio 2/3 1/3 muestran una efectividad competitiva con resultados optimos en comparativa con el primer algoritmo.

Tabla 5.4: Algoritmos ensamblados aplicados al dataset binario Normal-Benigno

Ensamblados(Meta)	DeepLearnig	2/3 – 1/3	CV-10	MR(41)
	Dlj4Mlp	73.4284	78.5462	66.0108
AttributeSelectedClassifier	Dlj4Mlp	83.9813	84.7409	85.9783
Staking	Dlj4Mlp	82.906	81.8494	80.2037
FilteredClassifier	Dlj4Mlp	82.3169	78.197	79.6837
25 algoritmos más ...	Dlj4Mlp

Tabla 5.5: Algoritmos ensamblados aplicados al dataset binario Normal-Benigno

Ensamblados(Meta)	DeepLearnig	2/3 – 1/3	CV-10	MR(22)
	Dlj4Mlp	44.4157	45.6934	51.5496
LogitBoost	Dlj4Mlp	55.2266	58.8607	58.5264
AttributeSelectedClassifier	Dlj4Mlp	52.8029	55.9725	59.96834
FilteredClassifier	Dlj4Mlp	55.1037	54.6614	58.4035
25 algoritmos más ...	Dlj4Mlp

5.2.7. Dataset Binario Benigno Maligno

En el análisis del dataset y contando con 110 líneas resultantes y dos estados posibles Maligno y Benigno, tenemos la (ver Tabla 5.5) en la cual se aprecia nuevamente al algoritmo LogitBoost en primer lugar compartiendo efectividad con el algoritmo FilteredClassifier en el criterio 2/3 1/3, que en retrospectiva a los análisis anteriores podemos declarar que para datos que pueden presentar confusión o multiclase es mejor aplicar el algoritmo LogitBoost en combinación con Dlj4Mlp.

Conclusiones y trabajos futuros

6.1. Conclusiones

El trabajo presentado en el presente manuscrito es conformado por una evolución de resultados conforme al tiempo de desarrollo y una disminución de tiempos a medida que se optimizan los procesos, inicialmente con resultados del 82 % en los datasets conformados con números brutos hasta llegar con un resultado del 94.187 % utilizando apendizaje profundo y 95.2512 % utilizando el método de Hiperparametrización, ello obtenido con base al aprendizaje de los anteriores clasificadores, la mejora se fue notando a medida que las imágenes se preprocesaban de manera mas efectiva: utilizando binarización, selección de mayor área, multiplicación, equalización de imagen, cortes ancho y largo, filtro de mediana y remoción del músculo pectoral, en suma con optimizaciones de cortes y el preprocesamiento de imágenes fueron incrementando la cantidad de porcentaje y su grado de efectividad. ,

En cuanto al uso de algoritmos la integración de ensamblados mejora en excelente medida el porcentaje de eficiencia a la hora de decidir el posible estado de las imágenes utilizadas, dentro de ellos destacan 3 que se conservan con los mejores resultados desde las primeras clasificaciones hasta la integración de preprocesamiento de imagen completo los cuales se mencionan a continuación:

- El algoritmo *LogitBoost+Dlj4Mlp* en general es bueno para clasificar datasets multiclase o que puedan presentar datos confusos en el análisis de nuestros datos derivados del histograma.

- El algoritmo *AttributeSelectedClassifier+Dlj4Mlp* demostró ser bueno en datasets cuyas características puedan separarse de manera sustancial, es decir, de acuerdo a los resultados obtenidos se pueda utilizar para definir si una mastografía presenta características normales o alguna microcalcificación.

- El algoritmo *FilteredClassifier+Dlj4Mlp* y *Staking+Dlj4Mlp* muestra tasas de efec-

6. CONCLUSIONES Y TRABAJOS FUTUROS

tividad buenas en algunos criterios de muestreo, los cuales se pueden utilizar como refuerzos para definir el diagnostico final.

Cabe destacar que a medida que las matrices de datos perdían peso estos algoritmos actuaban con mayor eficiencia, en suma con la disminución de tiempo de procesado con resultados proporcionalmente inversos al porcentaje de efectividad.

Para fines del presente trabajo es imprescindible que bajo los clasificadores y las metodologías seguidas los algoritmos desempeñan un trabajo de manera estable y mejor aún obteniendo resultados competitivos con investigaciones conforme a la limpieza de las imágenes trabajadas del dataset original.

La presente investigación tiene una repercusión positiva ya que bajo las métricas de las investigaciones anteriores las cuales se tomaron referencia en el presente trabajo son altamente efectivas y se colocan dentro del ranking en las primeras 3 posiciones cumpliendo así los objetivos planteados, corroborando la hipótesis y creando una nueva línea para el análisis de mastografías digitales a través de inteligencia artificial.

6.2. Trabajos Futuros

Tomando con referencia la clasificación número 4 es decir el mejoramiento de imagen y músculo pectoral, se pueden aplicar una mejoras al recorte de la imagen antes de su etapa final ya que se realiza a través de un polígono fijo, que, aunque es de gran ayuda para poder disminuir la cantidad de pérdida de información disponible en las mastografías puede provocar algunos micro-cortes que son innecesarios y en algunos casos inexactos tomando en cuenta un segmento muy bajo afectado, esto se podría solventar a través de una construcción de un polígono dinámico y que siga una tendencia de líneas curva tomando en cuenta los musculos pectorales que existen; con ello dar mayor exactitud a la hora de contrastarlo con otras matrices de imágenes y con ello disminuyendo la cantidad de error.

Como otro punto se puede mencionar la integración en de los diferentes procesos que se forman, ya que se encuentran en diferentes plataformas Python, Matlab, Excel entre otros donde se trasladan los datos de forma manual provocando cuellos de botella, limitando los tiempo y posible pérdida de información, por lo que convendría realizarlo en un solo proceso hasta llegar al final del proceso, sin embargo no se descarta la posibilidad de dar seguimiento futuro al presente proyecto en un proyecto de grado Doctoral.

Derivado de lo anterior, se propone con estas modificaciones a los procesos se alcance un porcentaje mayor al obtenido de 95.2512 % de tasa precisión para poder establecer un vínculo más competitivo con los trabajos relacionados en esta área.

Bibliografía

- [1] Abugattas, J., Manrique, J., and Vidaurre, T. (2015). Mamografía como instrumento de tamizaje en cáncer de mama. *Revista Peruana de Ginecología y Obstetricia*, 07(1):311–319. [9](#)
- [2] Al-masni, M. A., Al-antari, M. A., Park, J. M., Gi, G., Kim, T. Y., Rivera, P., Valarezo, E., Han, S. M., and Kim, T. S. (2017). Detection and classification of the breast abnormalities in digital mammograms via regional convolutional neural network. *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1230–1233. [18](#)
- [3] Aldaco, F., Pérez, . P., and Cervantes, L. (2018). Optimización de los filtros de mediana-gaussiano para una mejor convergencia del snake en la segmentación de imágenes médicas mortalidad por cáncer en México: actualización 2015. *Gaceta Mexicana de Oncología.*, 17(1). [2](#)
- [4] Arafí, A., Fajr, R., and Bouroumi, A. (2014). Breast cancer data analysis using support vector machines and particle swarm optimization. *Complex systems (wccs)*, Second world conference:1–6. [15](#), [18](#)
- [5] Arel, I., Rose, D. C., and Karnowski, T. P. (2010). Deep machine learning a new frontier in artificial intelligence research. *IEEE computational intelligence magazine*, 5(4):13–18. [10](#)
- [6] Arevalo, J., González, F., Pollán, R. R., Oliviera, J., and López, M. (2016). Representation learning for mammography mass lesion classification with convolutional neural networks. *Computer methods and programs in biomedicine*, 127(1):248–257. [14](#), [18](#)
- [7] Camacho, C. and Espíndola, V. (2018). Actualización de la nomenclatura bi-rads por mastografía y ultrasonido. *Anales de Radiología México*, 17(7). [10](#)
- [8] Carrera, J. A., Sánchez, E., Biswal, R. R., and Carreras, M. V. (2018). Categorización de anomalías cancerígenas en mastografías digitales aplicando aprendizaje profundo. *Research in Computing Science*, 147(7):1–12. [42](#)

BIBLIOGRAFÍA

- [9] Cataño, L. A. O. (2016). La célula como unidad de vida: Concepto clave para la educación y el aprendizaje en escuela nueva. Master's thesis, Universidad Nacional de Colombia. 1
- [10] Centellas, S. S. C. (2014). Método heurístico para el diagnóstico de cáncer de mama basado en minería de datos. *Revista PGI - Investigación, Científica y Tecnología*, 1(1):97–101. 18
- [11] CONACYT (2017). Desarrollan algoritmo para la detección precoz de cáncer de mama. newsnet.conacytprensa.mx/index.php/documentos/36532-desarrollan-algoritmo-para-la-deteccion-precoz-de-cancer-de-mama. 3
- [12] Cruz, M. C., Villaseñor, M. M., and Pérez, K. R. (2015). Mass segmentation in digital mammograms. *Ambient intelligence for health*, 9456(1):110–115. 2, 15, 17
- [13] Cunha, P., Franco, M. N., and Patrocínio, A. (2017). Breast density pattern characterization by histogram features and texture descriptors. *Research on Biomedical Engineering*, 33(1):69–77. 3
- [14] Dalmi, M., Mérida, A. G., Vreemann, S., and Karssemeijer, N. (2016). A computer-aided diagnosis system for breast dce-mri at high spatiotemporal resolution. *Medical physics*, 43(1):84–94. 14, 18
- [15] DL4J (2018). Deep learning for java. <https://deeplearning4j.org/>. 39
- [16] Gerazov, B. and Conceicao, R. C. (2017). Deep learning for tumour classification in homogeneous breast tissue in medical microwave imaging. *IEEE EUROCON 2017 -17th International Conference on Smart Technologies*, page 564–569. 18
- [17] INEGI (2018). Estadísticas a propósito del día mundial del cancer. http://www.beta.inegi.org.mx/contenidos/saladeprensa/aproposito/2018/cancer2018_Nal.pdf. 1, 3
- [18] Kottho, L., Thornton, C., Hoos, H. H., and Hutter, F. (2017). Auto-weka 2.0: Automatic model selection and hyperparameter optimization in weka. *Journal of Machine Learning Research*, 1(18):1–5. 11
- [19] Ling, P. (2016). A computer-aided diagnosis system for breast dce-mri at high spatiotemporal resolution. *Medical physics*, 43(1):84–94. 38
- [20] Lévy, D. and González, F. (2016). Breast mass classification from mammograms using deep convolutional neural networks. *CoRR*, abs/1612.00542(1). 18
- [21] Martínez, C., Herrera, M. A., Hall, M., and Pal, C. (2011). *Anatomía Patológica. Temas de Enfermería*. <https://clea.edu.mx/biblioteca/CirionLaHabana>, Cuba. 9
- [22] Moradkhania, F. and Bighamb, B. S. (2017). A new image mining approach for detecting micro-calcification in digital mammograms. *Applied Artificial Intelligence*, 31(5):411–424. XIII, 15, 17, 36

- [23] Neto, O. P. S., Carvalho, O., Sampaio, W., Correa, A., and Paiva, A. (2015). Automatic segmentation of masses in digital mammograms using particle swarm optimization and graph clustering. *International Conference on Systems, Signals and Image Processing (IWSSIP)*, Second world conference(1):109–112. [15](#), [17](#), [27](#)
- [24] Pedraza, A., Serrano, I., Fernández, M., Rojo, M. G., and González, L. (2016). Diagnóstico automático del her2 con deep learning. *Google Scholar*. [15](#), [17](#)
- [25] Quintana, L. and Sánchez, H. (2013). Optimización de los filtros de mediana-gaussiano para una mejor convergencia del snake en la segmentación de imágenes médicas. *Congreso Integracionista de las Ciencias y las Tecnologías Informáticas Santiago de Cuba*, 1(1). [32](#)
- [26] Raudales, I. (2014). Imágenes diagnósticas: Conceptos y generalidades. *Revista de la Facultad de Ciencias Médicas*, 1(1). [9](#)
- [27] Roa, A. C., Gilmore, H., Basavanhally, A., and Feldman, M. (2017). Accurate and reproducible invasive breast cancer detection in whole-slide images: A deep learning approach for quantifying tumor extent. *Scientific Reports*, pages 97–101. [18](#)
- [28] Rojas, A. and Cano, A. (2011). Trabajando con imágenes digitales en clase de matemáticas. *La Gaceta de la Real Sociedad Matemática Española*, 2(1):317–336. [32](#)
- [29] Seeböck, P. (2015). Deep learning in medical image analysis. Master’s thesis, Faculty of Informatics at the Vienna University of Technology. [2](#)
- [30] Witten, I., Frank, E., Hall, M., and Pal, C. (2017). *DATA MINING: Practical Machine Learning Tools and Techniques*. Springer-Verlag, Cambridge, United States. [38](#)
- [31] Wolpert, D. (1992). Stacked generalization. *Neural Networks*, 5(1):241–259. [39](#)
- [32] World Health Organization (2018). Who position paper on mammography screening. <http://www.who.int/cancer/publications>. [1](#)

Anexos

7.1. Código Python

7.1.1. Generación de CSV con ID por fila

Código de generación de CSV particulares por imagen que luego se comprimen en un solo archivo CSV para ser procesadas por la clasificación 1 y 2.

```
import os
from PIL import Image
from scipy import misc
import numpy as np
import csv

root = os.getcwd()
carpeta = "\miniMIAS"

ruta = root + carpeta
lista_imagenes = []
for (dirpath, dirnames, filenames) in os.walk(ruta):
    lista_imagenes.extend(filenames)
    break
print(lista_imagenes)

os.chdir(ruta)
matrix= []
matrixMax=[]

import string, random
lista = []
```

```
N=100
i=0
while i<1024:
    valor = ''.join(random.choices(string.ascii_uppercase + string.digits, k=3))
    if valor not in lista:
        lista.append(valor)
        i=i+1

lista.insert(0,"ID")

for i in range(len(lista_imagenes)):

    imagen = Image.open(lista_imagenes[i])
    matrix = misc.imread(lista_imagenes[i])

    ancho, alto =np.shape(matrix)
    ids=np.full(alto, i)
    print("imagen_:", i)
    matrix = np.vstack((ids, matrix))

    if i == 0:
        matrixMax = np.r_[lista]
        matrixMax = np.r_[matrix]
        print("Dimensiones", matrixMax.shape)
        print("Alto:_", alto)
        print("Ancho:_", ancho)
        b=np.ravel(matrixMax)
        print(b)
    else:
        matrixMax = np.r_[matrixMax, matrix]
        print("Dimensiones", matrixMax.shape)
        print("Alto:_", alto)
        print("Ancho:_", ancho)
        b=np.ravel(matrixMax)
        print(b)

print("Dimensiones", matrixMax.shape)
os.chdir(root)
np.savetxt("DatasetX.csv", b.reshape(1, b.shape[0]), fmt='%d', delimiter=',')
```


7.1.2. Generación de Histograma

Código de generación de CSV particulares que contiene únicamente el histograma de cada imagen por archivo que después se comprimen en un solo archivo csv que se emplean en las clasificaciones 3 y 4.

```
import cv2
import numpy as np
from matplotlib import pyplot as plt

a=0
img = cv2.imread('mdb001.pgm', cv2.IMREAD_GRAYSCALE)
cv2.imshow('mdb001.pgm', img)

hist = cv2.calcHist([img], [0], None, [256], [0, 256])
plt.plot(hist, color='red')

plt.xlabel('intensidad de iluminacion')
plt.ylabel('cantidad de pixeles')
plt.show()

r = hist.astype(numpy.int64)

aux=0

b=np.ravel(r)
print(b)

np.savetxt("mdb001.csv",b.reshape(1, b.shape[0]), fmt='%d', delimiter=',')

cv2.destroyAllWindows()
```

7.2. Código Matlab

7.2.1. Filtros de imagen

Este código se emplea en el mejoramiento general de la imagen los cuales son de carácter libre dentro de los cuales podemos mencionar la binarización, obtención de mayor área, ecualización de la imagen, filtro de mediana, conversión a escala de grises.

```
bw2 = im2bw(imag, (graythresh(imag))/3));
```

```
imwrite(bw2, 'mdb013_2binarizacion.jpg');

bwAreaOpenBW = bwareaopen(bw2,150000);
imwrite(bwAreaOpenBW, 'mdb013_3mayorarea.jpg');

imag2 = imadjust(imag3, stretchlim(imag), []);

J = imnoise(imc, 'salt & pepper', 0.02);
imwrite(J, 'mdb013_10salypimienta.jpg')
K = medfilt2(J);

ori_img = K;
imwrite(K, 'mdb013_11filtromediana.jpg')

if size(ori_img, 3) == 3 % imagen RGB
    ori_img = rgb2gray(ori_img);
    end
    ori_img1 = medfilt2(ori_img);
    ori_img1 = adapthisteq(ori_img1, 'ClipLimit', 0.01, 'Distribution', 'uniform');
    imagen1 = im2bw(ori_img1, 0.08);
    CC1 = bwconncomp(imagen1, 4);
```

7.2.2. Eliminación de objetos pequeños y ruido

Líneas de código que se encargan de realizar una limpieza de los elementos que no tienen relevancia dentro de la mastografía como manchones, letras y ruido iniciando con un cálculo de área de los componentes detectados.

```
S = regionprops(CC1, 'Area');

fig_areas = [S.Area];
[max_area, idx] = max(fig_areas);
extract_fig = false(size(imagen1));
extract_fig(CC1.PixelIdxList{idx}) = true;
f1 = extract_fig;
```

7.2.3. Recorte en región seleccionada

Código sobre el cual se selecciona una región de interés general en todas las imágenes y se discrimina el músculo pectoral.

```
I = proc_img1;
```

```
imwrite(blackMaskedImage1, 'mdb013_17Filtro.jpg');  
x=[0 400 300 220 100 20 0];  
y = [0 0 100 200 400 500 600];  
BW = roipoly(I,x,y);  
imwrite(BW, 'mdb013_16Filtro.jpg');  
ROI = I;  
ROI(BW == 0) = 0;  
imwrite(BW, 'mdb013_15Filtro.jpg');
```

7.3. Tablas de Resultados

En esta sección se anexan las tablas completas de todas las pruebas realizadas a cada uno de los archivos csv y que se generaron a partir de las diferentes clasificaciones hechas.

Tabla 7.1: Dataset 1-Multiclase

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(45)
	Dlj4Mlp	35.7698	48.8881	43.7714
AdaBoostM1	Dlj4Mlp	13.2245	38.2094	14.2286
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	42.1698	58.8931	54.2571
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	13.2245	38.2094	58.2286
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	54.1509	61.0943	58.8286
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	13.2245	33.178	58.2286
FilteredClassifier	Dlj4Mlp	13.2245	46.5579	60.4429
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	11.5245	36.3094	12.3429
LogitBost	Dlj4Mlp	57.4509	59.1253	59.0286
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	41.2264	54.805	47.9714
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	59.4509	43.2409	11.9429
Multisearch	Dlj4Mlp	12.3245	30.878	57.8286
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	42.6566	49.2025	52.9143
RandomCommitte	Dlj4Mlp	13.2245	52.3604	58.2286
RandomizableFilteredClassifier	Dlj4Mlp	47.1868		
RandomSubSpace	Dlj4Mlp	51.6038	58.5786	50.8286
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	56.4509	61.7943	58.2286
ThresholdSelector	Dlj4Mlp			
Vote	Dlj4Mlp	12.9245	32.478	58.8286
WeightedInstancesHandlerWrapper	Dlj4Mlp	11.5245	31.278	58.6286

Tabla 7.2: Dataset 1-Binario: Normal-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR (39.5)
	Dlj4Mlp	59.8791	66.3953	55.2
AdaBoostM1	Dlj4Mlp	12.7419	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	82.5756	77.1339	75.32
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	12.7419	71.0191	17.3806
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	78.5581	74.9447	72.5194
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	12.7419	52.7311	72.2194
FilteredClassifier	Dlj4Mlp	81.0465	70.7864	27.5581
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	11.0419	57.4459	17.7806
LogitBost	Dlj4Mlp	77.8581	75.0447	72.6794
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	12.4419	52.0311	72.8194
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	12.7419	58.5677	17.3806
Multisearch	Dlj4Mlp	11.8419	50.4311	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	59.8791	66.3953	55.2
RandomCommitte	Dlj4Mlp	77.8581	64.7934	72.2194
RandomizableFilteredClassifier	Dlj4Mlp	76.6953	71.0191	65.7677
RandomSubSpace	Dlj4Mlp	77.7209	76.1556	70.2387
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	77.7581	76.7447	70.4194
ThresholdSelector	Dlj4Mlp	77.8581	46.5054	71.9694
Vote	Dlj4Mlp	12.4419	51.6311	72.8194
WeightedInstancesHandlerWrapper	Dlj4Mlp	11.0419	50.8311	72.4194

Tabla 7.3: Dataset 1-Binario: Normal-Benigno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(41)
	Dlj4Mlp	63.2629	70.0809	57.8582
AdaBoostM1	Dlj4Mlp	71.9517	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	72.8517	75.9753	76.3322
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	72.2517	52.5052	70.7494
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	74.6517	71.9281	71.0494
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	73.9517	52.5052	70.7494
FilteredClassifier	Dlj4Mlp	73.6517	69.8318	71.2494
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	72.2517	28.5079	71.1494
LogitBost	Dlj4Mlp	16.6483	16.9719	19.3106
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	73.6517	51.8052	71.3494
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	73.9517	46.8873	70.7494
Multisearch	Dlj4Mlp	73.0517	50.2052	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	63.2629	70.0809	57.8582
RandomCommitte	Dlj4Mlp	73.9517	52.5052	70.7494
RandomizableFilteredClassifier	Dlj4Mlp	71.7045	65.2393	61.2557
RandomSubSpace	Dlj4Mlp	69.2809	73.9026	74.5139
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	73.8517	73.7281	68.9494
ThresholdSelector	Dlj4Mlp	16.6483	41.2693	18.6006
Vote	Dlj4Mlp	73.6517	51.4052	71.3494
WeightedInstancesHandlerWrapper	Dlj4Mlp	72.2517	50.6052	70.9494

Tabla 7.4: Dataset 1-Binario: Benigno-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(22)
	Dlj4Mlp	34.1405	36.6182	42.8744
AdaBoostM1	Dlj4Mlp	46.6514	48.6182	49.8512
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	42.8486	48.0182	51.134
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	44.9514	47.6091	49.4512
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	47.3514	48.9455	49.7512
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	46.6514	47.6091	49.4512
FilteredClassifier	Dlj4Mlp	46.3514	46.2091	49.4512
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	44.9514	45.7091	49.8512
LogitBost	Dlj4Mlp	46.6514	49.0455	50.9112
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	46.3514	46.9091	49.0512
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	46.6514	46.7	49.4512
Multisearch	Dlj4Mlp	45.7514	45.3091	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	34.1405	36.6182	42.8744
RandomCommitte	Dlj4Mlp	46.6514	38.5182	49.4512
RandomizableFilteredClassifier	Dlj4Mlp	38.5432	45.7909	42.4744
RandomSubSpace	Dlj4Mlp	42.6486	43.272	43.0744
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	46.5514	43.4727	47.6512
ThresholdSelector	Dlj4Mlp	46.6514	44.8818	39.8988
Vote	Dlj4Mlp	46.3514	46.5091	50.0512
WeightedInstancesHandlerWrapper	Dlj4Mlp	44.9514	45.7091	49.6512

Tabla 7.5: Dataset 2-Multiclase

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(45)
	Dlj4Mlp	40.5698	48.8881	43.9714
AdaBoostM1	Dlj4Mlp	13.6245	38.0094	14.4286
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	41.7698	58.7931	54.3571
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	13.7245	38.2094	58.4286
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	59.9509	60.8943	59.1286
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	13.1245	33.378	58.6286
FilteredClassifier	Dlj4Mlp	13.1245	46.6579	60.4429
GridSearch	Dlj4Mlp	48.4698	x	59.8571
IterativeClassifierOptimizer	Dlj4Mlp	11.5245	36.0094	12.6429
LogitBost	Dlj4Mlp	59.6509	61.5943	59.3286
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	41.0264	54.605	48.3714
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	59.2509	43.0409	12.1429
Multisearch	Dlj4Mlp	13.0245	30.878	58.1286
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	42.8566	49.3025	53.3143
RandomCommitte	Dlj4Mlp	13.2245	52.5604	57.7286
RandomizableFilteredClassifier	Dlj4Mlp	47.1868	52.8302	51.4286
RandomSubSpace	Dlj4Mlp	51.3038	58.8786	51.4286
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	59.7509	62.0943	58.3286
ThresholdSelector	Dlj4Mlp	x	x	x
Vote	Dlj4Mlp	13.1245	32.778	58.7286
WeightedInstancesHandlerWrapper	Dlj4Mlp	11.4245	31.478	58.6286

Tabla 7.6: Dataset 2-Binario: Normal-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR (39.5)
	Dlj4Mlp	59.9791	66.5953	55.4
AdaBoostM1	Dlj4Mlp	12.9419	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	82.6721	77.3339	75.6
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	12.9419	71.3191	17.7806
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	78.7581	75.3447	73.6194
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	12.9419	52.9311	72.4194
FilteredClassifier	Dlj4Mlp	78.0465	70.7864	27.9581
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	11.2419	57.5459	18.1806
LogitBost	Dlj4Mlp	77.9581	75.3447	73.4194
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	12.8419	51.7311	73.2194
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	12.7419	58.6677	17.1806
Multisearch	Dlj4Mlp	12.1419	50.4311	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	60.0791	66.6953	55.31
RandomCommitte	Dlj4Mlp	77.8581	64.9934	72.1771
RandomizableFilteredClassifier	Dlj4Mlp	76.9953	71.0191	65.9677
RandomSubSpace	Dlj4Mlp	77.0209	76.3556	70.2387
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	78.0581	77.0447	70.6194
ThresholdSelector	Dlj4Mlp	77.9581	46.7054	71.7494
Vote	Dlj4Mlp	12.6419	51.6311	73.1394
WeightedInstancesHandlerWrapper	Dlj4Mlp	11.2419	50.9311	72.3194

Tabla 7.7: Dataset 2-Binario: Normal-Benigno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(41)
	Dlj4Mlp	63.4629	70.3083	58.1582
AdaBoostM1	Dlj4Mlp	72.2517	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	72.9517	76.4893	76.5145
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	72.5517	52.8052	70.9784
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	74.7517	72.3411	71.3733
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	74.1817	52.8302	71.0574
FilteredClassifier	Dlj4Mlp	73.7917	70.3648	71.3213
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	72.5517	28.7079	71.1494
LogitBost	Dlj4Mlp	16.8483	16.9719	19.5987
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	73.6647	52.1342	71.5522
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	74.2415	47.0493	71.0324
Multisearch	Dlj4Mlp	73.1627	50.5392	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	63.6629	70.2809	58.2954
RandomCommitte	Dlj4Mlp	72.6797	52.6299	70.2064
RandomizableFilteredClassifier	Dlj4Mlp	72.0045	65.4053	61.4806
RandomSubSpace	Dlj4Mlp	69.3089	74.0506	74.6527
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	74.2267	74.1561	69.0784
ThresholdSelector	Dlj4Mlp	16.4203	41.4141	18.4086
Vote	Dlj4Mlp	73.8907	51.4632	71.1841
WeightedInstancesHandlerWrapper	Dlj4Mlp	72.4657	50.6712	70.9974

Tabla 7.8: Dataset 2-Binario: Benigno-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(22)
	Dlj4Mlp	34.3065	36.8459	42.9121
AdaBoostM1	Dlj4Mlp	46.9984	48.5319	49.9772
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	42.9933	48.174	51.383
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	45.1174	47.7339	49.6747
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	47.3642	49.0035	49.797
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	46.8984	47.6091	49.4512
FilteredClassifier	Dlj4Mlp	46.3514	46.3671	49.486
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	44.9874	45.9091	50.0862
LogitBost	Dlj4Mlp	47.3514	49.3001	51.3803
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	46.3514	46.9091	49.3759
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	46.8088	47.1766	49.7889
Multisearch	Dlj4Mlp	45.8968	45.7857	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	34.5743	37.1401	43.5217
RandomCommitte	Dlj4Mlp	46.975	38.6936	49.6512
RandomizableFilteredClassifier	Dlj4Mlp	38.5432	45.7909	42.4744
RandomSubSpace	Dlj4Mlp	42.7863	43.1486	43.3324
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	46.4853	43.9107	47.7859
ThresholdSelector	Dlj4Mlp	46.8091	44.3995	40.1736
Vote	Dlj4Mlp	46.3762	46.7438	50.1651
WeightedInstancesHandlerWrapper	Dlj4Mlp	45.2142	45.9091	49.8992

Tabla 7.9: Dataset 3-Multiclase

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(45)
	Dlj4Mlp	41.3698	49.4881	44.3714
AdaBoostM1	Dlj4Mlp	14.3245	38.8094	14.7766
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	42.8698	58.7931	54.8571
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	13.7245	38.8094	58.9044
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	60.4509	61.3943	60.1834
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	13.8245	33.778	58.6286
FilteredClassifier	Dlj4Mlp	14.0245	47.1579	61.1429
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	12.2245	37.0094	13.0429
LogitBost	Dlj4Mlp	60.5509	62.6943	59.7786
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	41.6264	54.805	49.0062
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	58.6509	43.9409	12.4906
Multisearch	Dlj4Mlp	13.0245	30.878	58.4286
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	44.0566	49.7025	54.2518
RandomCommitte	Dlj4Mlp	13.2245	52.8604	58.1536
RandomizableFilteredClassifier	Dlj4Mlp	48.3868	52.8302	51.4286
RandomSubSpace	Dlj4Mlp	52.4038	59.5786	51.4286
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	59.8509	62.4493	59.1973
ThresholdSelector	Dlj4Mlp	x	x	x
Vote	Dlj4Mlp	14.3245	32.978	58.9286
WeightedInstancesHandlerWrapper	Dlj4Mlp	12.5245	31.858	58.9286

Tabla 7.10: Dataset 3-Binario: Normal-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR (39.5)
	Dlj4Mlp	60.9791	67.5953	56.1236
AdaBoostM1	Dlj4Mlp	13.7419	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	83.7716	78.4539	76.1337
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	12.6419	71.3191	18.608
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	79.6581	76.0217	74.4194
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	13.7419	53.6824	72.6674
FilteredClassifier	Dlj4Mlp	79.0465	71.8134	28.2581
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	12.0419	57.5459	18.5806
LogitBost	Dlj4Mlp	78.8581	76.0107	73.793
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	13.6419	52.6961	72.5194
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	13.8419	59.3335	17.6076
Multisearch	Dlj4Mlp	12.1419	51.2771	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	60.6791	67.7305	56
RandomCommitte	Dlj4Mlp	77.8581	66.0934	72.6854
RandomizableFilteredClassifier	Dlj4Mlp	76.9953	71.4191	66.8335
RandomSubSpace	Dlj4Mlp	78.8209	76.7832	71.0844
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	79.2181	78.1205	71.4194
ThresholdSelector	Dlj4Mlp	79.0581	47.3834	72.4194
Vote	Dlj4Mlp	13.4656	52.4989	72.7474
WeightedInstancesHandlerWrapper	Dlj4Mlp	12.7589	51.2671	72.7474

Tabla 7.11: Dataset 3-Binario: Normal-Benigno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(41)
	Dlj4Mlp	64.5289	71.1574	58.4928
AdaBoostM1	Dlj4Mlp	72.7894	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	73.9317	76.913	77.0128
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	72.5517	53.0532	71.7154
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	75.7517	72.8411	71.674
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	73.6797	53.4632	71.6784
FilteredClassifier	Dlj4Mlp	73.7917	70.5318	71.5251
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	12.8997	28.9456	71.1981
LogitBost	Dlj4Mlp	16.8483	17.4719	19.8886
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	73.4997	52.8312	71.9494
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	74.6517	47.1873	71.9494
Multisearch	Dlj4Mlp	73.7805	50.9332	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	63.9499	70.4599	59.1352
RandomCommitte	Dlj4Mlp	73.0007	52.8052	71.3064
RandomizableFilteredClassifier	Dlj4Mlp	72.0045	65.5663	62.4806
RandomSubSpace	Dlj4Mlp	69.5189	74.1506	75.0139
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	74.6517	74.1561	72.0874
ThresholdSelector	Dlj4Mlp	16.6063	41.5693	18.6306
Vote	Dlj4Mlp	74.6517	52.4632	71.6041
WeightedInstancesHandlerWrapper	Dlj4Mlp	72.8955	50.6712	71.5965

Tabla 7.12: Dataset 3-Binario: Benigno-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(22)
	Dlj4Mlp	35.4171	37.8182	43.5744
AdaBoostM1	Dlj4Mlp	48.0884	49.0652	51.3772
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	43.7866	48.944	51.814
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	45.1174	48.1491	50.6512
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	48.3642	50.3145	51.5512
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	47.3514	48.2091	49.6892
FilteredClassifier	Dlj4Mlp	46.3514	46.9091	49.786
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	44.9874	45.6751	50.5282
LogitBost	Dlj4Mlp	47.8094	50.5455	52.5192
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	46.9394	47.1571	49.6512
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	48.02714	47.9	50.6747
Multisearch	Dlj4Mlp	45.8974	46.2571	50.8862
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	36.4405	37.8401	43.6744
RandomCommitte	Dlj4Mlp	47.3514	38.6936	49.8992
RandomizableFilteredClassifier	Dlj4Mlp	38.5432	45.7909	42.7124
RandomSubSpace	Dlj4Mlp	43.9233	43.5097	43.6744
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	47.3514	44.2727	48.6559
ThresholdSelector	Dlj4Mlp	47.7314	45.1995	40.5868
Vote	Dlj4Mlp	47.3514	47.1438	46.1651
WeightedInstancesHandlerWrapper	Dlj4Mlp	45.8284	46.4679	50.6512

Tabla 7.13: Dataset 4-Multiclase

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(45)
	Dlj4Mlp	42.2698	50.9561	45.2284
AdaBoostM1	Dlj4Mlp	14.9245	39.3794	15.4286
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	3.3698	59.1931	55.2228
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	13.9245	39.2971	59.6044
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	60.9509	61.8633	60.4934
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	14.2615	34.478	58.6286
FilteredClassifier	Dlj4Mlp	14.6815	47.5069	61.9087
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	13.0245	38.2784	13.2695
LogitBost	Dlj4Mlp	61.1509	63.0943	61.1936
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	42.3264	55.105	49.5714
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	59.2509	44.5409	13.0429
Multisearch	Dlj4Mlp	14.0245	32.178	58.7563
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	44.1566	50.1895	54.7143
RandomCommitte	Dlj4Mlp	13.5245	53.8374	58.5436
RandomizableFilteredClassifier	Dlj4Mlp	48.7868	40.5072	54.6626
RandomSubSpace	Dlj4Mlp	52.9038	59.9256	51.5286
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	61.1509	63.0943	59.6666
ThresholdSelector	Dlj4Mlp	x	x	x
Vote	Dlj4Mlp	15.7245	33.5436	59.1986
WeightedInstancesHandlerWrapper	Dlj4Mlp	13.2245	33.2497	59.4286

Tabla 7.14: Dataset 4- Binario:Normal-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR (39.5)
	Dlj4Mlp	62.1551	68.4613	56.8769
AdaBoostM1	Dlj4Mlp	15.0999	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	84.2741	78.5339	76.7587
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	12.9079	71.5891	19.3493
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	80.3271	76.4217	74.6564
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	14.4419	54.1081	73.4194
FilteredClassifier	Dlj4Mlp	79.7235	72.4864	28.3941
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	12.4419	57.8839	19.0182
LogitBost	Dlj4Mlp	79.3271	77.0027	74.4194
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	14.1969	53.2677	73.1872
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	14.4419	59.6447	18.5806
Multisearch	Dlj4Mlp	12.4699	51.7071	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	61.2791	68.4591	56.1358
RandomCommitte	Dlj4Mlp	78.5581	66.3404	73.4194
RandomizableFilteredClassifier	Dlj4Mlp	77.6433	72.0826	67.4414
RandomSubSpace	Dlj4Mlp	79.7209	76.5036	71.8387
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	79.5581	78.5447	74.5532
ThresholdSelector	Dlj4Mlp	79.5581	47.6824	73.7442
Vote	Dlj4Mlp	14.1666	52.8172	73.7474
WeightedInstancesHandlerWrapper	Dlj4Mlp	13.4419	51.8001	73.7474

Tabla 7.15: Dataset 4-Binario: Normal-Benigno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(41)
	Dlj4Mlp	64.8006	71.1574	58.4928
AdaBoostM1	Dlj4Mlp	73.4204	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	73.9317	76.913	77.1844
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	72.8894	53.0532	71.7154
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	76.6517	72.8411	71.674
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	73.6797	53.4632	71.6784
FilteredClassifier	Dlj4Mlp	73.7917	70.5318	71.5251
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	12.8997	28.9456	71.1981
LogitBost	Dlj4Mlp	16.8483	17.4719	19.8886
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	73.4997	52.8312	71.9494
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	74.6517	47.1873	71.9494
Multisearch	Dlj4Mlp	73.7805	50.9332	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	63.9499	70.4599	59.1352
RandomCommitte	Dlj4Mlp	73.0007	52.8052	71.3064
RandomizableFilteredClassifier	Dlj4Mlp	72.0045	65.5663	62.4806
RandomSubSpace	Dlj4Mlp	69.5189	74.1506	75.0139
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	74.6517	74.1561	72.0874
ThresholdSelector	Dlj4Mlp	16.6063	41.5693	18.6306
Vote	Dlj4Mlp	74.6517	52.4632	71.6041
WeightedInstancesHandlerWrapper	Dlj4Mlp	72.8955	50.6712	71.5965

Tabla 7.16: Dataset 4-Binario:Benigno-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(22)
	Dlj4Mlp	35.4171	37.8182	43.5744
AdaBoostM1	Dlj4Mlp	48.0884	49.0652	51.3772
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	43.7866	48.944	51.814
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	45.1174	48.1491	50.6512
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	48.3642	50.3145	51.5512
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	47.3514	48.2091	49.6892
FilteredClassifier	Dlj4Mlp	46.3514	46.9091	49.786
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	44.9874	45.6751	50.5282
LogitBost	Dlj4Mlp	47.8094	50.5455	52.5455
MetaCost	Dlj4Mlp			
MultiClassClassifier	Dlj4Mlp	46.9394	47.1571	49.6512
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	48.02714	47.9	50.6747
Multisearch	Dlj4Mlp	45.8974	46.2571	50.8862
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	36.4405	37.8401	43.6744
RandomCommitte	Dlj4Mlp	47.3514	38.6936	49.8992
RandomizableFilteredClassifier	Dlj4Mlp	38.5432	45.7909	42.7124
RandomSubSpace	Dlj4Mlp	43.9233	43.5097	43.6744
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	47.3514	44.2727	48.6559
ThresholdSelector	Dlj4Mlp	47.7314	45.1995	40.5868
Vote	Dlj4Mlp	47.3514	47.1438	46.1651
WeightedInstancesHandlerWrapper	Dlj4Mlp	45.8284	46.4679	50.6512

Tabla 7.17: Dataset 5-Multiclase

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(45)
	Dlj4Mlp	47.1698	54.0881	48.5714
AdaBoostM1	Dlj4Mlp	17.9245	41.5094	19.4286
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	47.1698	62.8931	58.8571
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	17.9245	41.5094	63.4286
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	64.1509	65.0943	63.4286
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	17.9245	36.478	63.4286
FilteredClassifier	Dlj4Mlp	17.9245	51.2579	65.1429
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	17.9245	41.5094	17.1429
LogitBost	Dlj4Mlp	64.1509	65.0943	63.4286
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	46.2264	58.805	52.5714
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	64.1509	46.5409	17.1429
Multisearch	Dlj4Mlp	17.9245	36.478	63.4286
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	49.0566	54.4025	57.7143
RandomCommitte	Dlj4Mlp	17.9245	55.6604	63.4286
RandomizableFilteredClassifier	Dlj4Mlp	51.8868	52.8302	51.4286
RandomSubSpace	Dlj4Mlp	56.6038	62.5786	55.4286
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	64.1509	65.0943	63.4286
ThresholdSelector	Dlj4Mlp	x	x	x
Vote	Dlj4Mlp	17.9245	36.478	63.4286
WeightedInstancesHandlerWrapper	Dlj4Mlp	17.9245	36.478	63.4286

Tabla 7.18: Dataset 5-Binario:Normal-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR (39.5)
	Dlj4Mlp	66.2791	71.5953	60
AdaBoostM1	Dlj4Mlp	17.4419	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	88.3721	80.9339	80
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	17.4419	74.3191	22.5806
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	82.5581	80.5447	77.4194
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	17.4419	56.0311	77.4194
FilteredClassifier	Dlj4Mlp	86.0465	75.4864	32.2581
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	17.4419	62.6459	22.5806
LogitBost	Dlj4Mlp	82.5581	80.5447	77.4194
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	17.4419	56.0311	77.4194
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	17.4419	61.8677	22.5806
Multisearch	Dlj4Mlp	17.4419	56.0311	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	66.2791	71.5953	60
RandomCommitte	Dlj4Mlp	82.5581	68.0934	77.4194
RandomizableFilteredClassifier	Dlj4Mlp	81.3953	74.3191	70.9677
RandomSubSpace	Dlj4Mlp	83.7209	80.1556	74.8387
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	82.5581	80.5447	77.4194
ThresholdSelector	Dlj4Mlp	82.5581	49.8054	77.4194
Vote	Dlj4Mlp	17.4419	56.0311	77.4194
WeightedInstancesHandlerWrapper	Dlj4Mlp	17.4419	56.0311	77.4194

Tabla 7.19: Dataset 5-Binario:Normal-Benigno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(41)
	Dlj4Mlp	69.6629	75.2809	62.6582
AdaBoostM1	Dlj4Mlp	78.6517	x	75.9494
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	78.6517	79.7753	81.0127
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	78.6517	55.8052	75.9494
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	78.6517	77.5281	75.9494
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	78.6517	55.8052	75.9494
FilteredClassifier	Dlj4Mlp	78.6517	74.5318	75.9494
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	78.6517	33.7079	75.9494
LogitBost	Dlj4Mlp	21.3483	22.4719	24.0506
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	78.6517	55.8052	75.9494
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	78.6517	50.1873	75.9494
Multisearch	Dlj4Mlp	78.6517	55.8052	75.9494
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	69.6629	75.2809	62.6582
RandomCommitte	Dlj4Mlp	78.6517	55.8052	75.9494
RandomizableFilteredClassifier	Dlj4Mlp	76.4045	68.5393	66.4557
RandomSubSpace	Dlj4Mlp	75.2809	77.9026	79.1139
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	78.6517	77.5281	75.9494
ThresholdSelector	Dlj4Mlp	21.3483	44.5693	24.0506
Vote	Dlj4Mlp	78.6517	55.8052	75.9494
WeightedInstancesHandlerWrapper	Dlj4Mlp	78.6517	55.8052	75.9494

Tabla 7.20: Dataset 5-Binario:Benigno-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(22)
	Dlj4Mlp	40.5405	41.8182	47.6744
AdaBoostM1	Dlj4Mlp	51.3514	51.8182	54.6512
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	48.6486	51.8182	55.814
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	51.3514	50.9091	54.6512
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	51.3514	54.5455	54.6512
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	51.3514	50.9091	54.6512
FilteredClassifier	Dlj4Mlp	51.3514	50.9091	54.6512
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	51.3514	50.9091	54.6512
LogitBost	Dlj4Mlp	51.3514	54.5455	54.6512
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	51.3514	50.9091	54.6512
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	51.3514	50	54.6512
Multisearch	Dlj4Mlp	51.3514	50.9091	54.6512
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	40.5405	41.8182	47.6744
RandomCommitte	Dlj4Mlp	51.3514	41.8182	54.6512
RandomizableFilteredClassifier	Dlj4Mlp	43.2432	49.0909	47.6744
RandomSubSpace	Dlj4Mlp	48.6486	47.272	47.6744
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	51.3514	47.2727	54.6512
ThresholdSelector	Dlj4Mlp	51.3514	48.1818	45.3488
Vote	Dlj4Mlp	51.3514	50.9091	54.6512
WeightedInstancesHandlerWrapper	Dlj4Mlp	51.3514	50.9091	54.6512

Tabla 7.21: Dataset 6-Multiclase

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(45)
	Dlj4Mlp	50.4963	57.4146	51.8979
AdaBoostM1	Dlj4Mlp	21.6797	45.2646	23.1838
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	51.146	66.8693	62.8333
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	20.3568	43.9417	65.8609
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	66.5381	67.4815	65.8158
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	21.1668	39.7203	66.6709
FilteredClassifier	Dlj4Mlp	21.1668	54.5002	68.1852
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	22.1708	45.7557	21.3892
LogitBost	Dlj4Mlp	68.2732	69.2166	67.5509
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	50.1787	62.7573	56.5237
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	67.6935	50.0835	20.6855
Multisearch	Dlj4Mlp	21.7897	40.3432	67.2938
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	53.7989	59.1448	62.4566
RandomCommitte	Dlj4Mlp	21.5668	59.3027	67.0709
RandomizableFilteredClassifier	Dlj4Mlp	55.541	56.4844	55.0828
RandomSubSpace	Dlj4Mlp	60.3458	66.3206	59.1706
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	68.2882	69.1316	67.5659
ThresholdSelector	Dlj4Mlp	x		
Vote	Dlj4Mlp	21.1668	39.7203	66.6709
WeightedInstancesHandlerWrapper	Dlj4Mlp	21.1668	39.7203	66.6709

Tabla 7.22: Dataset 6-Binario: Normal-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(39)
	Dlj4Mlp	71.1553	76.8112	64.7229
AdaBoostM1	Dlj4Mlp	22.2568	x	x
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	94.187	86.1992	85.3572
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	22.1919		
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	86.4234	85.2999	82.0714
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	22.0948	60.684	82.0723
FilteredClassifier	Dlj4Mlp	90.1699	79.1384	36.3815
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	21.5653		
LogitBost	Dlj4Mlp	86.6815	85.1974	82.1956
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	21.5653	60.0073	82.07363
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	21.5653	66.2159	26.704
Multisearch	Dlj4Mlp	21.5653	60.3793	x
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	70.4025	76.2529	63.8826
RandomCommitte	Dlj4Mlp	86.6815	72.47992	82.2846
RandomizableFilteredClassifier	Dlj4Mlp	84.654	78.1053	75.1677
RandomSubSpace	Dlj4Mlp	88.1522	84.5869	79.1039
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	86.3204	83.6427	80.6711
ThresholdSelector	Dlj4Mlp	86.5444	52.6527	80.9921
Vote	Dlj4Mlp	22.0392	59.8943	82.0817
WeightedInstancesHandlerWrapper	Dlj4Mlp	22.0392	59.8943	82.0817

Tabla 7.23: Dataset 6-Binario: Normal-Benigno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(41)
	Dlj4Mlp	73.4284	78.5462	66.0108
AdaBoostM1	Dlj4Mlp	82.2969	x	79.5946
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	83.9813	84.7409	85.9783
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	82.4172	59.5707	79.7149
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	82.4172	81.2936	79.5916
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	82.3169	59.4704	79.5816
FilteredClassifier	Dlj4Mlp	82.3169	78.197	79.6837
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	82.3169	37.3731	79.0836
LogitBost	Dlj4Mlp	24.5836	25.7072	28.3159
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	82.4172	59.5707	79.7149
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	82.4172	53.9528	79.7149
Multisearch	Dlj4Mlp	82.4172	59.5707	79.7149
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassfier	Dlj4Mlp	73.8402	79.4582	66.8355
RandomCommitte	Dlj4Mlp	83.6142	60.7677	80.9119
RandomizableFilteredClassifier	Dlj4Mlp	80.0471	72.1819	70.0983
RandomSubSpace	Dlj4Mlp	78.9232	81.5449	82.7562
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	82.906	81.8494	80.2037
ThresholdSelector	Dlj4Mlp	24.6026	48.8236	28.3049
Vote	Dlj4Mlp	82.1843	58.4395	79.482
WeightedInstancesHandlerWrapper	Dlj4Mlp	82.1843	59.3378	79.482

Tabla 7.24: Dataset 6-Binario: Benigno-Maligno

Ensamblados(Meta)	Deep learning	2/3-1/3	CV-10	MR(22)
	Dlj4Mlp	44.4157	45.6934	51.5496
AdaBoostM1	Dlj4Mlp	54.7056	55.1724	58.0054
AdditiveRegression	Dlj4Mlp	x	x	x
AttributeSelectedClassifier	Dlj4Mlp	52.8029	55.9725	59.9683
AutoWEKAClassifier	Dlj4Mlp	x	x	x
Bagging	Dlj4Mlp	54.3252	53.8829	57.625
ClassificationViaClustering	Dlj4Mlp	x	x	x
ClassificationViaRegretion	Dlj4Mlp	54.1052	57.2993	57.405
CostSensitiveClassifier	Dlj4Mlp	x	x	x
CVParameterSelection	Dlj4Mlp	55.1037	54.6614	58.4035
FilteredClassifier	Dlj4Mlp	55.1037	54.6614	58.4035
GridSearch	Dlj4Mlp	x	x	x
IterativeClassifierOptimizer	Dlj4Mlp	55.2266	54.7843	58.5264
LogitBost	Dlj4Mlp	55.2266	58.865	58.5264
MetaCost	Dlj4Mlp	x	x	x
MultiClassClassifier	Dlj4Mlp	55.3276	54.8853	58.6274
MultiClassClassifierUpdateable	Dlj4Mlp	x	x	x
MultiScheme	Dlj4Mlp	55.2266	53.8752	58.5264
Multisearch	Dlj4Mlp	55.2266	54.7843	58.5264
OneClassClassifier	Dlj4Mlp	x	x	x
OrdinalClassClassifier	Dlj4Mlp	44.4157	45.7854	51.5496
RandomCommitte	Dlj4Mlp	55.2266	45.6934	58.5264
RandomizableFilteredClassifier	Dlj4Mlp	47.1184	52.9661	51.5496
RandomSubSpace	Dlj4Mlp	52.5238	51.1472	51.5496
RegressionByDiscretization	Dlj4Mlp	x	x	x
Staking	Dlj4Mlp	55.6045	51.5258	58.9043
ThresholdSelector	Dlj4Mlp	55.6045	52.1549	49.6019
Vote	Dlj4Mlp	55.2266	54.7843	58.5264
WeightedInstancesHandlerWrapper	Dlj4Mlp	55.2266	54.7843	58.5264