



INSTITUTO TECNOLÓGICO SUPERIOR DE MISANTLA

CATEGORIZACIÓN DE ANORMALIDADES
CANCERÍGENAS EN MASTOGRAFÍAS DIGITALES
USANDO *DEEP LEARNING*

TESIS

QUE PARA OBTENER EL GRADO DE:
Maestra en Sistemas Computacionales

Presenta:

Marlenne Viridiana Roque Geronimo

Director:

Dr. Eddy Sánchez de la Cruz

Co-Director:

Dr. Rajesh Roshan Biswal

Misantla, Veracruz Junio de 2018

Dedicatoria

A mi madre Clara.

Gracias por su amor, consejos, su apoyo, por alentarme cuando las cosas no están bien. Siempre has sido la persona más importante en mi vida, es por ello que el culminar esta etapa es gracias a ti, te amo.

A mis hermanos.

Vanessa y Eduardo ustedes han sido un impulso a lograr este objetivo, si yo pude llegar a esta instancia, ustedes pueden obtener grandes logros, los quiero.

A Karim.

Estoy agradecida por haberte conocido, eres una excelente persona, agradezco su amistad, apoyo, solidaridad, confianza. Eres parte importante en mi vida.

A mi nueva familia.

Familia Quintero Rodríguez, agradecida con dios por haber permitido encontrar personas tan buenas como ustedes. Dra. Susana Rodríguez gracias por hacerme sentir parte de su familia y por brindarme su amistad incondicional, dios los bendiga.

Agradecimientos

Dr. Luis Alberto Morales Rosales agradezco a usted por creer en mí, el que ahora concluya la maestría en primera instancia es por que usted me aceptó como estudiante, gracias por los conocimientos técnicos y consejos de vida.

Dr. Eddy Sánchez de la Cruz gracias por haber sido mi asesor en este proyecto, por brindarme apoyo como docente a pesar que este proyecto se inició con varios contratiempos, gracias por su confianza.

Maestros del *Instituto Tecnológico Superior de Misantla*, agradezco los conocimientos impartidos, en particular a el Dr. Simón Pedro Arguijo y a el maestro Roberto Ángel Melendez.

A mis compañeros de maestría de cada una de las generaciones que me brindaron su amistad y apoyo a lo largo de esta travesía. Y a cada de las amistades en la misma institución.

Gracias al *Tecnológico Nacional de México* y al *Consejo Nacional de Ciencia y Tecnología* (Conacyt) por creer en mis capacidades.

Resumen

La presente investigación aborda el problema de categorizar anomalías cancerígenas en mastografías digitales. Siendo que el cáncer de mama es la primera causa de enfermedad entre las mujeres con una cifra de 1,671,149 de nuevos casos cada año y una prevalencia de 6,232,108, (Secretaría, 2015) es muy importante su estudio con métodos novedosos de inteligencia artificial como es *Deep learning*. El uso de esta técnica, según el estado del arte, en la detección y clasificación del cáncer de mama ha brindado resultados aceptables. El estudio médico para detectar cáncer de mama es mediante la mastografía o mamografía, aún cuando este estudio tradicional es bueno, tiene un margen de error visible (Krizhevsky, Sutskever, y Hinton, 2012), esto ocurre debido a las dificultades para apreciar cada detalle de la imagen mastográfica.

Este proyecto no suplente el diagnóstico ni la opinión del experto médico, solo pretende ser un apoyo. Para abordarlo, se utilizó Deep learning para categorizar anomalías cancerígenas en imágenes de mastografías digital, estas imágenes de tamaño 120x120 píxeles en formato pgm, con 176 imágenes obtenidas del repositorio (MIAS, 1994) (MIAS). Para aumentar el espectro de posibilidades experimentales, se utilizaron combinaciones de algoritmos ensamblados y un algoritmo deep learning, con tres datasets binarios: {Normal, Benigno}, {Normal, Maligno} y {Benigno, Maligno}; y un dataset multiclase: {Normal, Benigno, Maligno}. Después de realizar las pruebas, con tres criterios de muestreo, se obtuvo que el conjunto de datos binario {Normal, Maligno} generó el mejor resultado combinando *AttributeSelected Classifier + D4jMlpclassifier*: 83.7338 % instancias correctamente clasificadas, con ello se pudo observar que el clasificador distingue mejor las anomalías cuando se tienen sólo dos clases.

Dicho resultado, amplía el campo de estudio en relación de las técnicas convencionales, lo cual permite contribuir en el desarrollo tecnológico y social en beneficio de pacientes propensos a padecer cáncer de mama.

Abstract

In this work, we address the problem of categorizing carcinogenic abnormalities in digital mammograms. Breast cancer is the leading cause of disease among women with a number of 1,671,149 of new cases each year and a prevalence of 6,232,108 (Secretaría, 2015). It is very important to study with new methods of artificial intelligence such as deep learning. By using this technique, the detection and classification of breast cancer has provided acceptable results according to the state of the art. The medical studies to detect breast cancer is through mammography or also called mammography even though this traditional has good precision, but it has a margin of visible error (Krizhevsky y cols., 2012), this happens due to the difficulties to appreciate every detail of the mastographic image.

This project does not intend to replace the diagnosis or the opinion of the medical expert, it only intends to be a medical support to address it, to tackle it. Deep learning was used to categorize carcinogenic abnormalities in digital mammography images, these digital images of size 120x120 pixels in pgm format, and with 176 images obtained from the repository (MIAS, 1994)(MIAS) were used for this work. We used ensemble methods for increase the possibilities in the experiments to obtain better results. We used combinations of assembled algorithms and a deep learning algorithm, with three binary datasets: Normal, Benign, Normal, Malign and Benign, Malign; and a multi-class dataset: Normal, Benign, Malign. After performing the tests with three sampling criteria, we obtained that the binary data set Normal, Malignant generated the best result by combining *AttributeSelected Classifier + D4jMlpclassifier*: 83.7338% it was observed that the classifier distinguishes the abnormalities better when there are only two classes.

This result extends the field of study in relation to conventional techniques, which allows contributing to technological and social development for the benefit of patients prone to have breast cancer.

Índice general

Dedicatoria	II
Agradecimientos	III
Resumen	IV
Abstract	V
Índice de figuras	VIII
Índice de tablas	IX
Índice de algoritmos	XI
1. Generalidades	12
1.1. Introducción	13
1.2. Planteamiento del problema	14
1.2.1. Justificación	15
1.3. Objetivos	15
1.3.1. Objetivo general	15
1.3.2. Objetivos específicos	15
1.4. Hipótesis	16
1.5. Propuesta de solución	16
1.6. Alcances y limitaciones	17
1.7. Estructura de la tesis	18
2. Marco Teórico	19
2.1. Aprendizaje Automático	20
2.1.1. Algoritmos de aprendizaje automático	20
2.2. Deep learning (Aprendizaje profundo)	26
2.2.1. Redes neuronales artificiales	27
2.3. Métricas de evaluación	30
2.4. Imágenes médicas	30

2.4.1. Mastografía digital	31
2.4.2. Ruido en imágenes	31
2.5. Estado del Arte	32
3. Materiales y Métodos	34
3.1. Materiales	35
3.2. Método	37
3.2.1. Preprocesamiento de las imágenes	37
3.2.2. Algoritmos de aprendizaje profundo	38
4. Experimentos y resultados	41
4.1. Requerimientos	42
4.2. Pre procesamiento de las imágenes	43
4.3. Clasificación	44
5. Análisis de resultados	52
5.1. Análisis de algoritmos con mayor clasificación	53
6. Conclusiones y Trabajo a Futuro	55
6.1. Conclusiones	56
6.2. Trabajos Futuros	56
Referencias	57
Anexos	60
6.2.1. Anexo A	60
6.2.2. Criterio 2/3 - 1/3	61
6.2.3. Criterio Cross folder	65
6.2.4. Criterio Muestra Representativa	68

Índice de figuras

1.1. Diagrama de propuesta de solución.	17
2.1. Redes en deep learning, basado en el libro Advances in knowledge discovery and data mining (Fayyad y cols., 1996).	29
2.2. Descripción gráfica de una red neuronal artificial (López y Fernández, 2008).	30
2.3. Representación del ruido en una imagen (elaboración propia).	31
3.1. Imagen de 1024x1024 (Izq.) imagen recortada (Der.) a un tamaño de 120x120 pixeles	37
3.2. Red neuronal multilayer perceptrón (López y Fernandez, 2008).	39
4.1. Fragmento del dataset Multiclase {Normal, Benigno, Maligno} almacenado en un archivo en formato csv	44
4.2. Gráfica del dataset {Normal, Benigno} iteraciones.	46
4.3. Gráfica del dataset {Normal, Maligno} iteraciones.	46
4.4. Gráfica del dataset {Benigno, Maligno} iteraciones.	47
4.5. Gráfica del dataset Multiclase {Normal, Benigno, Maligno} iteraciones.	47
4.6. Gráfica del dataset {Normal, Benigno} semillas.	48
4.7. Gráfica del dataset {Normal, Maligno} semillas.	48
4.8. Gráfica del dataset {Benigno, Maligno} semillas.	49
4.9. Gráfica del dataset Multiclase {Normal, Benigno, Maligno} semillas.	49

Índice de tablas

2.1. Método bayesiano.	21
2.2. Método de funciones de clasificación.	22
2.3. Método de clasificación perezoso.	22
2.4. Meta clasificadores o algoritmos ensamblados parte uno.	23
2.5. Meta clasificadores o algoritmos ensamblados parte dos.	24
2.6. Método varios.	24
2.7. Método de generación de reglas.	25
2.8. Método de generación de árboles de decisión.	26
3.1. Tabla de datasets disponibles en internet.	35
4.1. Tabla de la combinación de algoritmos ensamblados y el algoritmo DI4Mlp, donde se muestra la combinación que obtuvo mayor clasificación de cada dataset	45
4.2. Matriz de confusión del dataset {Normal, Benigno}	50
4.3. Matriz de confusión del dataset {Normal, Maligno}	50
4.4. Matriz de confusión del dataset {Benigno, Maligno}	50
4.5. Matriz de confusión del dataset Multiclase {Normal, Benigno ,Maligno} . . .	51
5.1. Tabla de comparación entre estado del arte y resultados de la presente investigación	53
6.1. Dataset binario {Normal, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa	62
6.2. Dataset Binario {Normal, Benigno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa	63
6.3. Dataset {Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa	64
6.4. Dataset Multiclase {Normal, Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa	65

6.5. Dataset Binario Normal, Benigno y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de cross folder	66
6.6. Dataset Binario {Normal, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de cross folder	66
6.7. Dataset Binario {Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de cross folder	67
6.8. Dataset Multiclase {Normal, Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de cross folder	68
6.9. Dataset Binario {Normal, Benigno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa	69
6.10. Dataset Binario {Normal, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa	70
6.11. Dataset Binario {Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa	71
6.12. Dataset Multiclase {Normal, Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa	72

Índice de algoritmos

creacion.py	60
-----------------------	----

Capítulo 1

Generalidades

1.1. Introducción

(DelaCruz, Alpuín-Jiménez, Domínguez, y Parra, 2011), mencionan que el cáncer de mama es la neoplasia ¹ más común en la población mundial y la primera causa de enfermedad entre las mujeres con una cifra de 1,671,149 nuevos casos cada año y una prevalencia de 6,232,108,(Secretaría, 2015), es la causa de mayor mortalidad en la mujer con 521,907 defunciones anuales, lo que representa un 14.7% de la población total, de las cuales ocurren más en países desarrollados, en América latina, el cáncer de mama es la primera neoplasia en las mujeres con 152,059 casos anuales, la incidencia en la región es de 27 casos por 100,000 mujeres, alcanzando valores superiores a 50 en países como Argentina, Uruguay, Brasil y Guyana (Secretaría, 2015). En México, en el año 2014, se registraron 11,372 casos nuevos de cáncer de mama con una tasa de incidencia de 22.56 por 100,000 habitantes mayores de 10 años. En mujeres de 25 años, en el año 2015 se registraron 6,252 defunciones con una tasa cruda de 18 defunciones por 100,000 mujeres. Las entidades con mayor mortalidad por cáncer de mama son: Sonora (28.6), Nuevo León (26), Coahuila (25.7), Chihuahua (24.8), Cd. México (24.7) y Sinaloa (22.2) (Secretaría, 2015). La edad promedio de los casos de cáncer de mama es de 54.9 años, el porcentaje más alto en el número de casos se registra en el grupo de 50 a 59 años con el 45 % (Secretaría, 2015) de todos los casos.

La tecnología aplicada al área de la imagenología médica (en específico a las mastografías) y el análisis de estas imágenes por computadora, es de gran beneficio para establecer diagnósticos certeros en este campo, las obtenciones de las imágenes hacen que el especialista tenga una visualización de las características y medición de posibles anomalías que desencadenan en enfermedades. El uso de imágenes médicas para el diagnóstico es imprescindible, en particular, la adquisición de imágenes para la detección de cáncer de mama utilizando la mastografía también conocida como mamografía; estas imágenes apoyan al diagnóstico y tratamiento de enfermedades, la planeación de terapias y la planificación de procedimientos a los que debe someterse cada paciente según sea el caso (Krizhevsky y cols., 2012). Aún cuando los estudios de mastografía son ciertamente efectivos, existe la probabilidad de que exista una anomalía cancerígena que a simple vista no se pueda apreciar, es decir: el médico especialista (radiólogo) podría dar un diagnóstico al paciente erróneo, esto ocurre debido a las dificultades para apreciar cada detalle de la imagen.

En el estado del arte (Pérez, 2017), reporta trabajos previos relacionados a los avances en el campo de la computación y a la efectividad de las técnicas en el uso de las tareas de detección y segmentación de tumores, así como su seguimiento y control, la visualización y cuantificación del flujo sanguíneo, o la creación de sistemas de ayuda para interpretación de resultados médicos, es posible obtener porcentajes óptimos en la detección de anomalías cancerígenas en mastografías digitales; por lo tanto, se puede suponer que en un futuro éstas técnicas serán cada vez más utilizadas, convirtiéndose muchas de estas tareas en algo común de una computadora.

¹La neoplasia, es la formación anormal en alguna parte del cuerpo de un tejido nuevo de carácter tumoral de tipo benigno o maligno.

(Gerazov y Conceicao, 2017), describen que el deep learning (aprendizaje profundo) es una técnica muy utilizada en el área de imágenes biomédicas, incluso se le ha comparado a nivel humano, por la forma en que aprende, también enfatizan que los resultados reportados en la literatura al aplicarse al diagnóstico de cáncer de mama son aceptables y en muchos casos, competitivos. Se ha demostrado que las redes neuronales profundas (Deep Neural Network, por sus siglas en inglés de *DNN*) se basan en el proceso de las neuronas de un cerebro humano, al aprender con respecto a la entrada de información que se les da (Xu y cols., 2014). Las imágenes médicas son importantes para establecer diagnósticos certeros para la detección de diversas enfermedades, cuando se habla de imágenes estas pertenecen al área de visión por computadora, donde se han logrado resultados impresionantes, (Krizhevsky y cols., 2012), en el contexto de la mastografía, cuando se han empleado técnicas de Inteligencia Artificial (IA) ² ha permitido (Rautaray y Agrawal, 2015) detectar masas mamarias donde se encuentran posibles anomalías, aumentando el porcentaje de efectividad para un diagnóstico correcto. Con la presente investigación se puede ampliar el campo de estudio para ayudar en la categorización de anomalías en las imágenes de mastografías.

1.2. Planteamiento del problema

Según (Ríos y Avendaño, 2008), la mastografía es la “placa” que se utiliza para el estudio visual por parte del radiólogo, con la finalidad de detectar alguna anomalía cancerígena en el seno. Esta técnica ayuda a la correcta canalización de la paciente, para decidir si amerita la realización de una biopsia, con el fin de descartar o confirmar la anomalía. Siendo el cáncer de mama una de las principales enfermedades que afecta a las mujeres en el mundo (y en México), es importante detectar oportunamente esta enfermedad para disminuir el nivel de mortalidad ³.

Existen situaciones donde el médico especialista no logra detectar a simple vista la existencia de algún tumor cancerígeno, ya sea por factores externos como cansancio, estrés, mala iluminación, mala calidad de imagen, etc. Aunado a esto, el reconocimiento visual a través de la mastografía, es uno de los procedimientos más difíciles de interpretar según (Cruz y Vilella, 2006), por estas causas existen situaciones donde a el paciente se le da un mal diagnóstico.

Diversos métodos computacionales han sido utilizados para ayudar en el reconocimiento de características que ayuden a distinguir correctamente si existen o no anomalías. En esta presente investigación, se propone utilizar la combinación de algoritmos ensamblado con el deep learning para categorizar anomalías en una mastografía digital.

²Para la explicación de este trabajo se emplea el acrónimo de IA para referirse a Inteligencia Artificial

³El termino mortalidad se refiere en parte a la calidad de mortal; es decir, de lo que ha de morir o está sujeto a la muerte, todo lo contrario a la vida

1.2.1. Justificación

La mejor forma de combatir cualquier tipo de cáncer es la detección temprana, con el fin de poder erradicar esta enfermedad y evitar el fallecimiento de la persona. El uso de deep learning, en particular en imágenes médicas, ha ayudado en la detección de enfermedades con el uso de técnicas de IA que ayudan a abstraer información útil para el médico especialista. Cuando el médico (radiólogo, oncólogo, etc.) analiza la imagen de una mastografía y detecta en ella algún indicio de anormalidad, decide las indicaciones para el paciente. El beneficio principal es el diagnóstico oportuno, debido a que esta enfermedad al ser detectada en sus primeras etapas aumenta las probabilidades de erradicarlo y evitar fallecimientos. El resultado de la presente investigación pretende ampliar el campo de estudio en relación de las técnicas de IA y el categorizar anormalidades cancerígenas, este proyecto no suple el diagnóstico ni la opinión del experto.

Impacto social. El poder categorizar anormalidades en una imagen digital (mastografía) con técnicas computacionales y analizar un área específica para establecer si hay cáncer o no, sirve como apoyo para el médico especialista. Esto hace que la paciente reciba un certero diagnóstico y se le de indicaciones para el tratamiento de la enfermedad, dando como resultado la disminución de mortalidad.

Impacto Económico. Beneficia directamente al paciente, en el costo que implicaría repetir varias ocasiones una mastografía, dado el costo que representa y que la mayor parte de la población no cuenta con el recurso económico para solventar tales gastos, el traslado de su lugar de origen a la institución de salud.

Impacto Tecnológico. En el área de la salud se han desarrollado aplicaciones que apoyan de forma positiva a el diagnostico de padecimientos oportunos, el categorizar la anormalidad en la mastografía, permite aprovechar las tecnologías y avances en la IA en el apoyo de la medicina.

1.3. Objetivos

1.3.1. Objetivo general

Categorizar anormalidades cancerígenas en mastografías digitales combinando clasificadores ensamblados y un algoritmo de deep learning.

1.3.2. Objetivos específicos

- Obtener una base de datos de dominio público con mastografías digitales.
- Seleccionar las imágenes a usar, para construir los conjuntos de entrenamiento y prueba.

- Seleccionar una zona de interés de cada una imagen de acuerdo a las coordenadas donde se sitúa la anormalidad (matriz resultante).
- Aplicar aplanamiento a la matriz resultante, con la finalidad de obtener un vector por cada matriz.
- Concatenar cada vector para construir un dataset, donde se almacenan las 176 imágenes por vectores.
- A partir del dataset creado, discriminar combinando clasificadores ensamblados, con el algoritmo de aprendizaje profundo *Dl4JMp*, y obtener el mayor porcentaje de clasificación correcta.
- Al obtener resultados, variar el número de iteraciones y la semilla leatoria para obtener un averiguar si es posible obtener un porcentaje con mayor certeza.
- Analizar las métricas de matriz de confusión.

1.4. Hipótesis

Es posible, combinando clasificadores ensamblados con el algoritmo de deep learning *Dl4JMp*, aumentar la categorización correcta de anormalidades cancerígenas en mastografías digitales con un nivel de certeza aceptable en relación al reportado en el estado del arte.

1.5. Propuesta de solución

Para realizar la presente investigación, se analizó y determinó el método que pudiera ser factible para encontrar una solución adecuada al problema, por ello, se diseñó un diagrama en el cuál se explican las etapas para la culminación de esta investigación (ver Figura 1.1). En las siguientes secciones se detalla cada una de las etapas del diagrama.

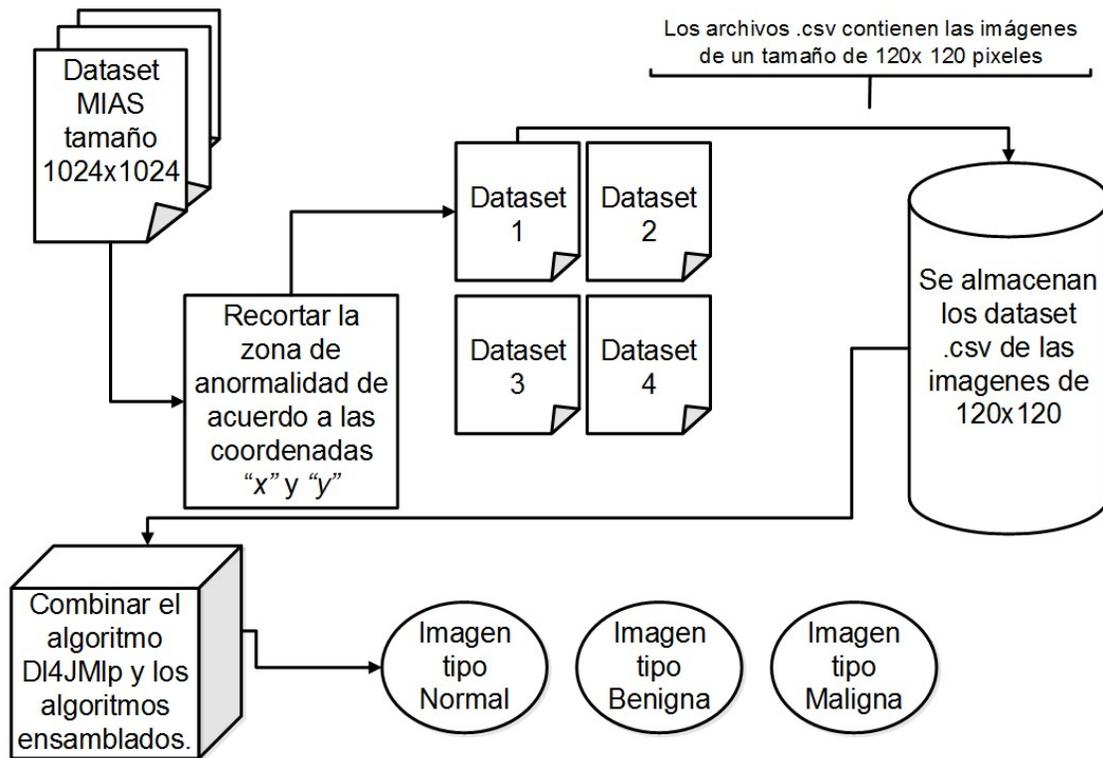


Figura 1.1: Diagrama de propuesta de solución.

Como primer paso se utilizó el dataset de dominio público (MIAS, 1994) para la obtención de imágenes de mastografías digitales que se usaron para esta investigación, este dataset contiene imágenes clasificadas en tres tipos: Normal, Benigna y Maligna; de este dataset se hizo un recorte a cada imagen, tomando en cuenta las coordenadas donde se sitúan las anomalías (extracción de características), en el caso de las imágenes Normales se toma cualquier punto ya que no se encuentra anomalía; las nuevas imágenes (recortes) se almacenan en cuatro distintos datasets, tres de tipo binario: {Normal, Benigno}, {Normal, Maligno} y {Benigno, Maligno} y un multiclase: {Normal, Benigno, Maligno}. Teniendo cada dataset, se aplica el proceso de aplanamiento esto con el fin de almacenar cada imagen en forma de vector para guardarse en un archivo csv. Ahora, cada dataset es implementado en la herramienta *Waikato Environment for Knowledge Analysis* (WEKA) combinando algoritmos ensamblados y el algoritmo DI4JMIp de deep learning. Con esto, se logra obtener la categorización de las imágenes contenidas en cada dataset. Como resultado se obtiene si es una imagen Normal, Benigna o Maligna.

1.6. Alcances y limitaciones

Alcances

- Categorizar anomalías cancerígenas en imágenes de mastografía digital utilizando deep learning.
- Ahorrar tiempo en el diagnóstico médico para el paciente.
- Aumentar las probabilidades de diagnóstico certeros por parte del médico especialista.
- Reducir costos de traslado para el paciente por estudios tales como como biopsias.

Limitaciones

- Acceso restringido a un repositorio de imágenes de la región de Misantla.

1.7. Estructura de la tesis

Este documento se encuentra organizado en seis capítulos que se describen a continuación:

En el capítulo uno, se explican las Generalidades, hacia donde se dirige el presente trabajo, en ella se detallan lo que se pretende resolver y la forma en que solucionará; la introducción explica sobre el tema a tratar, se establece el planteamiento del problema, objetivo general y objetivos específicos, hipótesis, propuesta de solución, alcances y limitaciones de la investigación.

El segundo capítulo describe el Marco teórico, donde se encuentra la nomenclatura utilizada para la comprensión del lector; hay dos secciones importantes en ella, primero se definen conceptos de la investigación y en estado del arte se encuentran los trabajos previos a este tema a abordar que sirven como sustento para la realización del presente trabajo.

El tercer capítulo Materiales y métodos se muestran el preprocesamiento de las imágenes así como la combinación implementada con la que se obtuvo mejor porcentaje: multilayer perceptrón.

Experimento y resultados es el cuarto capítulo, donde se describen los experimentos y la clasificación de los datos, de acuerdo a la metodología definida anteriormente.

El capítulo cinco contiene el análisis de resultados donde se muestran las matrices de confusión.

Conclusiones y trabajo a futuro es el capítulo seis, donde se plantean comentarios de la culminación de esta investigación y se establece lo que se puede realizar más adelante.

Capítulo 2

Marco Teórico

2.1. Aprendizaje Automático

Aprendizaje Automático ó Machine Learning, ¹ es parte de la disciplina de IA (De Raedt, 2008) para aprender y actuar sin ser específicamente programado para ello. (Corso, 2009). Los algoritmos del Aprendizaje Automático se centran en la mejora de los programas informáticos que son capaces de aprender por sí mismos y que pueden crecer y cambiar automáticamente, son capaces de analizar gran cantidad de datos para obtener resultados precisos y al mismo tiempo rápidos (Cambroner y Moreno, 2006). Existen dos enfoques para abordar un problema: el aprendizaje supervisado y aprendizaje no supervisado. El primero trabaja recibiendo un estímulo ó ejemplo (parseo de datos) lo procesa, extrae un modelo cada vez que se introduce un nuevo dato a la máquina y es capaz de dar una respuesta (Cambroner y Moreno, 2006). Una vez que sea entrena a la máquina, por ejemplo, en el caso de reconocer personas en una fotografía, primero se tendrá que introducir elementos que tienen que estar etiquetados, es decir la IA tiene que decir si tiene una cara o no (De Raedt, 2008). En el aprendizaje no supervisado no se introducen entradas, más bien el propio algoritmo tiene que aprender, a reconocer patrones y crear un modelo capaz de obtener conclusiones de conjuntos de datos. Hay muchos tipos de técnicas de Aprendizaje Automático que tratan de descubrir correlaciones sin ningún tipo de aportaciones externas, utilizando sólo los únicos datos en bruto disponibles (Cambroner y Moreno, 2006).

2.1.1. Algoritmos de aprendizaje automático

La minería de datos se vale de diferentes clasificadores para el descubrimiento de patrones, estos clasificadores se encuentran agrupados en métodos (Hall y cols., 2009), los clasificadores pueden combinarse para obtener resultados satisfactorios. En este apartado se mencionan los clasificadores que WEKA proporciona, los cuales fueron usados para hacer las pruebas para esta investigación.

Método Bayesiano

Existen varios clasificadores dentro de este método (ver Tabla 2.1.1), como se puede apreciar, todos basados en las redes bayesianas. El concepto de red bayesiana según (Gallegos-Funes, Gómez-Mayorga, Lopez-Bonilla, y Cruz-Santiago, 2009) es un grafo acíclico dirigido en el que cada nodo representa una variable y cada arco, representa una dependencia probabilística, en la cual se especifica la probabilidad condicional de cada variable dados sus padres, la variable a la que apunta el arco es dependiente (causa-efecto) de la que está en el origen de este. La topología o estructura de la red nos da información sobre las dependencias probabilísticas entre las variables pero también sobre las dependencias condicionales de una variable (o conjunto de variables) dada otra u otras variables, dichas dependencias, simplifican la representación del conocimiento (menos parámetros) y el razonamiento (propagación

¹Aprendizaje Automático es el término que se utiliza de aquí en adelante para la explicación de esta investigación

de las probabilidades).

Teorema de bayes. Sean A y B dos sucesos aleatorios cuyas probabilidades se denotan por $p(A)$ y $p(B)$ respectivamente, verificandose que $p(B) > 0$. Supongamos conocidas las probabilidades *a priori* de los sucesos A y B , es decir, $p(A)$ y $p(B)$, así como la probabilidad condicionada del suceso B dado el suceso A , es decir $p(B|A)$.

La probabilidad *a posteriori* del suceso A conocido que se verifica el suceso B , es decir $p(A|B)$, puede calcularse a partir de la siguiente fórmula:

$$p(A|B) = \frac{p(A, B)}{p(B)} = \frac{p(A)p(B|A)}{p(B)} = \frac{p(A)p(B|A)}{\sum_A p(A)p(B|A)} \quad (2.1)$$

Tabla 2.1: Método bayesiano.

Método	Clasificador	Función
Bayes	BayesNet	Redes bayesianas de aprendizaje
	Complement NaiveBayes	Construye un complemento de Naive Bayes
	DMNBtext	Clase para la construcción y el uso de un clasificador Naive Bayes discriminativo-Multinomial
	Naive Bayes	Probabilístico estándar de un clasificador bayesiano
	Naive Bayes Multinomial	Versión multinomial de Naive Bayes
	Naive Bayes Multinomial Updateable	Clasificador Naive Bayes incremental, aprende de un caso a la vez
	Naive BayesUpdate	Utiliza una precisión predeterminada de 0.1 para atributos numéricos.

Método de funciones de clasificación (Functions)

La categoría de las funciones de la (ver Tabla 2.1.1) incluye un grupo diverso de clasificadores que se pueden escribir como ecuaciones matemáticas de una manera bastante natural (Witten, Frank, Hall, y Pal, 2016).

Tabla 2.2: Método de funciones de clasificación.

Método	Clasificador	Función
Functions	Logistic	Construir los modelos lineales de regresión logística
	MultilayerPerceptron	Red neuronal propagación hacia atrás
	RBFNetwork	Implementa una red de funciones de base radial
	SimpleLogistic	Construir los modelos lineales de regresión logística con la selección de atributos incorporados
	SMOreg	Implementa un algoritmo de optimización de mínimos de Alex Smola y Bernhard Scholkopf para la formación de un modelo de regresión.

Método perezoso (Lazy)

Los clasificadores de éste método (ver Tabla 2.3) clasifican una nueva instancia, utilizando las k instancias más cercanas a un elemento $\langle a_1, \dots, a_n \rangle$ que deseamos clasificar. Se utiliza la función de distancia euclidiana (Witten y cols., 2016):

$$d(\langle a_1, \dots, a_n \rangle, \langle b_1, \dots, b_n \rangle) = (\sum (a_i - b_i)^2)^{1/2} \quad (2.2)$$

Tabla 2.3: Método de clasificación perezoso.

Método	Clasificador	Función
Lazy	IB1	Vecino más cercano basado en instancia de aprendizaje
	IBk	Clasificador k-vecino más cercano
	Kstar	Vecino más cercano con función de distancia generalizada
	LWL	Clasificador general de aprendizaje a nivel local ponderado

Metaclasificadores

Los metaclasificadores ó algoritmos ensamblados (ver Tabla 2.4 y 2.5), surgen con el objetivo de mejorar la precisión de las predicciones (Witten y cols., 2016) ha surgido un interés creciente en la definición de métodos que combinan varios modelos certificadoros. Se trata, entonces, de combinar las predicciones del conjunto de modelos, normalmente por votación, para clasificar nuevos ejemplos. La precisión obtenida por esta combinación supera, generalmente, la precisión de cada componente individual del conjunto (Witten y cols., 2016).

Tabla 2.4: Meta clasificadores o algoritmos ensamblados parte uno.

Método	Clasificador	Función
Meta	AdaBoostM1	Clase, genera un clasificador nominal utilizando el método AdaBoostM1
	AttributeSelectedClassifier	Reducir la dimensión de datos mediante selección de atributos
	Bagging	Clase para el ensacado de un clasificador para reducir la varianza, trabaja regresión.
	ClassificationViaClustering	Un simple meta clasificador que utiliza una agrupación para la clasificación
	ClasificaciónViaRegresión	Realiza la clasificación usando un método de regresión, binarizando la clase y el edifica un modelo de regresión para cada valor.
	CostSensitiveClassifier	Clasifica o predice la clase con el costo de clasificación erróneo mínimo esperado en lugar de la clase probablemente uno.
	CVParameterSelection	Realiza la selección de parámetros de validación cruzada
	Decorate	Construye conjuntos de clasificadores mediante el uso de ejemplos de entrenamiento artificiales
	FilteredClassifier	Ejecuta un clasificador de los datos filtrados
	IterativeClassifierOptimizer	Optimiza el número de iteraciones realizadas por clasificadores iterativos, usando la validación cruzada en los datos de entrenamiento.
	LogitBoost	Realiza una regresión logística aditiva.
	MultiClassClassifier	Usa un clasificador de dos clases para un conjunto de datos multiclase.

Tabla 2.5: Meta clasificadores o algoritmos ensamblados parte dos.

Método	Clasificador	Función
	MultiScheme	Usa validación cruzada para seleccionar un clasificador de varios candidatos
	OrdinalClassClassifier	Aplica algoritmos de clasificación estándar para problemas con valor de clase ordinal
	RandomCommittee	Crea un conjunto de clasificadores de base aleatoria
	RandomSubSpace	Hace un clasificador de árbol de decisión, donde mantiene mayor precisión en los datos de formación y mejor precisión de la generalización.
	Stacking	Combina varios clasificadores utilizando el método de apilamiento
	Vote	Combina clasificadores utilizando el promedio de las estimaciones de probabilidad.

Método varios (Misc)

En esta categoría (ver Tabla 2.6) se incluyen dos clasificadores simples: Hyperpipes, para problemas de clasificación discreta, y VFI (intervalos de votación característica) que clasifica los casos de prueba mediante el voto (Witten y cols., 2016).

Tabla 2.6: Método varios.

Método	Clasificador	Función
Misc	HyperPipes	Simple, aprende rápido sobre la base de hipervolumenes en el espacio de instancias
	VFI	Característica de votación: método de los intervalos, simple y rápido

Método de generación de reglas (Rules)

En estos clasificadores (ver Tabla 2.7) se descubren patrones de la forma: SI X ENTONCES Y (Witten y cols., 2016).

Tabla 2.7: Método de generación de reglas.

Método	Clasificador	Función
Rules	ConjunctiveRule	Aprendizaje con reglas sencillas conjuntivas
	Desición table	Crea una tabla simple de decisión de la mayoría de los clasificados
	DTNB	Clase para la construcción y el uso de una tabla de con clasificador bayesiano híbrido
	JRip	Usa el algoritmo RIPPER para reglas de inducción rápidas y eficaces
	NNge	Método del vecino más cercano de generación de reglas que utilicen ejemplares no interesados generalizados
	OneR	Clasificador 1R
	PART	Obtiene reglas de los árboles de decisión parcial construido con J4.8
	Ridor	Regla de aprendizaje modulación hacia abajo
	ZeroR	Predecir la clase de la mayoría (<i>si es nominal</i>) o el valor medio (<i>si es numérico</i>)

Método de generación de árboles de decisión (trees)

Un árbol de decisión (Witten y cols., 2016) es una estructura que se compone de nodos (internos y hojas) y de arcos. Sus nodos internos están caracterizados por uno o varios atributos de prueba y de estos nodos se desprenden uno o más arcos. Cada uno de estos arcos tiene asociado un valor del atributo de prueba y estos valores determinan que camino seguir en el recorrido del árbol. Los nodos hoja contienen información que permite determinar la pertenencia del objeto a una clase. Las características principales de un árbol de decisión son: construcción sencilla, no necesita determinar de antemano parámetros para su construcción, puede tratar problemas multiclase de la misma forma en que trabaja con problemas de dos clases, facilidad para ser representado mediante un conjunto de reglas y la fácil interpretación de su estructura (ver Tabla 2.8).

Tabla 2.8: Método de generación de árboles de decisión.

Método	Clasificador	Función
Trees	BFTree	Clase para la construcción de un clasificador de árbol de decisión primero-mejor
	DecisionStump	Construye árboles de decisión de un nivel
	FT	Clase para construir árboles de función, los cuales son árboles de clasificación que tienen funciones de regresión logística en los nodos internos y/o hojas
	JR48	Aprendizaje mediante árbol de decisión C4.5 (C4.5 implementa la revisión 8)
	J48graft	Clase para generar un injerto (podado o sin podar) de un árbol de decisión C4.5
	LADTree	Clase para generar un árbol de decisión multi-clase, utilizando el algoritmo LogitBoost
	LMT	Construye árboles de modelo logístico
	NBTree	Construye un árbol de decisión con clasificadores Naive Bayes en las hojas
	RandomForest	Construye bosques al azar
	RandomTree	Construya un árbol que tiene en cuenta un determinado número de rasgos al azar en cada nodo
	REPTree	Aprendizaje rápido de árboles que usa poda de error reducido
	SimpleCart	Clase que implementa poda de complejidad de costo mínimo

2.2. Deep learning (Aprendizaje profundo)

El término de aprendizaje profundo o deep learning, surge alrededor de los años ochenta, en Japón cuando el investigador Kunihiko Fukushima propuso un modelo neuronal de entre cinco y seis capas, al que denominó neocognitrón. Sin embargo, los problemas para el desarrollo de las alternativas a la propuesta fueron muy complejas y el coste para la investigación era sumamente elevado, lo que hizo que estas técnicas no se hayan vuelto a retomar con fuerza hasta hace una década. Por definición, deep learning es un subconjunto dentro del campo del aprendizaje automático, el cual se base en la idea del aprendizaje deprendizaje basado en ejemplos. Al hablar de deep learning se trata de una familia de algoritmos, con el objetivo de imitar el funcionamiento del cerebro humano al reconocer imágenes, sonidos, palabras, etc. Todo esto es posible gracias al uso de estructuras lógicas que se asemejan en mayor medida a la organización del sistema nervioso de los mamíferos, teniendo capas de unidades de proceso (neuronas artificiales) que se especializan en detectar determinadas características existentes en los objetos percibidos (Witten y cols., 2016).

En el área de visión artificial el uso de deep learning proporciona una mejora considerable en comparación con algoritmos tradicionales (García Navarro, 2015), en esta investigación se utilizan para la interpretación, detección y clasificación de anomalías cancerígenas en el área médica. El término de aprendizaje, en un contexto normal se define como la forma en que las personas aprendemos a realizar cambios adaptivos en el sistema para hacer las mismas tareas de la misma población de una manera más eficiente y efectiva la próxima vez (Roig-Vila y cols., 2016).

Para que una computadora pueda aprender no se le proporciona una lista enorme de reglas para solucionar un problema, se le enseña un modelo que pueda evaluar ejemplos y una pequeña colección de instrucciones para modificar este modelo cuando se produzcan errores (Witten y cols., 2016), con el tiempo se espera que esos modelos sean capaces de solucionar el problema de forma extremadamente precisa, gracias a que el sistema es capaz de extraer patrones (García Navarro, 2015). Aunque existen distintas técnicas para implementar deep learning, una de las más comunes es simular un sistema de redes artificiales de neuronas dentro del software de análisis de datos, esta generalización es un indicador de lo bien que funciona el algoritmo, se puede ver midiendo tanto el error de entrenamiento como el error de prueba (o error de generalización) y siempre se busca disminuirlos al máximo (Lévy y Jain, 2016). Como su propio nombre indica, el error de entrenamiento se mide en el conjunto de entrenamiento, y el error de prueba se mide en el conjunto de prueba (Hall y cols., 2009). Para hacer la división de los datos, se suele seguir una estrategia de generación de datos, en la cual se asume que los datos de cada conjunto son independientes y que están distribuidos de forma idéntica (Hall y cols., 2009), las principales líneas de trabajo en modelos de deep learning son las siguientes:

- Diseño de algoritmos de pre procesamiento de datos, señales, imágenes para deep learning, analizando el comportamiento de los procesos de aumento de datos (dataset).
- Diseño de modelos de ensamblamiento de algoritmos de deep learning.
- Análisis de la robustez de los algoritmos de deep learning, prestando atención a los problemas de ruido de clase asociados a los errores de equitación de las clases.
- Empleo de los algoritmos de deep learning en diferentes ámbitos aplicados: análisis de datos de alzheimer, detección de imágenes en vídeos (detección de armas), análisis de imágenes médicas, etc.

2.2.1. Redes neuronales artificiales

Una red neuronal es un sistema inteligente para el aprendizaje a partir de una base de datos de ejemplos de un problema para obtener el modelo asociado a estos (Gallegos-Funes y cols., 2009). Al igual que las neuronas biológicas del cerebro humano, las neuronas artificiales se enlazan para formar redes de neuronas artificiales, cada neurona artificial utiliza

una función de procesamiento que agrega la información de conexiones de entrada con otras neuronales artificiales, estas neuronas tienen una *función de activación* y una *función de transferencia* para dar a la neurona sus conexiones de salida. Para observar las redes neuronales artificiales se recomienda ver (Figura. 2.2.), donde se define que una red neuronal es en realidad una composición de perceptrones conectados (el perceptrón es la unidad básica de una red neuronal) de diferentes maneras, existen diferentes criterios de clasificación para estas redes (Gallegos-Funes y cols., 2009; Krizhevsky y cols., 2012)

- Caracterización temporal: se refiere a la caracterización temporal de las entradas.
- Continua en el tiempo: cuando la función de entrada es una función continua, este tipo de redes se utiliza en sistemas donde es necesario una clasificación inmediata de las entradas, como por ejemplo un sistema de alarma que se active ante la presencia de ciertos parámetros peligrosos. .
- Discreta: cuando las entradas se toman en determinados instantes de tiempo. Se utilizan en sistemas donde no es necesario un control inmediato de las entradas y sólo se busca ver su evolución sin saturar demasiado el sistema con una gran cantidad de datos .
- Entrada: se refiere a los valores que toman las entradas.
- Binaria: cero o uno. Se puede utilizar cuando las entradas son codificadas, por ejemplo, representando el intervalo sobre el que caen dentro de un rango. Si el rango se divide en x intervalos, cada entrada se representa con x valores, todos a cero menos uno, el que corresponde al intervalo de valores de la entrada.
- Continua: el valor de la entrada puede ser cualquiera dentro de un rango, es posible definir diferentes rangos de variación.
- Entrenamiento: se refiere al aprendizaje seguido por la red.
- Supervisado: cuando se vigila la evolución de la red, por ejemplo, cuando se le ofrecen a la red ejemplos señalando las salidas que se deberían obtener.
- Con recompensa/castigo: cuando la red acierta se le ofrece una recompensa, cuando falla se le castiga, así aprende cómo debe comportarse .
- No supervisado: no se le dice a la red lo que debe dar, esto se suele utilizar en aquellos casos en que tenemos una serie de entradas y no sabemos cómo clasificarlas, dependiendo de lo que se obtenga tras el entrenamiento tendremos un criterio de clasificación

Al utilizar deep learning también se utiliza una red neuronal artificial, esta se compone de un número de niveles jerárquicos, en el nivel inicial de la jerarquía la red aprende algo simple y luego envía esta información al siguiente nivel. El siguiente nivel toma esta información sencilla, la combina, compone una información algo un poco más compleja, y

se lo pasa al tercer nivel, y así sucesivamente (Krizhevsky y cols., 2012). Deep learning fue diseñado para trabajar con metadatos, almacenados en un dataset, permitiendo que estos datos puedan ser clasificados en base a patrones concurrentes (características similares entre ellos) (Xu y cols., 2014). El ámbito donde mayormente se aplica deep learning es en el análisis de imágenes médicas, dentro de este paradigma destacan modelos avanzados de redes con un número grande de capas de neuronas y con algoritmos muy sofisticados de procesamiento/optimización, dentro de los tipos de redes se pueden distinguir (Ver Figura. 2.1).

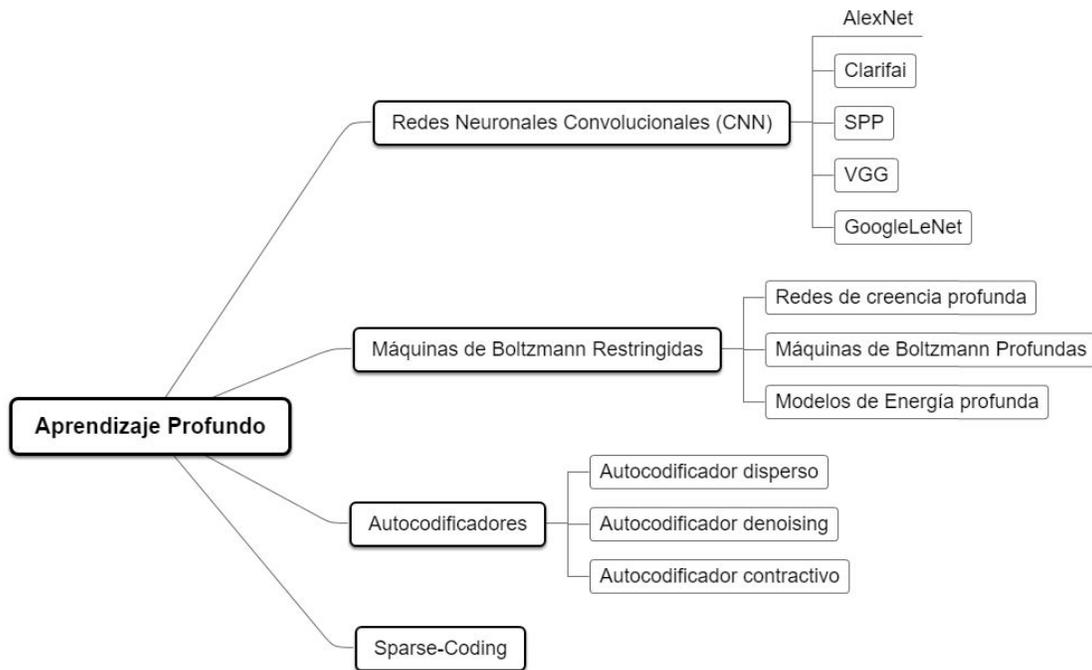


Figura 2.1: Redes en deep learning, basado en el libro *Advances in knowledge discovery and data mining* (Fayyad y cols., 1996).

Las redes neuronales no son la única forma para implementar deep learning, pero si es la más común como se muestra en la (Figura. 2.2)

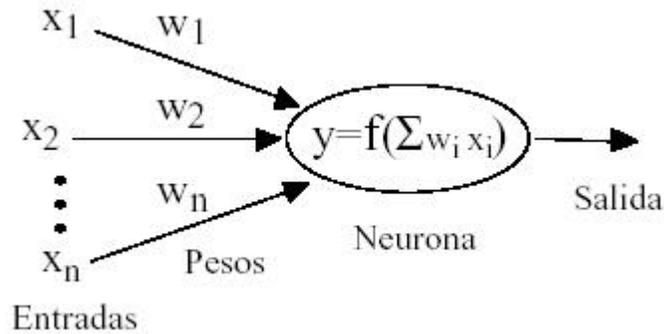


Figura 2.2: Descripción gráfica de una red neuronal artificial (López y Fernández, 2008).

2.3. Métricas de evaluación

Matriz de confusión. Es una herramienta estándar de evaluación de modelos estadísticos que ordena todos los casos del modelo en categorías, determinando si el valor de predicción coincide con el valor real. Se crea un gráfico comparando los valores reales con los valores de predicción para cada estado de predicción específico (msdn.microsoft, 2016).

Las filas de la matriz representan los valores de predicción para el modelo, mientras que las columnas representan los valores reales. Es una herramienta importante para evaluar los resultados de predicción, ya que hace que resulte fácil entender y explicar los efectos de las predicciones erróneas. Al ver la cantidad y los porcentajes en cada celda de la matriz, podrá saber en cuantas ocasiones ha sido exacta la predicción del modelo (msdn.microsoft, 2016). Dos medidas tradicionales que miden el valor diagnóstico de una prueba son las que a continuación se presentan:

2.4. Imágenes médicas

Una imagen médica se define como un conjunto de técnicas y procesos usados para crear imágenes del cuerpo humano ó partes de él con propósitos clínicos, esto es, procedimientos médicos que buscan revelar, diagnosticar o examinar enfermedades, o bien con propósitos científicos médicos, tales como el estudio de la anatomía física y metabólica (Lorca, Arzola, y Pereira, 2010). La disciplina médica más relacionada con la imagen médica es la Radiología, aunque también podemos destacar otras técnicas que no proceden implícitamente de la radiología, tales como la endoscopia, la fotografía médica y la microscopía.

2.4.1. Mastografía digital

Es la radiografía proyectiva correspondiente a las mamas y otros tejidos blandos, su uso fundamental es el diagnóstico del cáncer de mama, aunque puede tener otros usos, como la localización de tejidos sospechosos antes de realizar una biopsia (Lorca y cols., 2010). También se puede definir como un estudio de rayos “X” que se recomienda a mujeres de 40 a 69 años de edad, sin signos, ni síntomas de cáncer (asintomáticas) y tiene como propósito detectar anomalías en las mamas, que no se pueden percibir por la observación o la palpación.(IMSS, 1994). Pueden descubrir pequeños nódulos o masas que no los puede sentir una persona. Si se descubre alguna anomalía mediante la mastografía se requerirán estudios adicionales para descartar o confirmar el diagnóstico. En caso de que el resultado sea normal, se citará a una nueva mastografía en dos años (IMSS, 1994).

El estudio de mastografía 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. No todos los resultados anormales significan cáncer, *de cada 10 mujeres con mastografía anormal, sólo una tiene cáncer*. 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 (IMSS, 1994).

2.4.2. Ruido en imágenes

Cuando obtenemos una imagen digital y se ve un poco distorsionada, decimos que la imagen tiene ruido, según su definición, ruido es cualquier perturbación que sufre una señal en el proceso de adquisición, transmisión o almacenamiento. El ruido se modela usualmente como un proceso estocástico (un modelo probabilístico), se manifestará generalmente en píxeles aislados que toman valores distintos de los “reales”.

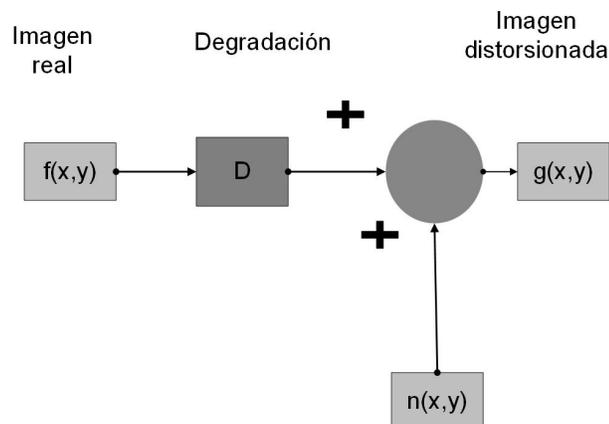


Figura 2.3: Representación del ruido en una imagen (elaboración propia).

2.5. Estado del Arte

En esta sección se describen 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 el problema que se quiere resolver, algunos son sobre predicción, clasificación, detección entre otras. La presente sección se encuentra ordenada de menor a mayor año:

(DelaCruz y cols., 2011), realizaron un sistema que consta de varias etapas, como primer paso, hacen que la imagen sea filtrada, posteriormente segmenta la zona de interés dada por el especialista médico. El algoritmo de clasificación utilizado es LogitBoost, con este método se obtienen 95 % de instancias correctamente clasificadas.

(Arafi, Fajr, y Bouroumi, 2014), implementaron un método para la detección de cáncer basado en Support Vector Machine (por sus siglas en inglés SVM), como técnica de aprendizaje supervisado para clasificar datos empíricos. Así se optimiza el rendimiento del clasificador resultante, de esta forma se tiene un resultado de 94.74 %

(Carreras-Cruz, de Lourdes Martínez-Villaseñor, y Rosas-Pérez, 2015), para atender la problemática de clasificar anormalidades cancerígenas, utilizaron el dataset (MIAS, 1994), en al 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.

(Neto, Carvalho, Sampaio, Corrêa, y Paiva, 2015), para automatizar la segmentación de masas en mamografías, utilizaron la optimización de enjambre de partículas (PSO) y graph clusters, logrando un 95.2 % de efectividad.

(Arevalo, González, Ramos-Pollán, Oliveira, y Lopez, 2016), utilizaron un enfoque híbrido donde las redes neuronales convolucionales se utilizan para aprender la representación de forma supervisada, obteniendo un porcentaje de 82 %.

(Lévy y Jain, 2016), implementaron un modelo integral de aprendizaje profundo para clasificar las masas mamarias pre-detectadas de mamografías, donde se utilizó la arquitectura AlexNet y GoogLeNet, obteniendo con la última el mayor porcentaje exactitud de 92.9 %.

(Gerazov y Conceicao, 2017), aplicaron métodos de aprendizaje profundo a un conjunto de datos de dominio de tiempo en mama homogénea del tejido adiposo. Aquí emplearon redes neuronales convolucionales, así como el clasificador de entrada SVM para una precisión de 93.44 %.

(Al-Masni y cols., 2017), el sistema CAD ² que se realizó para la detección de masas de

²Computer Aided Design (CAD) en español: Diseño Asistido por Ordenador

seno y clasificación de cáncer, implementó una CNN (Convolutional Neural Network), 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.

(Pedraza y cols., s.f.), entrenaron 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 proporcionó una precisión del 95.62 % para un conjunto de 5750 instancias.

(Dalmaz y cols., 2016), 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.

(Moradkhani y Bigam, 2017), la investigación aborda la problemática de detectar micro calcificaciones en mastografías del dataset de dominio público MIAS, aplicaron a las imágenes el Histograma Adaptativo Limitado por Contraste (CLAHE) por sus siglas en inglés, para la clasificación de las características de la imagen se hizo con la función de base radial SVM con un resultado de 97.35 % para clasificar correctamente las mastografías.

Capítulo 3

Materiales y Métodos

Este capítulo se divide en dos secciones: materiales y métodos, la sección de materiales describe el dataset así como el equipo necesario para resolver el problema de esta investigación; en la sección métodos se describen los procesos a los que se sometió las herramientas, todo ello con el fin de encontrar una solución factible para el objetivo de la investigación.

3.1. Materiales

La investigación se centró en la categorización de anomalías cancerígenas, en la cual se tuvo la necesidad de utilizar imágenes de mastografías en formato digital, como no fue posible obtener dichas imágenes de la región de Misantla, se optó por buscar en internet algún dataset que fuera de dominio público, varios son los repositorios que se encontraron durante la búsqueda. Se hizo énfasis en aquellos que anteriormente han sido utilizados en investigaciones, como lo muestra la Tabla 3.1, varios de ellos de acceso público; finalmente se escogió el dataset (MIAS, 1994), las razones fueron que es de dominio público, ha sido utilizado en varias investigaciones en el estado del arte, el tamaño de las imágenes contenidas en el dataset es aceptable comparado a otros. En la Tabla 3.1 se tomó del trabajo de tesis de (DelaCruz y cols., 2011), en el cual se puede apreciar una pequeña lista de los que se encuentran en la red:

Tabla 3.1: Tabla de datasets disponibles en internet.

Nombre	Descripción
DDSM	Digital Database for Screening mammography. Base de datos creada por el Hospital General de Massachusetts, la Universidad de Florida y Sandia National Laboratories. Fuente: http://marathon.csee.usf.edu/Mammography/Database.html
MIAS	Base de datos creada por la Mammographic Image Analysis Society, Reino Unido. Fuente: http://www.wiau.man.ac.uk/services/MIAS/MIASweb.html Existe una versión reducida que contiene las mismas imágenes que la versión original, pero reducidas a un tamaño de 1024 x 1024 pixeles, disponible en http://peipa.essex.ac.uk/info/mias.html
LLNL/UCSF	Base de datos creada por Lawrence Livermore National Laboratories (LLNL) y el Departamento de Radiología de la Universidad de California en San Francisco (UCSF). Fuente: mammo-db-help@llnl.gov
Nijmegen	Base de datos creada por el National Expert and Training Centre for Breast Cancer Screening y por el Departamento de Radiología de la Universidad de Nijmegen (Holanda). Fuente: nico@mbfys.kun.nl

El dataset MIAS, cuenta con 322 imágenes que contiene información acerca de anorma-

lidad presente en la imagen, como lo es:

- CALC (Calcificación)
- CIRC (Masas bien definidas / circunscritas)
- SPIC (Masas Spic)
- MISC (Otras masas mal definidas)
- ARCH (Distorsión arquitectónica)
- Asimetría (Asimétrica)
- NORM (Normal)

En el tipo de anormalidad se pueden distinguir tres tipos:

- B Benigno
- M Maligno
- N Normal

En la información obtenida, se encuentran las coordenadas x y y de cada imagen, además la longitud del radio que encierra la anormalidad de cada una de las imágenes. El tamaño de todas las imágenes es de 1024x1024 píxeles (ver Figura 3.1) . La lista está organizada en pares de películas, donde cada par representa los números de nombre izquierdo (incluso nombres de archivo) y mamografías correctas (números de nombre de archivo impares) de un solo paciente. El tamaño de todas las imágenes es de 1024 píxeles x 1024 píxeles, las imágenes se han centrado en la matriz, cuando hay calcificaciones, las ubicaciones centrales y los radios se aplican a los grupos en lugar de las calcificaciones individuales. El origen del sistema de coordenadas es la esquina inferior izquierda (MIAS, 1994), en algunos casos, las calcificaciones se distribuyen ampliamente en toda la imagen en lugar de concentrarse en un único sitio. En estos casos, las ubicaciones y los radios del centro son inapropiados y se han omitido (MIAS, 1994).

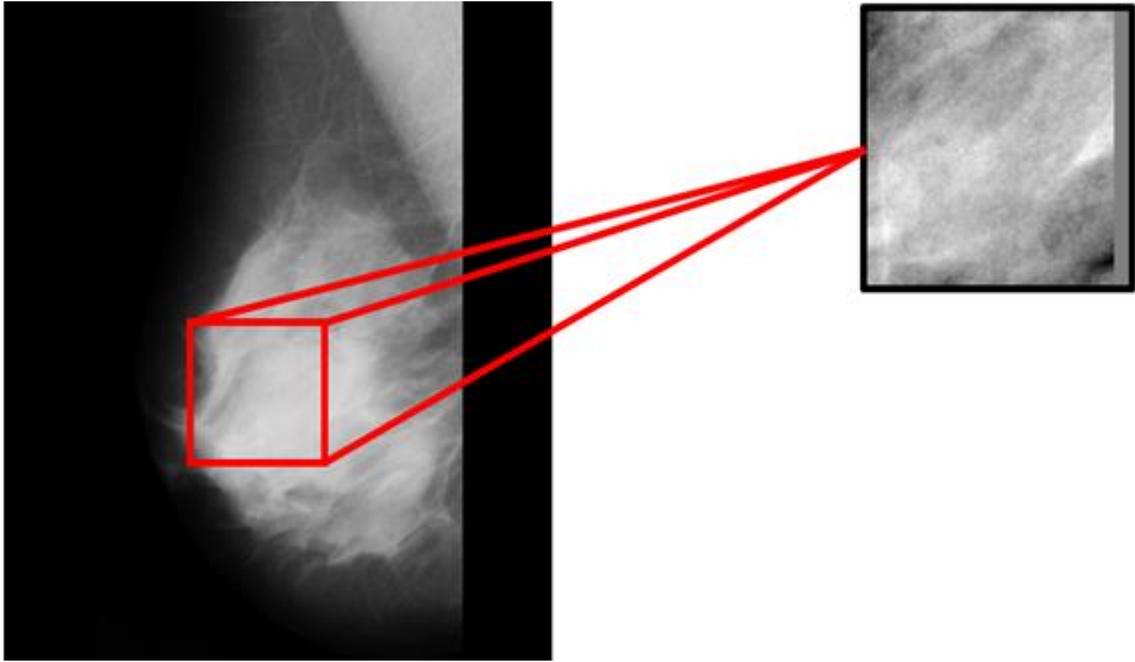


Figura 3.1: Imagen de 1024x1024 (Izq.) imagen recortada (Der.) a un tamaño de 120x120 pixeles

3.2. Método

Esta sección se divide en dos partes para una mejor comprensión: en el primero se explica el procesamiento de las imágenes 3.2.1 y en el segundo se muestra el algoritmo utilizado para solucionar la problemática 3.2.2.

3.2.1. Preprocesamiento de las imágenes

El tamaño de cada imagen se redujo a 120x120, (ver Figura 3.1) este criterio se aplicó al contemplar que no se pierdan las propiedades de la imagen y no se escapen anomalías que pueden ser excluidas. Al reducir las imágenes a el tamaño de 120x120, se obtiene como resultado una imagen con 14400 píxeles, mismos que son almacenado es un dataset csv, en forma de un vector, este dataset se prueba en la herramienta de minería de datos WEKA, con el algoritmo de deep learning DI4MlpClassifier; este algoritmo se combinó con todos los algoritmos ensamblados disponibles en WEKA. Contiene 322 imágenes clasificadas en Benigno (B) y Maligno (M) y Normal (N) están previamente clasificadas por un experto médico.

Todas las imágenes se digitalizan con una resolución de 1024x1024 píxeles y precisión de ocho bits (nivel de gris), también incluyen las coordenadas de imagen (x, y) de centro de anomalía (como el centro de un círculo que rodea el tumor), radio aproximado de

un círculo encierra la anormalidad, posición del pecho (izquierda o derecha), tipo de tejido mamario (graso, graso-glandular y denso) y tipo de tumor si existe (Benigno o Maligno). Cada imagen tiene dimensión de 1024x1024, ocupando la mayor parte de la imagen un fondo negro que genera ruido, dado que cada imagen se encuentra clasificada de acuerdo al criterio del especialista médico, se optó por recortar la imagen de acuerdo a las coordenadas donde se encuentra la anormalidad presente, en el caso de las imágenes clasificadas como normales puede tomar cualquier coordenada.

El tamaño de cada imagen fue de 120x120 considerando que tal tamaño conserva las anomalías detectadas, también no se pierden las propiedades de la imagen. Con el total de 117 imágenes utilizadas para realizar la investigación.

3.2.2. Algoritmos de aprendizaje profundo

En esta sección se describen los dos algoritmos con los cuales se obtuvieron mayor porcentaje de instancias correctamente clasificadas, y el funcionamiento de cada uno.

AttributeSelectedClassifier, este algoritmo ensamblado se combinó con el algoritmo de aprendizaje profundo Multilayer perceptron, que en la herramienta WEKA se nombra DI4JMLp (at University of Waikato, 2016), esta fue la combinación que mayor porcentaje de clasificación: **83.7338 %** de instancias correctamente clasificadas. A continuación se describe el algoritmo 1 **AttributeSelectedClassifier**:

Algoritmo 1 Attribute Selected Classifier (at University of Waikato, 2016).

Entrada: Conjunto de entrenamiento x ;

Método de selección de atributos β ;

Algoritmo de clasificación α

1: n : número de atributos

2: **mientras** $|X|$ **hacer**

3: Seleccionar un subconjunto i de características con método β

4: $r < -$ Clasificar sólo i características utilizando clasificador α

5: $i < -$ seleccionar i mejores anteriores con mejor predicción r

6: **fin mientras**

7: Salida: clasificación con los mejores atributos

Multilayer perceptrón. También conocido como perceptrón multicapa es una red neuronal artificial conformada por múltiples capas, esto le permite resolver problemas que no son linealmente separables, lo cual es la principal limitación del perceptrón (Tang, Deng, y Huang, 2016), el multilayer perceptrón puede ser totalmente o localmente conectado. En el primer caso cada salida de una neurona de la capa " i " es entrada de todas las neuronas de la capa " $i + 1$ ", mientras que en el segundo cada neurona de la capa " i " es entrada de una serie de neuronas (región) de la capa " $i + 1$ ". Los algoritmos basados en deep learning, contienen

muchas neuronas simples interconectadas entre sí (ver Figura 3.2) todas las redes neuronales artificiales reciben como información en conjuntos de entrada (Tang y cols., 2016).

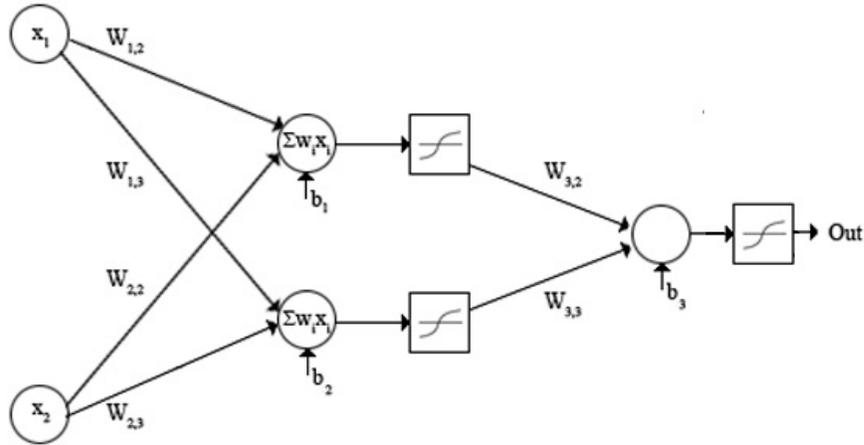


Figura 3.2: Red neuronal multilayer perceptrón (López y Fernandez, 2008).

El algoritmo multilayer perceptrón y su funcionamiento se muestra (ver Algoritmo 2) donde se aprecia de manera clara los pasos que se siguen para el entrenamiento de una red de este tipo:

Algoritmo 2 Multilayer perceptron (Witten y cols., 2016)

Entrada: x conjunto de entrenamiento

Entrada: $|X|$ Número de instancias en el conjunto de entrenamiento

Salida: cierto Inicializar pesos y umbral aleatoriamente.

1: **mientras** $|X|$ **hacer**

2: $p < -$ Producto punto de X con w

3: $sum < -$ Sumatoria de todos los pesos

4: Salida $< -$ Función de activación (sum)

5: **si** Salida $>$ a umbral **entonces**

6: **devolver**

Salida:

7: **si no**

8: Return 0

9: Actualizar pesos de acuerdo a :

10: $w_j(t + 1) = w_j(t)\eta(d - y)x$

11: Medir el rendimiento con:

12: $E = \frac{1}{2} \sum_{i=1}^p ||y^{(i)} - d^{(i)}||^2$

13: **fin si**

14: **fin mientras**

15: **devolver cierto**

Capítulo 4

Experimentos y resultados

La parte medular de la investigación son los resultados, para llegar a ellos se implementaron los métodos previamente definidos; estos permiten realizar una acción en particular y, luego, se observan y analizan los resultados con la intención de medir los efectos provocados. Este capítulo se divide en tres partes: en la sección de experimentos se encuentran los métodos implementados para dar solución a la investigación; la sección pre procesamiento explica los procesos a los cuales se sometieron las imágenes y, por último, la clasificación donde se da paso a los resultados obtenidos para analizarlos.

4.1. Requerimientos

Para llevar la realización de esta presente investigación, fue necesario reunir material de cómputo con características particulares que serán descritas a continuación.

Hardware

El equipo de cómputo utilizado para la realización de la investigación tiene las siguientes características:

- La computadora utilizada tiene una capacidad de 32 Gb de memoria RAM.
- Cuenta con un procesador de séptima generación (i7).
- Disco duro con una capacidad de almacenamiento de 2 Tb.
- Tiene una tarjeta de video Nvidia de GeForce
- Sistema operativo Windows 10 de 64 bits.

Software

A continuación se enlistan los programas que se utilizaron para este proyecto:

- La instalación de un visualizador de imágenes de extensión .pgm ¹ XnView, no todos permiten ver este tipo de imágenes.
- Software Microsoft Excel, para la visualización de los datasets creados, es una aplicación de hojas de cálculo que forma parte de la suite de oficina Microsoft Office, es utilizada en tareas financieras y contables, con fórmulas, gráficos y un lenguaje de programación.
- Instalación del programa de minería WEKA 3.9, es una plataforma de software para el aprendizaje automático y la minería de datos escrito en Java y desarrollado en la Universidad de Waikato, es software libre distribuido bajo la licencia GNU-GPL (at University of Waikato, 2016).

¹Portable Gray Map (PGM)

Herramienta de minería de datos WEKA

Los experimentos para la clasificación se realizaron en WEKA, esta contiene una colección de algoritmos para realizar análisis de datos y modelado predictivo, en particular, provee acceso a todas las técnicas de clasificación y regresión de estos datos, además provee una interfaz gráfica que unifica las herramientas para que estén a una mejor disposición (at University of Waikato, 2016).

4.2. Pre procesamiento de las imágenes

El tamaño original de cada una de las imágenes era de 1024x1024 pixeles, la matriz de cada una de 1,048,576 pixeles. Las imágenes, todas sin excepción, tienen un fondo negro que es de gran tamaño. Como el dataset utilizado (MIAS, 1994) cuenta con las dimensiones y el radio donde se encuentra la anomalía de cada imagen, se hizo un recorte de imagen para todas las contenidas en el dataset, garantizando que no se perdieran las propiedades de la imagen; el tamaño se redujo a 120x120, así se obtuvo una matriz de 14400 pixeles por cada imagen; a la matriz resultante se le aplicó un aplanamiento para obtener vectores y concatenar un vector sobre otro (ver Figura 4.1). La concatenación se refiere a unir un vector con otro vector, de tal forma que donde termina un vector se entrelaza con el siguiente.

02, 200, 205, 206, 204, 206, 206, 207, 209, 209, 211, 214, 216, 218, 221, 224, 229, 230, 232,
 30, 231, 233, 236, 235, 236, 236, 235, 234, 235, 235, 236, 235, 234, 232, 233, 233, 233, 232,
 32, 234, 231, 230, 232, 234, 232, 231, 229, 230, 229, 230, 230, 233, 238, 243, 245, 243, 243,
 35, 241, 243, 243, 245, 245, 247, 251, 250, 251, 250, 248, 250, 251, 251, 250, 248, 248, 243,
 36, 236, 234, 233, 230, 225, 224, 222, 217, 213, 215, 213, 205, 198, 144, 0, 0, 0, 0, 0, 189,
 98, 202, 202, 205, 204, 205, 206, 208, 207, 207, 206, 205, 208, 208, 211, 215, 214, 220, 221,
 31, 232, 234, 233, 234, 233, 232, 234, 234, 234, 234, 233, 233, 232, 231, 232, 231, 230, 229,
 32, 230, 229, 231, 230, 231, 229, 227, 230, 230, 231, 233, 236, 238, 234, 239, 238, 239, 238,
 45, 247, 247, 247, 245, 245, 245, 245, 242, 245, 245, 245, 247, 243, 243, 241, 240, 239, 236,
 35, 236, 234, 233, 231, 227, 222, 224, 224, 220, 217, 214, 208, 203, 195, 190, 188, 185, 187,
 , 0, 0, 0, 0, 198, 198, 200, 203, 208, 208, 211, 211, 214, 211, 213, 217, 222, 223, 222, 223,
 14, 213, 208, 214, 218, 221, 222, 223, 223, 225, 225, 227, 229, 230, 227, 226, 226, 227, 226,
 21, 220, 220, 218, 217, 216, 218, 221, 225, 225, 227, 231, 229, 224, 217, 220, 218, 227, 230,
 40, 240, 239, 243, 243, 243, 247, 245, 243, 243, 243, 242, 242, 241, 241, 240, 240, 235, 233,
 20, 220, 220, 222, 224, 224, 220, 217, 221, 220, 221, 216, 215, 217, 218, 214, 206, 206, 202,
 84, 181, 181, 185, 185, 187, 140, 0, 0, 0, 0, 0, 199, 200, 203, 203, 200, 202, 198, 200, 202,
 05, 205, 202, 206, 204, 204, 207, 211, 218, 217, 214, 217, 215, 218, 216, 217, 215, 213, 211,
 15, 215, 216, 215, 213, 214, 216, 216, 223, 224, 227, 226, 227, 227, 229, 229, 226, 229, 229,
 23, 223, 226, 226, 229, 232, 230, 231, 232, 233, 235, 232, 234, 234, 233, 232, 226, 224, 230,
 22, 222, 217, 211, 211, 215, 211, 211, 214, 213, 215, 221, 222, 218, 214, 207, 204, 202, 197,
 93, 192, 187, 187, 188, 185, 184, 184, 186, 188, 193, 194, 194, 143, 0, 0, 0, 0, 0, 179, 181,
 89, 190, 189, 187, 186, 186, 187, 185, 186, 189, 192, 195, 199, 204, 203, 200, 200, 205, 205,
 07, 207, 209, 207, 211, 208, 211, 208, 211, 213, 207, 206, 209, 211, 207, 207, 207, 205, 205,
 04, 202, 202, 206, 207, 211, 208, 209, 213, 213, 211, 213, 209, 207, 203, 202, 200, 198, 195,
 02, 200, 198, 195, 195, 188, 187, 187, 185, 185, 179, 179, 177, 183, 190, 193, 190, 185, 183,
 92, 190, 187, 185, 184, 181, 183, 179, 185, 188, 188, 192, 189, 192, 190, 188, 189, 187, 190,
 , 0, 0, 0, M
 189, 179, 192, 201, 195, 186, 174, 162, 168, 165, 179, 179, 179, 179, 179, 174, 174, 183, 195

Figura 4.1: Fragmento del dataset Multiclase {Normal, Benigno, Maligno} almacenado en un archivo en formato csv

4.3. Clasificación

Con el nuevo dataset (120x120 pixeles) se procedió a combinar los algoritmos ensamblados y el algoritmo de deep learning D14JMLp. Los experimentos de cada dataset tienen como parámetros: número de iteraciones diez y el número de semilla es uno. Los resultados mostrados (ver Tabla 4.3) son los registros de la combinación de algoritmos ensamblados con el algoritmo de deep learning D14JMLp que mayor porcentaje de instancias correctamente clasificadas obtuvieron de cada dataset, el resto de los resultados se pueden apreciar en el Anexo sección 2.2 .

Tabla 4.1: Tabla de la combinación de algoritmos ensamblados y el algoritmo D14JMLp, donde se muestra la combinación que obtuvo mayor clasificación de cada dataset

<i>Dataset</i>	<i>Algoritmo ensamblado</i>	<i>D14JMLp</i>	<i>Muestra representativa</i>	<i>2/3 1/3</i>	<i>Cross folder</i>
{N,M}	Attribute Selected Classifier	D14JMLp	63.9535	83.7338	80.9091
{N,B}	Randomizable Filtered Classifier	D14JMLp	74.7368	83.3333	78.4
{B,M}	Random Committee	D14JMLp	55.5556	69.0476	56.4103
{N,B,M}	MultiClass Classifier	D14JMLp	49.5868	62.5	56.25

Una vez que se obtuvieron los mejores resultados de cada combinación de algoritmos ensamblados con el algoritmo D14JMLp que mayor porcentaje de instancias correctamente clasificadas obtuvieron: *AttributeSelectedClassifier* con el dataset binario {Normal, Maligno}, *RandomizableFilteredClassifier* con el dataset binario {Normal, Benigno}, *RandomCommittee* con el dataset binario {Benigno, Maligno} y *MultiClassClassifier* para el dataset Multi-clase {Normal, Benigno, Maligno}. Para tener certeza que los resultados obtenidos son los de mayor precisión se modificaron los parámetros de las semillas e iteraciones de cada combinación de algoritmos, el comportamiento de estas modificaciones se registró por medio de gráficas, donde el eje de las abscisas representa el intervalo de iteraciones ó semillas (según sea el caso) y en el eje de las ordenadas el porcentaje de instancias correctamente clasificadas por la combinación de algoritmos.

La Figura 4.2 es la gráfica que se obtuvo al registrar las variaciones de el número de iteraciones del dataset binario {Normal, Benigno}, esto se hizo con el fin de saber si al variar las iteraciones fuese un factor que influyera a obtener un porcentaje mayor de categorización, como se aprecia en la gráfica no hubo ningún porcentaje que superara al mayor registrado de 83.3333 %.

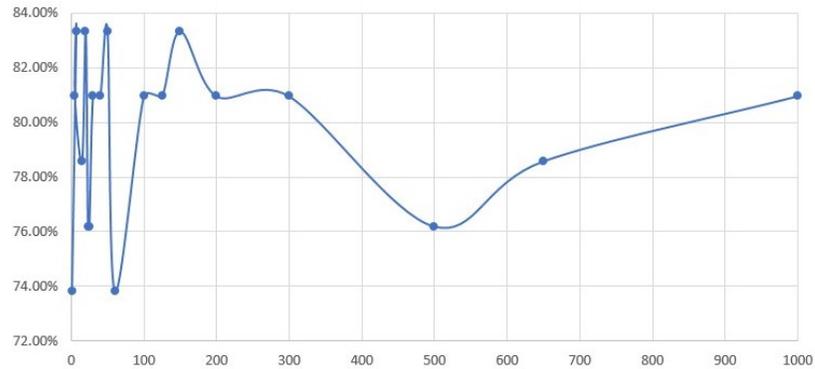


Figura 4.2: Gráfica del dataset {Normal, Benigno} iteraciones.

La Figura 4.3 muestra como al variar las iteraciones aumentaba el porcentaje hasta que con 500 épocas disminuye, aunque no supera al porcentaje anterior de 83.7338 %.

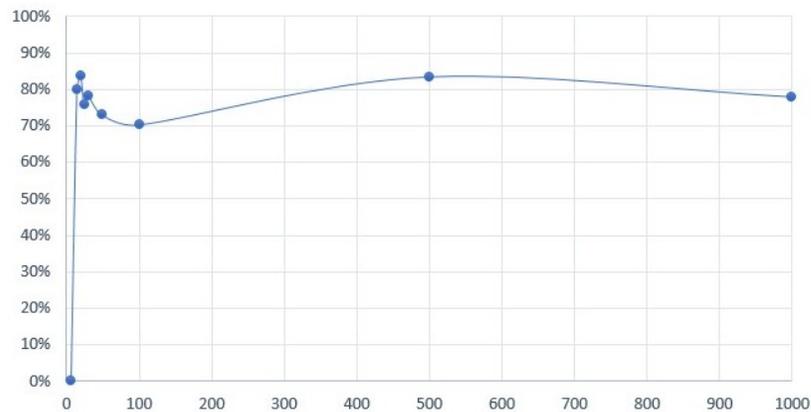


Figura 4.3: Gráfica del dataset {Normal, Maligno} iteraciones.

Al variar la iteracion a cinco e ir aumentado el número de ellas se observa como el porcentaje se mantiene igual hasta las 2000 épocas, como se aprecia en la Figura 4.4 donde no supera al 69.0476 %.

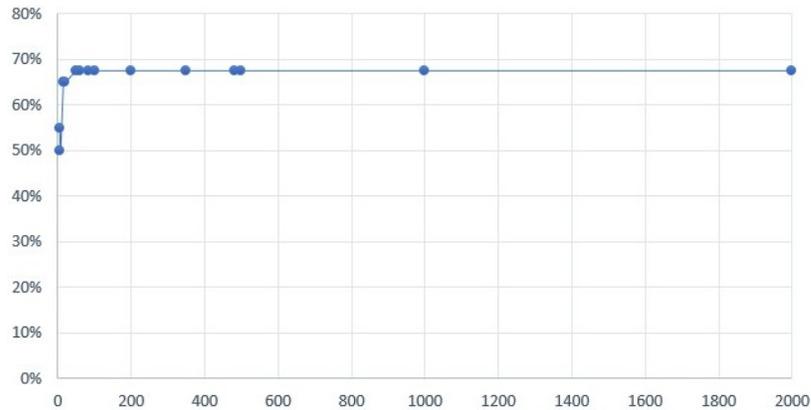


Figura 4.4: Gráfica del dataset {Benigno, Maligno} iteraciones.

Cuando se inicia en tres iteraciones hasta 15 mantiene un porcentaje de 47% y al cambiar de 50 hasta 1000 iteraciones el porcentaje de mantuvo Figura 4.5 sin llegar a superar el 62.5% que era el objetivo.

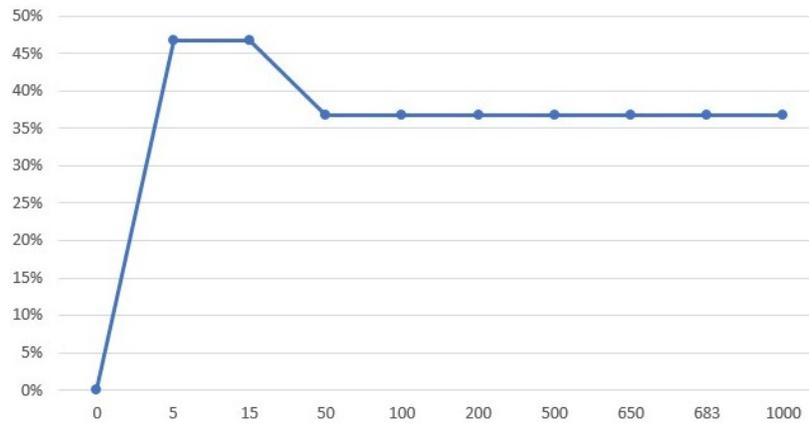


Figura 4.5: Gráfica del dataset Multiclasa {Normal, Benigno, Maligno} iteraciones.

Aparte de variar las iteraciones, se procedió a hacerlo de forma separada con las semillas tomando como número diez iteraciones y cinco semillas donde se obtuvo un porcentaje de 82.9% se me mantuvo hasta las 2000 semillas Ver Figura 4.8.

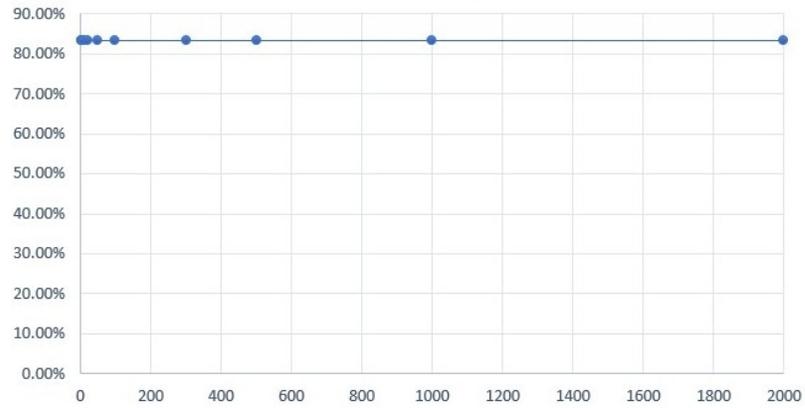


Figura 4.6: Gráfica del dataset {Normal, Benigno} semillas.

Cuando el número de la semilla fue 2 el porcentaje fue de de 78, al aumentar la semilla a 100 disminuyó el porcentaje para volver aumentar con 200, en este caso tampoco supera al 83.7338% Ver Figura 4.7.

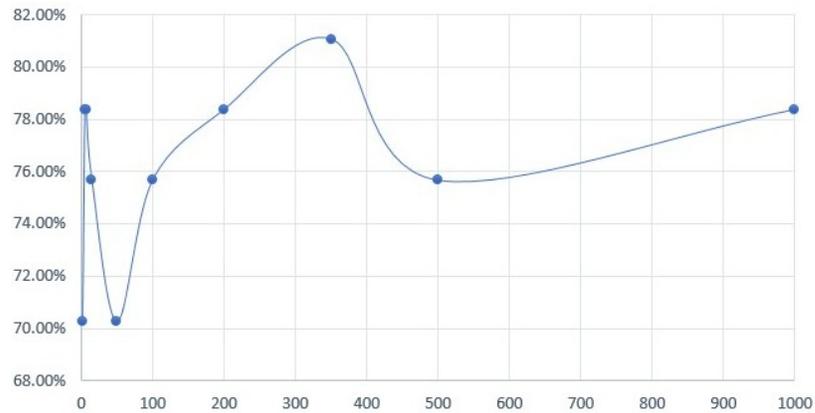


Figura 4.7: Gráfica del dataset {Normal, Maligno} semillas.

Cambiar la semilla a dos arrojó un porcentaje de 62% que se mantuvo hasta la semilla 1000 sin lograr superar al anterior porcentaje Ver Figura 4.8.

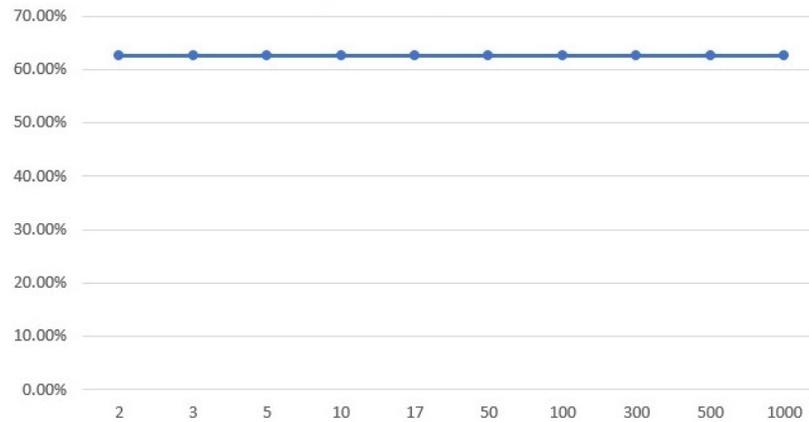


Figura 4.8: Gráfica del dataset {Benigno, Maligno} semillas.

La semilla con la que se inicia es dos, con esta se obtuvo 61.5 % de clasificación correctas, y varía el porcentaje aumentándolo hasta que llega a 100 semillas y disminuye a 40 % para aumentar este conforme se eleva el número de semillas y vuelve a disminuir cuando se llega a 1000 semillas con las que se obtiene 60 % Ver Figura 4.9.

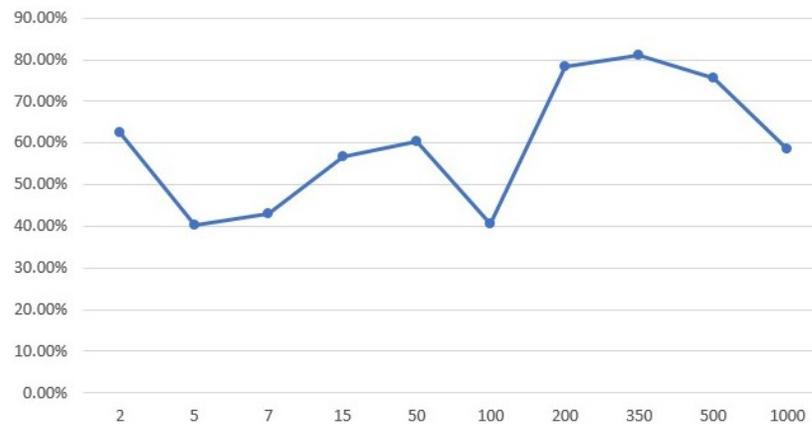


Figura 4.9: Gráfica del dataset Multiclase {Normal, Benigno, Maligno} semillas.

Como se pudo apreciar, la configuración de cada uno de los datasets con el número de diez iteraciones y una semilla es la mejor al quedar demostrado que aumentar las iteraciones y semillas no logran superar los mejores porcentajes registrados (ver Tabla 4.3), por lo que se quedan los datasets con esos parámetros.

Análisis de las matrices de confusión

Después de combinar los algoritmos ensamblados y el algoritmo de deep learning, se obtuvieron las matrices de confusión de las combinaciones más altas de cada uno de los

datasets binarios y dataset Multiclase, a continuación se describen cada uno de las matrices resultantes.

Para el conjunto de datos del dataset {Normal, Benigno} el clasificador arroja como resultado que 15 instancias representadas como a=Normal son correctamente clasificadas como Normales, mientras 20 instancias representadas como b=Benigno son clasificadas correctamente como Benigno. La clasificación por clase tuvo un rendimiento de 75 % para la clase Normal y un 90 % para la clase Benigno, el resultado obtenido es aceptable para un clasificador Ver Tabla 4.2.

Tabla 4.2: Matriz de confusión del dataset {Normal, Benigno}

a	b	classified as
15	5	a = N
2	20	b = B

Mientras, en la Tabla 4.3 del dataset {Normal, Maligno} las de instancias correctamente clasificadas son 15 veces donde a=Normal y 16 veces correctamente clasificadas para Maligno donde b=Maligno. Para la clasificación por clases 78 % corresponde para la clase Normal y 88 % para la clase Maligno.

Tabla 4.3: Matriz de confusión del dataset {Normal, Maligno}

a	b	classified as
15	4	a = N
2	16	b = M

En el dataset {Benigno, Maligno} su matriz de confusión muestra en la Tabla 4.4 que hay cero instancias correctamente clasificadas como Benigno donde a=Benigno y 17 instancias son clasificadas correctamente como Maligno donde b=Maligno, este clasificador tuvo un mal desempeño.

Tabla 4.4: Matriz de confusión del dataset {Benigno, Maligno}

a	b	classified as
0	23	a = Benigno
0	17	b = Maligno

En el dataset Multiclase {Normal, Benigno, Maligno} de la Tabla 4.5 muestra que 15 instancias son clasificadas correctamente como Benigno representadas como a=Benigno, 12 instancias son clasificadas como Normal representadas como b=Normal y dos instancias correctamente clasificadas como Maligno donde “c” representa a la clase Maligno. Para la clasificación por clases, la clase Benigno es de 62 % , la clase Normal 60 % y la clase Maligno 11 %, demostrando que para la clase Maligno no fue una buena clasificación.

Tabla 4.5: Matriz de confusión del dataset Multiclase {Normal, Benigno ,Maligno}

a	b	c	classified as
15	5	3	a = Benigno
8	12	0	a = Normal
0	17	5	a = Maligno

Capítulo 5

Análisis de resultados

Los resultados obtenidos en esta investigación son aceptables de acuerdo a la literatura, como se pudo apreciar el mejor resultado se obtuvo con el conjunto de datos {Normal, Maligno} y el algoritmo ensamblado *AttributeSelected Classifier* con un 83.7338 % de clasificación de instancias correctamente clasificadas, se pudo observar que el clasificador distingue mejor las anomalías cuando se tiene solo dos clases. Al observar que en el dataset Multi-clase {Normal, Benigno, Maligno} el clasificador se equivoca un número mayor de porcentaje, también que en esta categorización de anomalías no influyen parámetros como el número de épocas y semillas que se asignen a el clasificador, debido a que no fue posible obtener porcentajes que superaran a los resultados mostrados en la Tabla 4.3

(Cruz-Roa y cols., 2017) obtuvo un resultado menor al de este proyecto. Se demostró que cambiando los parámetros como las épocas y la semilla en el clasificador no hubo forma de que los resultados fuesen más altos a los obtenidos con anterioridad.

- Para el dataset binario {Normal, Maligno} el porcentaje mayor de efectividad se obtuvo con el criterio $2/1 - 1/3$ el resultado fue de 83.7338 % instancias correctamente clasificadas
- Para el dataset binario {Normal, Benigno} el porcentaje mayor de efectividad se obtuvo con el criterio $2/1 - 1/3$ el resultado fue de 83.3333 % instancias correctamente clasificadas.
- Para el dataset binario {Benigno-Maligno} el porcentaje mayor de efectividad se obtuvo con el criterio $2/1 - 1/3$ el resultado fue de 69.0476 % instancias correctamente clasificadas.
- Para el dataset multiclase {Normal, Benigno, Maligno} el porcentaje mayor de efectividad se obtuvo con el criterio $2/1 - 1/3$ el resultado fue de 62.5 % instancias correctamente clasificadas

Tabla 5.1: Tabla de comparación entre estado del arte y resultados de la presente investigación

<i>Año</i>	<i>Autores</i>	<i>Método</i>	<i>Resultado</i>
2016	(Pedraza y cols., s.f.)	Red neuronal convolucional , arquitectura GoogLe-Net	95.62 % de instancias correctamente clasificadas
2017	Marlenne Viridiana Roque , de la Cruz	Multilayer perceptrón con deep learning	83.7338 % instancias correctamente clasificadas

5.1. Análisis de algoritmos con mayor clasificación

A continuación se describen cada uno de los algoritmos con mayor porcentaje de instancias correctamente clasificadas en la presente investigación:

- I **AttributeSelectedClassifier**, dataset Binario {Normal, Maligno}, este algoritmo selecciona atributos (discretización supervisada) y después le aplica un algoritmo de clasificación, seleccionar atributos en conjunto de entrenamiento después evaluar lo sobre el conjunto de prueba, usa un clasificador de dos clases para conjuntos de datos multiclase, para estos problemas se utilizan cualquiera de estos métodos (at University of Waikato, 2016):
- i Uno versus todo el resto.
 - ii Clasificación por pares usando votación para predecir.
 - iii Grandes códigos de corrección de errores.
 - iv Códigos de corrección de errores seleccionados al azar, los vectores de código aleatorio tienen buenas propiedades de corrección de errores: el parámetro especifica la longitud del vector de código (en bits)
- II **RandomizableFilteredClassifier** dataset Binario {Normal, Benigno}. Clase para correr clasificadores sobre los datos de entrenamiento que han sido filtrados. Los filtros son aplicados al conjunto de entrenamiento y el la fase de prueba sin filtros (at University of Waikato, 2016).
- III **Random Committee** dataset Binario {Benigno, Maligno}. Este algoritmo construye un conjunto de clasificadores básicos y promedia sus predicciones, cada uno se basa en los mismos datos, pero utiliza una semilla de número aleatorio diferente. Esta solo tiene sentido si el clasificador base es aleatorio; de lo contrario, todos los clasificadores serían lo mismo (at University of Waikato, 2016) .
- IV **MultiClassClassifier**, dataset Multiclase {Normal, Benigno Maligno}. Es un meta clasificador para manejar dataset multiclase con dos clases de clasificadores, aplica corrección de errores. En caso de que quieras reducir o rankear datos, no solo la salida de los atributos rankeados, también se obtiene la clasificación o filtrado por lotes (at University of Waikato, 2016) .

Capítulo 6

Conclusiones y Trabajo a Futuro

6.1. Conclusiones

Se demostró que el aprendizaje profundo resulta aceptable en la categorización de anomalías cancerígenas, esto puede llegar a ser una contribución al especialista médico, para ser un apoyo en la interpretación de las imágenes, al mismo tiempo, esto permite contribuir en el desarrollo tecnológico y social en beneficio de pacientes propensos a padecer cáncer de mama.

El trabajo presentado, en comparación con el estado del arte, obtuvo resultados aceptables, estos resultados se podrían mejorar utilizando arquitecturas diferentes de redes neuronales como las que se emplean típicamente para desempeñar estas tareas de clasificación entrenadas de manera no supervisada, y CNNs, siendo esta última la más utilizada en los últimos años (un 76,6 % de los artículos publicados entre 2015 y 2017 lo confirman (Litjens y cols., 2017) además, el uso de CNNs ha demostrado tener un desempeño muy bueno, llegando a desafiar la precisión de expertos humanos (Pérez, 2017).

6.2. Trabajos Futuros

Como trabajo futuro se pretende mejorar los resultados obtenidos en esta investigación, para ello se pretende hacer los siguientes cambios:

- I Utilizar otros enfoques de aprendizaje profundo como las CNN.
- II Explorar arquitecturas de aprendizaje profundo que sean más flexibles de entrenar y que podrían proporcionar una interpretabilidad aún más concreta.
- III Poder desarrollar un sistema de detección y clasificación como el trabajo de (Carreras-Cruz y cols., 2015) utilizando un enfoque diferente.
- IV Utilizar un dataset de la región para corroborar que el trabajo realizado obtiene la misma precisión registrada.
- V Realizar la segmentación de cada una de las imágenes con el trabajo desarrollado.
- VI Utilizar un nuevo entorno de programación que pudiera ayudar a desarrollar un proyecto de mejor calidad y precisión.

Referencias

- Al-Masni, M., Al-Antari, M., Park, J., Gi, G., Kim, T., Rivera, P., ... Kim, T.-S. (2017). Detection and classification of the breast abnormalities in digital mammograms via regional convolutional neural network. En *Engineering in medicine and biology society (embc), 2017 39th annual international conference of the ieee* (pp. 1230–1233).
- Arafi, A., Fajr, R., y Bouroumi, A. (2014). Breast cancer data analysis using support vector machines and particle swarm optimization. En *Complex systems (wccs), 2014 second world conference on* (pp. 1–6).
- Arevalo, J., González, F. A., Ramos-Pollán, R., Oliveira, J. L., y Lopez, M. A. G. (2016). Representation learning for mammography mass lesion classification with convolutional neural networks. *Computer methods and programs in biomedicine*, 127, 248–257.
- at University of Waikato, M. L. G. (2016). *Weka 3 data mining software in java*. Descargado de [URL{https://www.cs.waikato.ac.nz/ml/weka}](https://www.cs.waikato.ac.nz/ml/weka) ([Web; accedido el 10-08-2017])
- Cambronero, C. G., y Moreno, I. G. (2006). Algoritmos de aprendizaje: knn & kmeans. *Inteligencia en Redes de Comunicación, Universidad Carlos III de Madrid*.
- Carreras-Cruz, M. V., de Lourdes Martínez-Villaseñor, M., y Rosas-Pérez, K. N. (2015). Mass segmentation in digital mammograms. En *Ambient intelligence for health* (pp. 110–115).
- Corso, C. L. (2009). Aplicación de algoritmos de clasificación supervisada usando weka. *Córdoba: Universidad Tecnológica Nacional, Facultad Regional Córdoba*.
- Cruz, M. V. C., y Vilella, P. R. (2006). Circumscribed mass detection in digital mammograms. En *Electronics, robotics and automotive mechanics conference, 2006* (Vol. 1, pp. 19–24).
- Cruz-Roa, A., Gilmore, H., Basavanhally, A., Feldman, M., Ganesan, S., Shih, N. N., ... Madabhushi, A. (2017). Accurate and reproducible invasive breast cancer detection in whole-slide images: A deep learning approach for quantifying tumor extent. *Scientific Reports*, 7, 46450.
- Dalmış, M. U., Gubern-Mérida, A., Vreemann, S., Karssemeijer, N., Mann, R., y Platel, B. (2016). A computer-aided diagnosis system for breast dce-mri at high spatiotemporal resolution. *Medical physics*, 43(1), 84–94.
- DelaCruz, E. S., Alpuín-Jiménez, H., Domínguez, H. d. J. O., y Parra, P. P. (2011). SDCA: System to Detect Cancerous Abnormalities. En *La-nmr* (pp. 115–122).
- De Raedt, L. (2008). *Logical and relational learning*. Springer Science & Business Media.

- Gallegos-Funes, F. J., Gómez-Mayorga, M. E., Lopez-Bonilla, J. L., y Cruz-Santiago, R. (2009). Rank m-type radial basis function (rmrbf) neural network for pap smear microscopic image classification. *Apeiron: Studies in Infinite Nature*, 16(4).
- García Navarro, B. (2015). Implementación de técnicas de deep learning.
- Gerazov, B., y Conceicao, R. C. (2017). Deep learning for tumour classification in homogeneous breast tissue in medical microwave imaging. En *Smart technologies, iee eurocon 2017-17th international conference on* (pp. 564–569).
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., y Witten, I. H. (2009). The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1), 10–18.
- IMSS. (1994). *cancer de mama*. Descargado de [URL{http://www.imss.gob.mx/salud-en-linea/cancer-mama/mastografia/ash}](http://www.imss.gob.mx/salud-en-linea/cancer-mama/mastografia/ash) ([Web; accedido el 10-08-2017])
- Krizhevsky, A., Sutskever, I., y Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. En *Advances in neural information processing systems* (pp. 1097–1105).
- Lévy, D., y Jain, A. (2016). Breast mass classification from mammograms using deep convolutional neural networks. *arXiv preprint arXiv:1612.00542*.
- Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical image analysis*, 42, 60–88.
- Lorca, G., Arzola, J., y Pereira, O. (2010). Segmentación de imágenes médicas digitales mediante técnicas de clustering. *Rev. Aporte Santiaguino*, 108–116.
- MIAS. (1994). *The mini mias database of mammograms*. Descargado de [URL{http://peipa.essex.ac.uk/info/mias.html/ash}](http://peipa.essex.ac.uk/info/mias.html/ash) ([Web; accedido el 10-08-2017])
- Moradkhani, F., y Bigham, B. S. (2017). A new image mining approach for detecting micro-calcification in digital mammograms. *Applied Artificial Intelligence*, 31(5-6), 411–424.
- msdn.microsoft. (2016). *Analysis services - minería de datos*. Descargado de [URL{https://msdn.microsoft.com/es-es/library/ms174811\(v=sql.120\).aspx}](https://msdn.microsoft.com/es-es/library/ms174811(v=sql.120).aspx) ([Web; accedido el 30-12-2017])
- Neto, O. P. S., Carvalho, O., Sampaio, W., Corrêa, A., y Paiva, A. (2015). Automatic segmentation of masses in digital mammograms using particle swarm optimization and graph clustering. En *Systems, signals and image processing (iwSSIP), 2015 international conference on* (pp. 109–112).
- Pedraza, A., Serrano, I., Fernandez-Carrobles, M., Rojo, M. G., González, L., y Bueno, G. (s.f.). Diagnóstico automático del her2 con deep learning.
- Peréz, E. P. (2017). *Diseño de una metodología para el procesamiento de imágenes mamográficas basada en técnicas de aprendizaje profundo* (Tesis Doctoral no publicada). Departamento de matemáticas aplicada a las tecnologías de la información y las comunicaciones.
- Rautaray, S. S., y Agrawal, A. (2015). Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, 43(1), 1–54.

- Ríos, N., y Avendaño, G. G. (2008). Reporte de lesiones no palpables diagnosticadas por mastografía en el hospital general de México. En *Anales de radiología México* (Vol. 4, pp. 248–251).
- Roig-Vila, R., y cols. (2016). Tecnología, innovación e investigación en los procesos de enseñanza-aprendizaje. *Barcelona: Editorial Octaedro*.
- Secretaría, S., de. (2015). *Estadísticas de cáncer de mama en México*. Descargado de [URL{https://www.gob.mx/salud/ash}](https://www.gob.mx/salud/ash) ([Web; accedido el 10-08-2017])
- Tang, J., Deng, C., y Huang, G.-B. (2016). Extreme learning machine for multilayer perceptron. *IEEE transactions on neural networks and learning systems*, 27(4), 809–821.
- Witten, I. H., Frank, E., Hall, M. A., y Pal, C. J. (2016). *Data mining: Practical machine learning tools and techniques*. Morgan Kaufmann.
- Xu, Y., Mo, T., Feng, Q., Zhong, P., Lai, M., Eric, I., y Chang, C. (2014). Deep learning of feature representation with multiple instance learning for medical image analysis. En *Acoustics, speech and signal processing (icassp), 2014 IEEE international conference on* (pp. 1626–1630).

Anexos

6.2.1. Anexo A

Código de aplanamiento.

```
1 import os
2 from PIL import Image
3 from scipy import misc
4 import numpy as np
5 import csv
6
7 root = os.getcwd()
8
9 carpeta = "\imagenes"
10 # leer nombre de los archivos de las imagenes
11 ruta = root + carpeta
12 lista_imagenes = []
13 for (dirpath, dirnames, filenames) in os.walk(ruta):
14     lista_imagenes.extend(filenames)
15     break
16 print(lista_imagenes)
17
18 os.chdir(ruta)
19 matrix= []
20 matrixMax=[]
21
22 import string, random
23 lista = []
24 N=100
25 i=0
26 while i<1024:
27     valor = ''.join(random.choices(string.ascii_uppercase + string.digits, k
28     =3))
29     if valor not in lista:
30         lista.append(valor)
31         i=i+1
32
33 lista.insert(0,"ID")
34 for i in range(len(lista_imagenes)):
35
36     # LEER IMÁGENES EN FORMA DE MATRIZ
37     imagen = Image.open(lista_imagenes[i])
38     matrix = misc.imread(lista_imagenes[i])
39     ancho, alto =np.shape(matrix)
40     # crear columna id
41     ids=np.full(alto, i)
42     # Agregar columna ID
```

```

43     print("imagen :",i)
44     matrix = np.c_[ids, matrix]
45
46     etiqueta = np.full(alto, random.choices(['B','M','N'],k=1))
47     matrix = np.c_[matrix, etiqueta]
48     # Agregar a matriz principal
49     if i == 0:
50         matrixMax = np.r_[lista]
51         matrixMax = np.r_[matrix]
52     else:
53         matrixMax = np.r_[matrixMax, matrix]
54
55     print("Dimensiones",matrixMax.shape)
56     #print(Image.fromstring('L', imagen.size))
57     # AGREGAR ID y guardar el formato CSV
58     os.chdir(root)
59     np.savetxt("base_conocimientos.csv", matrixMax.astype(int),fmt='%a', delimiter
    =",")

```

```

1 import string, random
2 from scipy import misc
3 import numpy as np
4 import csv
5 import os
6
7 lista = []
8 N=100
9 i=0
10 while i<1024:
11     valor = ''.join(random.choices(string.ascii_uppercase, k=3))
12     if valor not in lista:
13         lista.append(valor)
14         i=i+1
15
16 lista.insert(0,"ID")
17
18 #np.savetxt("ides.csv",lista)
19 import pandas
20 yourlist = lista
21
22 pd = pandas.DataFrame(np.array(yourlist).reshape(-1,len(yourlist)))
23 pd.to_csv("mylist.csv",header=False)

```

6.2.2. Criterio 2/3 - 1/3

Criterio muestra representativa

Dataset Binario {Normal, Maligno}

Tabla 6.1: Dataset binario {Normal, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4JMlpClassifier	62.1622 %
AdaBoost	DI4JMlpClassifier	70.2703 %
AttributeSelectedClassifier	DI4JMlpClassifier	83.7838 %
Bagging.	DI4JMlpClassifier	78.3784 %
ClassificationViaRegression	DI4JMlpClassifier	51.3514 %
CVParameterSelection	DI4JMlpClassifier	62.1622 %
FilteredClassifier	DI4JMlpClassifier	72.973 %
LogiBoost	DI4JMlpClassifier	51.3514 %
MultiClassClassifier	DI4JMlpClassifier	62.1622 %
RandomCommittee	DI4JMlpClassifier	59.4595 %
RandomizableFilteredClassifier	DI4JMlpClassifier	83.3333 %
RandomSubSpace	DI4JMlpClassifier	69.0476 %
Stacking	DI4JMlpClassifier	47.619 %
WeightedInstancesHandlerWrapper	DI4JMlpClassifier	66.6667 %

Dataset Binario {Normal, Benigno}

Tabla 6.2: Dataset Binario {Normal, Benigno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4JMlpClassifier	66.6667 %
AdaBoost	DI4JMlpClassifier	69.0476 %
AttributeSelectedClassifier	DI4JMlpClassifier	76.1905 %
Bagging.	DI4JMlpClassifier	69.0476 %
ClassificationViaRegression	DI4JMlpClassifier	52.381 %
CVParameterSelection	DI4JMlpClassifier	66.6667 %
FilteredClassifier	DI4JMlpClassifier	61.9048 %
LogiBoost	DI4JMlpClassifier	47.619 %
MultiClassClassifier	DI4JMlpClassifier	66.6667 %
RandomCommittee	DI4JMlpClassifier	69.0476 %
RandomizableFilteredClassifier	DI4JMlpClassifier	83.3333 %
RandomSubSpace	DI4JMlpClassifier	69.0476 %
Stacking	DI4JMlpClassifier	47.619 %
WeightedInstancesHandlerWrapper	DI4JMlpClassifier	66.6667 %

Dataset Binario {Benigno, Maligno}

Tabla 6.3: Dataset {Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	62.5 %
AdaBoost	DI4MlpClassifier	60 %
AttributeSelectedClassifier	DI4MlpClassifier	50 %
Bagging.	DI4MlpClassifier	45 %
ClassificationViaRegression	DI4MlpClassifier	57.5 %
CVParameterSelection	DI4MlpClassifier	62.5 %
FilteredClassifier	DI4MlpClassifier	47.5 %
LogiBoost	DI4MlpClassifier	57.5 %
MultiClassClassifier	DI4MlpClassifier	62.5 %
RandomCommittee	DI4MlpClassifier	69.0476 %
RandomizableFilteredClassifier	DI4MlpClassifier	45 %
RandomSubSpace	DI4MlpClassifier	52.5 %
Stacking	DI4MlpClassifier	57.5 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	62.5 %

Dataset Multiclase {Normal, Benigno, Maligno}

Tabla 6.4: Dataset Multiclase {Normal, Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	35 %
AdaBoost	DI4MlpClassifier	51.66 %
AttributeSelectedClassifier	DI4MlpClassifier	50 %
Bagging.	DI4MlpClassifier	45 %
ClassificationViaRegression	DI4MlpClassifier	57.5 %
CVParameterSelection	DI4MlpClassifier	62 %
FilteredClassifier	DI4MlpClassifier	47.5 %
LogiBoost	DI4MlpClassifier	57.5 %
MultiClassClassifier	DI4MlpClassifier	62.5 %
RandomCommittee	DI4MlpClassifier	60 %
RandomizableFilteredClassifier	DI4MlpClassifier	45 %
RandomSubSpace	DI4MlpClassifier	46.66 %
Stacking	DI4MlpClassifier	35 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	62.5 %

6.2.3. Criterio Cross folder

Dataset Binario {Normal, Benigno}

Tabla 6.5: Dataset Binario Normal, Benigno y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de cross folder

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4JMlpClassifier	74.4 %
AdaBoost	DI4MlpClassifier	76.8 %
AttributeSelectedClassifier	DI4MlpClassifier	76 %
Bagging.	DI4MlpClassifier	58 %
ClassificationViaRegression	DI4MlpClassifier	56.41 %
CVPParameterSelection	DI4MlpClassifier	74.4 %
FilteredClassifier	DI4MlpClassifier	55.55 %
LogiBoost	DI4MlpClassifier	47.2 %
MultiClassClassifier	DI4MlpClassifier	74.2 %
RandomCommittee	DI4MlpClassifier	75.2 %
RandomizableFilteredClassifier	DI4MlpClassifier	78.4 %
RandomSubSpace	DI4MlpClassifier	65 %
Stacking	DI4MlpClassifier	50.4 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	74.4 %

Dataset Binario {Normal, Maligno}

Tabla 6.6: Dataset Binario {Normal, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de cross folder

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	79.09 %
AdaBoost	DI4MlpClassifier	80 %
AttributeSelectedClassifier	DI4MlpClassifier	80.9091 %
Bagging.	DI4MlpClassifier	79 %
ClassificationViaRegression	DI4MlpClassifier	53.63 %
CVPParameterSelection	DI4MlpClassifier	67.45 %
FilteredClassifier	DI4MlpClassifier	64.54 %
LogiBoost	DI4MlpClassifier	53.63 %
MultiClassClassifier	DI4MlpClassifier	79.09 %
RandomCommittee	DI4MlpClassifier	80 %
RandomizableFilteredClassifier	DI4MlpClassifier	67.2 %
RandomSubSpace	DI4MlpClassifier	65 %
Stacking	DI4MlpClassifier	50.4 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	56 %

Dataset Binario {Benigno, Maligno}

Tabla 6.7: Dataset Binario {Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de cross folder

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	55.55 %
AdaBoost	DI4MlpClassifier	47.73 %
AttributeSelectedClassifier	DI4MlpClassifier	51.28 %
Bagging.	DI4MlpClassifier	44.44 %
ClassificationViaRegression	DI4MlpClassifier	56.4103 %
CVParameterSelection	DI4MlpClassifier	41.8803 %
FilteredClassifier	DI4MlpClassifier	55.5556 %
LogiBoost	DI4MlpClassifier	56.4003 %
MultiClassClassifier	DI4MlpClassifier	41.8803 %
RandomCommittee	DI4MlpClassifier	42.7956 %
RandomizableFilteredClassifier	DI4MlpClassifier	51 %
RandomSubSpace	DI4MlpClassifier	45 %
Stacking	DI4MlpClassifier	51.2821 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	41.8803 %

Dataset Multiclase {Normal, Benigno, Maligno}

Tabla 6.8: Dataset Multiclase {Normal, Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de cross folder

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	50.5682 %
AdaBoost	DI4MlpClassifier	48.8636 %
AttributeSelectedClassifier	DI4MlpClassifier	47.1591 %
Bagging.	DI4MlpClassifier	37.5 %
ClassificationViaRegression	DI4MlpClassifier	50.5682 %
CVPParameterSelection	DI4MlpClassifier	48 %
FilteredClassifier	DI4MlpClassifier	56.25 %
LogiBoost	DI4MlpClassifier	50 %
MultiClassClassifier	DI4MlpClassifier	36.9318 %
RandomCommittee	DI4MlpClassifier	50 %
RandomizableFilteredClassifier	DI4MlpClassifier	48.8636 %
RandomSubSpace	DI4MlpClassifier	46.5909 %
Stacking	DI4MlpClassifier	47.7273 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	41.8803 %

6.2.4. Criterio Muestra Representativa

Dataset Binario {Normal Benigno}

Tabla 6.9: Dataset Binario {Normal, Benigno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	73.68 %
AdaBoost	DI4MlpClassifier	71.5789 %
AttributeSelectedClassifier	DI4MlpClassifier	71.5789 %
Bagging.	DI4MlpClassifier	74.7368 %
ClassificationViaRegression	DI4MlpClassifier	48.4211 %
CVParameterSelection	DI4MlpClassifier	73.6842 %
FilteredClassifier	DI4MlpClassifier	67.3684 %
LogiBoost	DI4MlpClassifier	51.5789 %
MultiClassClassifier	DI4MlpClassifier	73.6842 %
RandomCommittee	DI4MlpClassifier	64.2105 %
RandomizableFilteredClassifier	DI4MlpClassifier	67.3684 %
RandomSubSpace	DI4MlpClassifier	74.7368 %
Stacking	DI4MlpClassifier	48.4211 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	73.6842 %

Dataset Binario {Normal, Maligno}

Tabla 6.10: Dataset Binario {Normal, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	61.6279 %
AdaBoost	DI4MlpClassifier	60.4651 %
AttributeSelectedClassifier	DI4MlpClassifier	59.3023 %
Bagging.	DI4MlpClassifier	60.4651 %
ClassificationViaRegression	DI4MlpClassifier	53.4884 %
CVParameterSelection	DI4MlpClassifier	61.6279 %
FilteredClassifier	DI4MlpClassifier	55.814 %
LogiBoost	DI4MlpClassifier	53.4884 %
MultiClassClassifier	DI4MlpClassifier	61.6279 %
RandomCommittee	DI4MlpClassifier	58.1395 %
RandomizableFilteredClassifier	DI4MlpClassifier	60.4651 %
RandomSubSpace	DI4MlpClassifier	63.9535 %
Stacking	DI4MlpClassifier	53.4884 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	61.6279 %

Dataset Binario {Benigno, Maligno}

Tabla 6.11: Dataset Binario {Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	41.1111 %
AdaBoost	DI4MlpClassifier	41.1111 %
AttributeSelectedClassifier	DI4MlpClassifier	40.1234 %
Bagging.	DI4MlpClassifier	45.5556 %
ClassificationViaRegression	DI4MlpClassifier	55.5556 %
CVParameterSelection	DI4MlpClassifier	41.1111 %
FilteredClassifier	DI4MlpClassifier	52.2222 %
LogiBoost	DI4MlpClassifier	41.1455 %
MultiClassClassifier	DI4MlpClassifier	47.7778 %
RandomCommittee	DI4MlpClassifier	37.7778 %
RandomizableFilteredClassifier	DI4MlpClassifier	32 %
RandomSubSpace	DI4MlpClassifier	38.8889 %
Stacking	DI4MlpClassifier	55.4556 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	41.2024 %

Dataset Multiclase {Normal, Benigno, Maligno}

Tabla 6.12: Dataset Multiclase {Normal, Benigno, Maligno} y todos los resultados de las combinaciones con los algoritmos ensamblados y el algoritmo de deep learning DI4Mlp con el criterio de muestra representativa

Clasificador	Algoritmo deep learning	Instancias correctas
DI4MlpClassifier	DI4MlpClassifier	38.843 %
AdaBoost	DI4MlpClassifier	48.7603 %
AttributeSelectedClassifier	DI4MlpClassifier	48.7603 %
Bagging.	DI4MlpClassifier	45.4545 %
ClassificationViaRegression	DI4MlpClassifier	30.5785 %
CVParameterSelection	DI4MlpClassifier	38.843 %
FilteredClassifier	DI4MlpClassifier	41.3223 %
LogiBoost	DI4MlpClassifier	30.5785 %
MultiClassClassifier	DI4MlpClassifier	41.1223 %
RandomCommittee	DI4MlpClassifier	47.1074 %
RandomizableFilteredClassifier	DI4MlpClassifier	39.669 %
RandomSubSpace	DI4MlpClassifier	49.5868 %
Stacking	DI4MlpClassifier	30.5785 %
WeightedInstancesHandlerWrapper	DI4MlpClassifier	38.843 %