



Universidad Simón Bolívar
Decanato de Estudios Profesionales
Coordinación de Ingeniería de la Computación

Título

Por:
Antonio Álvarez

Realizado con la asesoría de:
Emely Arráiz B.

PROYECTO DE GRADO
Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero de Computación

Sartenejas, septiembre de 2018

Resumen

Hola mundo

Índice general

Resumen	I
Índice de Figuras	IV
Lista de Tablas	V
Índice de algoritmos	X
Acrónimos y Símbolos	XI
Introducción	1
1. Marco teórico	2
1.1. Descubrimiento de Conocimiento y preprocesamiento de datos	2
1.2. Selección de Instancias y Selección de Prototipos	5
1.2.1. Taxonomía del problema de selección de prototipos	6
1.2.1.1. Dirección de búsqueda	7
1.2.1.2. Tipo de selección	7
1.2.1.3. Evaluación de la búsqueda	8
1.2.2. Heurísticas	9
1.2.2.1. Condensed Nearest Neighbor (CNN)	9
1.2.2.2. Edited Nearest Neighbor (ENN)	9
1.2.2.3. Relaxed Selective Subset (RSS)	9
1.3. Metaheurísticas	11
1.3.1. Metaheurísticas basadas en una única solución	12
1.3.2. Metaheurísticas basadas en una población	12
1.3.2.1. Algoritmos evolutivos	13
1.3.2.1.1. Algoritmo Genético Generacional (GGA) . . .	14
1.3.2.1.2. Algoritmo Genético Estacionario (SSGA) . . .	15
1.3.2.1.3. Algoritmo Memético (MA)	15
1.3.2.1.4. CHC Adaptative Search Algorithm	16
2. Marco metodológico	19
2.1. Representación del cromosoma	19

2.2. Función objetivo	20
2.3. Adaptaciones de los algoritmos evolutivos	21
2.4. Criterios para comparar los métodos de selección de prototipos	23
2.5. Conjunto de datos	25
2.6. Validación cruzada y estratificación	25
2.7. Entonación de las metaheurísticas	29
3. Evaluación experimental	31
3.1. Diseño experimental	31
3.1.1. Resultados de heurísticas	34
3.1.2. Resultados de la combinación entre heurísticas y metaheurísticas	34
3.1.2.1. Con número de iteraciones variable	34
3.1.2.2. Fijando el número de iteraciones	34
Conclusiones y Recomendaciones	43
Bibliografía	44
A. Resultados de todos los conjuntos	50
B. Pruebas estadísticas	98

Índice de figuras

1.1. Taxonomía para los métodos de selección de prototipos	8
2.1. Representación de un cromosoma y su respectivo conjunto reducido . .	20
2.2. Cruce de un punto	21
2.3. Validación cruzada	27
2.4. Estratificación	28

Índice de Tablas

2.1. Conjuntos de datos pequeños	26
2.2. Conjuntos de datos medianos	27
2.3. Conjuntos de datos grandes	27
3.1. Parámetros usados para <i>irace</i>	31
3.2. Rangos usados para los parámetros en la entonación	32
3.3. Parámetros usados para los conjuntos pequeños	32
3.4. Parámetros usados para los conjuntos medianos	32
3.5. Parámetros usados para los conjuntos grandes	33
3.6. Promedios de GGA para los conjuntos pequeños 1000 iteraciones	34
3.7. Promedios de GGA para los conjuntos medianos 1000 iteraciones	34
3.8. Promedios de GGA para los conjuntos grandes con 1000 iteraciones . .	34
3.9. Promedios de SSGA para los conjuntos pequeños 1000 iteraciones . . .	35
3.10. Promedios de SSGA para los conjuntos medianos 1000 iteraciones . . .	35
3.11. Promedios de SSGA para los conjuntos grandes con 1000 iteraciones . .	35
3.12. Promedios de MA para los conjuntos pequeños 1000 iteraciones	35
3.13. Promedios de MA para los conjuntos medianos 1000 iteraciones	36
3.14. Promedios de MA para los conjuntos grandes con 1000 iteraciones . . .	36
3.15. Promedios de CHC para los conjuntos pequeños 1000 iteraciones	36
3.16. Promedios de CHC para los conjuntos medianos con 1000 iteraciones . .	36
3.17. Promedios de CHC para los conjuntos grandes con 1000 iteraciones . .	37
3.18. Rangos de las metaheurísticas en <i>accuracy</i> , <i>kappa</i> y reducción para los conjuntos pequeños	37
3.19. Rangos de las metaheurísticas en <i>accuracy + reducción</i> , <i>kappa + reduc-</i> <i>ción</i> y tiempo para los conjuntos pequeños	38
3.20. Rangos de las metaheurísticas en <i>accuracy</i> , <i>kappa</i> y reducción para los conjuntos medianos	39
3.21. Rangos de las metaheurísticas en <i>accuracy + reducción</i> , <i>kappa + reduc-</i> <i>ción</i> y tiempo para los conjuntos medianos	40
3.22. Rangos de las metaheurísticas en <i>accuracy</i> , <i>kappa</i> y reducción para los conjuntos grandes	41
3.23. Rangos de las metaheurísticas en <i>accuracy + reducción</i> , <i>kappa + reduc-</i> <i>ción</i> y tiempo para los conjuntos grandes	42
A.1. Resultados de GGA para conjuntos pequeños con 1000 iteraciones fijas	50

A.2. Resultados de GGA para conjuntos medianos con 1000 iteraciones fijas	51
A.3. Resultados de GGA para conjuntos grandes con 1000 iteraciones fijas .	51
A.4. Resultados de CNN-GGA para conjuntos pequeños con 1000 iteraciones fijas	52
A.5. Resultados de CNN-GGA para conjuntos medianos con 1000 iteraciones fijas	53
A.6. Resultados de CNN-GGA para conjuntos grandes con 1000 iteraciones fijas	53
A.7. Resultados de ENN-GGA para conjuntos pequeños con 1000 iteraciones fijas	54
A.8. Resultados de ENN-GGA para conjuntos medianos con 1000 iteraciones fijas	55
A.9. Resultados de ENN-GGA para conjuntos grandes con 1000 iteraciones fijas	55
A.10. Resultados de RSS-GGA para conjuntos pequeños con 1000 iteraciones fijas	56
A.11. Resultados de RSS-GGA para conjuntos medianos con 1000 iteraciones fijas	57
A.12. Resultados de RSS-GGA para conjuntos grandes con 1000 iteraciones fijas	57
A.13. Resultados de CNN-RSS-GGA para conjuntos pequeños con 1000 iteraciones fijas	58
A.14. Resultados de CNN-RSS-GGA para conjuntos medianos con 1000 iteraciones fijas	59
A.15. Resultados de CNN-RSS-GGA para conjuntos grandes con 1000 iteraciones fijas	59
A.16. Resultados de ENN-RSS-GGA para conjuntos pequeños con 1000 iteraciones fijas	60
A.17. Resultados de ENN-RSS-GGA para conjuntos medianos con 1000 iteraciones fijas	61
A.18. Resultados de ENN-RSS-GGA para conjuntos grandes con 1000 iteraciones fijas	61
A.19. Resultados de SSGA para conjuntos pequeños con 1000 iteraciones fijas	62
A.20. Resultados de SSGA para conjuntos medianos con 1000 iteraciones fijas	63
A.21. Resultados de SSGA para conjuntos grandes con 1000 iteraciones fijas .	63
A.22. Resultados de CNN-SSGA para conjuntos pequeños con 1000 iteraciones fijas	64
A.23. Resultados de CNN-SSGA para conjuntos medianos con 1000 iteraciones fijas	65
A.24. Resultados de CNN-SSGA para conjuntos grandes con 1000 iteraciones fijas	65
A.25. Resultados de ENN-SSGA para conjuntos pequeños con 1000 iteraciones fijas	66

A.26.Resultados de ENN-SSGA para conjuntos medianos con 1000 iteraciones fijas	67
A.27.Resultados de ENN-SSGA para conjuntos grandes con 1000 iteraciones fijas	67
A.28.Resultados de RSS-SSGA para conjuntos pequeños con 1000 iteraciones fijas	68
A.29.Resultados de RSS-SSGA para conjuntos medianos con 1000 iteraciones fijas	69
A.30.Resultados de RSS-SSGA para conjuntos grandes con 1000 iteraciones fijas	69
A.31.Resultados de CNN-RSS-SSGA para conjuntos pequeños con 1000 iteraciones fijas	70
A.32.Resultados de CNN-RSS-SSGA para conjuntos medianos con 1000 iteraciones fijas	71
A.33.Resultados de CNN-RSS-SSGA para conjuntos grandes con 1000 iteraciones fijas	71
A.34.Resultados de ENN-RSS-SSGA para conjuntos pequeños con 1000 iteraciones fijas	72
A.35.Resultados de ENN-RSS-SSGA para conjuntos medianos con 1000 iteraciones fijas	73
A.36.Resultados de ENN-RSS-SSGA para conjuntos grandes con 1000 iteraciones fijas	73
A.37.Resultados de MA para conjuntos pequeños con 1000 iteraciones fijas	74
A.38.Resultados de MA para conjuntos medianos con 1000 iteraciones fijas	75
A.39.Resultados de MA para conjuntos grandes con 1000 iteraciones fijas	75
A.40.Resultados de CNN-MA para conjuntos pequeños con 1000 iteraciones fijas	76
A.41.Resultados de CNN-MA para conjuntos medianos con 1000 iteraciones fijas	77
A.42.Resultados de CNN-MA para conjuntos grandes con 1000 iteraciones fijas	77
A.43.Resultados de ENN-MA para conjuntos pequeños con 1000 iteraciones fijas	78
A.44.Resultados de ENN-MA para conjuntos medianos con 1000 iteraciones fijas	79
A.45.Resultados de ENN-MA para conjuntos grandes con 1000 iteraciones fijas	79
A.46.Resultados de RSS-MA para conjuntos pequeños con 1000 iteraciones fijas	80
A.47.Resultados de RSS-MA para conjuntos medianos con 1000 iteraciones fijas	81
A.48.Resultados de RSS-MA para conjuntos grandes con 1000 iteraciones fijas	81

A.49.Resultados de CNN-RSS-MA para conjuntos pequeños con 1000 iteraciones fijas	82
A.50.Resultados de CNN-RSS-MA para conjuntos medianos con 1000 iteraciones fijas	83
A.51.Resultados de CNN-RSS-MA para conjuntos grandes con 1000 iteraciones fijas	83
A.52.Resultados de ENN-RSS-MA para conjuntos pequeños con 1000 iteraciones fijas	84
A.53.Resultados de ENN-RSS-MA para conjuntos medianos con 1000 iteraciones fijas	85
A.54.Resultados de ENN-RSS-MA para conjuntos grandes con 1000 iteraciones fijas	85
A.55.Resultados de CHC para conjuntos pequeños con 1000 iteraciones fijas	86
A.56.Resultados de CHC para conjuntos medianos con 1000 iteraciones fijas	87
A.57.Resultados de CHC para conjuntos grandes con 1000 iteraciones fijas .	87
A.58.Resultados de CNN-CHC para conjuntos pequeños con 1000 iteraciones fijas	88
A.59.Resultados de CNN-CHC para conjuntos medianos con 1000 iteraciones fijas	89
A.60.Resultados de CNN-CHC para conjuntos grandes con 1000 iteraciones fijas	89
A.61.Resultados de ENN-CHC para conjuntos pequeños con 1000 iteraciones fijas	90
A.62.Resultados de ENN-CHC para conjuntos medianos con 1000 iteraciones fijas	91
A.63.Resultados de ENN-CHC para conjuntos grandes con 1000 iteraciones fijas	91
A.64.Resultados de RSS-CHC para conjuntos pequeños con 1000 iteraciones fijas	92
A.65.Resultados de RSS-CHC para conjuntos medianos con 1000 iteraciones fijas	93
A.66.Resultados de RSS-CHC para conjuntos grandes con 1000 iteraciones fijas	93
A.67.Resultados de CNN-RSS-CHC para conjuntos pequeños con 1000 iteraciones fijas	94
A.68.Resultados de CNN-RSS-CHC para conjuntos medianos con 1000 iteraciones fijas	95
A.69.Resultados de CNN-RSS-CHC para conjuntos grandes con 1000 iteraciones fijas	95
A.70.Resultados de ENN-RSS-CHC para conjuntos pequeños con 1000 iteraciones fijas	96
A.71.Resultados de ENN-RSS-CHC para conjuntos medianos con 1000 iteraciones fijas	97

A.72. Resultados de ENN-RSS-CHC para conjuntos grandes con 1000 iteraciones fijas	97
B.1. Pruebas de <i>Wilcoxon</i> entre GGA y las demás metaheurísticas	98

Índice de algoritmos

1.1. CNN	10
1.2. ENN	10
1.3. RSS	11
1.4. Algoritmo Genético Generacional	14
1.5. Algoritmo Genético Estacionario	15
1.6. Algoritmo Memético Estacionario	16
1.7. CHC	17
1.8. Recombinar	17
1.9. Reinicializar	18
2.1. Meme	24
2.2. IRACE	29

Acrónimos y Símbolos

KDD	Knowledge Discovery in Databases
DM	Data Mining
IS	Instance Selection
PS	Prototype Selection
NN	Nearest Neighbor
NE	Nearest Enemy
CNN	Condensed Nearest Neighbor
ENN	Edited Nearest Neighbor
RSS	Relaxed Selective Subset
GGA	Generational Genetic Algorithm
SGA	Steady-State Genetic Algorithm
CHC	CHC Adaptive Search Algorithm
MEM	Memetic Algorithm

\in	Relación de pertenencia, « <i>es un elemento de</i> »
\subseteq	Subconjunto
\setminus	Diferencia de conjuntos

Introducción

Introducción

Capítulo 1

Marco teórico

1.1. Descubrimiento de Conocimiento y preprocesamiento de datos

Hoy en día, existe una creciente necesidad de procesar grandes volúmenes de datos, estos datos son producto de la recolección de información de procesos y actividades de distintas índoles y se vuelven un material valioso para extraer información sobre posibles tendencias que puedan existir en dichos procesos. Es aquí donde entra el descubrimiento de conocimiento en bases de datos (KDD por su siglas en inglés) como disciplina encargada del procesamiento de datos para la extracción de información.

KDD es definida por *Smyth, P. et al.* [FSS96] como “el proceso no trivial de identificar patrones en los datos que sean válidos, novedosos, potencialmente útiles y finalmente entendibles”. Para este fin, KDD se subdivide en distintas etapas a llevar a cabo para lograr el fin último de identificar patrones, éstas son [GLH16]: especificación del problema, entendimiento del problema, preprocesamiento de los datos, minería de datos, evaluación de los resultados y explotación de los resultados. En este trabajo es de especial interés la etapa de preprocesamiento de datos.

El preprocesamiento de datos consiste en el conjunto actividades destinadas a preparar los datos para ser usado por un algoritmo de minería de datos (DM por sus siglas en inglés). Las actividades realizadas en el preprocesamiento pueden ser clasificadas como actividades para la preparación de los datos y la reducción de los mismos [GLH16].

La preparación de datos es un paso obligatorio en el preprocesamiento, ya que transforma los datos, que inicialmente no se pueden utilizar para el algoritmo de DM por asuntos como la presencia de atributos faltantes en instancias, datos erróneos y atributos con formatos no aceptables para el algoritmo a utilizar [GLH16]. Dependiendo del enfoque dado, estas actividades pueden clasificarse en:

- **Limpieza de datos [GLH16, KCH⁺03]:** incluye el tratamiento de los atributos faltantes y los datos erróneos, que si se dejan sin tratar resulta en un modelo de minería de datos poco confiable. Un atributo faltante en una instancia resulta de no haberlo introducido al momento del registro o por la pérdida en el proceso de almacenamiento. Los datos con atributos faltantes pueden tratarse de 3 maneras [FKP07]: la eliminación de las instancias que presenten el problema, utilizar métodos de estimación de máxima verosimilitud para calcular promedios y variancias y utilizar algoritmos del repertorio de *machine learning* como k-nn, k-means o *Support Vector Machine* para estimar el valor de los atributos faltantes.

Por su parte, los datos erróneos (también conocidos como datos ruidosos) pueden venir de dos formas [CAB11]: ruido de clase cuando la instancia está mal clasificada y ruido de atributo cuando uno o más valores de los atributos en una instancia están distorsionados y no representan la realidad. Para tratar los datos ruidosos se puede usar 3 métodos: construir algoritmos de DM que no se vean afectados en cierta medida ante el ruido (sean robustos), pulir los datos [Ten99] de tal manera que se corrijan los errores y por último se puede identificar los datos ruidosos para eliminarlos del conjunto y así quedarse sólo con datos correctos [BF99].

- **Transformación de datos [GLH16]:** se centra en aplicar fórmulas matemáticas a los valores de los atributos para así obtener valores sintéticos que pueden proporcionar más información respecto a la instancia y al conjunto que pertenecen, las transformaciones más comunes son la lineal y la cuadrática.
- **Integración de los datos [GLH16, BLN86]:** consiste en la unión de los conjuntos de datos provenientes de distintas fuentes en un único conjunto. La integración tiene que tomar en cuenta algunos aspectos que se pueden presentar

durante el proceso, entre ellos están la redundancia de atributos, la cual sucede cuando 2 atributos están fuertemente correlacionados. La redundancia de atributos puede traer consigo un sobre ajuste (*overfitting* en inglés) de los modelos predictivos, además de aumentar el tiempo de cómputo de los mismos, es por eso que se debe eliminar esta redundancia y para ello se usa una prueba de correlación χ^2 con el fin de identificar los atributos redundantes y así decidir con cual quedarse.

Continuando, con los problemas que se pueden presentar al momento de la integración, se tiene también la duplicación de instancias, problema que normalmente trae consigo la inconsistencia en los valores de los atributos, debido a las diferencias con las que se registran los valores. Para solucionar este asunto primero se tiene que identificar las instancias duplicadas usando técnicas que midan la similitud entre ellas, como la propuesta de *Fellegi, I. & Sunter, A.* [FS69] que lo modela como un problema de inferencia bayesiana o como en [CKLS01] donde se usan árboles de clasificación y regresión (CART por sus siglas en inglés) para cumplir este trabajo.

- **Normalización de datos [GLH16]:** busca cambiar la distribución de los datos originales de tal manera que se acoplen a las necesidades de los algoritmos predictivos. Dos de los tipos de normalización más usadas son la normalización min-max y la normalización *z-score*.

Pasando a la reducción de los datos, se tiene que engloba todas las técnicas que reducen el conjunto de datos original para obtener uno representativo con el cual trabajar en los modelos predictivos. La reducción de datos cobra especial importancia cuando se tienen conjuntos muy grandes que tienden a elevar en gran medida el tiempo de cómputo de los algoritmos que los van a usar. Las técnicas de reducción de datos son [GLH16]:

- **Discretización de datos [GLH16, GLS⁺13]:** es el proceso de transformar datos numéricos en datos categóricos, definiendo un número finito de intervalos que representan rangos entre distintos valores consecutivos con el fin de poder tratarlos como valores nominales. Es de especial importancia conseguir el número correcto de intervalos que mantengan la información original de los datos, ya que

muy pocos intervalos puede llegar a ocultar la relación existente entre un rango en específico y una clase dada y muchos intervalos puede llevar a un sobre ajuste [CPSK07]. El principal atractivo de la discretización es que permite utilizar un algoritmo de DM que trabaje principalmente con datos nominales como *Naïve Bayes* [YW09] a partir de datos numéricos. Para un estudio más completo de la discretización se referencia a [GLS⁺13].

- **Selección de características [GLH16, LM12]:** busca eliminar atributos que sean redundantes o irrelevantes de tal manera que el subconjunto de características restantes mantenga la distribución original de las clases. El proceso de selección de características tiene ventajas, como mantener e incluso mejorar la precisión de los modelos predictivos, reducir los tiempos de cómputo y reducir la complejidad de los modelos resultantes. La búsqueda de un subconjunto de atributos puede realizarse de 3 maneras: búsqueda exhaustiva, búsqueda heurística y métodos no determinísticos. La búsqueda exhaustiva cubre todo el espacio de soluciones, normalmente van probando todas las combinaciones posibles de atributos para conseguir el que mejor se acople a la métrica a optimizar, entre los métodos exhaustivos están *Focus* [AD91], *Automatic Branch & Bound* [LMD98], *Best First Search* [XYC88], entre otros. Por su parte, la búsqueda heurística busca una solución aproximada a la óptima en poco tiempo, entre sus métodos están los propuestos en [DL97, KS96, Bat94]. Por último, están los métodos no determinísticos, de entre los que destacan los algoritmos genéticos, recocido simulado y *Las Vegas Filter* [LS⁺96].
- **Selección de instancias [GLH16]:** consiste en elegir un subconjunto de las instancias totales manteniendo las características del conjunto original. Es el problema a tratar en este trabajo y se elabora más sobre el mismo en la siguiente sección.

1.2. Selección de Instancias y Selección de Prototipos

La selección de instancias (IS por sus siglas en inglés) consiste en reducir el conjunto de datos dado a un conjunto reducido que conserve las capacidades de representación

del conjunto original, para ser utilizado con un algoritmo de clasificación o regresión, manteniendo el desempeño del algoritmo como si se usara el conjunto original.

Definición 1. Dado un conjunto de datos X , se tiene que una instancia $X_i = (X_i^1, X_i^2, \dots, X_i^p)$ donde X_i^j es el atributo j para la instancia X_i con $X_i \in X$ y siendo p el número de atributos. La instancia X_i es de clase Y_j donde $Y_j \in Y$, siendo Y el conjunto de todas las clases definidas con $j \in (1 \dots q)$ donde q es el número de clases totales. Se divide el conjunto X en un conjunto TR de entrenamiento y un conjunto TS de prueba. El problema de **Selección de Instancias** consiste en conseguir un conjunto reducido $S \subseteq TR$ con el cual, al usarse con el clasificador M se mantenga o mejore la capacidad de representación del conjunto original [GLH16].

La respuesta óptima de un método de selección de instancias es un conjunto *consistente* y de cardinalidad mínima.

Definición 2. Un conjunto R es **consistente** con T , si y solo si toda instancia $t \in T$ es clasificada correctamente mediante el uso de un clasificador M y las instancias en R como conjunto de entrenamiento. [Ale14]

Sin embargo, conseguir la respuesta óptima es un problema NP-Duro (*NP-Hard*) como lo demuestra Zukhba, A. en [Zuk10]. Por lo tanto, la mayoría de los métodos propuestos hasta la fecha se enfocan en obtener una solución aproximada.

El problema de selección de instancias se puede enfocar como un problema de selección de prototipos (PS por sus siglas en inglés). PS es en esencia IS con el detalle de que el clasificador M usado es un clasificador basado en instancias [GLH16], de los cuales K Vecinos más Cercanos (KNN por sus siglas en inglés) es el más conocido. En este trabajo se usa 1-NN como clasificador.

1.2.1. Taxonomía del problema de selección de prototipos

En este trabajo se adopta la taxonomía propuesta por García, S. et al. en [GDCH12]. Sea TR el conjunto de entrenamiento y S el conjunto reducido, las propiedades son las siguientes:

1.2.1.1. Dirección de búsqueda

- **Incremental:** se empieza con un conjunto vacío S y se va añadiendo instancias de TR si cumple con cierto criterio. El orden de presentación de las instancias puede llegar a afectar el resultado final para muchos algoritmos, por eso se acostumbra a presentar los datos de manera aleatoria. Una búsqueda incremental tiene la ventaja de que puede seguir agregando instancias una vez finalizado un proceso de selección inicial, lo cual lo hace bastante atractivo para el aprendizaje continuo
- **Decremental:** la búsqueda empieza con $S = TR$ y se va seleccionando instancias para remover de S . El orden de presentación sigue siendo importante, pero a diferencia de los métodos incrementales, se tiene todo el conjunto desde el inicio. Los algoritmos decrementales tienden a presentar un mayor costo computacional que los incrementales.
- **Por lote:** se elige un grupo y se evalúan todos los elementos del mismo para su eliminación, los que no pasen la prueba seleccionada son desechados. El proceso se repite con distintos lotes hasta terminar.
- **Mixto:** S empieza como un subconjunto preseleccionado (puede ser de manera aleatoria o usando un proceso incremental/decremental) e iterativamente puede añadir o remover instancias que cumplan con criterios en específico.
- **Fijo:** el número final de instancias en S se fija al principio de la fase de aprendizaje y se aplica una búsqueda mixta hasta cumplir con dicha cuota.

1.2.1.2. Tipo de selección

- **Condensación:** se busca mantener los puntos bordes (aquellos que están cerca de las fronteras entre las clases). El razonamiento es que son los puntos bordes los que realmente determinan las fronteras, siendo más útiles al momento de clasificar una nueva instancia. Estos métodos tienden a reducir bastante el conjunto original ya que hay menos puntos bordes que interiores.

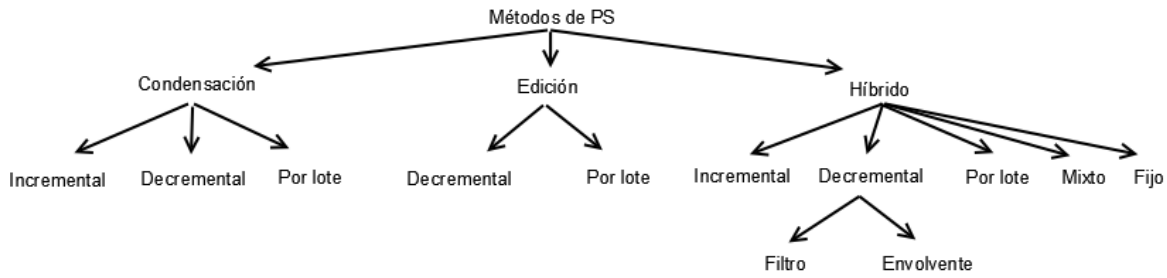


FIGURA 1.1: Taxonomía para los métodos de selección de prototipos

- **Edición:** los métodos de edición en cambio buscan remover los puntos bordes, suavizando las fronteras bajo la idea de que es el lugar donde se concentran la mayor cantidad de puntos ruidosos. Tienden a disminuir en menor medida el conjunto TR en comparación a los métodos de condensación.
- **Híbridos:** su principal objetivo es mantener la precisión del clasificador usando un conjunto lo más reducido posible. Para esto eliminan tanto puntos internos como los ruidosos en el borde, tomando las ideas principales de los métodos de condensación y edición.

1.2.1.3. Evaluación de la búsqueda

- **Filtro:** son los métodos que usan un conjunto parcial de datos para decidir cuáles remover o añadir sin usar un esquema de validación, donde se deja uno por fuera para probar con el resto de los datos en cada iteración del algoritmo.
- **Envolventes:** usan todo el conjunto TR en un proceso de validación cruzada. Son métodos más costosos que los filtros, pero tienden a obtener una precisión mayor al momento de generalizar usando un algoritmo de DM.

A continuación se presenta en la la figura 1.1 la clasificación que se le puede dar a los algoritmos. Para un estudio más extenso sobre los distintos algoritmos se recomienda leer [GLH16]

1.2.2. Heurísticas

En esta sección se exponen las heurísticas utilizadas en este trabajo. Citando a *Pearl, J.* en [Pea84]: “Una heurística es un criterio, método o principio para decidir cual, de entre varias alternativas de acciones a seguir, promete ser la más efectiva para alcanzar un objetivo”. Para el caso de PS, dicho objetivo es alcanzar un buen aproximado del conjunto de cardinalidad mínima y máxima precisión en la clasificación.

1.2.2.1. Condensed Nearest Neighbor (CNN)

Propuesto inicialmente por *Hart, P.* en [Har68], CNN es un método de condensación incremental. El conjunto S se construye de tal manera que cada elemento de TR está más cerca de un miembro de S de la misma clase que un miembro de S de clase distinta. El algoritmo empieza seleccionando una instancia aleatoria s y se coloca en S (inicialmente vacío), acto seguido se empieza a clasificar las instancias de TR sólo usando las instancias pertenecientes S ; si una instancia es clasificada incorrectamente, se agrega a S , asegurando así que en la siguiente vuelta sea clasificada correctamente. Una vez aumentado S se vuelve a probar cada instancia de TR y se agregan las que sean mal clasificadas. El proceso se repite hasta que no existan instancias en TR que se encuentren mal clasificadas. CNN se presenta en el algoritmo 1.1.

1.2.2.2. Edited Nearest Neighbor (ENN)

Propuesto por *Wilson, D.* en [Wil72] ENN es un método de edición decremental. Empieza con $S = TR$ y se va iterando sobre las instancias de S , removiendo aquellas que no concuerdan con la clase de la mayoría de sus k vecinos más cercanos. ENN se presenta en el algoritmo 1.2.

1.2.2.3. Relaxed Selective Subset (RSS)

Propuesto por *Flores, A. & Mount, D.* en [FM17] se tiene que RSS es un algoritmo híbrido incremental con la particularidad de que no es sensible al orden de

Algoritmo 1.1 CNN

Input: TR conjunto de entrenamiento, k número de vecinos a ser considerado en la clasificación

Output: S conjunto reducido

```

1:  $S \leftarrow$  instancia aleatoria  $s$ 
2:  $flag \leftarrow false$ 
3: while  $\neg flag$  do
4:   for all  $x \in TR$  do
5:      $Y \leftarrow k$  vecinos más cercanos a  $s$  pertenecientes a S
6:     Clasificar  $s$  con la misma clase que sea mayoría en Y
7:     if  $s$  está mal clasificada then
8:        $S \leftarrow S \cup \{s\}$ 
9:       Retornar a 3
10:  if Todas las instancias en TR fueron bien clasificadas then
11:     $flag \leftarrow true$ 
12: return S

```

Algoritmo 1.2 ENN

Input: TR conjunto de entrenamiento, k número de vecinos a ser considerado en la clasificación

Output: S conjunto reducido

```

1:  $S \leftarrow TR$ 
2: for  $s \in S$  do
3:    $Y \leftarrow k$  vecinos más cercanos a  $s$  pertenecientes a S
4:   if la clase de  $s$  es distinta a la clase mayoritaria en Y then
5:     Se elimina  $s$  de S
6: return S

```

presentación de las instancias, porque realiza un ordenamiento inicial de las mismas. El método primero ordena las instancias según la distancia que tengan a su enemigo más cercano (la instancia más cercana con clase distinta) de manera incremental (de la distancia más corta a la más larga). Luego, empezando con un conjunto S vacío, se van presentando las instancias y se agrega a S aquellas para las cuales no exista un punto $s \in S$ que esté a una distancia menor que la distancia que tiene s a su enemigo más cercano. Sea $d_{NE}(p)$ la distancia del punto p a su enemigo más cercano y sea $d(p_i, s)$ la distancia de un punto p_i a un punto s . Rss se presenta en el algoritmo 1.3.

Algoritmo 1.3 RSS

Input: TR conjunto de entrenamiento**Output:** S conjunto reducido

```

1:  $S \leftarrow \emptyset$ 
2: Sea  $\{p_i\}_{i=1}^n$  los puntos en TR ordenados de manera ascendente respecto a  $d_{NE}(p_i)$ 
3: for all  $p_i \in TR$  do
4:   if  $\neg \exists s \in S$  tal que  $d(p_i, r) < d_{NE}(r)$  then
5:      $S \leftarrow S \cup \{p_i\}$ 
6: return S

```

1.3. Metaheurísticas

Las metaheurísticas son una familia de algoritmos aproximados de propósito general y no determinístico; consistentes en procedimientos iterativos que guían una heurística subordinada. Al momento de diseñar una metaheurística se debe tomar en cuenta dos conceptos: intensificación y diversificación [Tal09]. En un proceso de intensificación, las regiones en el espacio de soluciones prometedoras son revisadas con la esperanza de conseguir mejores soluciones. En un proceso de diversificación, las regiones no exploradas son visitadas para poder abarcar distintos lugares en el espacio de soluciones y así evitar que la exploración se estanque en una región específica. Las metaheurísticas se pueden clasificar como metaheurísticas basadas en una única solución o metaheurísticas basadas en una población [Tal09]. Para estudiar los distintos métodos, primero se necesita definir una serie de conceptos que son comunes para todos:

Definición 3. La **representación del problema** es la manera de codificar las soluciones pertenecientes al espacio de soluciones. Debe ser acorde al problema de tal manera que cumpla con las siguientes características: debe ser completo, es decir, todas las soluciones deben poder ser codificadas; debe ser conexo, lo que se traduce a que debe haber un camino entre dos cualesquiera soluciones y por último, debe ser eficiente, de tal manera que la manipulación por los operadores de búsqueda tenga un costo en tiempo y espacio razonable [Tal09].

Definición 4. La **función objetivo** (también conocida como función de adaptabilidad o de utilidad) \mathcal{F} asocia a cada solución un valor real que mide la calidad de la solución: $\mathcal{F} : S \rightarrow \mathbb{R}$, donde S es el espacio de soluciones. Con la función objetivo se guía la búsqueda hacia “buenas” soluciones[Tal09].

Definición 5. La **vecindad** de una solución s es el conjunto de soluciones cercanas a s . Se obtienen realizando una pequeña perturbación a s con un operador de movimiento [Tal09].

1.3.1. Metaheurísticas basadas en una única solución

También conocidas como metaheurísticas de trayectoria, se centran en mejorar una solución que van cambiando a lo largo del curso del algoritmo; se puede ver como trayectorias de búsqueda en el espacio de soluciones, dichas trayectorias son trazadas por procesos iterativos que se mueven de una solución a otra dependiendo del criterio de aceptación particular de la metaheurística utilizada. Esta clase de metaheurísticas se enfocan principalmente en la explotación del espacio de soluciones. Entre ellas se encuentra la búsqueda local [Tal09, AL03], el recocido simulado [Tal09, KGV83], la búsqueda tabú [Tal09, Glo89], Búsqueda Local Iterada (ILS) [LMS03], Búsqueda de Vecindad Variable (VNS) [MH97], Búsqueda Local Guiada (GLS) [Vou98], GRASP [FR95], entre otros. De especial importancia está la búsqueda local, la cual juega un papel importante en la mayoría de las metaheurísticas de trayectoria y en algunas metaheurísticas poblacionales:

1.3.2. Metaheurísticas basadas en una población

Estas metaheurísticas empiezan con una población inicial de soluciones, que puede ser elegida de manera aleatoria o con heurísticas que introduzcan “buenas” soluciones, e iterativamente generan nuevos elementos que pueden llegar a suplantar los de la población actual según un criterio de selección. El proceso de generación y selección se repite hasta que se cumpla un criterio de parada, el cual puede ser un número

de iteraciones fijas o hasta que la población converga a una región sin mejoras pasado un número de iteraciones. Dichos procesos de generación y selección pueden ser sin memoria, es decir, solo dependen de la población actual, como el caso de los algoritmos genéticos tradicionales o pueden ser con memoria y usar información adquirida durante el proceso de búsqueda para dirigir la generación y selección a mejores resultados.[Tal09].

Entre las metaheurísticas basadas en una población se encuentran: *Scatter Search* [Tal09, Glo77], colonia de hormigas [Tal09, Dor92], optimización de enjambre de partículas [Tal09, ESK01], algoritmos de estimación de distribución [Tal09, LLIB06], Evolución Diferencial [Tal09, PSL06], Algoritmos evolutivos [Tal09], entre otros.

1.3.2.1. Algoritmos evolutivos

Los algoritmos evolutivos están basados en la competencia entre individuos de una población llamados cromosomas; la población se inicializa con cromosomas elegidos aleatoriamente o a través de heurísticas. Con esto, dado una función objetivo, se evalúa cuán bueno es cada cromosoma y con esta información se decide por medio de un proceso de selección cuáles serán los cromosomas que se van a cruzar, dando como resultado uno o más hijos que comparten características de sus padres. Luego del cruce, viene la mutación de los nuevos cromosomas con un operador definido que perturba ligeramente al cromosoma. Por último viene un proceso de reemplazo donde se decide si los hijos suplantán algún elemento de la población (esquema estacionario) o si se construye una nueva población con los hijos que va a suplantar totalmente a sus padres (esquema generacional) [Tal09].

El diseño de un algoritmo evolutivo viene dado con la toma de decisiones respecto a algunos componentes. Algunos comunes a todas las metaheurísticas como la representación del problema, el cual puede ser un vector de valores binarios, enteros, reales, una permutación, entre otros; la inicialización de la población, que puede ser por medio de heurísticas o aleatoria y la elección de una función objetivo que represente cuán buena es un cromosoma y el criterio de parada. Por otro lado, hay unos componentes

que son propios de los algoritmos evolutivos como el criterio de selección de cromosomas para reproducirse, el operador de cruce, el operador de mutación y la estrategia de reemplazo.

Los algoritmos evolutivos que fueron implementados en este trabajo fueron:

1.3.2.1.1 Algoritmo Genético Generacional (GGA)

Generational Genetic Algorithm (GGA) en inglés, es el esquema tradicional de algoritmos genéticos; los algoritmos genéticos fueron desarrollados por *Holland, H.* en [Hol75]. La versión generacional usa una estrategia de reemplazo en la cual se genera una población nueva de hijos en cada ciclo del algoritmo y ésta suplanta a la generación anterior. El algoritmo comienza con una población inicial aleatoria y va creando poblaciones nuevas en cada generación hasta que se cumpla una condición de parada. En medio del proceso está actuando un operador de cruce que mezcla los cromosomas seleccionados como padres y una operación de mutación que modifica algunos cromosomas de la nueva generación. GGA se presenta en el algoritmo 1.4 [Ale14].

Algoritmo 1.4 Algoritmo Genético Generacional

Input: *pop* tamaño de la población, *cp* probabilidad de cruce, *mp* probabilidad de mutación

Output: La solución al problema

- 1: $P \leftarrow$ Generar población aleatoria de *pop* cromosomas
 - 2: $s^* \leftarrow$ el *mejor* individuo en P
 - 3: **while** \neg Condición de parada **do**
 - 4: $P' \leftarrow \emptyset$
 - 5: **while** $|P'| < \text{pop}$ **do**
 - 6: $p_1 \leftarrow$ Seleccionar un cromosoma en P
 - 7: $p_2 \leftarrow$ Seleccionar un cromosoma en P
 - 8: $c_1, c_2 \leftarrow$ recombinar p_1 y p_2 con probabilidad *cp*
 - 9: Mutar c_1 y c_2 con probabilidad *mp*
 - 10: $P' \leftarrow P' \cup \{c_1, c_2\}$
 - 11: $P \leftarrow P'$
 - 12: **if** El *mejor* cromosoma en P es *mejor* que s^* **then**
 - 13: $s^* \leftarrow$ el *mejor* cromosoma en P
 - 14: **return** s^*
-

1.3.2.1.2 Algoritmo Genético Estacionario (SSGA)

Steady State Genetic Algorithm (SSGA) en inglés, es otra variación de los algoritmos genéticos. En este caso, la estrategia de reemplazo consiste en generar uno o dos hijos por iteración y decidir al momento si va a suplantarse algún elemento de la población; puede suplantarse a uno de los padres si es mejor que uno de ellos o puede suplantarse al peor elemento de la población. Al igual que GGA, se tiene que definir un operador de mutación y cruce. SSGA se presenta en el algoritmo 1.5 [Ale14].

Algoritmo 1.5 Algoritmo Genético Estacionario

Input: `pop` tamaño de la población, `cp` probabilidad de cruce, `mp` probabilidad de mutación

Output: La solución al problema

- 1: $P \leftarrow$ Generar población aleatoria de `pop` cromosomas
 - 2: $s^* \leftarrow$ el *mejor* cromosoma en P
 - 3: **while** \neg Condición de parada **do**
 - 4: $p_1 \leftarrow$ Seleccionar un cromosoma en P
 - 5: $p_2 \leftarrow$ Seleccionar un cromosoma en P
 - 6: $c_1, c_2 \leftarrow$ recombinar p_1 y p_2 con probabilidad `cp`
 - 7: Mutar c_1 y c_2 con probabilidad `mp`
 - 8: Seguir algún criterio de reemplazo de cromosomas en P por c_1 y c_2
 - 9: **if** El *mejor* cromosoma en P es *mejor* que s^* **then**
 - 10: $s^* \leftarrow$ el *mejor* cromosoma en P
 - 11: **return** s^*
-

1.3.2.1.3 Algoritmo Memético (MA)

Memetic Algorithm en inglés, es un algoritmo evolutivo basado en los algoritmos genéticos que tiene la peculiaridad de tener un proceso de optimización interno llamado “meme”, el cual es aplicado a todos o algunos cromosomas de la población en cada iteración; el meme más común es una búsqueda local [NC12]. El esquema clásico se basa en los GGA y primero genera una población nueva con los cruces y mutaciones propios de un GGA, para luego pasar a una fase de intensificación donde aplica el meme a todas las soluciones y se genera una nueva población optimizada que suplanta la generación anterior. Otro esquema se basa en los SSGA y en cada iteración se cruzan una serie de padres para generar uno o dos hijos que, luego de mutar con cierta

probabilidad dada, se decide si pasan a un proceso de optimización con el meme y el resultado se decide si se incorpora a la población. En el algoritmo 1.6 se presenta la versión estacionaria de los algoritmos meméticos (SSMA).

Algoritmo 1.6 Algoritmo Memético Estacionario

Input: *pop* tamaño de la población, *cp* probabilidad de cruce, *mp* probabilidad de mutación, *mem* meme usado

Output: La solución al problema

- 1: $P \leftarrow$ Generar población de *pop* cromosomas
 - 2: $s^* \leftarrow$ el *mejor* cromosoma en P
 - 3: **while** \neg Condición de parada **do**
 - 4: $p_1 \leftarrow$ Seleccionar un cromosoma en P
 - 5: $p_2 \leftarrow$ Seleccionar un cromosoma en P
 - 6: $c_1, c_2 \leftarrow$ recombinar p_1 y p_2 con probabilidad *cp*
 - 7: Mutar c_1 y c_2 con probabilidad *mp*
 - 8: Determinar si c_1 y c_2 van a ser optimizados con *mem* y almacenar el resultado en c'_1 y c'_2
 - 9: Seguir algún criterio de reemplazo de cromosomas en P por c'_1 y c'_2
 - 10: **if** El *mejor* cromosoma en P es *mejor* que s^* **then**
 - 11: $s^* \leftarrow$ el *mejor* cromosoma en P
 - 12: **return** s^*
-

1.3.2.1.4 CHC Adaptative Search Algorithm

Propuesto inicialmente por *Eshelman, L.* en [Esh91], es un algoritmo evolutivo generacional con la diferencia de que es totalmente elitista, ya que elige los mejores n elementos de entre la vieja y nueva población para conformar la nueva generación (n es el número de cromosomas en la población). También tiene la particularidad de que implementa un operador de cruce llamado HUX en el cual, dado dos padres, intercambia la mitad de los genes que no coincidan entre ellos de manera aleatoria con el fin de crear hijos lo más distinto posible de los padres. Además CHC tiene un mecanismo de prevención de incesto en el cual se usa la distancia de Hamming entre los dos posibles candidatos a ser padres para determinar si son lo suficientemente distintos para cruzarse, para esto usa un umbral que inicialmente es $L/4$ donde L es la longitud del cromosoma. Por último, no existe una operación de mutación y en cambio, cuando pasa una generación sin cromosomas nuevos, se disminuye el umbral de incesto en 1, hasta que llega a 0 y se toma la decisión de reinicializar la población, preservando el

mejor cromosoma encontrado hasta el momento y poblando los cromosomas restantes con variaciones del mejor, donde se perturban hasta un 35 % de los genes asociados al cromosoma. CHC se presenta en el algoritmo 1.7, donde t es la generación actual, d es el umbral de incesto, $P(t)$ es la población de la generación t , L es la longitud del cromosoma.

Algoritmo 1.7 CHC

Input: pop tamaño de la población

Output: La solución al problema

```

1:  $t = 0$ 
2:  $d = L/4$ 
3:  $P(t) \leftarrow$  Generar población de pop cromosomas
4:  $s^* \leftarrow$  el mejor cromosomas en  $P$ 
5: while  $\neg$  condición de parada do
6:    $t = t + 1$ 
7:    $C(t) \leftarrow P(t - 1)$ 
8:   recombinar los cromosomas en  $C(t)$  para formar  $C'(t)$ 
9:   evaluar los cromosomas en  $C'(t)$  con la función objetivo
10:  seleccionar  $P(t)$  de  $C'(t)$  y  $P(t-1)$  sólo con los mejores cromosomas
11:  if El mejor cromosoma en  $P$  es mejor que  $s^*$  then
12:     $s^* \leftarrow$  el mejor cromosoma en  $P$ 
13:  if  $P(t) = P(t - 1)$  then
14:     $d \leftarrow d - 1$ 
15:  if  $d < 0$  then
16:    Reinicializar  $P(t)$ 
17: return  $s^*$ 

```

Algoritmo 1.8 Recombinar

Input: $C(t)$ candidatos a padre, d umbral de incesto

Output: $C'(t)$ hijos

```

for all par de instancias en  $C(t)$   $x_1$  y  $x_2$  do
   $ham \leftarrow$  distancia de hamming entre  $x_1$  y  $x_2$ 
  if  $ham/2 > d$  then
    cambiar la mitad de elementos que difieran entre  $x_1$  y  $x_2$  de forma aleatoria
    para generar  $x'_1$  y  $x'_2$ 
     $C'(t) \leftarrow C'(t) \cup \{x'_1, x'_2\}$ 
  else
    borrar el par  $x_1$  y  $x_2$  de  $C(t)$ 
return  $C'(t)$ 

```

Algoritmo 1.9 Reinicializar

Input: $P(t-1)$ población anterior, s^* mejor solución, r porcentaje de genes a cambiar, d umbral de incesto

Output: $P(t)$ población renovada

Llenar $P(t)$ con copias de s^*

for all miembros $x_i \in P(t)$ excepto uno **do**

 Cambiar $r * L$ genes de manera aleatoria de x_i

 Evaluar x_i con la función objetivo

$d = L/4$

return $P(t)$

Capítulo 2

Marco metodológico

En este capítulo se detalla la representación utilizada para los cromosomas, la función objetivo, las adaptaciones particulares que se hizo a cada algoritmo evolutivo usado en el experimento, el proceso de validación cruzada, la técnica de estratificación, se presenta los conjuntos de datos usados para el experimento y se explica el método de entonación utilizado para ajustar los algoritmos evolutivos.

2.1. Representación del cromosoma

Sea T el conjunto de instancias a reducir de tamaño n , la representación usada para modelar el problema de selección de prototipos es el de un mapa de bits de tamaño n , donde cada bit representa una instancia $t_i \in T$; si el valor del bit i es 1, entonces la instancia $t_i \in S$, donde S es el conjunto reducido; si el bit i es 0, $t_i \notin S$. En este sentido, el conjunto S representado por el mapa de bits M se define como en la ecuación (2.1). Además un ejemplo se presenta en 2.1. Cabe destacar que en los algoritmos evolutivos, los mapas de bits se conocen como cromosomas y cada bit como gen.

$$S = \{t_i \in T \mid i = 1 \dots n \wedge m_i = 1 \wedge m_i \in M\} \quad (2.1)$$

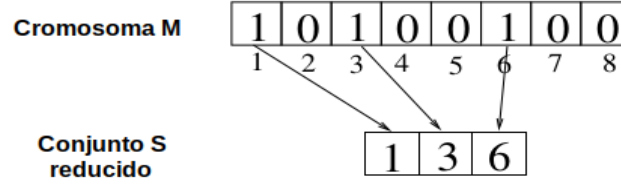


FIGURA 2.1: Representación de un cromosoma y su respectivo conjunto reducido

2.2. Función objetivo

Se necesita una función con la cual los algoritmos evolutivos puedan evaluar cuán buena es una solución dada, además de que dicha función debe permitir establecer una relación de orden entre las soluciones con el fin de decidir cuál cromosoma es mejor que otro. Como se explicó anteriormente, los algoritmos evolutivos buscan aproximarse al óptimo global, que en este caso es el conjunto reducido S con menor cardinalidad y mayor precisión en la clasificación de instancias nuevas. Es por eso que se adopta una función objetivo derivada del trabajo de *Cano, J.* en [dA04], la cual se presenta a continuación:

$$\mathcal{F}(S) = \alpha * error(S) + (1 - \alpha) * reduccion(S) \quad (2.2)$$

Donde $\mathcal{F} : S \rightarrow \mathbb{R}$ es la función objetivo, S es el conjunto reducido a evaluar, α es un parámetro que controla cuánta importancia se le da al error asociado a S con respecto a la tasa de reducción del segundo término de la ecuación (2.2), TR es el conjunto de entrenamiento original del cual se realizó la reducción, error(S) es el porcentaje de error al clasificar un conjunto de prueba TS usando 1-NN con S como conjunto de entrenamiento y reducción(S) es el porcentaje de instancias restantes en S en relación al conjunto original TR. El α usado es 0.5 como lo establecen en [dA04] para darle la misma importancia a la reducción de datos como a mantener bajo los porcentajes de error en la clasificación.

Dado esta función objetivo, la meta de todas las metaheurísticas implementadas se vuelve minimizar $\mathcal{F}(S)$, lo cual quiere decir que se busca tanto reducir $|S|$, como reducir error(S). Un conjunto S_i es mejor que un conjunto S_j si $\mathcal{F}(S_i) < \mathcal{F}(S_j)$.

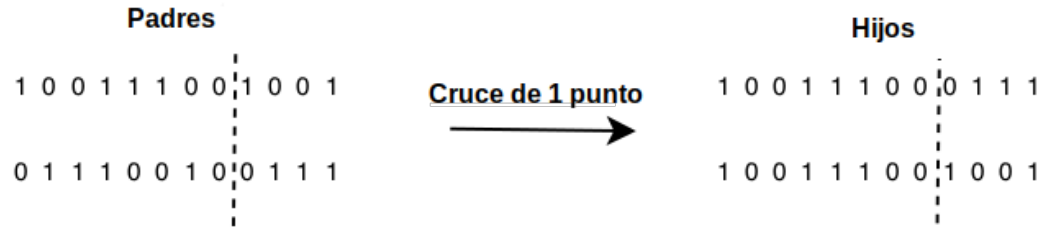


FIGURA 2.2: Cruce de un punto

2.3. Adaptaciones de los algoritmos evolutivos

Para aplicar los distintos algoritmos evolutivos implementados para este trabajo, es necesario determinar los operadores de cruce, mutación, el método de selección de los cromosomas que van a cruzarse, el criterio de selección de los cromosomas sobrevivientes y en caso del algoritmo memético el proceso de optimización interno (también conocido como meme) utilizado.

Para el caso del algoritmo genético estacionario y el algoritmo memético, se eligió como método de selección de cromosomas a cruzarse un proceso de torneo [Tal09], el cual consiste en seleccionar k cromosomas de manera aleatoria y elegir el mejor de los k . CHC por su parte elige dos cromosomas aleatorios y utiliza su mecanismo de prevención de incesto para elegir a los padres. El algoritmo genético generacional simplemente elige dos elementos aleatorios para realizar el cruce.

El operador de cruce utilizado en GGA, SSGA y MA es el cruce de un punto [Tal09], el cual consiste en definir un punto μ en el cual se va a dividir los dos cromosomas seleccionados como padres y luego se forman dos hijos a partir de la mezcla de las partes de los padres. CHC en cambio usa el operador HUX explicado anteriormente. En la figura 2.2 se muestra un ejemplo del cruce de un punto.

El operador de mutación para GGA, SSGA y MA consta de cambiar 5% de los genes del cromosoma de manera aleatoria. Se elige 5% basado en [Ale14] para que la mutación represente una variación en S , ya que si sólo se cambia un gen, el conjunto mutado sería para los efectos de la optimización casi idéntico al original. Sin embargo la probabilidad de que un cromosoma dado mute es baja, basado principalmente en los

resultados de *Cano, J.* en [dA04] donde obtienen mejores resultados experimentales con bajas probabilidades de mutación (menor al 1 % por cromosoma), justificándose en que con mayores valores, la búsqueda podría degenerar en una búsqueda aleatoria. CHC por su parte no tiene mutación.

El criterio de reemplazo para GGA es generar una población nueva de hijos P_i que va a suplantarse la generación anterior P_{i-1} excepto el mejor elemento en P_{i-1} . Por su parte, el criterio de reemplazo de SSGA es que dado dos padres y los dos hijos producidos por el operador de cruce, se eligen los 2 mejores cromosomas para permanecer dentro de la población. MA, en cambio usa un criterio de reemplazo en el cual los 2 hijos suplantados a los 2 peores elementos de la población y CHC se queda con los n mejores cromosomas entre P_i y P_{i-1} , ambos casos son totalmente elitistas.

El algoritmo memético es el que más adaptaciones tiene para adecuarse a PS, se usa una adaptación realizada por *Cano, J. et al.* en [GCH08]. Se basa en el algoritmo memético estacionario presentado anteriormente, con la peculiaridad de que para decidir si los hijos producidos en una iteración van a ser optimizados con el meme, se usa un parámetro P_{LS} que se determina de la siguiente forma:

$$P_{LS} = \begin{cases} 1 & \text{si } \mathcal{F}(S_{nuevo}) < \mathcal{F}(S_{peor}) \\ 0.0625 & \text{en caso contrario} \end{cases} \quad (2.3)$$

Donde \mathcal{F} es la función objetivo, S_{nuevo} es el conjunto reducido representado por uno de los cromosomas hijos y S_{peor} es el conjunto reducido representado por el peor cromosoma de la población. Es así como P_{LS} representa la probabilidad con la cual se va a decidir si se optimiza el cromosoma hijo; P_{LS} debe ser calculado para cada hijo creado en el cruce. La idea es que si el hijo es mejor que el peor cromosoma de la población, entonces vale la pena optimizarlo; en cambio, si es peor, se le da una probabilidad de optimización de 6,25 %.

El meme usado en MA es el que se presenta en el algoritmo 2.1. El procedimiento consiste en ir reduciendo progresivamente las instancias que se encuentran en el conjunto S , representado por el cromosoma M , sin que se pierda la precisión asociada a S . Para esto, se usa una lista U del primer vecino más cercano de cada gen en M ,

una lista R que contiene los genes que ya han sido puestos en 0 y que no generan una ganancia mayor al umbral de aceptación \mathfrak{t} , $\text{clase}(i)$ es la clase asociada a la instancia representada por el gen i del cromosoma M , **ganancia** representa cuánto mejora (en caso de que sea positiva) o cuánto empeora (en caso de ser negativa) la solución dada por el cromosoma M luego de cambiar un gen, fitness_M es el valor de evaluar la función objetivo con el cromosoma M y $\text{fitness}_{\text{ganancia}}$ se define como en la ecuación (2.4), donde L es el largo del cromosoma:

$$\text{fitness}_{\text{ganancia}} = \frac{\frac{\text{ganancia}}{L} * 100 + \frac{100}{L}}{2} \quad (2.4)$$

El meme intenta remover una instancia de S en cada iteración pra ver si la precisión mejora, se mantiene igual o empeora. Si la ganancia es positiva y está por encima del umbral de aceptación entonces se preservan los cambios; en cambio, si la ganancia está por debajo del umbral, entonces se vuelve a incluir la instancia eliminada y se etiqueta su respectivo gen como revisado.

2.4. Criterios para comparar los métodos de selección de prototipos

Al momento de comparar los distintos métodos de PS, se usan los siguiente criterios para evaluar las fortalezas y debilidades relativas de cada algoritmo [GLH16]:

- **Reducción:** se mide como la proporción existente entre la cardinalidad del conjunto reducido S entre el conjunto de entrenamiento; esto es $|S|/TR$. La reducción de las instancias trae consigo una disminución en los tiempos de cómputo al tener que revisar menos cromosomas en cada iteración para clasificar una nueva instancia.
- **Precisión de la clasificación:** se espera que aún con el conjunto reducido, se mantenga las tasas de acierto del clasificador o inclusive, mejoren. Un algoritmo

Algoritmo 2.1 Meme**Input:** M cromosoma a optimizar, τ umbral de aceptación**Output:** M cromosoma optimizado

```

1: Sea  $M = \{m_1, m_2, \dots, m_n\}$  el cromosoma a optimizar
2:  $R \leftarrow \emptyset$ 
3:  $U = \{u_1, u_2, \dots, u_n\}$  la lista de vecinos asociados, donde  $u_i$  es el vecino más cercano
   del gen  $i$ .
4: while  $(\exists m_i \in M \mid m_i = 1 \wedge i \notin R)$  do
5:   elegir  $j$  aleatoriamente de  $M$  tal que  $m_j = 1 \wedge j \notin R$ 
6:    $ganancia \leftarrow 0$ 
7:    $m_j \leftarrow 0$ 
8:   Copiar  $U$  a  $U'$ 
9:   for all  $u_i \in U \mid u_i = j$  do
10:     $u_i \leftarrow$  nuevo vecino más cercano con el nuevo  $M$ 
11:    if  $clase(i) = clase(u'_i) \wedge clase(i) \neq clase(u_i)$  then
12:       $ganancia \leftarrow ganancia - 1$ 
13:    else if  $clase(i) \neq clase(u'_i) \wedge clase(i) = clase(u_i)$  then
14:       $ganancia \leftarrow ganancia + 1$ 
15:    if  $ganancia \geq \tau$  then
16:       $fitness_M \leftarrow fitness_M + fitness_{ganancia}$ 
17:       $R \leftarrow \emptyset$ 
18:    else
19:      Recuperar  $U$  de  $U'$ 
20:       $m_j \leftarrow 1$ 
21:       $R \leftarrow R \cup j$ 
22: return  $M$ 

```

de PS debe poder mantener la precisión al momento de ser evaluado con el conjunto de prueba. La precisión se calcula dividiendo el número de clasificaciones hechas correctamente entre el total de clasificaciones.

- **Tiempo de cómputo:** involucra cuánto tiempo le lleva al algoritmo realizar la reducción de los datos, un factor importante al momento de escalar los métodos a conjuntos muy grandes. En este trabajo el tiempo de cómputo se mide en segundos.
- **Cohen's Kappa:** es una métrica que originalmente mide el nivel de acuerdo o desacuerdo entre dos clasificadores. Sin embargo se han hecho adaptaciones de esta métrica para ser usada por un sólo clasificador [GDCH12], ya que es más robusta que la precisión por tomar en cuenta la posibilidad de que una

clasificación sea hecha aleatoriamente. Esta métrica sirve para verificar si el clasificador está etiquetando las instancias correctamente de manera consistente o de una manera inestable con muchas decisiones aleatorias. *Cohen's kappa* se calcula a partir de la matriz de confusión como se muestra en la ecuación (1.1). Donde y_{ii} es el conteo de las celdas de la diagonal principal, N el número de instancias revisadas, Ω el número de clases presentes, $y_{i\cdot}$ es la suma de las celdas de la fila i y $Y_{\cdot i}$ es la suma de las celdas de la columna i .

$$kappa = \frac{N * \sum_{i=1}^{\Omega} y_{ii} - \sum_{i=1}^{\Omega} y_{i\cdot} * y_{\cdot i}}{N^2 - \sum_{i=1}^{\Omega} y_{i\cdot} * y_{\cdot i}} \quad (2.5)$$

2.5. Conjunto de datos

Los conjuntos de datos utilizados para validar el experimento provienen de *UCI Machine Learning Repository* [DKT17] y *KEEL Data-Mining Software Tool* [AFFL⁺11]. Se hace una separación como la establecida en [dA04] donde se considera como conjunto de datos pequeños aquellos con menos de 2000 instancias, los conjuntos medianos los que poseen entre 2000 y 20000 instancias y los conjuntos grandes aquellos con más de 20000 instancias. En la tabla 2.1 se detallan los conjuntos pequeños, en 2.2 los medianos y en 2.3 los grandes. Solo se eligió conjunto de datos numéricos para poder utilizar la distancia euclídea con 1-NN sin tener problemas en la preservación de información al convertir datos categóricos a numéricos.

2.6. Validación cruzada y estratificación

Dado un conjunto de datos T , el proceso de validación cruzada [K⁺95] consta de dividir T en k subconjuntos disjuntos T_1, T_2, \dots, T_k de aproximadamente el mismo tamaño, donde cada subconjunto mantiene la distribución de las clases presente en T . Luego se procede a probar el clasificador M , que en este caso es 1-NN, k veces, donde en cada prueba $t \in \{1, 2, \dots, k\}$ se utiliza como conjunto de entrenamiento $TR = T \setminus T_t$, se aplica el algoritmo de selección de prototipos a TR y el conjunto

CONJUNTO	INSTANCIAS	ATRIBUTOS	CLASES
Iris	150	4	3
Cleveland	297	13	5
Led7Digit	500	7	10
Pima	768	8	2
WDBC	569	30	2
Monk-2	432	6	2
Wisconsin	683	9	2
Wine	178	13	3
Glass	214	9	7
Banknote	1372	5	2
Appendicitis	106	7	2
Balance	625	4	3
Bands	539	19	2
Contraceptive	1473	9	3
Dermatology	366	34	6
Ecoli	336	7	8
Haberman	306	3	2
Hayes-roth	160	4	3
Heart	270	13	2
Hepatitis	155	19	2
Mammographic	961	5	2
Newthyroid	215	5	3
Tae	151	5	3
Vehicle	846	18	4
Vowel	990	13	11
Yeast	1484	8	10

TABLA 2.1: Conjuntos de datos pequeños

resultante S se valida usando $TS = T_t$ como conjunto de prueba. El porcentaje de aciertos del clasificador se calcula como el promedio de las k pruebas realizadas; pero como las metaheurísticas tienen un componente estocástico, se necesita repetir cada prueba varias veces. En la figura 2.3 se muestra un esquema de cómo se aplica la validación cruzada para el problema de selección de prototipos dado que la partición k es seleccionada como conjunto de prueba.

Por otra parte está la estratificación, la cual es una técnica propuesta por *Cano, J. et al.* en [CHL05] para solventar el problema de aplicación de los algoritmos de PS a conjunto de datos muy grandes. Dicho problema viene dado porque la mayoría de los algoritmos de PS y metaheurísticas utilizadas son $O(n^2)$, siendo n la cantidad de instancias del conjunto a procesar, y por lo tanto, para grandes volúmenes de

CONJUNTO	INSTANCIAS	ATRIBUTOS	CLASES
Banana	5300	2	2
Cardiotocography	2126	23	3
Eye-state	14980	15	2
Page-blocks	5473	10	5
Penbased	10992	16	10
Satimage	6435	36	7
Thyroid	7200	21	3
Segment	2310	19	7
Coil2000	9822	85	2
Magic	19020	10	2
Marketing	8993	13	9
Phoneme	5404	5	5
Ring	7400	20	2
Spambase	4597	57	2
Texture	5500	40	11
Titanic	2201	3	2
Twonorm	7400	20	2

TABLA 2.2: Conjuntos de datos medianos

CONJUNTO	INSTANCIAS	ATRIBUTOS	CLASES
Credit-card	30000	24	2
Shuttle	58000	9	7

TABLA 2.3: Conjuntos de datos grandes

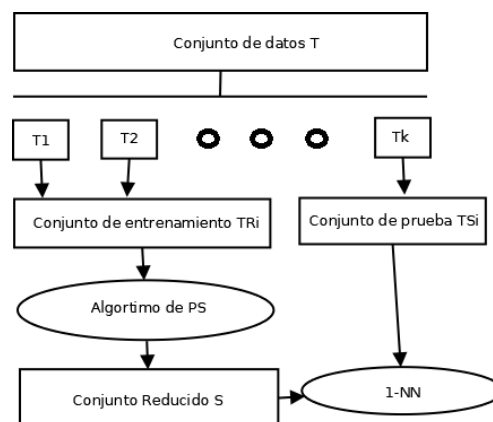


FIGURA 2.3: Validación cruzada

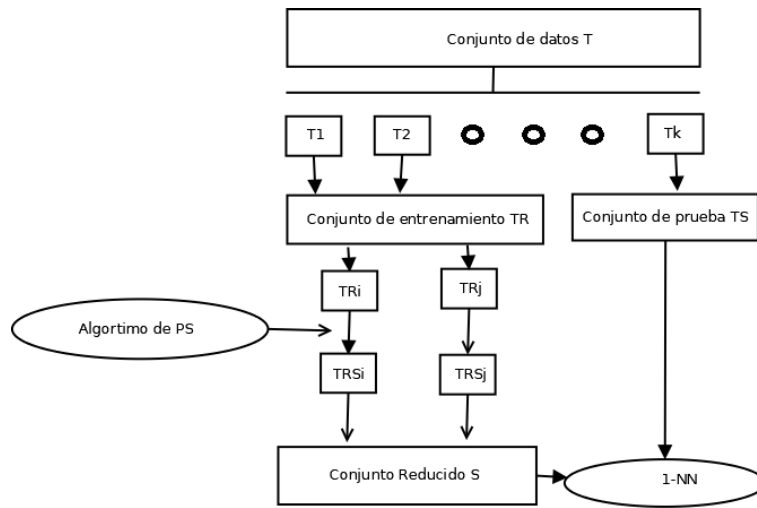


FIGURA 2.4: Estratificación

datos estos algoritmos empiezan a tardar mucho en calcular una solución, lo cual los vuelve poco útiles al momento de hacer preprocesamiento de datos. Es así que la estratificación se adopta como una técnica que lleva a tiempos aceptables el cómputo con conjuntos de muchas instancias.

Dado un conjunto de datos TR producto de la división hecha por un proceso de validación cruzada, la estratificación empieza dividiendo TR en k subconjuntos disjuntos, llamados estratos, TR_1, TR_2, \dots, TR_k de aproximadamente el mismo tamaño. Luego, se aplica el algoritmo de PS directamente a cada uno de los k subconjuntos seleccionados TR_i para el entrenamiento, formando entonces subconjuntos reducidos $TR_{S1}, TR_{S2}, \dots, TR_{Sk}$; acto seguido, se juntan todos los TR_{Si} para formar el conjunto reducido S que va a ser usado por 1-NN para clasificar TS . La estratificación prueba ser un método efectivo, como lo demuestran en [CHL05], ya que reduce la cantidad de instancias que debe tratar el algoritmo de PS a TR/k , por lo que la elección del número de estratos k se vuelve de especial importancia. En la figura 2.4 se muestra un esquema de cómo se aplica la estratificación, donde el estrato k es seleccionado como conjunto de prueba.

2.7. Entonación de las metaheurísticas

Para la entonación de las metaheurísticas se usó *irace* [LIDLC⁺16], el cual es un paquete de R que implementa el método de entonación automática conocido como *iterated F-race*. La esencia de *iterated F-race* es hacer varias carreras, las cuales son procesos internos del algoritmo que ponen a competir un conjunto de vectores de parámetros (también conocidos como configuraciones) con un repertorio de problemas, con el fin de recopilar datos referentes al desempeño de cada configuración, para poder hacer pruebas estadísticas que determinarán las configuraciones que van a ser elegidas como las mejores y además descartar las peores en una etapa temprana de la carrera.

Iterated F-race en específico trata de un método que consiste en tres pasos: (1) elegir varias configuraciones posibles de acuerdo a una distribución en particular y así formar una población, (2) elegir una población de configuraciones (3) actualizar la distribución de la población de tal manera que sea sesgado a favor de las mejores configuraciones. Estos tres pasos son repetidos hasta que pase un número de iteraciones dadas por el usuario. El algoritmo de *iterated F-race* se presenta en 2.2.

Algoritmo 2.2 IRACE

Input: I conjunto de instancias del problema a entonar, X espacio de configuraciones, U función de utilidad, B número de iteraciones con el que se cuenta

Output: θ^{elite} mejores configuraciones encontradas

- 1: $\theta_1 \leftarrow$ una muestra uniforme de X
 - 2: $\theta^{elite} \leftarrow$ resultado de una **carrera** usando θ_1 como población y con número de iteraciones B_1
 - 3: $j \leftarrow 1$
 - 4: **while** $B^{usado} \leq B$ **do**
 - 5: $j \leftarrow j + 1$
 - 6: $\theta^{nueva} \leftarrow$ muestra sesgada hacia θ^{elite} de X
 - 7: $\theta_j \leftarrow \theta^{nueva} \cup \theta^{elite}$
 - 8: $\theta^{elite} \leftarrow$ resultado de una **carrera** usando θ_j como población y con presupuesto B_j
 - 9: **return** θ^{elite}
-

Cabe acotar que *irace* asume que el problema sobre el cual se va a entonar es un problema de minimización. Por lo tanto, siempre se busca menores valores de la función de utilidad U para cada configuración; además, luego de cada carrera asigna

un rango r_z dependiendo de cómo se compare la configuración z con respecto al resto, a menor rango, mejor es la configuración, por lo tanto, el conjunto de elites θ^{elite} está conformado por los k elementos con menor rango. El número k es calculado al principio de la corrida de *irace* y en este trabajo se deja su cálculo automático por defecto.

La prueba estadística utilizada por la carrera para ir eliminando las peores configuraciones y preservando las elites es la prueba Friedman no paramétrica o la prueba *t-test*. Por recomendación de los autores se usa una prueba *t-test* con un valor significativo de 0.05 ya que, según sus criterios, es la más adecuada para la entonación de parámetros de valores continuos.

Irace, además, implementa un método de reinicialización cuando la población de configuraciones converge prematuramente. En este proceso se mantienen las configuraciones elites de la última carrera y empieza de nuevo según ciertas consideraciones. Aunado a esto, *irace* también implementa un sistema de carrera elitista, el cual evita que las mejores configuraciones encontradas hasta el momento se pierdan en una carrera producto de una serie desfavorable de evaluaciones en un momento dado. En este trabajo se usa ambas funciones como configuración por defecto que tiene *irace*. Para más información de todas las funciones y utilidades que presenta esta herramienta, se recomienda leer [LIDLC⁺16].

Capítulo 3

Evaluación experimental

3.1. Diseño experimental

En este capítulo se describe todas las decisiones tomadas con respecto a la experimentación; esto incluye los parámetros usados para la entonación, las particiones hechas en la validación cruzada y la estratificación, el diseño de los experimentos combinando las heurísticas con las metaheurísticas y finalmente se presentan los resultados de cada prueba realizada.

Lo primero que se hace es entonar los algoritmos evolutivos para que devuelvan el mejor valor posible al probarse con los conjuntos de datos expuestos en la sección 2.4. Para lograr la entonación, se usa *irace*, expuesto en la sección 2.6, con los parámetros de la tabla 3.1. Se consigue 1 configuración de parámetros para los problemas pequeños, medianos y grandes respectivamente.

Parámetros	<i>irace</i>
Iteraciones	1000, 400 y 100
Número decimales significativos	4
Prueba estadística	t-test
Nivel de confianza para prueba estadística	0.95
Frecuencia de la prueba estadística	1 iteración
Número de configuraciones elites	automática
Reinicialización por convergencia prematura	Sí
Modo elitista	Sí

TABLA 3.1: Parámetros usados para *irace*

Parámetros	Tipo de dato	Rangos
Iteraciones sin mejoras	entero	[10,1000]
Población	entero	[10,150]
Probabilidad de cruce	real	[0,1]
Probabilidad de mutación	real	[0,0.01]
Número del torneo	entero	[1,10]

TABLA 3.2: Rangos usados para los parámetros en la entonación

PARÁMETROS	ALGORITMOS			
	GGA	SGA	MA	CHC
Iteraciones fijas	1000	1000	1000	1000
Iteraciones sin mejoría	215	682	516	134
Población	70	90	21	33
Prob. de Cruce	0.4837	0.9848	0.9496	-
Prob. de Mutación	0.0001	0.0057	0.0071	-
Número del torneo	-	3	1	-

TABLA 3.3: Parámetros usados para los conjuntos pequeños

PARÁMETROS	ALGORITMOS			
	GGA	SGA	MA	CHC
Iteraciones fijas	1000	1000	1000	1000
Iteraciones sin mejoría	574	973	393	263
Población	88	132	32	27
Prob. de Cruce	0.5779	0.9859	0.9549	-
Prob. de Mutación	0.0001	0.0001	0.0004	-
Número del torneo	-	1	3	-

TABLA 3.4: Parámetros usados para los conjuntos medianos

Los parámetros a entonar son: condición de parada, cardinalidad de la población, probabilidad de cruce, probabilidad de mutación y número del torneo. Los rangos válidos para cada parámetro se presentan en la tabla 3.2. La elección de los rangos se hace en base a los trabajos [dA04, dA04, GDCH12, GCH08, Tal09]. Los resultados de la entonación se presentan en las tablas 3.3, 3.4 y 3.5.

Para la validación cruzada se usa $k = 10$ y se repite cada prueba 3 veces basándose en el trabajo de *Cano, J.* en [dA04]. Este esquema de validación cruzada se aplica a los conjuntos de tamaño pequeño como lo hacen en [dA04]. Para la estratificación se adopta $k = 10$ para los conjuntos medianos y $k = 50$ para los conjuntos grandes, tal y como se determinan en [CHL05], cuya idea es hacer que el algoritmo de PS no trabaje con más de 2000 instancias por estrato para reducir la cantidad a un conjunto

PARÁMETROS	ALGORITMOS			
	GGA	SGA	MA	CHC
Iteraciones fijas	1000	1000	1000	1000
Iteraciones sin mejoría	530	593	514	221
Población	102	122	35	37
Prob. de Cruce	0.5158	0.9554	0.9698	-
Prob. de Mutación	0.0001	0.0078	0.0049	-
Número del torneo	-	7	3	-

TABLA 3.5: Parámetros usados para los conjuntos grandes

de tamaño pequeño según la clasificación anteriormente expuesta. Además, al igual que en la validación cruzada, las metaheurísticas utilizadas son estocásticas y por lo tanto, cada una de las k pruebas realizadas se repite 3 veces, regresando el promedio de todas las pruebas realizadas como resultado.

Una vez obtenido las distintas configuraciones para cada algoritmo evolutivo. Se procede con el experimento principal, el cual consta de obtener los resultados de reducción, precisión y kappa evaluados en el conjunto de entrenamiento y en el conjunto de prueba y tiempo en segundos de cada heurística y metaheurística. Primero se obtienen los resultados de CNN, RSS y ENN por sí solas. Luego el experimento se divide en dos modalidades, la primera consta de probar las metaheurísticas utilizando como criterio de parada el paso de un número de iteraciones sin mejorías, de tal manera que cada metaheurística tenga la posibilidad de converger a una solución sin restricciones de tiempo, a esta fase se le nombra como modalidad A; por otra parte la modalidad B prueba las metaheurísticas fijando el número de iteraciones en el cual corren, esto con el sentido de poder comparar el desempeño de las metaheurísticas dado los mismos recursos de número de iteraciones, en este caso se fijan las iteraciones a 1000 para todas las metaheurísticas.

Los experimentos fueron hechos con un procesador Intel(R) Core(TM) i5-3470 CPU @ 3.20GHz, 4 procesadores y 4GB de memoria RAM. El preprocesamiento de los datos fue hecho con Python v.3.6.5, la entonación fue hecha con *Irace* v.2.4 y R v.3.4.4, los experimentos principales fueron realizados con GCC v.7.3.0.

En las siguientes secciones se van presentando los experimentos realizados con sus respectivos resultados.

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
GGA	0.8312	0.7525	0.7017	0.5557	0.5532	12.8250
CNN-GGA	0.5736	0.4911	0.3525	0.2151	0.7491	44.6228
ENN-GGA	0.8533	0.7891	0.7327	0.6093	0.2611	14.6024
RSS-GGA	0.6626	0.6177	0.4044	0.3263	0.7717	21.5376
CNN-RSS-GGA	0.7883	0.6809	0.6353	0.4472	0.6221	17.2401
ENN-RSS-GGA	0.8973	0.7872	0.8182	0.6063	0.1994	14.6176

TABLA 3.6: Promedios de GGA para los conjuntos pequeños 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
GGA	0.8702	0.7982	0.7375	0.5812	0.5641	110.0812
CNN-GGA	0.7990	0.7051	0.5952	0.4426	0.7745	232.8342
ENN-GGA	0.8475	0.8071	0.6949	0.5965	0.2862	125.8465
RSS-GGA	0.8027	0.7520	0.5575	0.4547	0.6733	137.0381
CNN-RSS-GGA	0.8932	0.7890	0.7571	0.5516	0.5514	123.7841
ENN-RSS-GGA	0.9141	0.8155	0.8246	0.6168	0.1981	125.7187

TABLA 3.7: Promedios de GGA para los conjuntos medianos 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
GGA	0.9316	0.8644	0.7994	0.6014	0.5050	672.0273
CNN-GGA	0.7106	0.6431	0.4478	0.3094	0.8409	8473.8825
ENN-GGA	0.9357	0.8890	0.7944	0.6453	0.1502	668.0400
RSS-GGA	0.9150	0.8719	0.7227	0.5821	0.7083	1197.0505
CNN-RSS-GGA	0.9455	0.8588	0.8410	0.9455	0.6089	1058.65
ENN-RSS-GGA	0.9574	0.9574	0.9574	0.6238	0.1025	681.3705

TABLA 3.8: Promedios de GGA para los conjuntos grandes con 1000 iteraciones

3.1.1. Resultados de heurísticas

3.1.2. Resultados de la combinación entre heurísticas y metaheurísticas

3.1.2.1. Con número de iteraciones variable

3.1.2.2. Fijando el número de iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
SSGA	0.8654	0.7716	0.7635	0.5917	0.8432	0.6655
CNN-SSGA	0.8791	0.7695	0.7840	0.5815	0.8425	1.2187
ENN-SSGA	0.8733	0.7919	0.7683	0.6112	0.7823	0.8601
RSS-SSGA	0.8581	0.7726	0.7460	0.5905	0.8958	1.0116
CNN-RSS-SSGA	0.8808	0.7742	0.7884	0.5974	0.8373	0.9554
EEN-RSS-SSGA	0.8843	0.7890	0.7904	0.6134	0.7586	0.8282

TABLA 3.9: Promedios de SSGA para los conjuntos pequeños 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
SSGA	0.8431	0.8029	0.6676	0.5871	0.8392	3.5589
CNN-SSGA	0.8540	0.8007	0.6965	0.5923	0.8547	6.4410
ENN-SSGA	0.8200	0.8009	0.6324	0.5793	0.7459	5.6722
RSS-SSGA	0.8302	0.8045	0.6389	0.5854	0.8690	5.9946
CNN-RSS-SSGA	0.8586	0.8047	0.7043	0.5977	0.8121	5.2934
ENN-RSS-SSGA	0.8550	0.8078	0.7042	0.6004	0.7049	5.2147

TABLA 3.10: Promedios de SSGA para los conjuntos medianos 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
SSGA	0.8911	0.8743	0.6953	0.6201	0.8056	27.6637
CNN-SSGA	0.8897	0.8512	0.7002	0.5939	0.8807	105.1004
ENN-SSGA	0.9097	0.8892	0.7113	0.6441	0.6823	34.9024
RSS-SSGA	0.8968	0.8826	0.6743	0.6271	0.8927	57.7995
CNN-RSS-SSGA	0.9002	0.8634	0.7140	0.6069	0.8434	49.5815
ENN-RSS-SSGA	0.9112	0.8805	0.7245	0.6272	0.6668	39.2984

TABLA 3.11: Promedios de SSGA para los conjuntos grandes con 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
MA	0.8570	0.7918	0.7440	0.6216	0.9561	4.1047
CNN-MA	0.8653	0.7987	0.7519	0.6307	0.9486	5.9638
ENN-MA	0.8666	0.7930	0.7576	0.6215	0.9534	4.2491
RSS-MA	0.8446	0.7795	0.7137	0.5903	0.9630	4.6391
CNN-RSS-MA	0.8639	0.7848	0.7538	0.6090	0.9507	5.9923
ENN-RSS-MA	0.8736	0.7942	0.7716	0.6268	0.9474	5.5961

TABLA 3.12: Promedios de MA para los conjuntos pequeños 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
MA	0.8057	0.7908	0.5825	0.5564	0.9624	73.3461
CNN-MA	0.8623	0.8523	0.6374	0.6125	1.0171	60.9810
ENN-MA	0.7938	0.7925	0.5747	0.5608	0.9707	67.5310
RSS-MA	0.7924	0.7893	0.5611	0.5449	0.9671	69.5927
CNN-RSS-MA	0.8051	0.8015	0.5962	0.5808	0.9654	80.8147
ENN-RSS-MA	0.8070	0.7971	0.5886	0.8070	0.9633	74.6872

TABLA 3.13: Promedios de MA para los conjuntos medianos 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
MA	0.8904	0.8999	0.6499	0.6510	0.9973	256.1432
CNN-MA	0.8925	0.8926	0.6360	0.6341	0.9884	612.7902
ENN-MA	0.8807	0.8902	0.6072	0.6145	0.9976	288.6525
RSS-MA	0.9000	0.9002	0.6347	0.6343	0.9746	527.9115
CNN-RSS-MA	0.8857	0.8951	0.6176	0.6160	0.9707	640.2634
ENN-RSS-MA	0.8746	0.8912	0.6292	0.6386	0.9969	295.3335

TABLA 3.14: Promedios de MA para los conjuntos grandes con 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
CHC	0.8446	0.7843	0.7172	0.6084	0.9466	0.5266
CNN-CHC	0.8495	0.7812	0.7269	0.6027	0.9385	0.7891
ENN-CHC	0.8429	0.7863	0.7179	0.6132	0.9424	0.7445
RSS-CHC	0.8383	0.7779	0.6963	0.5836	0.9546	0.7137
CNN-RSS-CHC	0.8442	0.7794	0.7099	0.5864	0.9437	0.7429
ENN-RSS-CHC	0.8484	0.7846	0.7224	0.6030	0.9333	0.7149

TABLA 3.15: Promedios de CHC para los conjuntos pequeños 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
CHC	0.8313	0.8115	0.6347	0.5986	0.9455	2.8843
CNN-CHC	0.8303	0.8081	0.6337	0.5925	0.9352	4.6572
ENN-CHC	0.8107	0.8032	0.6084	0.5834	0.9356	5.0976
RSS-CHC	0.8151	0.8058	0.6074	0.5851	0.9535	4.7450
CNN-RSS-CHC	0.8276	0.8073	0.6280	0.5871	0.9319	4.5240
ENN-RSS-CHC	0.8255	0.8074	0.6355	0.5944	0.9154	4.6856

TABLA 3.16: Promedios de CHC para los conjuntos medianos con 1000 iteraciones

ALGORITMO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
CHC	0.8961	0.8934	0.6614	0.6506	0.9615	16.8665
CNN-CHC	0.8903	0.8871	0.6554	0.6437	0.9771	38.3314
ENN-CHC	0.8993	0.8964	0.6746	0.6631	0.9233	27.3520
RSS-CHC	0.8972	0.8962	0.6591	0.6526	0.9819	41.1142
CNN-RSS-CHC	0.8945	0.8910	0.8945	0.6435	0.9719	33.7200
ENN-RSS-CHC	0.8975	0.8922	0.6721	0.6532	0.9099	31.6004

TABLA 3.17: Promedios de CHC para los conjuntos grandes con 1000 iteraciones

ACCURACY			KAPPA			REDUCCIÓN		
Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor
ENN-SSGA	7.80	1	CNN-MA	8.42	3	RSS-MA	2.84	11
CNN-MA	8.15	3	ENN-SSGA	8.42	1	RSS-CHC	3.80	7
ENN-GGA	8.38	2	ENN-GGA	9.03	1	MA	5.92	1
ENN-MA	9.11	0	ENN-RSS-MA	9.26	2	CHC	6.76	2
ENN-RSS-MA	9.34	1	ENN-MA	9.38	0	ENN-MA	6.88	1
MA	9.53	3	MA	9.61	3	CNN-RSS-CHC	6.96	2
ENN-RSS-GGA	9.65	4	ENN-RSS-GGA	9.69	5	CNN-RSS-MA	7.00	1
CHC	10.42	1	ENN-RSS-SSGA	9.88	1	ENN-CHC	7.57	1
ENN-RSS-SGGA	10.42	1	ENN-CHC	10.65	0	CNN-MA	7.76	0
ENN-CHC	10.65	0	CHC	11.00	2	ENN-RSS-MA	8.00	0
ENN-RSS-CHC	11.23	2	RSS-MA	11.61	3	CNN-CHC	8.03	0
RSS-MA	11.53	4	ENN-RSS-CHC	11.73	0	ENN-RSS-CHC	8.53	0
CNN-RSS-MA	12.03	0	CNN-RSS-MA	11.80	0	RSS-SSGA	13.03	0
RSS-CHC	12.19	0	CNN-RSS-SSGA	12.11	2	CNN-SSGA	15.50	0
CNN-RSS-CHC	12.91	1	SSGA	12.38	0	SSGA	15.65	0
CNN-CHC	12.92	0	CNN-CHC	13.15	0	CNN-RSS-SSGA	15.80	0
SSGA	13.00	0	RSS-CHC	13.23	0	RSS-GGA	17.30	0
CNN-RSS-SSGA	13.03	0	RSS-SSGA	13.42	2	CNN-GGA	17.42	0
RSS-SSGA	13.73	2	CNN-RSS-CHC	13.53	0	ENN-SSGA	17.57	0
CNN-SSGA	14.50	1	CNN-SSGA	14.50	1	ENN-RSS-SSGA	18.88	0
GGA	16.15	0	GGA	15.26	0	CNN-RSS-GGA	20.69	0
CNN-RSS-GGA	19.03	0	CNN-RSS-GGA	18.03	0	GGA	21.23	0
RSS-GGA	21.23	0	RSS-GGA	20.73	0	ENN-GGA	22.80	0
CNN-GGA	23.26	0	CNN-GGA	23.07	0	ENN-RSS-GGA	23.96	0

TABLA 3.18: Rangos de las metaheurísticas en *accuracy*, *kappa* y reducción para los conjuntos pequeños

ACCURACY + REDUCCIÓN			KAPPA + REDUCCIÓN			TIEMPO		
Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor
MA	5.38	5	CNN-MA	1.30	22	CHC	1.53	20
ENN-MA	5.46	2	MA	6.03	2	SSGA	3.07	3
CNN-MA	5.57	4	ENN-MA	6.03	0	RSS-CHC	4.07	0
RSS-MA	6.07	6	ENN-RSS-MA	6.38	1	ENN-RSS-CHC	5.11	0
ENN-RSS-MA	6.30	3	RSS-MA	7.23	0	CNN-RSS-CHC	6.00	0
RSS-CHC	6.80	1	CNN-RSS-MA	7.38	0	ENN-CHC	6.19	0
CNN-RSS-MA	6.96	0	ENN-CHC	7.76	0	CNN-CHC	6.84	0
CHC	7.07	0	CHC	7.80	0	ENN-RSS-SSGA	7.34	1
ENN-CHC	7.53	0	RSS-CHC	8.23	0	ENN-SSGA	8.11	1
CNN-RSS-CHC	7.65	2	CNN-RSS-CHC	8.30	0	CNN-RSS-SSGA	8.96	0
ENN-RSS-CHC	7.88	2	ENN-RSS-CHC	8.61	0	RSS-SSGA	10.19	1
CNN-CHC	8.73	1	CNN-CHC	8.92	1	CNN-SSGA	10.53	0
RSS-SSGA	12.07	0	RSS-SSGA	11.96	0	MA	14.42	0
CNN-SSGA	14.53	0	SSGA	14.34	0	ENN-MA	15.00	0
SSGA	14.88	0	CNN-RSS-SSGA	14.38	0	RSS-MA	15.15	0
CNN-RSS-SSGA	15.07	0	CNN-SSGA	14.53	0	ENN-RSS-MA	15.61	0
ENN-SSGA	16.11	0	ENN-SSGA	15.26	0	CNN-RSS-MA	16.23	0
ENN-RSS-SSGA	17.26	0	ENN-RSS-SSGA	16.46	0	CNN-MA	16.76	0
RSS-GGA	19.11	0	GGA	20.15	0	GGA	20.46	0
GGA	20.73	0	RSS-GGA	20.19	0	CNN-RSS-GGA	20.73	0
CNN-RSS-GGA	20.84	0	CNN-RSS-GGA	20.88	0	ENN-GGA	21.23	0
CNN-GGA	21.30	0	CNN-GGA	21.69	0	ENN-RSS-GGA	21.34	0
ENN-GGA	22.73	0	ENN-GGA	22.53	0	RSS-GGA	22.07	0
ENN-RSS-GGA	23.84	0	ENN-RSS-GGA	23.53	0	CNN-GGA	22.96	0

TABLA 3.19: Rangos de las metaheurísticas en *accuracy + reducción*, *kappa + reducción* y tiempo para los conjuntos pequeños

ACCURACY			KAPPA			REDUCCIÓN		
Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor
ENN-RSS-GGA	5.64	7	ENN-RSS-GGA	4.82	9	RSS-MA	4.05	4
ENN-GGA	5.94	2	ENN-GGA	6.11	1	CNN-RSS-MA	4.70	2
ENN-SSGA	7.88	0	ENN-RSS-SSGA	8.64	0	ENN-RSS-MA	4.76	3
ENN-RSS-SSGA	8.17	1	ENN-SSGA	8.88	0	ENN-MA	4.76	3
CHC	9.23	1	GGA	9.64	0	MA	5.41	1
ENN-CHC	10.64	1	CHC	9.88	1	CNN-MA	6.00	3
ENN-RSS-CHC	10.64	0	ENN-RSS-CHC	10.05	0	RSS-CHC	6.29	1
GGA	11.29	0	SSGA	10.64	1	CHC	7.52	0
SSGA	11.52	0	CNN-RSS-SSGA	10.64	0	CNN-CHC	8.35	0
CNN-RSS-SSGA	11.82	0	ENN-CHC	11.05	0	CNN-RSS-CHC	9.23	0
CNN-CHC	11.94	0	CNN-RSS-MA	12.41	0	ENN-CHC	9.23	0
CNN-RSS-MA	12.41	1	CNN-SSGA	12.76	0	ENN-RSS-CHC	10.64	0
RSS-CHC	12.88	0	CNN-CHC	13.23	0	RSS-SSGA	13.64	0
RSS-SSGA	12.88	0	CNN-RSS-GGA	13.70	1	CNN-SSGA	13.70	0
CNN-MA	13.00	1	RSS-SSGA	13.70	0	SSGA	15.35	0
CNN-RSS-CHC	13.05	1	RSS-CHC	13.94	0	CNN-GGA	16.05	0
ENN-MA	13.70	1	MA	14.05	0	CNN-RSS-SSGA	16.29	0
MA	13.94	0	ENN-MA	14.11	2	ENN-SSGA	17.05	0
CNN-SSGA	14.23	0	CNN-RSS-CHC	14.23	1	ENN-RSS-SSGA	18.82	0
ENN-RSS-MA	14.41	0	CNN-MA	14.52	0	RSS-GGA	19.41	0
CNN-RSS-GGA	14.64	0	ENN-RSS-MA	14.76	0	GGA	21.23	0
RSS-MA	18.35	0	RSS-MA	18.47	0	CNN-RSS-GGA	21.47	0
RSS-GGA	20.58	1	RSS-GGA	19.58	1	ENN-GGA	22.11	0
CNN-GGA	21.11	0	CNN-GGA	20.05	0	ENN-RSS-GGA	23.82	0

TABLA 3.20: Rangos de las metaheurísticas en *accuracy*, *kappa* y reducción para los conjuntos medianos

ACCURACY + REDUCCIÓN			KAPPA + REDUCCIÓN			TIEMPO		
Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor
CNN-RSS-MA	4.76	5	CNN-MA	1.94	13	CHC	1.52	12
ENN-MA	5.05	4	CHC	5.58	1	SSGA	2.05	3
CHC	5.52	2	ENN-MA	5.94	1	CNN-CHC	5.52	0
CNN-MA	5.94	3	CNN-RSS-MA	6.00	1	ENN-RSS-CHC	5.64	0
ENN-RSS-MA	5.94	1	RSS-CHC	6.94	0	CNN-RSS-CHC	5.76	0
RSS-CHC	5.94	0	ENN-RSS-MA	7.47	1	RSS-CHC	6.64	0
MA	6.29	0	CNN-CHC	7.58	0	ENN-RSS-SSGA	6.82	1
RSS-MA	7.05	1	MA	7.64	0	ENN-CHC	7.11	0
CNN-CHC	7.35	0	ENN-CHC	7.82	0	CNN-RSS-SSGA	8.05	1
ENN-CHC	8.41	0	CNN-RSS-CHC	8.58	0	ENN-SSGA	8.11	0
CNN-RSS-CHC	8.70	0	ENN-RSS-CHC	8.76	0	CNN-SSGA	10.05	0
ENN-RSS-CHC	10.00	1	RSS-MA	8.82	0	RSS-SSGA	10.64	0
RSS-SSGA	13.47	0	CNN-SSGA	13.00	0	MA	14.76	0
CNN-SSGA	13.47	0	RSS-SSGA	13.17	0	RSS-MA	15.35	0
SSGA	14.70	0	SSGA	14.29	0	CNN-RSS-MA	16.05	0
CNN-RSS-SSGA	15.82	0	CNN-RSS-SSGA	15.11	0	CNN-MA	16.29	0
ENN-SSGA	16.64	0	ENN-SSGA	16.82	0	ENN-MA	16.41	0
CNN-GGA	18.17	0	ENN-RSS-SSGA	18.00	0	ENN-RSS-MA	16.76	0
ENN-RSS-SSGA	18.47	0	CNN-GGA	18.52	0	GGA	19.82	0
RSS-GGA	19.94	0	GGA	20.47	0	ENN-RSS-GGA	20.11	0
GGA	21.00	0	RSS-GGA	20.52	0	ENN-GGA	20.41	0
CNN-RSS-GGA	21.41	0	CNN-RSS-GGA	21.23	0	CNN-RSS-GGA	21.00	0
ENN-GGA	22.05	0	ENN-GGA	22.05	0	RSS-GGA	21.82	0
ENN-RSS-GGA	23.82	0	ENN-RSS-GGA	23.64	0	CNN-GGA	23.17	0

TABLA 3.21: Rangos de las metaheurísticas en *accuracy + reducción*, *kappa + reducción* y tiempo para los conjuntos medianos

ACCURACY			KAPPA			REDUCCIÓN		
Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor
GGA	11.00	0	ENN-GGA	6.50	0	ENN-MA	2.00	1
CNN-GGA	24.00	0	ENN-SSGA	7.00	0	MA	3.00	0
ENN-GGA	9.50	0	ENN-RSS-GGA	8.50	1	ENN-RSS-MA	4.50	0
RSS-GGA	15.00	0	ENN-RSS-CHC	8.50	0	RSS-MA	5.50	1
CNN-RSS-GGA	12.50	0	ENN-RSS-SSGA	9.00	0	CNN-MA	6.00	0
ENN-RSS-GGA	9.00	1	CNN-MA	10.00	0	RSS-CHC	6.50	0
SSGA	12.50	0	ENN-CHC	10.00	1	CNN-RSS-MA	7.00	0
CNN-SSGA	16.00	0	GGA	11.00	0	CNN-CHC	7.00	0
ENN-SSGA	8.50	0	CHC	11.00	0	CHC	9.50	0
RSS-SSGA	14.50	0	SSGA	11.50	0	CNN-RSS-CHC	9.50	0
CNN-RSS-SSGA	15.50	0	RSS-CHC	11.50	0	CNN-GGA	11.50	0
ENN-RSS-SSGA	10.00	0	MA	12.00	0	CNN-SSGA	11.50	0
MA	11.50	1	CNN-RSS-CHC	12.00	0	RSS-SSGA	13.00	0
CNN-MA	8.00	0	CNN-CHC	12.50	0	ENN-CHC	13.00	0
ENN-MA	13.50	0	CNN-RSS-GGA	13.00	0	ENN-RSS-CHC	14.00	0
RSS-MA	8.50	0	RSS-MA	13.00	0	CNN-RSS-SSGA	15.00	0
CNN-RSS-MA	9.00	0	RSS-SSGA	14.00	0	SSGA	16.00	0
ENN-RSS-MA	15.50	0	CNN-RSS-SSGA	14.50	0	RSS-GGA	18.50	0
CHC	12.00	0	CNN-SSGA	15.00	0	ENN-SSGA	19.00	0
CNN-CHC	16.00	0	CNN-RSS-MA	15.00	0	ENN-RSS-SSGA	19.00	0
ENN-CHC	11.00	0	ENN-RSS-MA	15.50	0	CNN-RSS-GGA	20.50	0
RSS-CHC	12.50	0	RSS-GGA	17.00	0	GGA	21.50	0
CNN-RSS-CHC	13.50	0	ENN-MA	18.00	0	ENN-GGA	23.50	0
ENN-RSS-CHC	11.00	0	CNN-GGA	24.00	0	ENN-RSS-GGA	23.50	0

TABLA 3.22: Rangos de las metaheurísticas en *accuracy*, *kappa* y reducción para los conjuntos grandes

ACCURACY + REDUCCIÓN			KAPPA + REDUCCIÓN			TIEMPO		
Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor	Algoritmo	Rango	Mejor
MA	1.50	1	CNN-MA	2.00	1	CHC	1.00	2
ENN-MA	3.00	0	MA	2.00	0	ENN-CHC	4.00	0
ENN-RSS-MA	3.50	1	ENN-RSS-MA	3.00	1	SSGA	4.00	0
CNN-MA	5.00	0	RSS-CHC	4.50	0	ENN-RSS-CHC	4.00	0
RSS-MA	5.50	0	CNN-CHC	6.50	0	CNN-RSS-CHC	4.50	0
RSS-CHC	7.00	0	ENN-MA	8.00	0	CNN-CHC	5.00	0
CNN-RSS-MA	7.50	0	RSS-MA	8.00	0	RSS-CHC	8.00	0
CNN-CHC	7.50	0	CHC	8.00	0	ENN-SSGA	8.00	0
CHC	9.00	0	CNN-RSS-CHC	8.50	0	ENN-RSS-SSGA	8.00	0
CNN-RSS-CHC	9.00	0	ENN-CHC	9.50	0	CNN-RSS-SSGA	9.00	0
CNN-SSGA	11.00	0	CNN-SSGA	10.50	0	CNN-SSGA	11.00	0
ENN-CHC	11.50	0	CNN-RSS-MA	11.50	0	RSS-SSGA	11.50	0
RSS-SSGA	12.50	0	ENN-RSS-CHC	12.50	0	MA	13.50	0
ENN-RSS-CHC	13.50	0	RSS-SSGA	12.50	0	ENN-MA	16.00	0
CNN-RSS-SSGA	15.00	0	CNN-RSS-SSGA	15.00	0	ENN-RSS-MA	16.00	0
SSGA	15.50	0	SSGA	15.50	0	CNN-RSS-MA	17.00	0
RSS-GGA	17.50	0	ENN-SSGA	17.50	0	RSS-MA	17.50	0
ENN-SSGA	18.00	0	RSS-GGA	18.00	0	CNN-MA	17.50	0
ENN-RSS-SSGA	19.00	0	ENN-RSS-SSGA	18.50	0	GGA	19.00	0
CNN-GGA	19.50	0	CNN-RSS-GGA	20.00	0	ENN-GGA	19.00	0
CNN-RSS-GGA	20.00	0	CNN-GGA	20.50	0	ENN-RSS-GGA	20.50	0
GGA	21.50	0	GGA	21.00	0	RSS-GGA	21.00	0
ENN-GGA	23.50	0	ENN-GGA	23.50	0	CNN-RSS-GGA	21.50	0
ENN-RSS-GGA	23.50	0	ENN-RSS-GGA	23.50	0	CNN-GGA	23.50	0

TABLA 3.23: Rangos de las metaheurísticas en *accuracy + reducción*, *kappa + reducción* y tiempo para los conjuntos grandes

Conclusiones y Recomendaciones

Conclusiones

Bibliografía

- [AD91] Hussein Almuallim and Thomas G Dietterich. Learning with many irrelevant features. In *AAAI*, volume 91, pages 547–552, 1991.
- [AFFL⁺11] Jesús Alcalá-Fdez, Alberto Fernández, Julián Luengo, Joaquín Derrac, Salvador García, Luciano Sánchez, and Francisco Herrera. Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic & Soft Computing*, 17, 2011.
- [AL03] Emile HL Aarts and Jan Karel Lenstra. *Local search in combinatorial optimization*. Princeton University Press, 2003.
- [Ale14] Flores Alejandro. *Metaheurísticas Bio-Inspiradas para Selección de Instancias*. PhD thesis, Undergraduate thesis, Departamento de Ciencias de la Computación, Universidad Simón Bolívar, Venezuela, 2014.
- [Bat94] Roberto Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on neural networks*, 5(4):537–550, 1994.
- [BF99] Carla E Brodley and Mark A Friedl. Identifying mislabeled training data. *Journal of artificial intelligence research*, 11:131–167, 1999.
- [BLN86] Carlo Batini, Maurizio Lenzerini, and Shamkant B. Navathe. A comparative analysis of methodologies for database schema integration. *ACM computing surveys (CSUR)*, 18(4):323–364, 1986.

- [CAB11] Cagatay Catal, Oral Alan, and Kerime Balkan. Class noise detection based on software metrics and roc curves. *Information Sciences*, 181(21):4867–4877, 2011.
- [CHL05] José Ramón Cano, Francisco Herrera, and Manuel Lozano. Stratification for scaling up evolutionary prototype selection. *Pattern Recognition Letters*, 26(7):953–963, 2005.
- [CKLS01] Munir Cochinwala, Verghese Kurien, Gail Lalk, and Dennis Shasha. Efficient data reconciliation. *Information Sciences*, 137(1-4):1–15, 2001.
- [CPSK07] Krzysztof J Cios, Witold Pedrycz, Roman W Swiniarski, and Lukasz Andrzej Kurgan. *Data mining: a knowledge discovery approach*. Springer Science & Business Media, 2007.
- [dA04] José Ramón Cano de Amo. *Reducción de datos basada en Selección Evolutiva de Instancias para Minería de Datos*. PhD thesis, PhD thesis, Departamento de Ciencias de la Computación e Inteligencia Artificial, Universidad de Granada, Spain, 2004.
- [DKT17] Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.
- [DL97] Manoranjan Dash and Huan Liu. Feature selection for classification. *Intelligent data analysis*, 1(3):131–156, 1997.
- [Dor92] Marco Dorigo. Optimization, learning and natural algorithms. *PhD Thesis, Politecnico di Milano*, 1992.
- [Esh91] Larry J Eshelman. The chc adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination. In *Foundations of genetic algorithms*, volume 1, pages 265–283. Elsevier, 1991.
- [ESK01] Russell C Eberhart, Yuhui Shi, and James Kennedy. *Swarm intelligence*. Elsevier, 2001.

- [FKP07] Alireza Farhangfar, Lukasz A Kurgan, and Witold Pedrycz. A novel framework for imputation of missing values in databases. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37(5):692–709, 2007.
- [FM17] Alejandro Flores and David M Mount. Nearest neighbor condensation with guarantees. 2017.
- [FR95] Thomas A Feo and Mauricio GC Resende. Greedy randomized adaptive search procedures. *Journal of global optimization*, 6(2):109–133, 1995.
- [FS69] Ivan P Fellegi and Alan B Sunter. A theory for record linkage. *Journal of the American Statistical Association*, 64(328):1183–1210, 1969.
- [FSS96] Usama M Fayyad, Gregory P Shapiro, and Padhraic Smyth. From data mining to knowledge discovery: An overview. 1996.
- [GCH08] Salvador García, José Ramón Cano, and Francisco Herrera. A memetic algorithm for evolutionary prototype selection: A scaling up approach. *Pattern Recognition*, 41(8):2693–2709, 2008.
- [GDCH12] Salvador Garcia, Joaquin Derrac, Jose Cano, and Francisco Herrera. Prototype selection for nearest neighbor classification: Taxonomy and empirical study. *IEEE transactions on pattern analysis and machine intelligence*, 34(3):417–435, 2012.
- [GLH16] Salvador García, Julián Luengo, and Francisco Herrera. *Data preprocessing in data mining*. Springer, 2016.
- [Glo77] Fred Glover. Heuristics for integer programming using surrogate constraints. *Decision sciences*, 8(1):156–166, 1977.
- [Glo89] Fred Glover. Tabu search—part i. *ORSA Journal on computing*, 1(3):190–206, 1989.
- [GLS⁺13] Salvador Garcia, Julian Luengo, José Antonio Sáez, Victoria Lopez, and Francisco Herrera. A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 25(4):734–750, 2013.

- [Har68] Peter Hart. The condensed nearest neighbor rule (corresp.). *IEEE transactions on information theory*, 14(3):515–516, 1968.
- [Hol75] John H Holland. Adaptation in natural and artificial systems. an introductory analysis with application to biology, control, and artificial intelligence. *Ann Arbor, MI: University of Michigan Press*, pages 439–444, 1975.
- [K⁺95] Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.
- [KCH⁺03] Won Kim, Byoung-Ju Choi, Eui-Kyeong Hong, Soo-Kyung Kim, and Doheon Lee. A taxonomy of dirty data. *Data mining and knowledge discovery*, 7(1):81–99, 2003.
- [KGV83] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.
- [KS96] Daphne Koller and Mehran Sahami. Toward optimal feature selection. Technical report, Stanford InfoLab, 1996.
- [LIDLC⁺16] Manuel López-Ibáñez, Jérémie Dubois-Lacoste, Leslie Pérez Cáceres, Mauro Birattari, and Thomas Stützle. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58, 2016.
- [LLIB06] Jose A Lozano, Pedro Larrañaga, Iñaki Inza, and Endika Bengoetxea. *Towards a new evolutionary computation: advances on estimation of distribution algorithms*, volume 192. Springer, 2006.
- [LM12] Huan Liu and Hiroshi Motoda. *Feature selection for knowledge discovery and data mining*, volume 454. Springer Science & Business Media, 2012.
- [LMD98] Huan Liul, Hiroshi Motoda, and Manoranjan Dash. A monotonic measure for optimal feature selection. In *European conference on machine learning*, pages 101–106. Springer, 1998.

- [LMS03] Helena R Lourenço, Olivier C Martin, and Thomas Stützle. Iterated local search. In *Handbook of metaheuristics*, pages 320–353. Springer, 2003.
- [LS⁺96] Huan Liu, Rudy Setiono, et al. A probabilistic approach to feature selection—a filter solution. In *ICML*, volume 96, pages 319–327. Cite-seer, 1996.
- [MH97] Nenad Mladenović and Pierre Hansen. Variable neighborhood search. *Computers & operations research*, 24(11):1097–1100, 1997.
- [NC12] Ferrante Neri and Carlos Cotta. Memetic algorithms and memetic computing optimization: A literature review. *Swarm and Evolutionary Computation*, 2:1–14, 2012.
- [Pea84] Judea Pearl. Heuristics: intelligent search strategies for computer problem solving. 1984.
- [PSL06] Kenneth Price, Rainer M Storn, and Jouni A Lampinen. *Differential evolution: a practical approach to global optimization*. Springer Science & Business Media, 2006.
- [Tal09] El-Ghazali Talbi. *Metaheuristics: from design to implementation*, volume 74. John Wiley & Sons, 2009.
- [Ten99] Choh-Man Teng. Correcting noisy data. In *ICML*, pages 239–248. Cite-seer, 1999.
- [Vou98] Christos Voudouris. Guided local search—an illustrative example in function optimisation. *BT Technology Journal*, 16(3):46–50, 1998.
- [Wil72] Dennis L Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man, and Cybernetics*, (3):408–421, 1972.
- [XYC88] Lei Xu, Pingfan Yan, and Tong Chang. Best first strategy for feature selection. In *Pattern Recognition, 1988., 9th International Conference on*, pages 706–708. IEEE, 1988.

-
- [YW09] Ying Yang and Geoffrey I Webb. Discretization for naive-bayes learning: managing discretization bias and variance. *Machine learning*, 74(1):39–74, 2009.
- [Zuk10] AV Zuhba. Np-completeness of the problem of prototype selection in the nearest neighbor method. *Pattern Recognition and Image Analysis*, 20(4):484–494, 2010.

Apéndice A

Resultados de todos los conjuntos

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9578	0.9333	0.9367	0.9000	0.4860	1.6052
cleveland	0.7383	0.5825	0.5962	0.3419	0.5485	6.5904
led7digit	0.6893	0.6480	0.6540	0.6072	0.5488	9.0854
pima	0.8317	0.6768	0.6240	0.2754	0.5402	17.3229
wdbc	0.9733	0.9492	0.9420	0.8889	0.5409	23.3391
monk2	0.8673	0.7822	0.7336	0.5629	0.5329	8.2115
wisconsin	0.9730	0.9504	0.9394	0.8886	0.5313	15.1765
wine	0.9632	0.9353	0.9445	0.9030	0.5129	2.5872
glass	0.7006	0.6131	0.5633	0.4205	0.6154	3.4891
banknote	0.9984	0.9957	0.9967	0.9912	0.5497	29.7193
appendicitis	0.8901	0.8738	0.6201	0.5506	0.5321	1.0743
balance	0.8964	0.8034	0.8158	0.6503	0.5368	12.4007
bands	0.7987	0.6696	0.5676	0.3031	0.5471	10.1985
contraceptive	0.6950	0.4366	0.5287	0.1354	0.5538	38.9566
dermatology	0.9709	0.9496	0.9631	0.9364	0.5212	15.6820
ecoli	0.8591	0.8110	0.7956	0.7203	0.5548	6.6674
haberman	0.8072	0.6772	0.4823	0.1404	0.5516	4.6479
hayes-roth	0.5444	0.5125	0.2700	0.2143	0.6838	1.8741
heart	0.8407	0.7593	0.6745	0.5086	0.5300	5.5938
hepatitis	0.8611	0.8250	0.4622	0.2541	0.5100	1.0918
mammographic	0.8450	0.7373	0.6895	0.4739	0.5405	17.2283
newthyroid	0.9618	0.9504	0.9073	0.8607	0.5135	2.7401
tae	0.4696	0.4042	0.2095	0.0909	0.7205	1.8484
vehicle	0.8158	0.6925	0.7541	0.5897	0.5595	26.8813
vowel	0.9247	0.8697	0.9172	0.8567	0.5658	28.9557
yeast	0.7370	0.5263	0.6575	0.3821	0.5566	40.4827

TABLA A.1: Resultados de GGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.9336	0.8670	0.8660	0.7317	0.5339	50.4962
cardiotocography	0.9359	0.8954	0.8168	0.6954	0.5731	19.7200
eye-state	0.9002	0.8108	0.7984	0.6177	0.5512	209.4679
page-blocks	0.9760	0.9573	0.8652	0.7563	0.5420	60.4759
penbased	0.9950	0.9913	0.9944	0.9904	0.5456	155.4689
satimage	0.9440	0.8931	0.9310	0.8684	0.5439	114.2495
thyroid	0.9697	0.9384	0.7598	0.4854	0.5408	99.9336
segment	0.4049	0.1468	0.3057	0.0045	0.7051	23.9123
coil2000	0.9448	0.9002	0.4752	0.0612	0.5412	320.5003
magic	0.9008	0.8061	0.7793	0.5617	0.5466	257.4847
marketing	0.5472	0.2673	0.4840	0.1647	0.6214	93.4540
phoneme	0.9310	0.8701	0.8307	0.6803	0.5455	53.9916
ring	0.7438	0.6525	0.4848	0.2999	0.6361	98.3467
spambase	0.9463	0.8900	0.8873	0.7689	0.5521	98.5384
texture	0.9880	0.9781	0.9868	0.9759	0.5505	104.2189
titanic	0.7621	0.7624	0.3333	0.3332	0.5249	10.2480
twonorm	0.9698	0.9426	0.9395	0.8852	0.5358	100.8737

TABLA A.2: Resultados de GGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8641	0.7299	0.6008	0.2059	0.5073	505.3758
shuttle	0.9990	0.9990	0.9979	0.9969	0.5026	838.6788

TABLA A.3: Resultados de GGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.3030	0.3200	-0.0456	-0.0200	0.9240	8.4874
cleveland	0.5293	0.3679	0.3537	0.1336	0.6963	7.9552
led7digit	0.6047	0.5360	0.5591	0.4826	0.4976	10.0936
pima	0.7962	0.6137	0.5746	0.1733	0.6272	20.3269
wdbc	0.6694	0.6784	0.3348	0.3413	0.9313	58.6600
monk2	0.7477	0.7167	0.4912	0.4271	0.7556	10.7442
wisconsin	0.3132	0.3147	-0.1204	-0.1146	0.9529	363.5220
wine	0.3945	0.3988	0.0050	0.0191	0.9202	9.9247
glass	0.4090	0.3240	0.1886	0.0510	0.8065	5.6923
banknote	0.5184	0.5102	0.0812	0.0678	0.9848	373.8600
appendicitis	0.1747	0.1563	-0.0476	-0.0932	0.9358	4.0006
balance	0.9204	0.7651	0.8647	0.5975	0.6248	19.5066
bands	0.7986	0.6423	0.5860	0.2476	0.5633	9.7889
contraceptive	0.9028	0.4222	0.8500	0.1121	0.3169	38.6017
dermatology	0.4545	0.4624	0.3400	0.3427	0.9212	31.4329
ecoli	0.5530	0.5303	0.4009	0.3704	0.8086	11.0493
haberman	0.6193	0.5517	0.2413	0.0795	0.7000	6.2717
hayes-roth	0.5792	0.5875	0.3234	0.3317	0.6288	1.7985
heart	0.6284	0.5444	0.2698	0.1018	0.7552	6.9793
hepatitis	0.2208	0.1750	0.0044	-0.0382	0.8950	3.1597
mammographic	0.7618	0.6639	0.5233	0.3270	0.6719	25.0182
newthyroid	0.3228	0.3650	0.1216	0.1462	0.9470	22.7798
tae	0.4702	0.3988	0.2035	0.1067	0.6715	1.7293
vehicle	0.7324	0.6024	0.6424	0.4689	0.6746	29.7396
vowel	0.5860	0.6020	0.5446	0.5622	0.8578	39.0387
yeast	0.9031	0.5188	0.8737	0.3691	0.4065	40.0306

TABLA A.4: Resultados de CNN-GGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8370	0.7930	0.6838	0.5855	0.8698	143.1180
cardiotocography	0.7095	0.6793	0.4598	0.3887	0.8567	60.1878
eye-state	0.8868	0.7900	0.7828	0.5787	0.6282	231.2230
page-blocks	0.7852	0.7752	0.3919	0.3442	0.9472	694.1560
penbased	0.9485	0.9540	0.9438	0.9488	0.9346	515.6620
satimage	0.8691	0.8292	0.8420	0.7911	0.8576	238.5030
thyroid	0.8393	0.8158	0.4006	0.2771	0.8964	280.0980
segment	0.7321	0.1606	0.6891	0.0207	0.3306	21.6793
coil2000	0.7481	0.7008	0.2124	0.0491	0.8571	339.5150
magic	0.8244	0.7278	0.6557	0.4453	0.7179	426.6730
marketing	0.7031	0.2532	0.6639	0.1514	0.3992	87.8192
phoneme	0.7686	0.7270	0.5190	0.4164	0.8306	243.0430
ring	0.8275	0.7330	0.6640	0.4633	0.7424	197.2490
spambase	0.8256	0.7844	0.6574	0.5640	0.8252	93.0783
texture	0.8071	0.8042	0.7888	0.7846	0.8905	151.4730
titanic	0.5624	0.5702	-0.0657	-0.0621	0.7318	31.2225
twonorm	0.9095	0.8884	0.8291	0.7768	0.8511	203.4820

TABLA A.5: Resultados de CNN-GGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7000	0.5640	0.3824	0.1055	0.6966	563.6650
shuttle	0.7212	0.7222	0.5131	0.5134	0.9852	16384.1000

TABLA A.6: Resultados de CNN-GGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9541	0.9533	0.9311	0.9300	0.1413	1.4803
cleveland	0.6738	0.5659	0.4545	0.2666	0.4650	7.4858
led7digit	0.7144	0.6660	0.6819	0.6276	0.3834	10.2158
pima	0.8063	0.7271	0.5610	0.3728	0.3194	17.8755
wdbc	0.9670	0.9576	0.9275	0.9062	0.1381	38.7962
monk2	0.8400	0.7310	0.6777	0.4589	0.2706	8.2817
wisconsin	0.9761	0.9651	0.9469	0.9223	0.1329	17.8421
wine	0.9494	0.9471	0.9237	0.9203	0.1455	4.4621
glass	0.8043	0.7164	0.7185	0.5900	0.2916	3.5409
banknote	0.9979	0.9978	0.9957	0.9956	0.1019	26.7934
appendicitis	0.9581	0.9438	0.7735	0.6611	0.1613	1.1819
balance	0.9092	0.8688	0.8337	0.7590	0.2474	11.3776
bands	0.7811	0.6753	0.5171	0.2805	0.3200	13.0032
contraceptive	0.6226	0.4706	0.4174	0.1848	0.5267	41.4551
dermatology	0.9640	0.9462	0.9541	0.9313	0.1369	25.6686
ecoli	0.8810	0.8522	0.8270	0.7833	0.2595	6.8828
haberman	0.8010	0.7122	0.4381	0.1458	0.3216	3.8786
hayes-roth	0.8153	0.6250	0.7060	0.3870	0.2931	1.6927
heart	0.8617	0.8000	0.7196	0.5942	0.2667	7.2303
hepatitis	0.8931	0.8875	0.5329	0.4000	0.2188	1.6808
mammographic	0.8454	0.7711	0.6909	0.5418	0.2852	17.0408
newthyroid	0.9623	0.9456	0.9118	0.8662	0.1340	2.6122
tae	0.7579	0.5571	0.6337	0.3278	0.3450	1.6246
vehicle	0.7817	0.6847	0.7089	0.5792	0.3258	30.7541
vowel	0.9892	0.9758	0.9881	0.9733	0.1097	35.4269
yeast	0.6795	0.5729	0.5786	0.4377	0.4485	41.3776

TABLA A.7: Resultados de ENN-GGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.9326	0.8866	0.8743	0.7714	0.2005	40.9813
cardiotocography	0.9373	0.9015	0.8397	0.7162	0.2209	20.6367
eye-state	0.9088	0.8282	0.8270	0.6547	0.3007	207.7310
page-blocks	0.9690	0.9656	0.8694	0.8018	0.1413	57.1816
penbased	0.9844	0.9924	0.9837	0.9915	0.1169	169.9620
satimage	0.9241	0.8960	0.9086	0.8715	0.2172	144.5530
thyroid	0.9431	0.9451	0.5049	0.4152	0.1509	114.0900
segment	0.2854	0.1437	0.1679	0.0010	0.8145	25.7382
coil2000	0.9377	0.9314	0.2758	0.0670	0.1968	465.3530
magic	0.8914	0.8303	0.7665	0.6106	0.2775	250.8440
marketing	0.4381	0.2958	0.3566	0.1914	0.6979	102.7230
phoneme	0.9308	0.8858	0.8481	0.7263	0.2446	45.7861
ring	0.6693	0.6020	0.3447	0.1980	0.4323	95.1795
spambase	0.9383	0.9067	0.8814	0.8027	0.2322	134.9760
texture	0.9771	0.9827	0.9758	0.9810	0.1475	140.3440
titanic	0.7680	0.7746	0.4346	0.4355	0.3282	9.3024
twonorm	0.9723	0.9526	0.9545	0.9051	0.1461	114.0080

TABLA A.8: Resultados de ENN-GGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8728	0.7793	0.5922	0.2944	0.2770	541.2280
shuttle	0.9985	0.9987	0.9965	0.9962	0.0233	794.8520

TABLA A.9: Resultados de ENN-GGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.5400	0.5467	0.3100	0.3200	0.8833	5.5023
cleveland	0.6995	0.5729	0.5127	0.2824	0.6000	6.3971
led7digit	0.5527	0.5160	0.5020	0.4613	0.8240	11.9904
pima	0.8002	0.6972	0.5542	0.3114	0.6133	18.1228
wdbc	0.8521	0.8519	0.6553	0.6578	0.8538	23.9792
monk2	0.6772	0.6479	0.3484	0.2897	0.8639	12.4095
wisconsin	0.7849	0.7956	0.4227	0.4501	0.9149	39.3925
wine	0.5652	0.5539	0.3467	0.3296	0.8933	8.3321
glass	0.5492	0.5655	0.3256	0.3274	0.7813	4.4381
banknote	0.7390	0.7355	0.4367	0.4300	0.9587	157.4090
appendicitis	0.6647	0.6575	0.0856	0.0546	0.7925	1.7964
balance	0.4461	0.4325	0.1961	0.1813	0.9686	48.2580
bands	0.7586	0.6377	0.4917	0.2357	0.5882	9.8358
contraceptive	0.6986	0.4528	0.5339	0.1546	0.5468	37.5165
dermatology	0.6459	0.6480	0.5522	0.5552	0.9059	16.5175
ecoli	0.7478	0.7249	0.6330	0.5976	0.7229	6.8582
haberman	0.7321	0.6872	0.3222	0.1714	0.6794	5.1684
hayes-roth	0.4563	0.4500	0.1113	0.1064	0.7663	2.1052
heart	0.7185	0.7148	0.4190	0.4175	0.7233	5.4765
hepatitis	0.5903	0.5875	0.1514	0.1471	0.7925	1.3769
mammographic	0.7645	0.6952	0.5278	0.3888	0.7325	21.4182
newthyroid	0.7107	0.7255	0.0031	0.0395	0.8977	6.3003
tae	0.4747	0.3958	0.2144	0.0949	0.6967	1.7932
vehicle	0.7788	0.6904	0.7048	0.5867	0.6264	25.0300
vowel	0.5505	0.5333	0.5056	0.4867	0.8702	42.3341
yeast	0.7308	0.5445	0.6486	0.4052	0.5676	40.2190

TABLA A.10: Resultados de RSS-GGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8050	0.8013	0.6016	0.5828	0.7280	84.1568
cardiotocography	0.8283	0.8282	0.4189	0.3822	0.7138	22.3631
eye-state	0.7666	0.7295	0.5189	0.4281	0.6508	233.4130
page-blocks	0.9259	0.9258	0.5842	0.5230	0.8192	108.7890
penbased	0.9352	0.9390	0.9291	0.9323	0.5503	199.4800
satimage	0.8348	0.8182	0.8007	0.7783	0.5464	189.0800
thyroid	0.8317	0.8125	0.4056	0.2997	0.8765	168.3580
segment	0.4871	0.1623	0.4032	0.0227	0.5987	21.4625
coil2000	0.9171	0.8938	0.3499	0.0775	0.6987	266.3050
magic	0.8032	0.7798	0.5325	0.4562	0.6845	319.2850
marketing	0.5438	0.2746	0.4821	0.1740	0.5708	94.6385
phoneme	0.7567	0.7530	0.2580	0.2201	0.6511	58.1494
ring	0.9608	0.8303	0.9315	0.6595	0.5230	150.5110
spambase	0.8336	0.8186	0.6403	0.5936	0.6711	90.7451
texture	0.8847	0.8831	0.8742	0.8714	0.6330	117.4850
titanic	0.6750	0.6820	0.0236	0.0245	0.7902	20.4498
twonorm	0.8566	0.8520	0.7232	0.7039	0.7399	184.9760

TABLA A.11: Resultados de RSS-GGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8316	0.7452	0.4495	0.1684	0.5853	490.8110
shuttle	0.9983	0.9985	0.9959	0.9958	0.8312	1903.2900

TABLA A.12: Resultados de RSS-GGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.6281	0.6667	0.4422	0.5000	0.8380	3.6577
cleveland	0.8638	0.5521	0.7875	0.3042	0.3933	7.1737
led7digit	0.6580	0.6000	0.6187	0.5526	0.5338	9.7518
pima	0.9100	0.6925	0.8002	0.3161	0.4408	17.4020
wdbc	0.9307	0.9275	0.8474	0.8418	0.7239	20.4897
monk2	0.8403	0.7683	0.6786	0.5332	0.6391	8.8170
wisconsin	0.9308	0.9190	0.8324	0.8039	0.7564	20.1675
wine	0.6436	0.7031	0.4442	0.5351	0.8416	5.8451
glass	0.6767	0.5648	0.5309	0.3691	0.6570	3.7065
banknote	0.8327	0.8440	0.6475	0.6723	0.9466	116.1180
appendicitis	0.4336	0.4575	-0.0203	-0.0329	0.8519	1.7216
balance	0.7175	0.6850	0.5520	0.4907	0.8419	24.1511
bands	0.8965	0.6867	0.7794	0.3293	0.4145	11.7027
contraceptive	0.8889	0.4386	0.8283	0.1333	0.2884	38.4136
dermatology	0.7799	0.7905	0.7184	0.7307	0.8101	13.5112
ecoli	0.9068	0.7804	0.8654	0.6772	0.5366	6.2739
haberman	0.8447	0.7222	0.6102	0.2778	0.4918	4.2080
hayes-roth	0.7438	0.6500	0.5903	0.4316	0.4844	1.6753
heart	0.8687	0.7222	0.7330	0.4367	0.5144	5.3110
hepatitis	0.6750	0.6500	0.2622	0.1001	0.7163	1.4540
mammographic	0.8582	0.7289	0.7161	0.4570	0.5357	17.7520
newthyroid	0.7647	0.7434	0.2932	0.2836	0.8488	5.5369
tae	0.6836	0.5096	0.5249	0.2564	0.4914	1.5996
vehicle	0.9108	0.6951	0.8810	0.5931	0.4475	27.9084
vowel	0.6726	0.6646	0.6399	0.6311	0.7999	35.2918
yeast	0.9344	0.5412	0.9148	0.4037	0.3317	38.6023

TABLA A.13: Resultados de CNN-RSS-GGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8739	0.8308	0.7500	0.6505	0.6487	76.9952
cardiotocography	0.9095	0.8802	0.7463	0.6270	0.6335	20.1874
eye-state	0.8796	0.7870	0.7624	0.5597	0.5011	214.1540
page-blocks	0.9520	0.9430	0.7694	0.6651	0.7332	82.0846
penbased	0.9808	0.9852	0.9797	0.9835	0.5059	183.8240
satimage	0.9151	0.8732	0.8981	0.8443	0.4645	167.4070
thyroid	0.9475	0.9071	0.7441	0.4328	0.7163	127.4800
segment	0.7420	0.1506	0.7007	0.0091	0.3236	21.6527
coil2000	0.9282	0.8717	0.5406	0.0891	0.6014	288.4990
magic	0.9017	0.8006	0.7898	0.5419	0.5272	267.3640
marketing	0.7786	0.2765	0.7492	0.1750	0.2644	86.4194
phoneme	0.8723	0.8347	0.6715	0.5446	0.5467	53.2051
ring	0.9709	0.8181	0.9518	0.6350	0.4729	128.3230
spambase	0.9388	0.8771	0.8811	0.7381	0.5176	103.3830
texture	0.9712	0.9698	0.9694	0.9668	0.5565	118.9760
titanic	0.6843	0.6910	0.0796	0.0818	0.7211	30.0544
twonorm	0.9382	0.9164	0.8864	0.8327	0.6395	134.3210

TABLA A.14: Resultados de CNN-RSS-GGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8922	0.7187	0.6845	0.1801	0.4155	507.4450
shuttle	0.9988	0.9989	0.9974	0.9968	0.8023	1609.8700

TABLA A.15: Resultados de CNN-RSS-GGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9630	0.9467	0.9444	0.9200	0.1340	1.4924
cleveland	0.8096	0.5459	0.6995	0.2790	0.2906	8.0685
led7digit	0.7038	0.6360	0.6693	0.5930	0.3022	10.2075
pima	0.8747	0.7025	0.7175	0.3308	0.2320	18.0607
wdbc	0.9701	0.9576	0.9348	0.9068	0.1351	39.1326
monk2	0.8843	0.7333	0.7665	0.4622	0.2285	8.5056
wisconsin	0.9774	0.9593	0.9500	0.9095	0.1291	17.6931
wine	0.9619	0.9471	0.9422	0.9205	0.1348	4.6680
glass	0.8577	0.7657	0.7982	0.6660	0.2369	4.0254
banknote	0.9980	0.9978	0.9959	0.9956	0.1018	26.8554
appendicitis	0.9602	0.9238	0.8214	0.6144	0.1481	1.2215
balance	0.9193	0.8466	0.8508	0.7211	0.2227	11.3091
bands	0.8959	0.7068	0.7714	0.3534	0.2093	14.3717
contraceptive	0.7880	0.4583	0.6708	0.1612	0.3020	38.5490
dermatology	0.9637	0.9548	0.9538	0.9423	0.1355	25.9527
ecoli	0.9111	0.8205	0.8712	0.7389	0.2012	6.9918
haberman	0.8627	0.7128	0.6145	0.1416	0.2448	4.0245
hayes-roth	0.8472	0.6500	0.7582	0.4311	0.2231	1.5606
heart	0.9058	0.8074	0.8091	0.6068	0.2044	7.3741
hepatitis	0.9431	0.8750	0.7678	0.3314	0.1613	1.9428
mammographic	0.8665	0.7554	0.7329	0.5107	0.2234	16.4364
newthyroid	0.9737	0.9504	0.9379	0.8781	0.1242	2.4817
tae	0.8021	0.5567	0.6999	0.3246	0.2417	1.6441
vehicle	0.8735	0.7045	0.8313	0.6058	0.2326	32.8126
vowel	0.9921	0.9798	0.9914	0.9778	0.1078	35.3741
yeast	0.8246	0.5715	0.7714	0.4405	0.2779	39.3012

TABLA A.16: Resultados de ENN-RSS-GGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.9534	0.8830	0.9159	0.7632	0.1627	39.4244
cardiotocography	0.9491	0.9085	0.8736	0.7388	0.1989	20.8852
eye-state	0.9507	0.8331	0.9105	0.6622	0.2012	204.8410
page-blocks	0.9725	0.9655	0.8916	0.8053	0.1363	57.0765
penbased	0.9883	0.9941	0.9881	0.9934	0.1079	169.4890
satimage	0.9493	0.8982	0.9397	0.8744	0.1695	148.3270
thyroid	0.9850	0.9457	0.9524	0.5330	0.1088	116.3590
segment	0.6367	0.1515	0.5778	0.0101	0.4432	20.5384
coil2000	0.9461	0.9223	0.4612	0.0722	0.1737	466.5530
magic	0.9156	0.8274	0.8216	0.6036	0.2259	239.1560
marketing	0.7100	0.2909	0.6706	0.1907	0.3459	86.5193
phoneme	0.9505	0.8967	0.8929	0.7445	0.1946	44.5310
ring	0.9572	0.7403	0.9243	0.4778	0.1505	112.5310
spambase	0.9611	0.9086	0.9291	0.8071	0.1779	142.4090
texture	0.9843	0.9867	0.9837	0.9854	0.1262	143.2200
titanic	0.7520	0.7596	0.3175	0.3208	0.3143	9.3336
twonorm	0.9785	0.9514	0.9670	0.9027	0.1296	116.0250

TABLA A.17: Resultados de ENN-RSS-GGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.9158	0.7605	0.7359	0.2502	0.1812	563.7580
shuttle	0.9990	0.9991	0.9978	0.9974	0.0239	798.9830

TABLA A.18: Resultados de ENN-RSS-GGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9719	0.9467	0.9578	0.9200	0.9167	0.1425
cleveland	0.7433	0.5907	0.5781	0.3256	0.8357	0.2904
led7digit	0.7111	0.6220	0.6783	0.5789	0.8304	0.3914
pima	0.8255	0.7143	0.6052	0.3413	0.8090	0.9706
wdbc	0.9744	0.9472	0.9445	0.8847	0.8499	0.8070
monk2	0.9069	0.7849	0.8135	0.5688	0.8164	0.4609
wisconsin	0.9777	0.9593	0.9508	0.9099	0.8466	0.8056
wine	0.9862	0.9471	0.9788	0.9187	0.9112	0.1887
glass	0.8510	0.7061	0.7855	0.5725	0.8374	0.2000
banknote	0.9995	0.9978	0.9990	0.9956	0.8233	1.9947
appendicitis	0.9707	0.9338	0.8649	0.6594	0.9519	0.1038
balance	0.8780	0.8420	0.7765	0.7118	0.8299	0.6654
bands	0.8143	0.6714	0.5932	0.2956	0.8222	0.3919
contraceptive	0.6222	0.4354	0.4154	0.1299	0.7909	2.3099
dermatology	0.9737	0.9427	0.9667	0.9272	0.8665	0.4319
ecoli	0.8783	0.7956	0.8234	0.7013	0.8503	0.3654
haberman	0.8297	0.7556	0.4794	0.2531	0.8471	0.2898
hayes-roth	0.7875	0.5813	0.6665	0.3331	0.8156	0.1207
heart	0.8914	0.8111	0.7794	0.6165	0.8556	0.2568
hepatitis	0.9153	0.7625	0.6882	0.2213	0.9300	0.0680
mammographic	0.8491	0.7855	0.6979	0.5704	0.8157	1.0376
newthyroid	0.9721	0.9628	0.9328	0.9065	0.8981	0.1924
tae	0.7520	0.5046	0.6276	0.2569	0.8278	0.1282
vehicle	0.7992	0.6505	0.7322	0.5338	0.7987	1.1757
vowel	0.9398	0.8838	0.9338	0.8722	0.7589	1.2075
yeast	0.6798	0.5276	0.5812	0.3804	0.7868	2.3061

TABLA A.19: Resultados de SSGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.9009	0.8797	0.7996	0.7566	0.8460	2.0155
cardiotocography	0.9010	0.8794	0.7177	0.6550	0.8952	0.8157
eye-state	0.8318	0.7782	0.6597	0.5512	0.7998	8.1422
page-blocks	0.9607	0.9511	0.7809	0.7240	0.8591	2.3354
penbased	0.9910	0.9868	0.9900	0.9853	0.8266	4.9590
satimage	0.9130	0.8916	0.8924	0.8659	0.8399	2.9289
thyroid	0.9471	0.9370	0.5434	0.4430	0.8547	3.4614
segment	0.3352	0.1377	0.2244	-0.0061	0.8025	0.8695
coil2000	0.9291	0.9155	0.2023	0.0364	0.8408	7.0551
magic	0.8479	0.8044	0.6583	0.5567	0.8081	11.0371
marketing	0.4385	0.2958	0.3576	0.1942	0.8008	3.1245
phoneme	0.8770	0.8431	0.7047	0.6212	0.8438	2.1808
ring	0.8423	0.7698	0.6837	0.5376	0.8069	3.3822
spambase	0.9028	0.8821	0.7962	0.7531	0.8468	2.1174
texture	0.9719	0.9648	0.9691	0.9613	0.8382	2.3282
titanic	0.7844	0.7838	0.4532	0.4497	0.9170	0.4438
twonorm	0.9581	0.9480	0.9162	0.8960	0.8395	3.3044

TABLA A.20: Resultados de SSGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7834	0.7498	0.3935	0.2439	0.8070	17.8910
shuttle	0.9988	0.9987	0.9971	0.9963	0.8041	37.4363

TABLA A.21: Resultados de SSGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9741	0.9400	0.9611	0.9100	0.9407	0.1854
cleveland	0.8058	0.5818	0.6817	0.3238	0.7593	0.3769
led7digit	0.6373	0.5680	0.5959	0.5179	0.7878	0.5826
pima	0.8325	0.6664	0.6258	0.2451	0.7874	1.1970
wdbc	0.9691	0.9362	0.9333	0.8622	0.9364	1.5751
monk2	0.9059	0.8447	0.8116	0.6898	0.8435	0.5582
wisconsin	0.9764	0.9579	0.9481	0.9075	0.9621	5.2854
wine	0.9813	0.9548	0.9716	0.9310	0.9438	0.2875
glass	0.9081	0.7267	0.8674	0.6058	0.7790	0.2505
banknote	0.9973	0.9905	0.9946	0.9808	0.9780	6.2513
appendicitis	0.9759	0.9600	0.8770	0.6889	0.9585	0.1973
balance	0.8669	0.7695	0.7620	0.5936	0.8496	1.1170
bands	0.8499	0.6426	0.6746	0.2241	0.7751	0.5191
contraceptive	0.6974	0.4489	0.5335	0.1527	0.6686	2.8633
dermatology	0.9659	0.9278	0.9571	0.9089	0.9254	0.8536
ecoli	0.9025	0.8114	0.8576	0.7221	0.8435	0.4702
haberman	0.8453	0.6850	0.5383	0.0956	0.8183	0.3448
hayes-roth	0.7889	0.6313	0.6684	0.4175	0.7813	0.1321
heart	0.8934	0.7667	0.7839	0.5287	0.8581	0.3429
hepatitis	0.9167	0.8500	0.5894	0.3057	0.9550	0.0871
mammographic	0.8481	0.7422	0.6960	0.4842	0.8157	1.4781
newthyroid	0.9840	0.9361	0.9624	0.8492	0.9442	0.4025
tae	0.7991	0.5700	0.6965	0.3420	0.7722	0.1270
vehicle	0.8318	0.6569	0.7755	0.5421	0.7615	1.6007
vowel	0.9608	0.9202	0.9569	0.9122	0.7652	1.6396
yeast	0.7415	0.5222	0.6636	0.3770	0.6951	2.9597

TABLA A.22: Resultados de CNN-SSGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8885	0.8645	0.7846	0.7259	0.8845	3.2397
cardiotocography	0.8965	0.8788	0.7236	0.6507	0.9112	1.2179
eye-state	0.8339	0.7759	0.6746	0.5472	0.7736	11.8877
page-blocks	0.9549	0.9532	0.7970	0.7406	0.9531	10.7988
penbased	0.9773	0.9826	0.9759	0.9807	0.9243	10.0667
satimage	0.9108	0.8825	0.8922	0.8551	0.8776	5.3487
thyroid	0.9257	0.9158	0.5344	0.4023	0.9175	6.8629
segment	0.4875	0.1455	0.4037	0.0030	0.6338	0.8590
coil2000	0.8935	0.8766	0.2694	0.0713	0.8878	11.5184
magic	0.8408	0.7857	0.6621	0.5253	0.8168	19.2789
marketing	0.5485	0.2730	0.4871	0.1712	0.6361	3.6459
phoneme	0.8794	0.8488	0.7219	0.6320	0.8583	4.4701
ring	0.9086	0.8707	0.8270	0.7409	0.8571	7.7641
spambase	0.9001	0.8819	0.8018	0.7521	0.8753	2.7725
texture	0.9668	0.9667	0.9645	0.9634	0.8820	3.4171
titanic	0.7687	0.7755	0.4367	0.4387	0.9235	0.6296
twonorm	0.9368	0.9339	0.8836	0.8678	0.9175	5.7192

TABLA A.23: Resultados de CNN-SSGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7810	0.7039	0.4041	0.1919	0.7816	21.0698
shuttle	0.9984	0.9986	0.9963	0.9960	0.9797	189.1310

TABLA A.24: Resultados de CNN-SSGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9674	0.9600	0.9511	0.9400	0.8693	0.1472
cleveland	0.7258	0.6011	0.5452	0.3391	0.8253	0.4247
led7digit	0.7618	0.6880	0.7346	0.6516	0.7902	0.6063
pima	0.8420	0.7403	0.6424	0.4055	0.7372	1.1757
wdbc	0.9799	0.9597	0.9564	0.9119	0.7274	0.9593
monk2	0.8987	0.7587	0.7964	0.5142	0.7498	0.5481
wisconsin	0.9793	0.9593	0.9543	0.9087	0.7158	0.9413
wine	0.9776	0.9588	0.9658	0.9375	0.8472	0.2131
glass	0.8588	0.6863	0.7976	0.5411	0.7921	0.2460
banknote	0.9997	0.9986	0.9993	0.9971	0.6555	2.5586
appendicitis	0.9476	0.9238	0.7417	0.5986	0.9245	0.0900
balance	0.8924	0.8675	0.8001	0.7526	0.7298	0.7682
bands	0.8313	0.6850	0.6312	0.3014	0.7701	0.4592
contraceptive	0.6334	0.4609	0.4340	0.1708	0.7903	3.2616
dermatology	0.9702	0.9387	0.9624	0.9226	0.7684	0.5990
ecoli	0.8939	0.8422	0.8460	0.7688	0.7890	0.4199
haberman	0.8279	0.7311	0.4790	0.1877	0.8108	0.3237
hayes-roth	0.8167	0.7063	0.7128	0.5367	0.7875	0.1310
heart	0.9008	0.8185	0.7980	0.6307	0.8019	0.3099
hepatitis	0.9236	0.8375	0.6029	0.1971	0.9425	0.0762
mammographic	0.8625	0.7855	0.7249	0.5703	0.7265	1.2844
newthyroid	0.9783	0.9399	0.9487	0.8453	0.8260	0.2164
tae	0.7550	0.5438	0.6298	0.3072	0.8278	0.1260
vehicle	0.8144	0.6929	0.7526	0.5905	0.7273	1.4909
vowel	0.9787	0.9343	0.9765	0.9278	0.6441	1.7673
yeast	0.6890	0.5701	0.5911	0.4355	0.7624	3.2192

TABLA A.25: Resultados de ENN-SSGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8991	0.8864	0.8065	0.7708	0.7288	2.1969
cardiotocography	0.8910	0.8766	0.7018	0.6409	0.8060	0.8030
eye-state	0.8455	0.8005	0.6997	0.5990	0.7017	13.5620
page-blocks	0.9597	0.9620	0.8156	0.7792	0.7061	2.5222
penbased	0.9824	0.9908	0.9816	0.9898	0.6497	8.3334
satimage	0.9037	0.8884	0.8833	0.8620	0.7211	4.5211
thyroid	0.9397	0.9456	0.4656	0.4177	0.6880	4.4753
segment	0.2051	0.1325	0.0742	-0.0121	0.9342	1.2889
coil2000	0.9307	0.9321	0.1334	0.0155	0.6953	16.2075
magic	0.8506	0.8219	0.6713	0.5890	0.6946	19.2229
marketing	0.3668	0.2908	0.2718	0.1827	0.8889	4.5905
phoneme	0.8783	0.8540	0.7228	0.6493	0.7404	2.3454
ring	0.6895	0.6382	0.3855	0.2716	0.7558	4.1515
spambase	0.8997	0.8862	0.7988	0.7587	0.7447	3.3582
texture	0.9684	0.9733	0.9662	0.9706	0.6984	3.7807
titanic	0.7742	0.7819	0.4504	0.4549	0.8467	0.5041
twonorm	0.9562	0.9545	0.9225	0.9089	0.6800	4.5633

TABLA A.26: Resultados de ENN-SSGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8205	0.7796	0.4251	0.2915	0.7354	21.8477
shuttle	0.9989	0.9988	0.9975	0.9966	0.6292	47.9571

TABLA A.27: Resultados de ENN-SSGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9763	0.9667	0.9644	0.9500	0.9527	0.2457
cleveland	0.7247	0.5631	0.5525	0.3053	0.8508	0.3319
led7digit	0.7180	0.6620	0.6859	0.6230	0.9114	0.6265
pima	0.8157	0.7164	0.5872	0.3737	0.8465	1.2026
wdbc	0.9643	0.9413	0.9217	0.8718	0.9545	1.3163
monk2	0.8809	0.8167	0.7623	0.6332	0.9153	0.6187
wisconsin	0.9745	0.9651	0.9436	0.9237	0.9630	1.9268
wine	0.9862	0.9725	0.9792	0.9582	0.9573	0.2952
glass	0.8308	0.7267	0.7611	0.6090	0.8710	0.2159
banknote	0.9960	0.9956	0.9920	0.9911	0.9622	4.4683
appendicitis	0.9623	0.9138	0.8455	0.6281	0.9623	0.1313
balance	0.8597	0.8335	0.7399	0.6901	0.9483	1.3446
bands	0.8149	0.6245	0.5968	0.1818	0.8323	0.4596
contraceptive	0.6243	0.4583	0.4174	0.1633	0.7886	2.9997
dermatology	0.9659	0.9364	0.9566	0.9194	0.9453	0.6027
ecoli	0.8641	0.7814	0.8034	0.6854	0.8979	0.4327
haberman	0.8094	0.7189	0.3919	0.0827	0.9007	0.3774
hayes-roth	0.7903	0.5938	0.6735	0.3597	0.8531	0.1580
heart	0.8774	0.7963	0.7516	0.5891	0.9052	0.3431
hepatitis	0.9222	0.8875	0.6251	0.4514	0.9638	0.0922
mammographic	0.8380	0.7916	0.6758	0.5825	0.8937	1.4353
newthyroid	0.9716	0.9485	0.9274	0.8601	0.9600	0.3301
tae	0.7586	0.4708	0.6371	0.1988	0.8238	0.1111
vehicle	0.7849	0.6482	0.7130	0.5304	0.8395	1.4432
vowel	0.9210	0.8384	0.9131	0.8222	0.7868	1.6671
yeast	0.6775	0.5197	0.5787	0.3699	0.8041	3.1249

TABLA A.28: Resultados de RSS-SSGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8901	0.8908	0.7860	0.7769	0.8852	2.8499
cardiotocography	0.8781	0.8771	0.6600	0.6389	0.9514	1.1030
eye-state	0.8008	0.7667	0.6007	0.5196	0.8122	13.0538
page-blocks	0.9450	0.9512	0.7343	0.7219	0.9671	5.5642
penbased	0.9805	0.9885	0.9795	0.9873	0.8067	8.6334
satimage	0.8792	0.8716	0.8536	0.8420	0.8119	5.1614
thyroid	0.9050	0.8997	0.4760	0.3767	0.9155	5.7662
segment	0.3390	0.1498	0.2305	0.0081	0.7916	0.9479
coil2000	0.9204	0.9246	0.1088	0.0262	0.9177	12.1242
magic	0.8165	0.8027	0.5834	0.5351	0.8615	19.2977
marketing	0.4480	0.2737	0.3721	0.1717	0.7637	3.9325
phoneme	0.8366	0.8301	0.5915	0.5566	0.8713	2.8382
ring	0.9224	0.8915	0.8547	0.7827	0.8250	7.4582
spambase	0.8903	0.8849	0.7799	0.7573	0.8722	2.9937
texture	0.9568	0.9602	0.9534	0.9562	0.8459	3.3894
titanic	0.7629	0.7696	0.4047	0.4065	0.9804	1.2767
twonorm	0.9410	0.9441	0.8920	0.8881	0.8932	5.5177

TABLA A.29: Resultados de RSS-SSGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7956	0.7669	0.3534	0.2589	0.8467	22.9713
shuttle	0.9980	0.9983	0.9951	0.9953	0.9387	92.6277

TABLA A.30: Resultados de RSS-SSGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9719	0.9067	0.9578	0.8600	0.9440	0.1770
cleveland	0.7897	0.5900	0.6680	0.3612	0.7660	0.3547
led7digit	0.6960	0.6360	0.6611	0.5938	0.7896	0.5439
pima	0.8491	0.6987	0.6628	0.3313	0.7544	1.1701
wdbc	0.9690	0.9436	0.9326	0.8779	0.9295	1.2770
monk2	0.9051	0.8259	0.8102	0.6525	0.8477	0.5597
wisconsin	0.9749	0.9678	0.9448	0.9287	0.9416	1.6467
wine	0.9813	0.9529	0.9716	0.9287	0.9388	0.2556
glass	0.8988	0.7577	0.8527	0.6446	0.7921	0.2154
banknote	0.9988	0.9956	0.9975	0.9911	0.9557	4.2250
appendicitis	0.9769	0.9538	0.8837	0.6995	0.9528	0.1550
balance	0.8661	0.8038	0.7580	0.6512	0.8680	1.1141
bands	0.8429	0.6627	0.6643	0.2748	0.7595	0.4785
contraceptive	0.6965	0.4299	0.5307	0.1265	0.6687	2.9117
dermatology	0.9733	0.9398	0.9658	0.9223	0.9176	0.6511
ecoli	0.8972	0.8068	0.8509	0.7170	0.8262	0.4092
haberman	0.8490	0.6972	0.5479	0.1525	0.8183	0.3336
hayes-roth	0.8132	0.6625	0.7068	0.4650	0.7725	0.1315
heart	0.8951	0.7889	0.7874	0.5744	0.8378	0.3180
hepatitis	0.9208	0.8125	0.6281	0.3032	0.9475	0.0752
mammographic	0.8518	0.7675	0.7032	0.5344	0.8037	1.2999
newthyroid	0.9850	0.9571	0.9637	0.8975	0.9447	0.3174
tae	0.7918	0.5304	0.6832	0.2761	0.7755	0.1328
vehicle	0.8264	0.6728	0.7684	0.5635	0.7580	1.4625
vowel	0.9357	0.8566	0.9293	0.8422	0.7768	1.6761
yeast	0.7448	0.5117	0.6676	0.3623	0.6821	2.9498

TABLA A.31: Resultados de CNN-RSS-SSGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.9000	0.8734	0.8081	0.7442	0.8382	2.5317
cardiotocography	0.8969	0.8868	0.7267	0.6805	0.9040	0.9384
eye-state	0.8451	0.7846	0.6955	0.5618	0.7367	11.6751
page-blocks	0.9573	0.9565	0.8093	0.7561	0.9377	4.9004
penbased	0.9825	0.9902	0.9817	0.9891	0.7955	7.5879
satimage	0.9093	0.8825	0.8905	0.8551	0.7698	4.3521
thyroid	0.9283	0.9178	0.5506	0.4085	0.8944	5.1864
segment	0.4875	0.1494	0.4038	0.0076	0.6400	0.8468
coil2000	0.9012	0.8839	0.2681	0.0486	0.8554	10.9299
magic	0.8494	0.7970	0.6760	0.5430	0.7777	17.0458
marketing	0.5589	0.2727	0.4990	0.1714	0.6266	3.5809
phoneme	0.8828	0.8472	0.7270	0.6239	0.8041	2.4486
ring	0.9165	0.8703	0.8428	0.7400	0.7973	6.4466
spambase	0.9046	0.8869	0.8102	0.7617	0.8179	2.6913
texture	0.9709	0.9716	0.9689	0.9688	0.8216	3.1957
titanic	0.7669	0.7760	0.4271	0.4354	0.9272	0.7884
twonorm	0.9387	0.9326	0.8875	0.8651	0.8609	4.8417

TABLA A.32: Resultados de CNN-RSS-SSGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8019	0.7282	0.4316	0.2178	0.7528	20.3867
shuttle	0.9985	0.9986	0.9964	0.9959	0.9341	78.7762

TABLA A.33: Resultados de CNN-RSS-SSGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9719	0.9400	0.9578	0.9100	0.8660	0.1378
cleveland	0.7714	0.5521	0.6285	0.2937	0.7741	0.3678
led7digit	0.7518	0.7080	0.7235	0.6742	0.7688	0.5984
pima	0.8511	0.7071	0.6623	0.3430	0.7053	1.0767
wdbc	0.9814	0.9650	0.9600	0.9244	0.7286	0.9349
monk2	0.9108	0.8679	0.8213	0.7356	0.7308	0.4828
wisconsin	0.9834	0.9593	0.9632	0.9099	0.7044	1.1348
wine	0.9812	0.9332	0.9714	0.8985	0.8534	0.2051
glass	0.8858	0.7394	0.8314	0.6143	0.7715	0.2164
banknote	0.9998	0.9978	0.9995	0.9956	0.6522	2.6565
appendicitis	0.9581	0.9338	0.7988	0.6732	0.9255	0.0928
balance	0.8928	0.8572	0.8017	0.7367	0.7210	0.7768
bands	0.8444	0.6999	0.6599	0.3276	0.7326	0.4967
contraceptive	0.6781	0.4434	0.5010	0.1394	0.6894	2.9036
dermatology	0.9811	0.9553	0.9758	0.9429	0.7623	0.5566
ecoli	0.9018	0.7977	0.8564	0.7027	0.7619	0.3897
haberman	0.8464	0.7283	0.5532	0.2168	0.7784	0.2862
hayes-roth	0.8229	0.6938	0.7211	0.5151	0.7706	0.1423
heart	0.8979	0.7963	0.7930	0.5848	0.7889	0.3207
hepatitis	0.9097	0.8000	0.6258	0.1505	0.9188	0.0698
mammographic	0.8581	0.7904	0.7159	0.5801	0.7120	1.1591
newthyroid	0.9788	0.9533	0.9491	0.8851	0.8228	0.2067
tae	0.7741	0.5163	0.6589	0.2699	0.7742	0.1266
vehicle	0.8415	0.6824	0.7886	0.5763	0.6903	1.4604
vowel	0.9786	0.9384	0.9764	0.9322	0.6408	1.7282
yeast	0.7376	0.5568	0.6558	0.4158	0.6778	3.0048

TABLA A.34: Resultados de ENN-RSS-SSGA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.9097	0.8842	0.8273	0.7651	0.7089	2.0860
cardiotocography	0.9044	0.8889	0.7396	0.6723	0.7899	0.7867
eye-state	0.8605	0.8037	0.7280	0.6027	0.6566	12.4354
page-blocks	0.9618	0.9644	0.8329	0.7988	0.7063	2.7204
penbased	0.9840	0.9904	0.9833	0.9893	0.6511	8.0300
satimage	0.9152	0.8953	0.8976	0.8707	0.6979	4.2810
thyroid	0.9529	0.9424	0.6829	0.4985	0.6739	4.2335
segment	0.3945	0.1390	0.2952	-0.0045	0.7362	0.8872
coil2000	0.9282	0.9192	0.2377	0.0296	0.6844	13.9876
magic	0.8613	0.8201	0.6963	0.5853	0.6673	17.2626
marketing	0.4879	0.2724	0.4171	0.1692	0.7186	3.6747
phoneme	0.8945	0.8664	0.7535	0.6678	0.7110	2.2860
ring	0.8581	0.7457	0.7256	0.4890	0.6467	4.4507
spambase	0.9139	0.8884	0.8297	0.7645	0.7159	3.0871
texture	0.9740	0.9758	0.9724	0.9734	0.6861	3.5499
titanic	0.7798	0.7883	0.4323	0.4388	0.8499	0.5530
twonorm	0.9547	0.9481	0.9194	0.8962	0.6822	4.3375

TABLA A.35: Resultados de ENN-RSS-SSGA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8234	0.7621	0.4512	0.2574	0.7019	20.4495
shuttle	0.9990	0.9989	0.9979	0.9969	0.6318	58.1472

TABLA A.36: Resultados de ENN-RSS-SSGA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9770	0.9067	0.9656	0.8600	0.9740	0.2593
cleveland	0.7171	0.5838	0.5237	0.3102	0.9306	0.9515
led7digit	0.7644	0.7420	0.7376	0.7122	0.9696	1.2289
pima	0.8131	0.7659	0.5680	0.4569	0.9674	6.3130
wdbc	0.9734	0.9505	0.9423	0.8921	0.9830	3.5505
monk2	0.9532	0.9328	0.9062	0.8653	0.9757	2.4957
wisconsin	0.9766	0.9578	0.9486	0.9073	0.9927	7.3022
wine	0.9831	0.9292	0.9744	0.8933	0.9680	0.3459
glass	0.7996	0.6545	0.7033	0.4836	0.9145	0.5640
banknote	0.9982	0.9978	0.9964	0.9956	0.9892	12.6341
appendicitis	0.9644	0.9600	0.8391	0.7352	0.9745	0.3151
balance	0.8843	0.8627	0.7835	0.7431	0.9738	3.8908
bands	0.7885	0.6582	0.5250	0.2308	0.9449	1.2039
contraceptive	0.6194	0.4819	0.4061	0.1940	0.9452	18.6463
dermatology	0.9746	0.9559	0.9677	0.9438	0.9645	1.2560
ecoli	0.8608	0.8012	0.7961	0.7062	0.9625	1.3572
haberman	0.8018	0.7089	0.3817	0.1279	0.9768	1.6556
hayes-roth	0.7688	0.6688	0.6381	0.4826	0.9088	0.2636
heart	0.8811	0.8296	0.7572	0.6524	0.9696	0.8221
hepatitis	0.9222	0.8750	0.6535	0.3771	0.9638	0.1015
mammographic	0.8459	0.8265	0.6914	0.6529	0.9854	9.7776
newthyroid	0.9747	0.9313	0.9417	0.8320	0.9735	0.7552
tae	0.6799	0.5567	0.5189	0.3293	0.9106	0.2531
vehicle	0.7965	0.6621	0.7286	0.5490	0.9221	5.1786
vowel	0.8875	0.8253	0.8763	0.8078	0.8701	5.6765
yeast	0.6766	0.5623	0.5732	0.4211	0.9472	19.9237

TABLA A.37: Resultados de MA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8886	0.8846	0.7740	0.7658	0.9484	27.6695
cardiotocography	0.8783	0.8682	0.6553	0.6240	0.9869	3.1308
eye-state	0.7385	0.7190	0.4705	0.4308	0.9717	143.5858
page-blocks	0.9423	0.9394	0.6565	0.6373	0.9959	16.9769
penbased	0.9826	0.9809	0.9807	0.9788	0.9032	448.9862
satimage	0.8715	0.8628	0.8410	0.8303	0.9672	35.6554
thyroid	0.9447	0.9437	0.4543	0.4416	0.9954	25.3644
segment	0.2546	0.1546	0.1304	0.0136	0.8941	3.3403
coil2000	0.9358	0.9340	0.0739	0.0454	0.9975	81.9335
magic	0.8206	0.8078	0.6008	0.5718	0.9699	237.1728
marketing	0.3404	0.2877	0.2414	0.1808	0.9419	25.6661
phoneme	0.8127	0.8007	0.5439	0.5128	0.9817	16.7353
ring	0.7169	0.7041	0.4306	0.4047	0.9704	32.5529
spambase	0.8696	0.8617	0.7279	0.7113	0.9854	9.3396
texture	0.9623	0.9586	0.9586	0.9545	0.8660	110.9760
titanic	0.7839	0.7828	0.4563	0.4506	0.9902	3.0523
twonorm	0.9534	0.9523	0.9067	0.9046	0.9954	24.7465

TABLA A.38: Resultados de MA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8043	0.8035	0.3163	0.3128	0.9970	97.4400
shuttle	0.9764	0.9962	0.9835	0.9892	0.9976	414.8464

TABLA A.39: Resultados de MA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9593	0.9533	0.9389	0.9300	0.9713	0.2560
cleveland	0.7340	0.5976	0.5674	0.3410	0.9178	1.1617
led7digit	0.7636	0.7640	0.7369	0.7370	0.9688	1.7061
pima	0.8181	0.7408	0.5853	0.4129	0.9616	6.3759
wdbc	0.9692	0.9517	0.9332	0.8968	0.9835	4.3126
monk2	0.9601	0.9513	0.9201	0.9022	0.9722	2.2859
wisconsin	0.9749	0.9666	0.9449	0.9272	0.9918	10.8397
wine	0.9670	0.9235	0.9497	0.8824	0.9674	0.4105
glass	0.8380	0.7259	0.7628	0.5999	0.9005	0.5582
banknote	0.9960	0.9949	0.9918	0.9897	0.9897	19.3378
appendicitis	0.9622	0.9338	0.8197	0.6226	0.9726	0.3812
balance	0.8704	0.8306	0.7579	0.6843	0.9618	4.4479
bands	0.7952	0.6745	0.5346	0.2540	0.9318	1.4442
contraceptive	0.6397	0.4874	0.4369	0.1983	0.9238	34.5453
dermatology	0.9711	0.9359	0.9635	0.9185	0.9620	1.6582
ecoli	0.8770	0.8189	0.8208	0.7371	0.9438	1.6602
haberman	0.7988	0.7850	0.3414	0.3042	0.9830	1.6235
hayes-roth	0.7951	0.6625	0.6817	0.4761	0.9063	0.3040
heart	0.8695	0.8074	0.7336	0.6065	0.9748	0.9108
hepatitis	0.9250	0.8750	0.6103	0.3771	0.9625	0.1547
mammographic	0.8473	0.8205	0.6945	0.6408	0.9807	10.7792
newthyroid	0.9731	0.9408	0.9343	0.8521	0.9767	0.9724
tae	0.7277	0.5238	0.5888	0.2769	0.8795	0.2448
vehicle	0.8039	0.6320	0.7386	0.5085	0.9063	6.7557
vowel	0.9759	0.9141	0.9735	0.9056	0.8363	10.6054
yeast	0.6867	0.5546	0.5887	0.4161	0.9371	31.3284

TABLA A.40: Resultados de CNN-MA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8740	0.8594	0.7565	0.7170	0.9243	23.6071
cardiotocography	0.8781	0.8798	0.6685	0.6528	0.9776	3.6315
eye-state	0.7818	0.7592	0.5695	0.5136	0.9208	184.2650
page-blocks	0.9395	0.9461	0.6854	0.6734	0.9905	29.4964
penbased	0.9802	0.9871	0.9791	0.9856	0.9392	109.8440
satimage	0.8773	0.8664	0.8510	0.8352	0.9362	36.4186
thyroid	0.9321	0.9408	0.4125	0.4144	0.9969	30.0824
segment	0.1780	0.1489	0.0427	0.0071	0.9731	3.7275
coil2000	0.9033	0.9409	0.0000	0.0000	0.9991	74.6177
magic	0.7938	0.7951	0.5462	0.5362	0.9934	263.4050
marketing	0.4412	0.2817	0.3632	0.1794	0.8123	57.3855
phoneme	0.8260	0.8223	0.5917	0.5671	0.9715	21.4618
ring	0.8409	0.8426	0.6912	0.6844	0.9867	49.4540
spambase	0.8803	0.8743	0.7595	0.7359	0.9675	11.3417
texture	0.9570	0.9625	0.9537	0.9588	0.9030	48.3416
titanic	0.7763	0.7842	0.4455	0.4498	0.9869	3.5772
twonorm	0.9366	0.9449	0.8832	0.8897	0.9942	25.0392

TABLA A.41: Resultados de CNN-MA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7874	0.7867	0.2759	0.2721	0.9969	98.1104
shuttle	0.9976	0.9986	0.9962	0.9961	0.9800	1127.4700

TABLA A.42: Resultados de CNN-MA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9741	0.9267	0.9611	0.8900	0.9707	0.2700
cleveland	0.7042	0.5521	0.5089	0.2664	0.9461	0.8643
led7digit	0.7660	0.7300	0.7393	0.6991	0.9708	1.4853
pima	0.8289	0.7174	0.6114	0.3523	0.9574	7.0557
wdbc	0.9766	0.9416	0.9493	0.8743	0.9821	3.9045
monk2	0.9072	0.8562	0.8146	0.7135	0.9743	2.2931
wisconsin	0.9776	0.9608	0.9504	0.9131	0.9918	9.4304
wine	0.9832	0.9666	0.9745	0.9492	0.9708	0.3391
glass	0.8084	0.7291	0.7159	0.5946	0.9318	0.4199
banknote	0.9989	0.9964	0.9977	0.9926	0.9894	12.2694
appendicitis	0.9665	0.9375	0.8231	0.6294	0.9811	0.3498
balance	0.8905	0.8769	0.7944	0.7688	0.9733	3.8255
bands	0.8162	0.6680	0.5940	0.2643	0.9332	1.3530
contraceptive	0.6406	0.4773	0.4409	0.1889	0.9330	16.9860
dermatology	0.9789	0.9289	0.9733	0.9098	0.9690	1.5115
ecoli	0.8926	0.8254	0.8429	0.7448	0.9417	1.2752
haberman	0.8134	0.7578	0.3781	0.1659	0.9745	1.6995
hayes-roth	0.7847	0.6813	0.6616	0.4987	0.9038	0.2399
heart	0.8840	0.8037	0.7630	0.5949	0.9607	0.8990
hepatitis	0.9208	0.8625	0.6728	0.4790	0.9588	0.1211
mammographic	0.8535	0.8145	0.7067	0.6281	0.9793	8.4882
newthyroid	0.9736	0.9551	0.9396	0.8972	0.9753	0.8561
tae	0.7182	0.5104	0.5728	0.2575	0.9139	0.2509
vehicle	0.8080	0.6633	0.7440	0.5511	0.9204	6.3843
vowel	0.9774	0.9111	0.9752	0.9022	0.8437	9.8755
yeast	0.6891	0.5681	0.5917	0.4336	0.9423	18.0281

TABLA A.43: Resultados de ENN-MA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8633	0.8692	0.7302	0.7319	0.9896	17.1335
cardiotocography	0.8858	0.8889	0.6940	0.6839	0.9746	3.0763
eye-state	0.7977	0.7742	0.6026	0.5449	0.9110	270.1560
page-blocks	0.9342	0.9408	0.6493	0.6349	0.9959	16.5685
penbased	0.9723	0.9798	0.9703	0.9776	0.9655	109.4700
satimage	0.8632	0.8615	0.8334	0.8289	0.9789	32.1135
thyroid	0.9365	0.9440	0.4528	0.4337	0.9964	30.7640
segment	0.1909	0.1489	0.0577	0.0071	0.9684	3.8059
coil2000	0.9099	0.9146	0.0526	0.0135	0.9952	91.0547
magic	0.8211	0.8115	0.6112	0.5762	0.9630	316.5210
marketing	0.2945	0.2837	0.1790	0.1649	0.9923	34.3821
phoneme	0.7932	0.7953	0.5156	0.5060	0.9915	17.3694
ring	0.6782	0.6812	0.3626	0.3585	0.9962	37.9116
spambase	0.8732	0.8777	0.7438	0.7424	0.9873	12.6539
texture	0.9563	0.9616	0.9529	0.9578	0.8216	113.3510
titanic	0.7730	0.7814	0.4506	0.4552	0.9894	3.5486
twonorm	0.9509	0.9577	0.9119	0.9154	0.9846	38.1463

TABLA A.44: Resultados de ENN-MA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7956	0.7950	0.2746	0.2715	0.9971	103.3980
shuttle	0.9658	0.9854	0.9398	0.9575	0.9982	473.9070

TABLA A.45: Resultados de ENN-MA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9726	0.9400	0.9589	0.9100	0.9727	0.2681
cleveland	0.6992	0.6059	0.4858	0.3448	0.9556	1.0823
led7digit	0.7669	0.7280	0.7407	0.6967	0.9742	1.4398
pima	0.8080	0.7400	0.5662	0.4072	0.9727	6.1474
wdbc	0.9670	0.9398	0.9286	0.8712	0.9919	3.9374
monk2	0.9264	0.8863	0.8527	0.7721	0.9782	2.3380
wisconsin	0.9769	0.9665	0.9492	0.9264	0.9937	8.1878
wine	0.9737	0.9294	0.9600	0.8931	0.9787	0.3840
glass	0.7976	0.6688	0.7010	0.5058	0.9322	0.4620
banknote	0.9945	0.9934	0.9888	0.9867	0.9914	16.3687
appendicitis	0.9434	0.8938	0.7176	0.4961	0.9717	0.2145
balance	0.8823	0.8770	0.7797	0.7696	0.9891	4.7406
bands	0.7808	0.6224	0.5058	0.1368	0.9496	1.3912
contraceptive	0.6280	0.4801	0.4201	0.1944	0.9347	20.4189
dermatology	0.9450	0.9146	0.9297	0.8913	0.9768	1.4266
ecoli	0.8568	0.8189	0.7914	0.7371	0.9560	1.2293
haberman	0.8101	0.7300	0.4011	0.1513	0.9752	1.7327
hayes-roth	0.6833	0.5875	0.5102	0.3519	0.9406	0.3744
heart	0.8691	0.8333	0.7328	0.6602	0.9778	0.8880
hepatitis	0.8958	0.8500	0.4275	0.1914	0.9750	0.0999
mammographic	0.8463	0.8313	0.6924	0.6623	0.9878	8.7594
newthyroid	0.9498	0.9275	0.8730	0.8136	0.9847	0.7880
tae	0.7042	0.5300	0.5526	0.2789	0.9026	0.2229
vehicle	0.7782	0.6560	0.7043	0.5415	0.9349	6.7902
vowel	0.8299	0.7475	0.8128	0.7222	0.8881	11.5964
yeast	0.6746	0.5689	0.5729	0.4341	0.9510	19.3274

TABLA A.46: Resultados de RSS-MA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8694	0.8760	0.7437	0.7468	0.9740	23.4139
cardiotocography	0.8226	0.8371	0.4657	0.4626	0.9920	2.9228
eye-state	0.7464	0.7364	0.4971	0.4665	0.9637	152.6790
page-blocks	0.9288	0.9360	0.6430	0.6366	0.9949	25.5404
penbased	0.9665	0.9728	0.9639	0.9698	0.8778	422.8700
satimage	0.8535	0.8552	0.8214	0.8211	0.9694	42.2098
thyroid	0.9333	0.9421	0.4113	0.4047	0.9964	32.8508
segment	0.2506	0.1372	0.1274	-0.0066	0.8956	3.6726
coil2000	0.9089	0.9131	0.0702	0.0289	0.9938	75.3100
magic	0.8068	0.8080	0.5762	0.5662	0.9845	221.9630
marketing	0.2934	0.2611	0.1928	0.1544	0.9611	27.8332
phoneme	0.7951	0.8005	0.5122	0.5107	0.9918	19.1806
ring	0.8637	0.8573	0.7370	0.7140	0.9596	42.5625
spambase	0.8600	0.8681	0.7192	0.7258	0.9902	10.0464
texture	0.9277	0.9316	0.9215	0.9248	0.9064	50.5114
titanic	0.6995	0.7342	0.2370	0.2324	0.9951	3.0863
twonorm	0.9444	0.9520	0.8987	0.9041	0.9946	26.4229

TABLA A.47: Resultados de RSS-MA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8022	0.8023	0.2749	0.2739	0.9979	100.2660
shuttle	0.9978	0.9982	0.9945	0.9948	0.9513	955.5570

TABLA A.48: Resultados de RSS-MA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9726	0.9333	0.9589	0.9000	0.9707	0.3046
cleveland	0.7303	0.5741	0.5455	0.3040	0.9205	1.1551
led7digit	0.7738	0.7460	0.7482	0.7170	0.9698	1.7640
pima	0.8281	0.7213	0.6135	0.3763	0.9512	6.6283
wdbc	0.9664	0.9469	0.9273	0.8852	0.9880	4.3109
monk2	0.9535	0.9465	0.9067	0.8928	0.9706	2.5570
wisconsin	0.9779	0.9577	0.9513	0.9072	0.9933	8.2103
wine	0.9819	0.9409	0.9725	0.9107	0.9685	0.3847
glass	0.8364	0.7236	0.7614	0.5906	0.9107	0.5561
banknote	0.9969	0.9934	0.9938	0.9867	0.9914	18.7481
appendicitis	0.9528	0.9338	0.7774	0.6732	0.9708	0.3227
balance	0.8683	0.8387	0.7533	0.6981	0.9723	4.0537
bands	0.7912	0.6412	0.5318	0.1805	0.9419	1.5031
contraceptive	0.6450	0.4889	0.4451	0.2005	0.9210	39.5381
dermatology	0.9684	0.9318	0.9600	0.9132	0.9626	1.4353
ecoli	0.8879	0.8172	0.8378	0.7320	0.9494	1.4201
haberman	0.8119	0.7183	0.4173	0.1254	0.9703	1.4346
hayes-roth	0.7681	0.6438	0.6383	0.4330	0.9225	0.2822
heart	0.8778	0.8222	0.7511	0.6378	0.9719	0.7823
hepatitis	0.9125	0.8125	0.6414	0.3098	0.9688	0.1244
mammographic	0.8430	0.8060	0.6857	0.6117	0.9861	10.2855
newthyroid	0.9742	0.9381	0.9410	0.8504	0.9753	0.8506
tae	0.7425	0.5158	0.6071	0.2551	0.8682	0.2351
vehicle	0.8167	0.6563	0.7556	0.5415	0.9046	7.4441
vowel	0.8846	0.8000	0.8731	0.7800	0.8721	5.9631
yeast	0.6983	0.5573	0.6035	0.4204	0.9259	35.5053

TABLA A.49: Resultados de CNN-RSS-MA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8774	0.8792	0.7629	0.7564	0.9715	16.4339
cardiotocography	0.8714	0.8728	0.6403	0.6243	0.9831	3.1510
eye-state	0.6811	0.6801	0.3634	0.3510	0.9900	167.5720
page-blocks	0.9316	0.9397	0.6278	0.6243	0.9961	19.5174
penbased	0.9784	0.9854	0.9771	0.9838	0.8672	463.2490
satimage	0.8731	0.8696	0.8451	0.8384	0.9782	32.5058
thyroid	0.9372	0.9458	0.4726	0.4685	0.9973	28.5697
segment	0.2104	0.1511	0.0805	0.0096	0.9353	4.0036
coil2000	0.9116	0.9158	0.0981	0.0429	0.9927	134.2930
magic	0.8169	0.8191	0.6003	0.5924	0.9819	282.5090
marketing	0.3113	0.2670	0.2147	0.1628	0.9541	30.1330
phoneme	0.8108	0.8101	0.5538	0.5364	0.9805	18.4183
ring	0.9140	0.9130	0.8378	0.8258	0.9623	49.4138
spambase	0.8712	0.8705	0.7397	0.7274	0.9822	10.7736
texture	0.9648	0.9658	0.9622	0.9624	0.8575	84.0142
titanic	0.7756	0.7823	0.4509	0.4514	0.9906	3.2848
twonorm	0.9493	0.9574	0.9085	0.9149	0.9923	26.0074

TABLA A.50: Resultados de CNN-RSS-MA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7923	0.7917	0.2394	0.2364	0.9974	94.9568
shuttle	0.9791	0.9985	0.9958	0.9956	0.9440	1185.5700

TABLA A.51: Resultados de CNN-RSS-MA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9741	0.9200	0.9611	0.8800	0.9673	0.2614
cleveland	0.7426	0.5886	0.5728	0.3320	0.9189	0.9653
led7digit	0.7782	0.7480	0.7530	0.7191	0.9708	1.4491
pima	0.8345	0.7023	0.6241	0.3260	0.9490	6.9891
wdbc	0.9793	0.9546	0.9552	0.9030	0.9821	3.8590
monk2	0.9558	0.9233	0.9114	0.8463	0.9718	2.7427
wisconsin	0.9785	0.9665	0.9528	0.9262	0.9927	8.5964
wine	0.9876	0.9489	0.9811	0.9221	0.9657	0.3832
glass	0.8256	0.6903	0.7478	0.5461	0.9164	0.4121
banknote	0.9981	0.9927	0.9961	0.9852	0.9885	12.1268
appendicitis	0.9665	0.9175	0.8359	0.5183	0.9802	0.3499
balance	0.8932	0.8627	0.7995	0.7423	0.9723	3.7977
bands	0.8143	0.6457	0.5849	0.2262	0.9271	1.5688
contraceptive	0.6525	0.4936	0.4589	0.2119	0.9182	29.9721
dermatology	0.9767	0.9260	0.9702	0.9054	0.9670	1.5526
ecoli	0.8856	0.8072	0.8347	0.7188	0.9342	1.1927
haberman	0.8032	0.7283	0.3933	0.2118	0.9742	1.5164
hayes-roth	0.8069	0.6688	0.6974	0.4788	0.8844	0.2362
heart	0.8798	0.8259	0.7551	0.6425	0.9656	0.8167
hepatitis	0.9153	0.8875	0.6817	0.5556	0.9738	0.1274
mammographic	0.8499	0.8012	0.6996	0.6020	0.9782	10.5598
newthyroid	0.9773	0.9628	0.9471	0.9100	0.9744	0.9485
tae	0.7395	0.5567	0.6065	0.3210	0.8788	0.2661
vehicle	0.8216	0.6482	0.7622	0.5308	0.9051	8.2951
vowel	0.9851	0.9242	0.9836	0.9167	0.8381	9.1675
yeast	0.6930	0.5576	0.5964	0.4175	0.9369	37.3473

TABLA A.52: Resultados de ENN-RSS-MA para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8828	0.8751	0.7728	0.7470	0.9516	18.3825
cardiotocography	0.8543	0.8578	0.5909	0.5811	0.9899	2.9532
eye-state	0.7765	0.7584	0.5581	0.5113	0.9334	314.6240
page-blocks	0.9353	0.9424	0.6634	0.6540	0.9962	20.1270
penbased	0.9687	0.9751	0.9663	0.9723	0.9754	103.8270
satimage	0.8637	0.8628	0.8339	0.8303	0.9808	33.3027
thyroid	0.9329	0.9419	0.3756	0.3770	0.9969	29.5335
segment	0.2306	0.1437	0.1041	0.0010	0.9178	3.7915
coil2000	0.9212	0.9281	0.0616	0.0345	0.9977	81.3723
magic	0.8246	0.8172	0.6176	0.5883	0.9633	408.7460
marketing	0.3801	0.2915	0.2915	0.1881	0.8971	36.9086
phoneme	0.7936	0.7868	0.5115	0.4801	0.9845	20.0665
ring	0.7960	0.7976	0.6011	0.5941	0.9922	52.3051
spambase	0.8730	0.8718	0.7446	0.7316	0.9679	17.5936
texture	0.9634	0.9642	0.9607	0.9606	0.8504	87.0645
titanic	0.7758	0.7837	0.4488	0.4537	0.9898	3.3379
twonorm	0.9468	0.9527	0.9036	0.9054	0.9906	35.7473

TABLA A.53: Resultados de ENN-RSS-MA para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7724	0.7857	0.2868	0.2864	0.9964	102.9950
shuttle	0.9769	0.9968	0.9716	0.9908	0.9974	487.6720

TABLA A.54: Resultados de ENN-RSS-MA para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9585	0.9000	0.9378	0.8500	0.9700	0.1097
cleveland	0.7003	0.5659	0.5024	0.2849	0.9465	0.2684
led7digit	0.7476	0.7300	0.7193	0.6992	0.9682	0.3557
pima	0.8132	0.7283	0.5688	0.3752	0.9405	0.7985
wdbc	0.9631	0.9354	0.9199	0.8599	0.9865	0.6287
monk2	0.9100	0.8797	0.8196	0.7588	0.9611	0.4123
wisconsin	0.9730	0.9651	0.9405	0.9233	0.9914	0.7332
wine	0.9606	0.9527	0.9398	0.9272	0.9798	0.1367
glass	0.7784	0.6482	0.6734	0.4813	0.9266	0.1676
banknote	0.9981	0.9993	0.9961	0.9985	0.9527	1.3379
appendicitis	0.9288	0.9338	0.6988	0.6424	0.9792	0.1030
balance	0.8690	0.8462	0.7555	0.7145	0.9626	0.6150
bands	0.7591	0.6825	0.4559	0.2739	0.9537	0.3133
contraceptive	0.6411	0.4746	0.4423	0.1849	0.8444	1.5983
dermatology	0.9532	0.9118	0.9412	0.8891	0.9718	0.3737
ecoli	0.8648	0.8228	0.8007	0.7366	0.9610	0.3069
haberman	0.7956	0.7683	0.3214	0.2651	0.9820	0.2920
hayes-roth	0.7111	0.6375	0.5504	0.4333	0.9388	0.1336
heart	0.8646	0.8111	0.7232	0.6142	0.9756	0.2325
hepatitis	0.9194	0.8625	0.6624	0.3505	0.9663	0.0733
mammographic	0.8333	0.8048	0.6667	0.6097	0.9665	0.9039
newthyroid	0.9633	0.9628	0.9126	0.9129	0.9814	0.1926
tae	0.6409	0.5154	0.4583	0.2499	0.9404	0.1154
vehicle	0.7888	0.6587	0.7185	0.5450	0.9007	0.9109
vowel	0.9327	0.8535	0.9259	0.8389	0.8156	0.9711
yeast	0.6909	0.5398	0.5953	0.3979	0.8494	1.6069

TABLA A.55: Resultados de CHC para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8986	0.8916	0.7941	0.7799	0.9654	1.8342
cardiotocography	0.8928	0.8855	0.6854	0.6626	0.9747	0.8028
eye-state	0.8250	0.7829	0.6460	0.5609	0.8324	5.6788
page-blocks	0.9585	0.9542	0.7596	0.7342	0.9861	2.5129
penbased	0.9848	0.9829	0.9831	0.9809	0.9332	3.5643
satimage	0.8970	0.8844	0.8726	0.8571	0.9501	2.3742
thyroid	0.9431	0.9412	0.4272	0.4032	0.9882	3.0773
segment	0.2311	0.1515	0.1029	0.0101	0.9260	0.8781
coil2000	0.9377	0.9359	0.0584	0.0255	0.9819	5.4810
magic	0.8440	0.8106	0.6482	0.5707	0.8601	8.2651
marketing	0.3807	0.2867	0.2882	0.1797	0.8911	2.4493
phoneme	0.8373	0.8168	0.6045	0.5535	0.9586	2.0545
ring	0.9091	0.8983	0.8180	0.7963	0.9530	3.7638
spambase	0.8875	0.8775	0.7643	0.7436	0.9719	1.5940
texture	0.9635	0.9584	0.9599	0.9543	0.9285	1.7402
titanic	0.7829	0.7810	0.4586	0.4515	0.9911	0.5162
twonorm	0.9593	0.9559	0.9186	0.9117	0.9808	2.4462

TABLA A.56: Resultados de CHC para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7944	0.7888	0.3286	0.3070	0.9740	12.2479
shuttle	0.9977	0.9980	0.9942	0.9941	0.9490	21.4851

TABLA A.57: Resultados de CHC para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9452	0.9400	0.9178	0.9100	0.9667	0.1280
cleveland	0.6824	0.5735	0.4687	0.2898	0.9508	0.3327
led7digit	0.7407	0.7180	0.7114	0.6863	0.9576	0.4912
pima	0.8177	0.7273	0.5885	0.3820	0.9167	1.0538
wdbc	0.9567	0.9311	0.9051	0.8525	0.9851	0.9464
monk2	0.9295	0.9075	0.8589	0.8153	0.9671	0.5052
wisconsin	0.9723	0.9593	0.9389	0.9093	0.9924	2.3531
wine	0.9631	0.9706	0.9439	0.9551	0.9702	0.1832
glass	0.8172	0.6815	0.7366	0.5386	0.9150	0.1969
banknote	0.9938	0.9869	0.9875	0.9734	0.9827	2.1826
appendicitis	0.9602	0.9175	0.8315	0.5920	0.9755	0.1196
balance	0.8535	0.8053	0.7281	0.6410	0.9581	0.8653
bands	0.7610	0.6755	0.4572	0.2639	0.9438	0.4083
contraceptive	0.6360	0.4610	0.4381	0.1721	0.8007	2.3267
dermatology	0.9526	0.9050	0.9396	0.8789	0.9707	0.5386
ecoli	0.8526	0.7865	0.7871	0.6893	0.9530	0.4642
haberman	0.7894	0.7389	0.3108	0.1839	0.9856	0.3480
hayes-roth	0.7292	0.6250	0.5779	0.4102	0.9344	0.1447
heart	0.8580	0.7963	0.7123	0.5873	0.9759	0.2957
hepatitis	0.9181	0.8500	0.6438	0.3686	0.9650	0.0828
mammographic	0.8276	0.8120	0.6552	0.6242	0.9696	1.2071
newthyroid	0.9716	0.9608	0.9311	0.9058	0.9767	0.3146
tae	0.6542	0.4646	0.4782	0.1843	0.9272	0.1390
vehicle	0.7987	0.6744	0.7315	0.5656	0.8805	1.2140
vowel	0.9621	0.9030	0.9583	0.8933	0.8136	1.2354
yeast	0.7427	0.5392	0.6628	0.3971	0.7664	2.4397

TABLA A.58: Resultados de CNN-CHC para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8872	0.8851	0.7817	0.7671	0.9564	2.4352
cardiotocography	0.8710	0.8752	0.6482	0.6389	0.9755	0.9279
eye-state	0.8377	0.7796	0.6823	0.5550	0.7871	9.4798
page-blocks	0.9485	0.9538	0.7531	0.7341	0.9829	5.0899
penbased	0.9786	0.9867	0.9774	0.9852	0.9360	5.5268
satimage	0.8871	0.8811	0.8626	0.8528	0.9473	3.8257
thyroid	0.9355	0.9443	0.4300	0.4248	0.9930	4.8042
segment	0.2910	0.1506	0.1744	0.0091	0.8691	0.8686
coil2000	0.9302	0.9389	0.0163	0.0049	0.9874	9.2518
magic	0.8239	0.7935	0.6211	0.5393	0.8866	15.0822
marketing	0.4467	0.2704	0.3690	0.1658	0.7859	3.2887
phoneme	0.8315	0.8249	0.6065	0.5757	0.9627	3.3927
ring	0.9092	0.9034	0.8283	0.8066	0.9561	6.2863
spambase	0.8681	0.8658	0.7342	0.7184	0.9771	2.1910
texture	0.9547	0.9575	0.9511	0.9532	0.9215	2.4389
titanic	0.7753	0.7810	0.4491	0.4481	0.9898	0.5828
twonorm	0.9391	0.9464	0.8882	0.8927	0.9841	3.7002

TABLA A.59: Resultados de CNN-CHC para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7829	0.7763	0.3168	0.2937	0.9700	15.7268
shuttle	0.9976	0.9978	0.9940	0.9937	0.9841	60.9359

TABLA A.60: Resultados de CNN-CHC para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9526	0.9200	0.9289	0.8800	0.9720	0.1291
cleveland	0.6704	0.5825	0.4417	0.3001	0.9630	0.3670
led7digit	0.7624	0.7440	0.7357	0.7146	0.9682	0.5451
pima	0.8141	0.7239	0.5739	0.3599	0.9289	1.0341
wdbc	0.9615	0.9523	0.9166	0.8956	0.9761	0.9131
monk2	0.8855	0.8515	0.7710	0.7034	0.9637	0.4827
wisconsin	0.9732	0.9592	0.9410	0.9110	0.9717	0.9064
wine	0.9652	0.9294	0.9470	0.8937	0.9758	0.1929
glass	0.7674	0.6870	0.6668	0.5421	0.9318	0.2286
banknote	0.9979	0.9971	0.9957	0.9941	0.8972	2.1298
appendicitis	0.9623	0.9438	0.7982	0.6792	0.9717	0.1422
balance	0.8855	0.8657	0.7851	0.7476	0.9522	0.7496
bands	0.7729	0.6596	0.4840	0.2229	0.9408	0.4344
contraceptive	0.5895	0.4678	0.3628	0.1731	0.9041	2.5653
dermatology	0.9634	0.9445	0.9531	0.9287	0.9721	0.5299
ecoli	0.8505	0.7960	0.7829	0.7055	0.9589	0.3897
haberman	0.7905	0.7611	0.3168	0.2320	0.9863	0.3135
hayes-roth	0.7361	0.6125	0.5920	0.3955	0.9219	0.1379
heart	0.8646	0.8148	0.7237	0.6222	0.9756	0.2903
hepatitis	0.9125	0.8750	0.6779	0.4848	0.9663	0.0864
mammographic	0.8454	0.8060	0.6904	0.6117	0.9370	1.0858
newthyroid	0.9612	0.9399	0.9068	0.8557	0.9819	0.2263
tae	0.6689	0.4892	0.5023	0.2263	0.9285	0.1184
vehicle	0.7833	0.6747	0.7111	0.5660	0.8928	1.2502
vowel	0.9364	0.8828	0.9300	0.8711	0.7685	1.5126
yeast	0.6426	0.5642	0.5311	0.4273	0.8965	2.5965

TABLA A.61: Resultados de ENN-CHC para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8875	0.8851	0.7821	0.7671	0.9598	2.2316
cardiotocography	0.8724	0.8672	0.6601	0.6284	0.9721	0.8639
eye-state	0.7984	0.7754	0.6033	0.5467	0.8776	11.5479
page-blocks	0.9490	0.9560	0.7510	0.7428	0.9809	2.8736
penbased	0.9773	0.9850	0.9758	0.9833	0.8884	7.0029
satimage	0.8864	0.8855	0.8618	0.8583	0.9375	4.0794
thyroid	0.9358	0.9442	0.4030	0.3912	0.9758	4.4285
segment	0.1882	0.1407	0.0545	-0.0025	0.9614	1.0229
coil2000	0.9228	0.9289	0.0935	0.0487	0.9579	14.3025
magic	0.8528	0.8260	0.6778	0.6012	0.7430	16.7927
marketing	0.3476	0.2971	0.2503	0.1907	0.9427	3.5624
phoneme	0.8321	0.8242	0.6084	0.5759	0.9494	2.3561
ring	0.7809	0.7707	0.5702	0.5393	0.9521	4.4712
spambase	0.8729	0.8732	0.7418	0.7311	0.9653	2.9880
texture	0.9503	0.9540	0.9464	0.9494	0.9139	3.2882
titanic	0.7763	0.7833	0.4493	0.4514	0.9898	0.5746
twonorm	0.9520	0.9573	0.9140	0.9146	0.9383	4.2723

TABLA A.62: Resultados de ENN-CHC para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.8010	0.7950	0.3551	0.3325	0.9622	17.9675
shuttle	0.9976	0.9978	0.9940	0.9937	0.8843	36.7365

TABLA A.63: Resultados de ENN-CHC para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9689	0.9467	0.9533	0.9200	0.9753	0.1232
cleveland	0.6603	0.5582	0.4330	0.2698	0.9623	0.2902
led7digit	0.7684	0.7500	0.7422	0.7214	0.9746	0.4270
pima	0.8060	0.7304	0.5625	0.3914	0.9478	0.9977
wdbc	0.9557	0.9347	0.9031	0.8573	0.9886	0.8044
monk2	0.9035	0.8818	0.8066	0.7638	0.9799	0.4537
wisconsin	0.9740	0.9636	0.9427	0.9200	0.9928	1.2788
wine	0.9700	0.9353	0.9541	0.9001	0.9730	0.1844
glass	0.7596	0.6808	0.6439	0.5252	0.9509	0.2138
banknote	0.9930	0.9898	0.9859	0.9793	0.9821	2.4199
appendicitis	0.9422	0.9038	0.7179	0.4875	0.9755	0.1132
balance	0.8688	0.8709	0.7541	0.7582	0.9848	0.8330
bands	0.7455	0.7061	0.4134	0.3239	0.9636	0.3674
contraceptive	0.6579	0.4773	0.4676	0.1875	0.8264	2.3649
dermatology	0.9473	0.9369	0.9333	0.9196	0.9749	0.3946
ecoli	0.8466	0.8089	0.7733	0.7152	0.9649	0.3800
haberman	0.7908	0.7317	0.3173	0.1412	0.9853	0.2903
hayes-roth	0.7035	0.5188	0.5274	0.2341	0.9400	0.1418
heart	0.8527	0.7963	0.7000	0.5835	0.9807	0.2551
hepatitis	0.9000	0.8500	0.3634	0.1257	0.9775	0.0793
mammographic	0.8340	0.8133	0.6680	0.6262	0.9753	1.1150
newthyroid	0.9669	0.9438	0.9199	0.8531	0.9814	0.2253
tae	0.6748	0.5238	0.5093	0.2817	0.9219	0.1172
vehicle	0.7819	0.6408	0.7091	0.5206	0.9058	1.0728
vowel	0.8688	0.7899	0.8557	0.7689	0.8494	1.2532
yeast	0.6538	0.5411	0.5463	0.3986	0.8846	2.3610

TABLA A.64: Resultados de RSS-CHC para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8809	0.8832	0.7687	0.7631	0.9713	2.4651
cardiotocography	0.8730	0.8765	0.6594	0.6501	0.9763	0.8185
eye-state	0.7984	0.7672	0.5977	0.5229	0.8514	10.3898
page-blocks	0.9387	0.9466	0.6831	0.6805	0.9928	3.6991
penbased	0.9728	0.9798	0.9709	0.9776	0.9293	6.9140
satimage	0.8643	0.8622	0.8349	0.8299	0.9627	3.9961
thyroid	0.9340	0.9421	0.3929	0.3818	0.9919	4.2709
segment	0.2340	0.1528	0.1080	0.0116	0.9190	0.9589
coil2000	0.9246	0.9320	0.0466	0.0223	0.9901	9.2795
magic	0.8127	0.8075	0.5797	0.5532	0.9313	15.2582
marketing	0.3830	0.2894	0.2947	0.1862	0.8809	3.3203
phoneme	0.8035	0.8066	0.5439	0.5374	0.9815	2.7217
ring	0.9204	0.9134	0.8507	0.8266	0.9381	6.3558
spambase	0.8635	0.8660	0.7260	0.7205	0.9794	2.4941
texture	0.9387	0.9442	0.9335	0.9386	0.9372	2.7477
titanic	0.7672	0.7751	0.4304	0.4353	0.9918	0.7916
twonorm	0.9478	0.9546	0.9055	0.9092	0.9841	4.1837

TABLA A.65: Resultados de RSS-CHC para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7973	0.7948	0.3250	0.3120	0.9836	18.1931
shuttle	0.9970	0.9976	0.9933	0.9932	0.9801	64.0352

TABLA A.66: Resultados de RSS-CHC para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9644	0.9400	0.9467	0.9100	0.9727	0.1323
cleveland	0.6924	0.5969	0.4606	0.2823	0.9535	0.3325
led7digit	0.7267	0.7000	0.6959	0.6659	0.9640	0.4951
pima	0.8095	0.7296	0.5673	0.3839	0.9271	0.9923
wdbc	0.9566	0.9303	0.9055	0.8497	0.9905	0.8764
monk2	0.9252	0.9049	0.8501	0.8100	0.9653	0.5571
wisconsin	0.9730	0.9679	0.9404	0.9287	0.9931	1.2903
wine	0.9719	0.9471	0.9569	0.9182	0.9725	0.1798
glass	0.7954	0.7109	0.7021	0.5814	0.9215	0.2090
banknote	0.9958	0.9942	0.9915	0.9881	0.9802	2.3484
appendicitis	0.9498	0.9138	0.7707	0.5601	0.9708	0.1231
balance	0.8514	0.8099	0.7218	0.6446	0.9603	0.8365
bands	0.7756	0.6735	0.4974	0.2610	0.9381	0.4440
contraceptive	0.5845	0.4671	0.3566	0.1746	0.8599	2.2979
dermatology	0.9528	0.9198	0.9398	0.8978	0.9684	0.4987
ecoli	0.8519	0.7839	0.7869	0.6901	0.9580	0.4320
haberman	0.7850	0.7622	0.2966	0.2072	0.9876	0.3616
hayes-roth	0.7278	0.5625	0.5742	0.3116	0.9269	0.1477
heart	0.8617	0.8000	0.7184	0.5894	0.9807	0.2763
hepatitis	0.9000	0.7875	0.4285	0.0026	0.9700	0.0713
mammographic	0.8335	0.7928	0.6667	0.5849	0.9600	1.1245
newthyroid	0.9659	0.9447	0.9213	0.8775	0.9786	0.2718
tae	0.6674	0.5571	0.4990	0.3363	0.9278	0.1215
vehicle	0.8088	0.6459	0.7450	0.5272	0.8680	1.1788
vowel	0.9519	0.9051	0.9470	0.8956	0.8132	1.3003
yeast	0.6715	0.5163	0.5697	0.3669	0.8268	2.4174

TABLA A.67: Resultados de CNN-RSS-CHC para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8792	0.8785	0.7654	0.7538	0.9557	2.2206
cardiotocography	0.8781	0.8766	0.6643	0.6402	0.9725	0.9069
eye-state	0.8138	0.7754	0.6328	0.5445	0.8348	9.5799
page-blocks	0.9441	0.9512	0.7199	0.7132	0.9873	4.1491
penbased	0.9785	0.9860	0.9773	0.9844	0.9193	6.0296
satimage	0.8847	0.8768	0.8596	0.8474	0.9368	3.5942
thyroid	0.9303	0.9392	0.3410	0.3393	0.9924	4.8284
segment	0.2703	0.1416	0.1503	-0.0015	0.8742	0.8990
coil2000	0.9257	0.9327	0.0725	0.0277	0.9833	9.3002
magic	0.8329	0.7904	0.6400	0.5303	0.8337	14.2995
marketing	0.4297	0.2765	0.3504	0.1737	0.8099	3.3324
phoneme	0.8373	0.8227	0.6201	0.5707	0.9532	2.5038
ring	0.9208	0.9159	0.8515	0.8318	0.9416	5.8020
spambase	0.8743	0.8734	0.7503	0.7380	0.9593	2.3094
texture	0.9611	0.9656	0.9582	0.9622	0.9154	2.6238
titanic	0.7675	0.7746	0.4285	0.4301	0.9906	0.7138
twonorm	0.9416	0.9477	0.8933	0.8954	0.9829	3.8145

TABLA A.68: Resultados de CNN-RSS-CHC para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7912	0.7839	0.3192	0.2925	0.9655	15.9920
shuttle	0.9979	0.9981	0.9947	0.9945	0.9784	51.4480

TABLA A.69: Resultados de CNN-RSS-CHC para conjuntos grandes con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
iris	0.9674	0.9400	0.9511	0.9100	0.9700	0.1274
cleveland	0.6887	0.5646	0.4708	0.2789	0.9492	0.3316
led7digit	0.7700	0.7500	0.7440	0.7213	0.9650	0.5179
pima	0.8145	0.7104	0.5781	0.3404	0.9069	0.9370
wdbc	0.9652	0.9454	0.9246	0.8819	0.9740	0.8264
monk2	0.8819	0.8244	0.7642	0.6501	0.9477	0.4550
wisconsin	0.9763	0.9607	0.9478	0.9150	0.9659	1.0389
wine	0.9650	0.9294	0.9465	0.8910	0.9725	0.1927
glass	0.8033	0.7046	0.7063	0.5670	0.9327	0.2082
banknote	0.9985	0.9964	0.9969	0.9926	0.8961	2.1539
appendicitis	0.9560	0.9075	0.8038	0.4920	0.9736	0.1182
balance	0.8695	0.8549	0.7554	0.7281	0.9526	0.7160
bands	0.7678	0.6398	0.4652	0.1757	0.9496	0.4576
contraceptive	0.6172	0.4582	0.4071	0.1685	0.8392	2.4165
dermatology	0.9553	0.9346	0.9433	0.9168	0.9704	0.5322
ecoli	0.8605	0.8275	0.7988	0.7507	0.9604	0.3802
haberman	0.7938	0.7506	0.3261	0.2283	0.9843	0.3200
hayes-roth	0.7431	0.6500	0.6008	0.4526	0.9244	0.1371
heart	0.8601	0.8222	0.7153	0.6362	0.9767	0.2643
hepatitis	0.9083	0.8875	0.5886	0.4257	0.9638	0.0837
mammographic	0.8446	0.7976	0.6891	0.5953	0.9237	0.9804
newthyroid	0.9633	0.9628	0.9088	0.9072	0.9842	0.2442
tae	0.6799	0.4963	0.5171	0.2344	0.9238	0.1370
vehicle	0.7766	0.6744	0.7021	0.5656	0.8819	1.1214
vowel	0.9300	0.8737	0.9230	0.8611	0.7779	1.4420
yeast	0.7014	0.5367	0.6082	0.3927	0.8005	2.4467

TABLA A.70: Resultados de ENN-RSS-CHC para conjuntos pequeños con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
banana	0.8825	0.8779	0.7720	0.7525	0.9508	2.1428
cardiotocography	0.8731	0.8734	0.6457	0.6284	0.9742	0.8664
eye-state	0.8119	0.7798	0.6305	0.5553	0.8348	10.5138
page-blocks	0.9459	0.9512	0.7476	0.7303	0.9752	2.9450
penbased	0.9807	0.9886	0.9797	0.9874	0.8829	6.7662
satimage	0.8899	0.8873	0.8663	0.8608	0.9385	3.8026
thyroid	0.9368	0.9442	0.4603	0.4439	0.9591	4.1253
segment	0.2481	0.1472	0.1245	0.0051	0.9100	0.9087
coil2000	0.9235	0.9287	0.1108	0.0501	0.9295	11.9566
magic	0.8575	0.8194	0.6889	0.5860	0.7224	15.2480
marketing	0.4103	0.2797	0.3281	0.1774	0.8467	3.2276
phoneme	0.8372	0.8244	0.6229	0.5780	0.9477	2.3684
ring	0.8877	0.8574	0.7852	0.7143	0.8887	4.2800
spambase	0.8648	0.8636	0.7258	0.7117	0.9597	2.7554
texture	0.9577	0.9642	0.9545	0.9606	0.9136	3.0292
titanic	0.7758	0.7837	0.4500	0.4549	0.9902	0.7179
twonorm	0.9503	0.9543	0.9105	0.9086	0.9387	4.0010

TABLA A.71: Resultados de ENN-RSS-CHC para conjuntos medianos con 1000 iteraciones fijas

CONJUNTO	ACCURACY		KAPPA		REDUCCIÓN	TIEMPO PROMEDIO (SEG)
	Training	Test	Training	Test		
credit-card	0.7967	0.7860	0.3481	0.3110	0.9372	16.2408
shuttle	0.9983	0.9984	0.9960	0.9955	0.8826	46.9600

TABLA A.72: Resultados de ENN-RSS-CHC para conjuntos grandes con 1000 iteraciones fijas

Apéndice B

Pruebas estadísticas

ALGORITMO	GGA					
	ACCURACY		KAPPA	REDUCCIÓN		
	W	p -valor		W	p -valor	
GGA	46	3.574×10^{-1}	1	1.221×10^{-4}	1.221×10^{-4}	
SGA	27	5.945×10^{-2}	0	6.104×10^{-5}	6.104×10^{-5}	
CHC	6	8.545×10^{-4}	14	6.714×10^{-3}	6.714×10^{-3}	
PSO	6	8.545×10^{-4}	0	6.104×10^{-5}	6.104×10^{-5}	

TABLA B.1: Estadístico W y p -valor de pruebas de rangos con signo de *Wilcoxon* para determinar si PBIL es mejor que el resto de las metaheurísticas, en función del error de validación y el tamaño de las soluciones encontradas.