UNIVERSIDAD DISTRITAL "FRANCISCO JOSÉ CALDAS"
FACULTAD DE INGENIERÍA

MAESTRIA EN CIENCIAS DE LA INFORMACION Y LAS COMUNICACIONES
ENFASIS EN "SISTEMA DE INFORMACION GEOFRAFICA"

ESTUDIO COMPARATIVO DE ALGORITMOS DE AGRUPACIÓN BASADOS EN LA LEY DE GRAVITACIÓN UNIVERSAL APLICADOS A LA SEGMENTACIÓN DE IMÁGENES.

AUTOR

ANGÉLICA JOANA SUAREZ PORRAS

Bogotá

2015

UNIVERSIDAD DISTITAL "FRANCISCO JOSÉ CALDAS"
FACULTAD DE INGENIERÍA

MAESTRIA EN CIENCIAS DE LA INFORMACION Y LAS COMUNICACIONES
ENFASIS EN "SISTEMA DE INFORMACION GEOFRAFICA"

ESTUDIO COMPARATIVO DE ALGORITMOS DE AGRUPACIÓN BASADOS EN LA LEY DE GRAVITACIÓN UNIVERSAL APLICADOS A LA SEGMENTACIÓN DE IMÁGENES.

Tesis presentada en opción al grado de Magister en Ciencias de la Información y las Comunicaciones.

AUTOR
ANGÉLICA JOANA SUAREZ PORRAS

TUTOR
JORGE ENRIQUE RODRÍGUEZ RODRÍGUEZ

Bogotá

2015

A Dios por darme la oportunidad de obtener este título y
enriquecer mi formación profesional;
a mis Padres y hermanas por acompañarme en este proceso;
a mi tutor Jorge Enrique Rodríguez por brindarme sus conocimientos.

AGRADECIMIENTOS

Agradezco a Dios por guiarme en la realización de este proyecto como herramienta de aprendizaje, por llenar de perseverancia, sabiduría, fortaleza en mi proceso de formación académica en la Universidad Distrital Francisco José de Caldas y por el crecimiento personal durante este tiempo.

También agradezco a mi familia por acompañarme en el cumplimiento de mis metas y de los cuales recibí un apoyo incondicional.

Y finalmente agradezco a los docentes que apoyaron y colaboraron al desarrollo de este proceso y en especial a mi tutor el Ingeniero Msc. Jorge Enrique Rodríguez, quien oriento y dio seguimiento con sus conocimientos a esta investigación.

TABLA DE CONTENIDO

INDICE DE TABLA

INDICE DE ILUSTRACIONES

1. RESUMEN

Esta investigación tuvo como objetivo principal resolver el problema de la segmentación de imágenes a través del uso del algoritmo de agrupación basados en la ley de gravitación universal llamado "Algoritmo de búsqueda gravitacional", se realizó un análisis comparativo entre los resultados obtenidos por este y los obtenidos por los algoritmos de agrupación "convencionales", cuyo propósito fue evaluar si este resuelve las debilidades identificadas en los algoritmos convencionales para este campo de aplicación, para tal efecto se planteó: :

- Seleccionar herramientas y/o implementar algoritmos de agrupación gravitacional y convencional para su aplicación en la segmentación de imágenes con el propósito de comparar los resultados obtenidos por cada uno.
- Definir el conjunto de datos a utilizar en el estudio, que consistirá en un conjunto de imágenes digitales sintéticas.
- Aplicar los algoritmos al conjunto de datos definido, con el fin de medir su efectividad para la segmentación de imágenes.
- Medir la efectividad de cada uno de los algoritmos utilizados y validar si las debilidades existentes en los algoritmos convencionales para la segmentación de imágenes fueron resueltas por los algoritmos gravitacionales.

2. ABSTRACT

This study's main objective was to solve the problem of image segmentation through the use of clustering algorithms based on the law of universal gravitation called "gravitational search algorithm with heuristics" and a comparative analysis of the results was carried out by this and those obtained by conventional algorithms, whose purpose was to assess whether this resolves the weaknesses identified in conventional algorithms for this field of application to that effect was raised:

- Select tools and / or implement algorithms and conventional gravitational clustering for application to image segmentation in order to compare the results obtained by each.
- Define the data set used in the study, which will consist of a set of synthetic digital images.
- Apply defined set of algorithms to data in order to measure its effectiveness for image segmentation.
- Measure the performance of each of the algorithms used and validate whether the weaknesses in conventional algorithms for image segmentation were resolved by gravitational algorithms

3. INTRODUCCIÓN

En la actualidad la utilización de técnicas de modelamiento aplicadas para el descubrimiento de conocimiento es cada vez más popular entre organizaciones de diferentes tipos que desean explotar su información, la minería de datos se ha convertido en la herramienta predilecta para la extracción de conocimiento. Por otra parte, una de las tareas de la ciencia de la computación que en los últimos años ha ido ganando terreno, es sin duda el procesamiento digital de imágenes también de la mano con la sociedad de la información; donde la información cada vez se presenta de manera gráfica, de forma abundante y con un grado de importancia mayor. Surge así la necesidad del desarrollo de técnicas que permitan el tratamiento adecuado de este tipo de información, adicionalmente el procesamiento digital de imágenes de manera similar a la minería de datos aplica técnicas para extraer conocimiento de las imágenes de interés, por lo anterior este proyecto genera un aporte de conocimiento que en la actualidad es de gran interés para la comunidad científica, ya que desarrolla temáticas que están vigentes.

La segmentación de imágenes, es uno de los principales problemas en el análisis de imágenes por computador [1]. La cual tiene como objetivo principal simplificar una imagen en segmentos que tienen una fuerte correlación con objetos en el mundo real. Regiones homogéneas de una imagen que contienen características comunes se agrupan en un solo segmento, la segmentación de imágenes se ha aplicado ampliamente en varias áreas como la biomedicina, los sistemas de transporte inteligentes y el análisis de imágenes aéreas y satelitales [2], en los últimos años se han desarrollado varias iniciativas que resuelven el problema de segmentación de imágenes mediante el uso de algoritmos de agrupación [1], [3], [4].

Adicionalmente, es una realidad que ninguna técnica es óptima en cualquier situación, el estudio comparativo de la variedad de algoritmos implementados en este proyecto posibilita la comparación de algoritmos de agrupación basados en la ley de gravitación universal con algoritmos convencionales para determinar si con este tipo de algoritmos se resuelven las debilidades identificadas para los algoritmos convencionales en este campo de aplicación (Segmentación de imágenes).

Por otra parte la aplicación de las conclusiones generadas para este proyecto tiene un amplio campo de áreas en las que pueden ser utilizadas como la biomédica, procesamiento digital de imágenes, extracción de patrones de conocimiento, y genera un aporte interesante proponiendo el uso de algoritmos novedosos en el área de segmentación de imágenes.

4. ALCANCE Y DEFINICIÓN DEL PROBLEMA DE INVESTIGACIÓN

4.1 HIPÓTESIS

La pregunta de investigación que resume el núcleo de este proyecto de investigación es la siguiente: ***"Si se usa una técnica de agrupación basada en la ley de gravitación universal para la segmentación de imágenes se está empleando una técnica que minimice las debilidades evidenciadas al utilizar las técnicas convencionales de agrupación"***.

4.2 OBJETIVOS

4.2.1 OBJETIVO GENERAL

Realizar un análisis comparativo del algoritmo metaheurístico de búsqueda gravitacional GSA con los algoritmos de agrupación convencionales K-medias y SOM, para establecer si el GSA logra ser una buena alternativa para la segmentación de imágenes y conocer más a fondo el desempeño de este en este tipo de problemas y así establecer si es más efectivo y eficiente que los dos primeros algoritmos convencionales.

4.2.2 OBJETIVOS ESPECIFICOS

- Identificar el estado del arte alrededor del uso de algoritmos basados en la ley de gravitación universal en el área de segmentación de imágenes o áreas relacionadas.
- Implementar el algoritmo gravitacional seleccionado para la segmentación de imágenes.
- Recopilar y pre-procesar un conjunto de datos del área de segmentación de imágenes con el objeto de aplicar los algoritmos en mención.

- Realizar análisis de pruebas y resultados de los algoritmos (gravitacional vs. convencionales) con el fin de realizar una comparación de la efectividad lograda por los algoritmos considerados para la segmentación de imágenes.

5. MARCO DE REFERENCIA

En este capítulo se definen y establecen conceptos necesarios para la comprensión y realización del presente proyecto, se consideraron los antecedentes de esta investigación y contenidos en relación a los conceptos de minería de datos, agrupación, segmentación de imágenes, ley de gravitación universal y algoritmos considerados.

5.1 MARCO TEÓRICO

Actualmente con la llamada sociedad de la información, gran cantidad de datos son almacenados en las bases de datos de las organizaciones cuyo volumen se está multiplicando considerablemente, se pensaría que este aumento de datos debería traer consigo un aumento del conocimiento, pero esto no es del todo posible, debido a que el procesamientos datos en volúmenes elevados es difícil con los métodos clásicos. Con relación a lo anterior y en respuesta a esta limitación en los últimos años se han venido desarrollando técnicas y métodos que facilitan el procesamiento y el análisis de grandes cantidades de datos de manera automática y semiautomática. Estas técnicas son agrupadas dentro del concepto de descubrimiento de conocimiento a partir de bases de datos, del inglés *Knowledge Discovery from Databases* (KDD), que tiene como idea principal de descubrir conocimiento de alto nivel desde datos que se encuentren sin procesar.

Es decir el KDD, es una herramienta que considera la selección, limpieza, transformación, relleno de los datos faltantes, análisis, evaluación e interpretación de los patrones para convertirlos en conocimiento y hacer el conocimiento disponible para su uso. Una etapa importante dentro de KDD es la minería de datos que puede ser entendida como "el proceso de extraer conocimiento útil y comprensible, previamente desconocido, desde grandes cantidades de datos almacenados en distintos formatos" [5]. Para dar uso de la minería de datos es importante conocer que existen principalmente dos tipos de tareas: la primera conocida como predictivas que tienen como objetivo estimar el valor de las variables de interés a través de un modelo y la segunda llamada tareas descriptivas, las cuales producen modelos que explican los datos generando una agrupación.

De esta manera, la agrupación es entonces una tarea descriptiva fundamental de aprendizaje que consiste en la separación de los datos en subgrupos o clases de tal manera que los datos asignados al mismo grupo tienen alta similitud, mientras que la similitud entre los datos asignados con otros grupos es baja. Algunas de las técnicas se basan en las mediciones de distancia, densidad y aquellas inspiradas en la naturaleza. Estas últimas técnicas han sido útiles para resolver el problema de agrupación, entre los que se encuentran algoritmos basados en la ley de gravitación de newton que simulan la atracción entre objetos como sucede el mundo real. Su funcionamiento se basa en calcular la atracción gravitatoria entre un objeto, es decir un registro con el resto de objetos y la **primitiva** que el grado de pertenencia de un objeto a una clase aumenta conforme la fuerza de gravedad influye entre dicho objeto y el resto de objetos de dicha clase.

Dicho lo anterior y haciendo referencia al presente proyecto de investigación se hará integración de los conceptos de agrupación y el algoritmo de optimización basado en la ley de gravitación universal llamado "Algoritmo de búsqueda gravitacional" para resolver el problema de la segmentación de imágenes, a través de comparación de los algoritmos de la ley de gravitación con los resultados obtenidos por los algoritmos convencionales, y así determinar si resuelven las debilidades identificadas para estos últimos en este campo de aplicación.

5.1.1 MINERÍA DE DATOS

La minería de datos es un proceso que se emplea para detectar la información en grandes conjuntos de datos. En esta se ha uso de análisis matemáticos para generar patrones y tendencias que existen entre los datos. Por lo general, aquellos patrones no se pueden hallar mediante la exploración tradicional de los datos, debido a que las relaciones existentes son muy complejas o al gran volumen de datos.

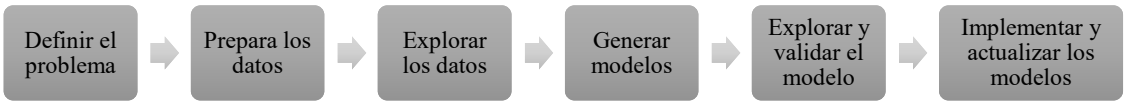
Aquellos patrones y tendencias se pueden recopilar y definir mediante un modelo de minería de datos. Estos modelos son aplicados generalmente en escenarios como la previsión, riesgo

y probabilidad, recomendaciones, búsqueda de secuencias y finalmente la agrupación que agrupa los elementos relacionados, analizar y predecir afinidades.

Aunque hay diferentes definiciones para minería de datos, una muy simple podría considerarse como que es el estudio y tratamiento de datos masivos para extraer conclusiones e información importante de ellos. [6]

Para la generación de un modelo de minería de datos es importante conocer que estos parten de la formulación de preguntas acerca de los datos, hasta la implementación del modelo en un entorno de trabajo. Este proceso se puede definir mediante los seis pasos básicos como se presentan en la Ilustración 1:

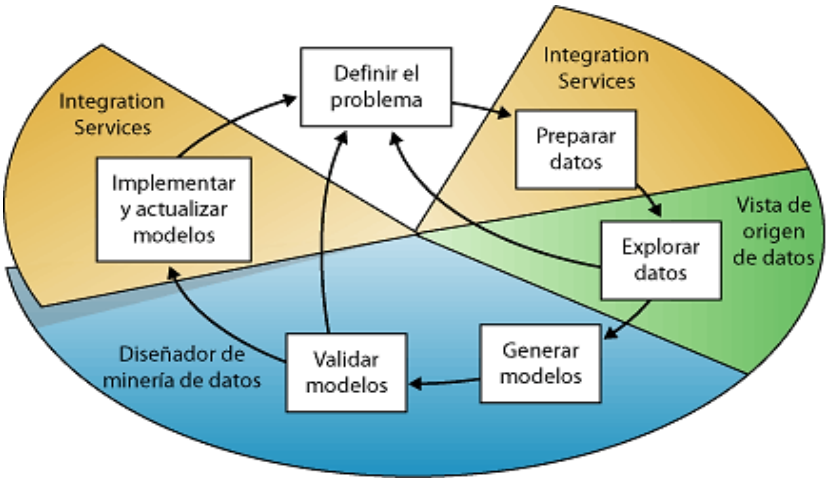
Ilustración 1 Pasos para la generación de un modelo de minería de datos.



Fuente generación propia.

Los pasos mencionados anteriormente se describen en la ilustración 2, en el cual se muestran las relaciones existentes entre cada uno de los proceso [7].

Ilustración 2 Microsoft. Conceptos de minería de datos.



Fuente Microsoft, Concepto de minería de datos,» Microsoft Developer Network, Octubre 2013. [7]

5.1.1.1 Utilidad de la minería de datos

Resumiendo lo expuesto hasta ahora, se puede decir que la funcionalidad de la minería de datos puede ser:

a) Predictiva. (p.ej. caso del banco, hospital): sirve para predecir cosas.

- En base a una **clasificación**: por ejemplo si el cliente pagará o no pagará, o el tipo de dolencia que puede tener un paciente.
- En base a una **regresión**: por ejemplo calcular el tiempo previsible que se empleará en corregir los errores de un desarrollo de software**.**

b) "Descriptiva:

- **Agrupamiento** (clustering): clasificar individuos en grupos en base a sus características. Por ejemplo, clasificar pacientes del hospital en base a los datos de sus analíticas.

- **Reglas de asociación:** conocer cómo se relacionan los datos o campos. Por ejemplo conocer en el hipermercado que un cliente que compra leche muy probablemente comprará también pan.
- **Secuenciación:** intentar predecir el valor de una variable en función del tiempo. Por ejemplo la demanda de energía eléctrica**.** [7]

5.1.1.2 Campos y aplicación de la minería de datos

La minería de datos tiene muchos campos de aplicación pues puede ser útil en prácticamente todas las facetas de la actividad humana. A continuación se indicaran algunos posibles campos de aplicación de la minería de datos:

a) La minería de datos tiene **utilidad empresarial**: las empresas pueden optimizar procesos y mejorar sus productos y ventas utilizando minería de datos.

b) Existen pocos especialistas o empresas especializadas en minería de datos. Teniendo en cuenta su importancia, es un **campo de trabajo** para emprendedores.

c) La minería de datos es una disciplina que se está desarrollando cada vez con mayores capacidades gracias al avance en tecnología y a la cada vez más alta capacidad de computación de los ordenadores. Constituye un campo amplio de **investigación** en el que cada vez trabajan más investigadores y equipos de investigación [8].

Existen diferentes ambientes en los que los flujos de datos pueden ser utilizados, algunas de estas aplicaciones son:

a) Los sistemas satelitales tienen la característica de manejar grandes cantidades de datos que se generan todo el tiempo y que deben ser procesados. Por ejemplo se encuentran el Centro Nacional para la Investigación Atmosférica (NCAR) que tiene datos en tiempo real del clima, maneja imágenes satelitales de la superficie terrestre, un sistema de predicción del clima, entre otras aplicaciones. Y el Centro de Ingeniería y Ciencia Espacial (SSEC) de la universidad de Wisconsin, también mantiene una galería de

imágenes en tiempo real de la Antártica, imágenes sobre huracanes y tormentas tropicales, un Interferómetro de la Radiación Atmosférica Emitida (AERI), entre otros.

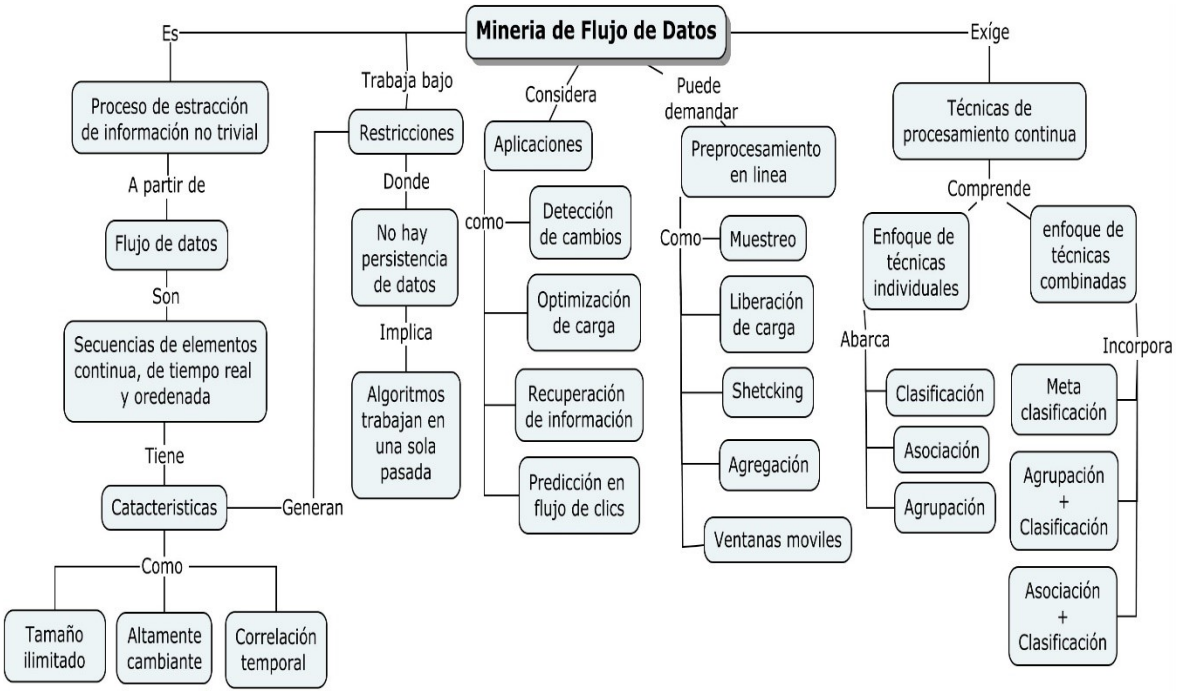
b) Detección de cambios. Una aplicación importante de la minería de flujos de datos está en el análisis de la detección del cambio y el monitoreo. Un ejemplo son los sensores, generalmente este tipo de datos se encuentran en aplicaciones en tiempo real que monitorean la actividad del ambiente en el que se encuentran. La técnica más usada para sensores de redes es agrupación.

c) **"**Optimización de carga. Un ejemplo de este es el Análisis de tráfico de redes: El tráfico IP tiene un interés particular en el monitoreo de direcciones fuentes y de destino para detectar patrones de tráfico que consuman el ancho de banda y generen condiciones críticas. Por ejemplo, sistemas Ad-hoc que analizan tráfico en Internet en tiempo real son usados para contabilizar estadísticas de tráfico y detectar condiciones de congestión o negación de servicio. La técnica más usada para esta aplicación es agrupación y generalmente los ejemplos que se encuentran en la literatura son algoritmos de partición.

d) **"**Recuperación de información (IR). Dentro de esta amplia categoría, se encuentra la búsqueda en bases de datos, ya sea a través de internet, intranet, para textos, imágenes, sonido o datos de otras características, de manera pertinente y relevante. Un ejemplo, son las aplicaciones de comercio electrónico, puesto que los clientes son la principal preocupación de cualquier negocio, la mayoría de los análisis de datos de minería están en el nivel del cliente. Es decir, cada registro de un conjunto de datos en la etapa final de análisis es la que contiene toda la información sobre él. Cada cliente puede tener múltiples filas en estos niveles. Para hacer que esta información detallada sea útil en el análisis, la agregación de los atributos son muy importantes [7].

En la siguiente ilustración se puede visualizar el funcionamiento de la minería de flujo de datos, en esta también se describe a su vez las características, consideraciones, como trabaja, las posibles aplicaciones y que tipo de restricciones posee.

Ilustración 3 Mapa conceptual de los tópicos más representativos en la minería de los flujos de datos.



Fuente. A. Rojas, Modelo Basado en Minería de Flujos de Datos para el Análisis de CLICS en un Sitio WEB. [9]

5.1.1.3 Restricciones de la minería de flujos de datos

Cuando se aplica minería en flujos de datos existen condiciones que marcan diferencias con los conjuntos de datos convencionales, es decir, aquellos que residen en disco y permiten varias lecturas completas al conjunto de datos. Como el volumen de los flujos de datos es tan grande y su tasa de generación es tan rápida y variable, no es factible pensar en almacenarlos, ni mucho menos, en procesarlos eficientemente utilizando procedimientos que requieran usar un dato muchas veces; por lo que los algoritmos de minería deben ser diseñados o adaptados para trabajar con los datos en tan solo una pasada. Adicionalmente debe ser tenido en cuenta el componente temporal implícito que existe entre un dato y otro. Esto último se debe a que los datos tienden a evolucionar en el tiempo; por ello el diseño de los algoritmos debe estar orientado a modelar con el paso del tiempo, lo que implica que el modelo debe ser dinámico**”**.

“Esas características marcan restricciones en los sistemas que trabajan con este tipo de datos, las cuales se mencionan a continuación:

1. Los flujos de datos tienen potencialmente un tamaño ilimitado.
2. No es factible almacenar estos datos. Cuando un dato es procesado, este debe ser descartado.
3. Los datos llegan en línea y se generan continuamente.
4. No hay manera de controlar el orden en el que llegan los datos.
5. Los datos dentro del flujo tienen una correlación temporal, lo que marca un orden de llegada específico para cada uno.
6. Son datos altamente cambiantes que presuponen una evolución implícita con el paso del tiempo [7].

5.1.1.4 Condiciones de los algoritmos de minería de flujos de datos

Con respecto a los algoritmos en la minería de datos es importante resaltar que son un conjunto de cálculos y reglas heurísticas, que permiten generar un modelo de minería a partir de datos. Para crear un modelo, el algoritmo analiza los datos proporcionados, en búsqueda de tipos específicos de patrones o tendencias; bajo ciertas condiciones que se presentan a continuación [7]:

- *La naturaleza de la alta velocidad de secuencias de datos:* Una de las características inherentes en los flujos de datos es su alta velocidad. Los algoritmos deben ser capaces de adaptarse a la naturaleza de alta velocidad de transmisión de la información. La tasa de construcción de una modelo de clasificación debe ser superior a la tasa de datos. Por otra parte, no es posible escanear los datos más de una vez. Esto se conoce como la restricción de una sola pasada.

- *Requisitos de memoria sin cotas:* Las técnicas de clasificación necesitan datos que residan en memoria para construir el modelo. Las enormes cantidades de flujos de datos generados

rápidamente dictan la necesidad de la memoria sin límites. Este reto se ha afrontado mediante la liberación de carga, muestreo, la agregación, y la creación de sinopsis de datos. El problema de memoria es una motivación importante detrás de muchas de las técnicas desarrolladas en este campo.

- *Concepto de variabilidad de los datos con el tiempo (Drifting):* Este concepto cambia los resultados del clasificador con el tiempo. Esto es debido al cambio en los patrones de datos subyacente. También es denominado evolución del flujo de datos, que se traduce en un modelo que se vuelve obsoleto y menos relevante en el tiempo. La captura de dichos cambios fomenta la renovación del modelo clasificador eficazmente, mientras que la utilización de un modelo desactualizado podría conducir a una precisión muy baja en la clasificación.

- *Equilibrio entre precisión y eficiencia:* La desventaja principal de los algoritmos de minería de flujos de datos está en la exactitud de la salida con respecto a la aplicación, el tiempo y la complejidad del espacio. En muchos casos, los algoritmos de aproximación puede garantizar los límites de error, manteniendo al mismo tiempo un alto nivel de eficiencia**.**

- *Desafíos en las aplicaciones distribuidas:* Un número significativo de las fuentes de flujos de datos son aplicaciones que deben ejecutarse con un ancho de banda limitado, tales como redes de sensores y dispositivos de mano. Así, la representación de la estructura del conocimiento es una cuestión importante. Después de extraer modelos y patrones a nivel local a partir de los generadores de flujo de datos o receptores, es importante la transferencia de los resultados de la minería de datos para el usuario. Esto es a menudo un desafío debido a los límites de ancho de banda en la transferencia de datos.

- *Visualización de los resultados de la minería de flujos de datos:* La visualización de los datos y resultados de la minería tradicional ha sido un tema de investigación de hace más de una década. Por ejemplo, visualizar datos en una PDA (Personal Digital Assistant), es un verdadero desafío y un problema abierto de investigación, debido a los recursos

limitados de estos dispositivos. Dado un escenario en el que un empresario en movimiento está viendo y analizando datos en su PDA, dichos resultados de este análisis deben ser desplegados de manera diciente en una forma que le permitan tomar una decisión rápida**.**

- *Modelado de cambio de los resultados de la minería en el tiempo:* En algunos casos, el usuario no se interesa en ver los resultados de la minería de datos, sino en cómo estos resultados están cambiando en el tiempo. Los cambios durante la clasificación podrían ayudar a comprender la transformación de los flujos de datos en el tiempo.

Las condiciones mencionadas anteriormente, representan una parte importante para el funcionamiento eficiente del modelo de minería de datos, la selección cual radica en el mejor algoritmo para una tarea analítica específica, pero esto, no quiere decir que solo se deba usar una clase de algoritmos, se pueden usar diferentes algoritmos para realizar la misma tarea, cada uno de ellos genera un resultado diferente, y algunos pueden generar más de un tipo de resultado. Para este caso de investigación hizo uso algoritmos de agrupación y de búsqueda que se definen a continuación.

5.1.2 ALGORITMOS DE AGRUPACIÓN

La clasificación no supervisada o agrupamiento, es usualmente entendida como el proceso de minería de datos que tiene como finalidad esencial “revelar concentraciones de datos (casos o variables) para su agrupamiento eficiente en grupos según su homogeneidad” [10]. En otras palabras, se busca particional un conjunto de objetos de tal manera que cada grupo contiene objetos que son más parecidos entre sí y a su vez diferentes a los objetos de otros grupos, de acuerdo con alguna medida de proximidad o similaridad [11]. Estos métodos relacionan un patrón a un grupo siguiendo algún criterio de similaridad. Tales medidas de similaridad deben ser aplicables entre patrones, grupos y finalmente entre pares de grupos. Generalmente las medidas de similaridad que se emplean son las *métricas de distancia* [12].

Es importante resaltar que la agrupación se trata de una tarea de agrupación no supervisada, en donde los grupos se crean en función de la naturaleza de los datos, es decir estamos tratado con una técnica *post hoc,* que puede verse como un análisis estadístico multivalente de agrupación automática a partir de un conjunto de datos que trata de asignarlos a grupos homogéneos no conocidos de antemano pero si sugeridos por la esencia de los datos, su objetivo es otorgar al usuario un listado de los elementos que conforman cada uno de los grupos que se obtienen. Estos agrupamientos detectados dependen del algoritmo empleado, del valor dado a sus parámetros, de los datos utilizados y de la medida de similaridad adoptada. Se han propuesto cientos de algoritmos de agrupamiento más o menos específicos que pueden ser de tipo directo o indirecto (basados en aproximaciones heurísticas) de los algoritmos indirectos o **por optimización** [12].

En ocasiones el número de grupos es también un problema a resolver. En general, se puede enfrentar a dos tipos fundamentales de problemas de agrupación: (1) determinar las características de la proximidad que se debe utilizar para agrupar los objetos y (2) encontrar una buena medida de proximidad para resolver un problema en particular [11].

Matemáticamente, un problema de agrupamiento de datos es definido como $O = \{O_1, \ldots, O_n\}$ donde O es un conjunto finito de n objetos (vector) en un espacio de elementos S, como se mencionó el objetivo de un problema de clasificación de datos es hallar la partición optima de objetos O, $C = \{C_1, \ldots C_D\}$, $O = \cup_{i=1}^{D} C_i$, $y\ C_i \cap C_j = \varphi; para\ i \neq j$, Donde C_i representa la *i-ésima* grupo de la partición C, de tal manera que los datos que pertenecen al mismo grupo son similares mientras que son lo más diferentes posible a los datos que pertenecen a otros grupos en términos de una función de medición de distancia.

Como se mencionó, la medida de distancia es uno de los elementos clave en la solución de problemas de agrupación de datos, por lo general la similaridad entre dos objetos diferentes O_i y O_j está relacionada a una medida de distancia en un espacio S, Existen diferentes tipos de medidas de similitud utilizadas para este tipo de problemas, donde una de las más comunes es la *distancia euclidiana* [13].

5.1.2.1 Medidas de Similitud

Según la clasificación de Sneath y Sokal existen cuatro grande tipos de medidas de similitud [10]:

- *Distancias*: Se trata de las distintas medidas entre los puntos del espacio definidos por los individuos, miden la disimilitud entre elementos y el ejemplo clásico de este tipo es la distancia euclidiana.

- *Coeficientes de Asociación*: estas son utilizadas cuando se trabaja con datos cualitativos, aunque es posible utilizarlos con datos cuantitativos, permiten medir la concordancia o conformidad entre los estados de dos columnas de datos.

- *Coeficientes Angulares:* Permite medir la proporcionalidad e independencia entre los vectores que definen los individuos. El más común es el coeficiente de correlación aplicado a variables continuas.

- *Coeficientes de similitud probabilística:* miden la homogeneidad del sistema por particiones o subparticiones del conjunto de los individuos e incluyen información estadística. La idea de utilizar estos coeficientes se basa en relacionarlos con estadísticos. Sus propiedades principales es que son aditivos, se distribuyen como Chi-cuadrado y son probabilísticas. Esta última propiedad permite en algunos casos establecer una hipótesis nula y constatarla por los métodos estadísticos tradicionales.

Tabla 1 Definición de Distancias utilizadas para Medidas de Similitud

$$Distancias \begin{cases} Distancia\ Euclidiana\ al\ cuadrado\ d(i,j)^2 = \sum_k \left(x_{ik} - x_{jk}\right)^2 \\ Distancia\ euclidiana\ d(i,j) = \sqrt{\sum_k \left(x_{ik} - x_{jk}\right)^2} \\ Distancia\ de\ Minkowski\ \ d_q(i,j) = \left(\sum_k \left|x_{ik} - x_{jk}\right|^q\right)^{\frac{1}{q}} \\ Distancia\ City - Block\ o\ de\ Manhatan\ d_I(i,j) = \sum_k \left|x_{ik} - x_{jk}\right| \\ Distancia\ de\ Chebichev\ d_\infty(i,j) = Max_k\left(\left|x_{ik} - x_{jk}\right|\right) \\ Distancia\ de\ Canberra\ d_{CANB}(i,j) = \sum_k \frac{\left|x_{ik} - x_{jk}\right|}{(x_{ik} + x_{jk})} \end{cases}$$

Fuente A. Adaptado de C. Pérez, Minería de Datos: Técnicas y Herramientas [10]

Ahora, se procede a profundizar en los coeficientes de asociación ya que uno de estos es el que es utilizados en algoritmo de búsqueda gravitacional para agrupación, algoritmo que es el eje central de esta tesis, Los *coeficientes asociación* son utilizados comúnmente en variables cualitativas y especialmente en datos binarios, para explicar su funcionamiento se va a hacer uso de una tabla de frecuencias de 2x2 en la que el número de elementos de la población en los que se constata la presencia o ausencia del carácter de estudio.

Tabla 2 Tabla de Frecuencias (Coeficientes de Asociación)

Variable 1 → *Variable 2 ↓*	*Presencia*	*Ausencia*
Presencia	*a*	*b*
Ausencia	*c*	*d*

Fuente A. Adaptado de C. Pérez, Minería de Datos: Técnicas y Herramientas [10]

Tabla 3 Defincón de los Coeficientes de Asociación

$$Coeficientes\ de\ Asociación \begin{cases} Jaccard - Sneath\ S_i = \dfrac{a}{a+u} = \dfrac{a}{(a+b+c)} \\ Coeficiente\ de\ emparejamiento\ simple \\ S_{Sm} = \dfrac{m}{(m+u)} = \dfrac{m}{n} = \dfrac{(a+d)}{(a+b+c+d)} \\ Coeficiente\ de\ Yule\ S_Y = \dfrac{(ad-bd)}{(ad+bc)} \end{cases}$$

Fuente A. Adaptado de C. Pérez, Minería de Datos : Técnicas y Herramientas [10]

El **coeficiente de Jaccard** es el coeficiente utilizados por el algoritmo de agrupación basado en la ley de gravitación universal considerado en esta investigación, es uno de los coeficientes más sencillos y no tiene en cuenta los emparejamientos negativos, puede entenderse como el número de emparejamientos positivos entre la suma de los emparejamientos positivos y desacuerdos. El coeficiente de Jaccard tiende a cero cuando *a/u* tiende a cero y es igual a cero cuando el número de emparejamientos positivos coincide con el número de desacuerdos, así mismo este coeficiente tiende a 1 cuando *u* tiende a cero, es decir, S_j vale uno cuando no hay de acuerdos.

5.1.2.2 Técnicas en el análisis de agrupación

Como se mencionó anteriormente el problema de agrupación de datos tiene como objetivo definir un conjunto de particiones de un conjunto de datos en base a determinadas características de los mismos, Estas características estarán definidas por la puntuaciones que cada uno de ellos tiene con relación a diferentes variables [10]. Si se logra el objetivo anterior, se consigue que dos individuos son similares si pertenecen al mismo grupo, así mismo todos los individuos que pertenecen al mismo grupo se parecerán entre si y serán diferentes de los individuos que pertenecen a otros grupos, por lo tanto, los miembros de un mismo grupo gozan de características comunes que los diferencian de los miembros de otro

grupo, estas características deberán por definición ser genéricas y difícilmente una única característica podrá definir un grupo.

Los diferentes métodos para realizar análisis de agrupación se generan desde las diferentes firmas de adelantar la agrupación de individuos, es decir, dependido del algoritmo que se utilice para hallar a cabo la agrupación de individuos se obtienen diferentes métodos de análisis de agrupación, Una clasificación de los métodos de análisis de conglomerados basada en los algoritmos de agrupación de individuos podría ser la siguiente [10]

- *Métodos Aglomérativos - Divisivos:* un método es aglomerativo si considera tantos grupos como individuos y sucesivamente va fusionando los dos grupos más similares, hasta llegar a una clasificación determinada; mientras que un método es divisivo si parte de un solo grupo formado por todos los individuos y va separando los individuos de los grupos previos formando así nuevo grupos.

- *Métodos Jerárquicos – No jerárquicos:* un método es jerárquico si consiste en una secuencia $g+1$ grupos: $G_0, \ldots, G_g$ en la que G_0 es la partición disjunta de todos los individuos y G_g es el conjunto partición. El número de partes de cada una de ñas particiones disminuye progresivamente, lo que hace que estas sean cada vez más amplias y menos homogéneas. Por el contrario, un método es llamado No – jerárquicos cuando se forman grupos homogéneos sin establecer relaciones de orden o jerarquía.

- *Métodos Solapados – Exclusivos:* un método es solapado si admite que un individuo pueda pertenecer a dos grupos simultáneamente en alguna de las etapas de agrupación, y es exclusivo si ningún individuo puede pertenecer simultáneamente a dos grupos en la misma etapa.

- *Métodos Secuenciales – Simultáneos:* un método es secuencial si a cada grupo se le aplica el mismo algoritmo de forma recursiva, mientras que los métodos simultáneos son aquellos en los que la agrupación se logra por una simple y no reiterada operación sobre los individuos.

- *Métodos Monotéticos – Politéticos:* un método se dice monotético si está basado en una característica única de los objetos, mientras que es politético si se basa en varias características de los mismos. Sin exigir que todos los objetos las posean, aunque si las suficientes como para poder justificar la analogía entre los miembros de una misma clase.

- *Métodos Directos – Iterativos:* un método es directo si utiliza algoritmos en los que una vez asignado un individuo a un grupo ya no se saca del mismo, mientras que los métodos iterativos corrigen las asignaciones previas volviendo a comprobar en posteriores iteraciones si la asignación de un individuo a un grupo es óptima, llevando acabo un nuevo reagrupamiento de los individuos si es necesario.

- *Métodos Ponderados – No Ponderados:* los métodos no ponderados son aquellos que establecen el mismo peso a todas las características de los individuos, mientras que los métodos ponderados hacen recaer mayor peso en determinadas características.

- *Métodos Adaptativos – No Adaptativos:* los métodos no adaptativos son aquellos para los que el algoritmo utilizado se dirige hacia una solución en la que el método de formación de conglomerados es fijo y esta predeterminado, mientras que los adaptativos son aquellos en los que de alguna manera aprenden durante el proceso de formación de grupos y modifican el criterio de optimización a la medida de similitud a utilizar.

5.1.3 ALGORITMOS DE AGRUPACIÓN CONVENCIONALES (CLÁSICOS)

Por su popularidad y efectividad en diferentes campos de la agrupación son llamados algoritmos de agrupación convencionales el algoritmo K–medias y los mapas Auto-organizativos de Kohonen SOM [14] [15], estos serán estudiados en las siguientes secciones:

5.1.3.1 Algoritmo K-medias

El algoritmo de las K medias es probablemente el algoritmo de agrupamiento más utilizado gracias a su sencillez y eficiencia, es un algoritmo no jerárquico que se basa en la minimización de la distancia interna de los individuos asignados a un grupo. Es un método de agrupamiento heurístico con un número de clases llamadas *K* [16]. Está destinado a situaciones en las cuales todas las variables son del tipo cuantitativo y utiliza la distancia **cuadrática Euclidiana** como medida de similaridad.

$$d(X_i, X_{i'}) = \sum_{j=1}^{p} (X_{ij} - X_{i'j})^2 = \|X_i - X_{i'}\|^2 \quad (1)$$

Donde, se puede notar que los valores en la distancia *Euclidiana* pueden ser usados redefiniendo a los valores x_{ij} [17].

Los puntos de dispersión pueden ser escritos como:

$$W(C) = \sum_{k=1}^{k} \sum_{C(i)=k} \sum_{C(i')=k} \|X_i - X_{i'}\|^2 \quad (2)$$

$$W(C) = \sum_{k=1}^{k} Nk \sum_{C(i)=k} \|X_i - \overline{X}_K\|^2 \quad (3)$$

Donde, $\overline{X}k = (\overline{X}_1k, \ldots\ldots, \overline{X}_pk)$, es el vector de medias asociado con el *k-ésimo* grupo o cluster, y $Nk = PN\ i=1\ I\ (C\ (i) = k)$. Así, el criterio es asignar las *N* observaciones a los *K* clusters de modo que dentro de cada cluster el promedio de las diferencias de cada observación a la media del cluster, definido por los puntos del cluster, sea mínima [17].

A. Principios del algoritmo K-medias (K-means)

El algoritmo está basado en la minimización de la distancia interna (la suma de las distancias de los individuos asignados a un agrupamiento al centroide de dicho agrupamiento). De hecho, este algoritmo minimiza la suma de las distancias al cuadrado de cada patrón al centroide de su cluster [16]. El algoritmo K-medias es considerado como un algoritmo de clasificación no supervisado. Este requiere que de antemano se especifique los k grupos que se quieren obtener.

- Si supone que se tiene un juego de datos compuesto por *n* casos o instancias.
- Se podrá llamar *X* a este juego de datos $X = [X_1, X_2, \ldots X_i, \ldots X_n]$ donde cada X_i podría ser un cliente con atributos, $X_i = (X_{i1}, X_{i2}, \ldots X_{im})$ [18].

Para clasificar el juego de datos *X* mediante el algoritmo *k*-medias se deben seguir los siguientes 5 pasos:

1) De los n casos se selecciona *k*, que se llamará semillas y se denotara por C_j, $j = 1, \ldots k$. Cada semilla C_j identificará su cluster C_j Es decir, $(X_i, C_j) = min\ [d(X_i, C_j)], t = 1, \ldots k$.

2) Asignar el caso X_i al cluster C_j cuando la distancia entre el caso X_i y la semilla C_j sea la menor entre todas las semillas.

3) Calcular la mejora que se produciría si asignará un caso a un cluster al que no pertenece actualmente. Se debería seleccionar un criterio apropiado, para medir esta mejora. Un ejemplo podría ser el de minimizar la distancia de las distintas instancias o casos a sus respectivos centros.

4) Hacer el cambio que proporciona una mayor mejora.

5) Repetir los pasos 3 y 4 hasta que ningún cambio sea capaz de proporcionar alguna mejora.

B. Funcionamiento del algoritmo K-medias (K-means)

Algoritmo de las k medias:

Seleccionar arbitrariamente una configuración inicial de los clusters y repetir:

- Calcular los centros de los K clusters.
- Redistribuir los patrones entre los clusters utilizando la mínima distancia euclidiana al cuadrado como clasificador.
- Hasta que no cambien los centros de los clusters [16].

A continuación se presentara un ejemplo sencillo en el que se tomaran para simplificar como medida de distancia el cuadrado de la distancia euclidiana, y como criterio de mejora, la minimización de la suma de distancias de cada caso a su semilla correspondiente [18].

Se toma el juego de datos:

$$X = \begin{pmatrix} 1 & 1 \\ 2 & 2 \\ 3 & 0 \\ 7 & 3 \\ 8 & 4 \end{pmatrix} \begin{matrix} \_caso\_1 \\ \_caso\_2 \\ \_caso\_3 \\ \_caso\_4 \\ \_caso\_5 \end{matrix} \tag{4}$$

Aplicando el algoritmo k-medias tomando k=2 y tomando como semillas iniciales los casos 1 y 3.

Siguiendo con nuestra notación $C_1 = (1,1)$ y $C_2 = (3,0)$, tenemos que y serán respectivamente los centros de los clusteres C_1 y C_2.

Partición 0

Para los casos 2, 4 y 5 decidiremos a cuál de los clusteres pertenecen:

$$\left.\begin{matrix} d^2(X_2, c_1) = (2-1)^2 + (2-1)^2 = 2 \\ d^2(X_2, c_2) = (2-3)^2 + (2-0)^2 = 5 \end{matrix}\right\} De\ modo\ que\ X_2 \in C_1$$

$$\left.\begin{array}{l} d^2(X_4, c_1) = (7-1)^2 + (3-1)^2 = 40 \\ d^2(X_4, c_2) = (7-3)^2 + (3-0)^2 = 25 \end{array}\right\} \textit{De modo que } X_4 \in C_2$$

$$\left.\begin{array}{l} d^2(X_5, c_1) = (8-1)^2 + (4-1)^2 = 58 \\ d^2(X_5, c_2) = (8-3)^2 + (4-0)^2 = 41 \end{array}\right\} \textit{De modo que } X_5 \in C_2$$

(5)

Resumiendo, tenemos que la partición está formada por 2 clusteres que contienen los siguientes casos: $C_1 = [X_1, X_2]$ y $C_2 = [X_3, X_4, X_5]$ [18].

Partición 1

"En primer lugar calcularemos las medias o centros de los 2 clusteres:

Donde $\overline{X}_1 = (1{,}5\ ;\ 1{,}5)$, $\overline{X}_2 = (6;\ 2{,}3)$

El criterio para valorar si las siguientes particiones son mejores o no será el de minimizar la distancia de los casos a sus respectivos centros.

Calculemos entonces el valor del criterio S para la Partición P_o:

$$S(P_o) = (1-1{,}5\ ;\ 1-1{,}5)^2 + (2-1{,}5\ ;\ 2-1{,}5)^2 + (3-6\ ;\ 0-2{,}3)^2 + (7-6\ ;\ 3-2{,}3)^2 + (8-6\ ;\ 4-2{,}3)^2 = 23{,}7$$

(6)

Ahora deberemos cambiar cada caso de cluster siempre que el cambio suponga una mejora en el valor Si (P_o).

"Por ejemplo, si asignamos el caso 3 al cluster c_1 podemos ver cómo se produce una mejora significativa en el valor del criterio, donde P_1 sería la nueva partición formada por el cluster $C1 = \{X_1, X_2, X_3\}$ y $C1 = \{X_4, X_5\}$ y donde, $\overline{X}_1 = (2{,}1)$, $\overline{X}_2 = (7.5,\ 3.5)$ serían las nuevas medias o centros:

$$S(P_1) = (1 - 2{,}5\,;\, 1 - 1)^2 + (2 - 2\,;\, 2 - 1)^2 + (3 - 2\,;\, 0 - 1)^2 + (7 - 7\,;\, 3 - 3{,}5)^2 + (8 - 7{,}5\,;\, 4 - 3{,}5)^2 = 5 \tag{7}$$

Como este cambio mejora el valor del criterio S, lo daríamos por bueno.

Después de haber desarrollado numéricamente este ejemplo tan simple, es fácil entender que uno de los problemas que presenta k-medias es el gran número de cálculos que requiere [18].

C. Inconvenientes del algoritmo K-medias (K-means)

Como se puede percibir, el algoritmo k-medias puede presentar numerosas versiones en función de la métrica de distancia que se utilice y en función del criterio que se seleccione para medir la mejora producida por la asignación de un caso a un cluster distinto del actual. Normalmente el criterio tratará de minimizar alguna función. También es fácil distinguir que el algoritmo no siempre alcanzará un óptimo global, puede darse el caso de que antes encuentre un óptimo local. Además, también es posible que el algoritmo no sea capaz de encontrar ningún óptimo, bien porque simplemente el juego de datos no presente ninguna estructura de clusteres, bien porque no se haya escogido correctamente el número k de cluster a construir.

Y finalmente el algoritmo k-medias contiene un **paso de optimización**. Se trata de la optimización de la mejora obtenida al encontrar nuevos centros. Como todo proceso de optimización, cuando encontramos un extremo debemos preguntarnos si se trata de un extremo local o absoluto [18].

D. Comentarios sobre ventajas y debilidades sobre las K-medias

Ventajas

- Relativamente eficiente
- Generalmente termina con un óptimo local.

Debilidad

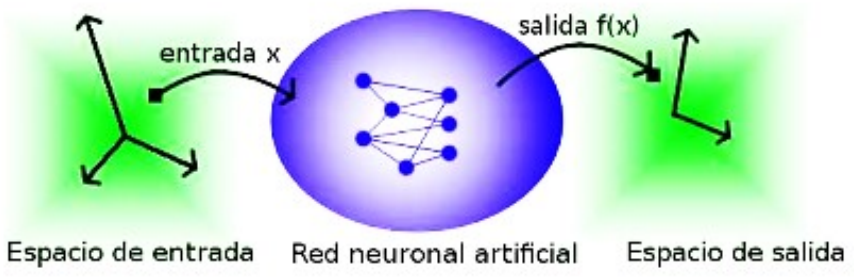
- Solo es aplicable cuando la media está definida. Es decir que no es funcional para datos categóricos.
- Se necesita especificar K de antemano.
- No es capaz de tratar con ruido No es apropiado para descubrir cluster que no tengan formas no convexas [19].
- "El algoritmo es sencillo y eficiente. Además, procesa los patrones secuencialmente (por lo que requiere un almacenamiento mínimo). Sin embargo, está sesgado por el orden de presentación de los patrones (los primeros patrones determinan la configuración inicial de los agrupamientos) y su comportamiento depende enormemente del parámetro K [16].

5.1.3.2 Mapas auto-organizativos de Kohonen SOM

Los mapas auto-organizativos fueron introducidos por T. Kohonen en 1982 y son un tipo especial de redes neuronales artificiales de aprendizaje no supervisado que ha sido exitosamente aplicado en Minería de Datos (Data Mining) y en metodología de Descubrimiento de Conocimiento en Base de Datos (Knowledge Discovery Database - KDD) con una gran variedad de aplicaciones de ingeniería tales como reconocimiento de patrones, análisis de imágenes, monitoreo de procesos y detección de fallas por nombrar algunas [20].

Esta red o mapa debe descubrir rasgos comunes, regularidades, correlaciones o categorías en los datos de entrada, e incorporarlos a su estructura interna de conexiones. Entonces, se dice que las neuronas deben auto-organizarse en función de los estímulos (datos) procedentes del exterior [21] como se observa en la ilustración 7.

Ilustración 4 Red neuronal artificial como una función.



Fuente A. Díaz R, Redes Neuronales No Supervisadas con Topología Dinámica de la Segmentación de Imágenes a Color [22]

Estos mapas tiene entonces la función de transformar patrones de dimensión arbitraria como respuesta de los patrones de una o dos dimensiones de neuronas en arreglo que cambian adaptándose en función de los rasgos característicos de entrada. Se puede afirmar qué los algoritmos de los mapas de auto-organizativos pueden definirse para visualización de datos multidimensionales [23].

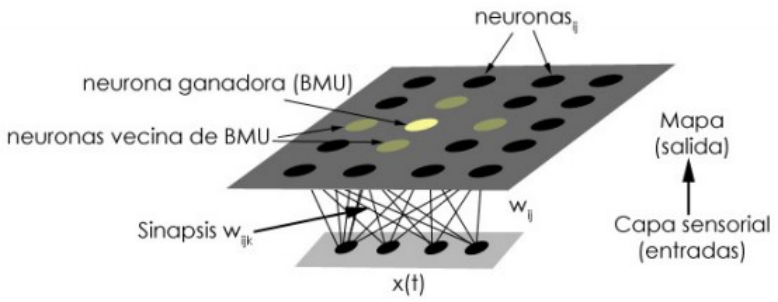
A. Estructura del SOM

Un modelo SOM está compuesto por los siguientes elementos [24]:

- *Matriz de neuronas:* La capa de entrada (formada por N neuronas, una por cada variable de entrada) se encarga de recibir y transmitir a la capa de salida la información procedente del exterior. La capa de salida (formada por M neuronas) es la encargada de procesar la información y formar el mapa de rasgos. Normalmente, se organizan en forma de mapa bidimensional que pueden ser rectangulares o hexagonales.
- *Relación entre neuronas*: Entre todas las neuronas hay una relación de vecindad que es la clave para conformar el mapa durante la etapa de entrenamiento. Esta relación viene dada por una función.

En el ilustración 8, se detallan los elementos que componen a un mapa de Auto organización.

Ilustración 5 Estructura de mapas de auto organizados SOM



Fuente J. Serrano, Aplicación de Mapas Autoorganizados (SOM) a la Visualización de datos [24]

B. El algoritmo del SOM

El proceso de aprendizaje del SOM es el siguiente:

Paso 1. Un vector x es seleccionado al azar del conjunto de datos y se calcula su distancia (similitud) a los vectores del codebook, usando, por ejemplo, la distancia Euclidiana:

$$\|x - m_c\| = \min\{\|x - m_j\|\} \tag{8}$$

Paso 2. Una vez que se ha encontrado el vector más próximo el resto de vectores del codebook es actualizado. El vector más próximo y sus vecinos (en sentido topológico) se mueven cerca del vector x en el espacio de datos. La magnitud de dicha atracción está regida por la tasa de aprendizaje. Mientras se va produciendo el proceso de actualización y nuevos vectores se asignan al mapa, la tasa de aprendizaje decrece gradualmente hacia cero. Junto con ella también decrece el radio de vecindad también. La regla de actualización para el vector de referencia dado i es la siguiente:

$$m_j(t+1) = \begin{cases} m_j(t) + \alpha(t)\left(x(t) - m_j * (t)\right) & j \in N_c(t) \\ m_j(t) & j \in N_c(t) \end{cases}$$

(9)

Los pasos 1 y 2 se van repitiendo hasta que el entrenamiento termina. El número de pasos de entrenamiento se debe fijar antes a priori, para calcular la tasa de convergencia de la función de vecindad y de la tasa de aprendizaje. Una vez terminado el entrenamiento, el mapa ha de ordenarse en sentido topológico: n vectores topológicamente próximos se aplican en n neuronas adyacentes o incluso en la misma neurona [21].

C. Funcionamiento de la red SOM

El funcionamiento de la red es relativamente simple cuando se realiza la entrada de la información $E_k = \left({e_1}^{(K)},,,,,,,,{e_n}^{(K)}\right)$ cada una de las M neuronas de la capa de salida la recibe a través de *feedbook* que son la conexiones de peso W_{ij}. De la misma forma, las neuronas reciben las correspondientes entradas de conexiones laterales o de salida cuya influencia dependerá de la distancia a la que se encuentran.

De este modo, la salida generada por una neurona de salida *j* ante un vector de entrada E_K seria:

$$S_1(t+1) = f\left(\sum_{i=1}^{N} W_{ij}{e_i}^{(k)} + \sum_{p=1}^{M} Int_{pj}{S_p}^{(k)}\right)$$

(10)

Donde Int_{pj} es una función que representa la influencia lateral de la neurona p sobre la neurona j.

Es evidente que se trata de una red de tipo competitivo, dado que al presentar una entrada E_k la red evoluciona hasta una situación estable en la que se activa una neurona de salida, la vencedora. Para ello, la formulación matemática puede simplificarse a la siguiente expresión:

$$S_1 = \{1\,\mathrm{MIN}\|E_k - w_j\| = \min\left(\sqrt{\sum_{i=1}^{N}\left(e_i{}^{(k)} - W_{ij}\right)^2}\right)$$

(11)

Donde $\|E_k - w_j\|$ es una medida (por ejemplo distancia eucledía) dela diferencia del vector de entrada $E_k = \left(e_1{}^{(K)},,,,,,,,e_n{}^{(K)}\right)$ y del vector de los pesos $E_k = \left(e_1{}^{(K)},,,,,,,,e_n{}^{(K)}\right)$ de las conexiones entre cada una de las neuronas de entrada y de salida j. En estos pesos se registran los datos almacenados en la red durante el proceso de aprendizaje. En la fase de funcionamiento se pretende encontrar el dato más similar al de la entrada para investigar que neurona se activara y en qué zona del espacio bidimensional de salida se encuentra [23].

D. Ventajas de mapas SOM

Las ventajas de los mapas autoorganizativos radican en que son capaces de preservar la topología del espacio de los datos, proyectan datos altamente dimensionales a un esquema de representación de baja dimensión y tienen la habilidad de encontrar similitudes en los datos [20].

5.1.4 SEGMENTACIÓN DE IMÁGENES

Las imágenes están en todos lados y como humanos casi toda la información que procesamos está en forma de imágenes, por ejemplo cuando miramos una fotografía, vemos televisión, admiramos una pintura o leemos un libro. Más aún, nuestra visión es el más eficiente de nuestros sentidos.

Se realizan una gran cantidad de tareas de procesamiento de imágenes. Por ejemplo, cuando miramos algo, la primera imagen que nuestros ojos envían al cerebro esta posiblemente fuera de foco. El cerebro intenta corregir esto ajustando los lentes oculares, entonces una nueva

imagen es enviada de los ojos al cerebro [25]. Este proceso de retroalimentación es tan rápido que no se puede percibir ni sentir. Otro ejemplo es la estereovisión, en donde nuestros ojos envían dos imágenes bidimensionales al cerebro y este es capaz de fusionarlas en una imagen tridimensional, todo esto de manera instantánea.

El procesamiento de imágenes combina esta forma natural de como los humanos usan las imágenes con la matemática. Esto produce una mezcla única, ya que las imágenes y el procesamiento de imágenes son descritos con rigor matemático pero sin perder el carácter intuitivo [25].

Pero en realidad, el procesamiento de imágenes tiene una naturaleza interdisciplinaria, porque antes de que nosotros podamos procesar una imagen, necesitamos conocer como la señal digital está relacionada a las características de los objetos en la imagen. La comprensión de estos procesos recae en la física. Luego, un sensor convierte la irradiación incidente en una forma de señal eléctrica para luego convertir esa señal en números digitales y ser procesada por una computadora digital para extraer información relevante. En esta cadena de procesos, intervienen muchas áreas: Física, Ciencia de la Computación y Matemática.

Más aún, las tareas del procesamiento de imágenes pueden ser parcialmente vistas como un problema de medida, el cual es parte de la ciencia conocida como metrología. Asimismo, las tareas de reconocimiento de patrones son muchas veces incorporadas dentro del procesamiento de imágenes. Existen otras disciplinas con conexiones relacionadas como: las Redes Neuronales, Inteligencia Artificial y la Percepción Visual.

El segundo aspecto importante de la naturaleza interdisciplinaria del procesamiento de imágenes es su extenso campo de aplicación. No existe campo en las ciencias naturales o

disciplinas técnicas en donde el procesamiento de imágenes no sea aplicado. Esta es una de las causas por las que ´este campo ha ganado terreno y se hace notorio su rápido progreso.

El interés en el procesamiento digital de imágenes proviene de dos principales áreas de aplicación: el mejoramiento de información pictórica para interpretación humana y el procesamiento de imágenes de datos para almacenar, transmitir y representar información en la percepción de máquinas autónomas [25].

Para la segmentación de imágenes. El objetivo principal de segmentación de imágenes es la generación de un dominio de particiones independientes (domain independent partitioning) de una imagen, este es un conjunto de regiones disjuntos que son visualmente distintas, homogéneas y significativas con respecto a algunas características [26], en otras palabras el objetivo de la segmentación de imágenes consiste en agrupar píxeles en áreas de la imagen que tienen un comportamiento similar significativo con respecto a una aplicación particular y en la actualidad es uno de los principales desafíos del procesamiento digital de imágenes.

Algoritmos de agrupamiento han sido utilizados para segmentación de imágenes [27], uno de los algoritmos de agrupamiento más populares es el algoritmo k-medias, este algoritmo numérico, no supervisado, no determinista e iterativo ha sido usado ampliamente en diferentes campos incluida la segmentación de imágenes, sin embargo, en general se han identificado las siguientes debilidades:

- El número de grupos, K, se debe determinar antes de implementar el algoritmo. Este procedimiento consume tiempo y es demasiado subjetiva para diferentes usuarios.

- El algoritmo es sensible a las condiciones iniciales (es decir, diferentes condiciones iniciales pueden producir diferentes resultados de cluster). Además, el algoritmo puede ser atrapado en el óptimo local. Como resultado, los racimos o centros atrapados podrían representar grupos erróneas de datos.

- Los datos que están aislados lejos de los centros pueden tirar de los centros fuera de su ubicación óptima. Esto podría conducir a una mala representación de los datos.

- A consecuencia de las anteriores debilidades han sido propuestos varios algoritmos de agrupación que pretenden superarlas (por ejemplo el hard c – mean y el fuzzy c-means)

Los algoritmos convencionales de agrupación aplicados para la segmentación de imágenes. Muchos algoritmos de segmentación como ya se ha visto se han desarrollado para diversas aplicaciones en algunos casos con resultados insatisfactorios.

"El término de segmentación (a menudo referido como agrupación o partición), abarca una amplia serie de procesos a través de los cuales se obtiene la división de la imagen en distintas regiones espaciales disjuntas (ver ilustración 9), atendiendo a una cierta homogeneidad de éstas. Dicha homogeneidad se establece según una serie de criterios tales como:

- Características de bajo nivel (por ejemplo, el color o la textura).

- Conocimiento previo sobre la escena o imagen (por ejemplo, suavidad de los bordes).

- Conocimientos de alto nivel (estos son los denominados, modelos semánticos)

- O incluso la interacción de usuarios [28].

Ilustración 6 Un ejemplo de segmentación automática.

Fuente. E Ortiz, Contribuciones a técnicas de segmentación de imágenes. Universidad Autónoma de Madrid. Escuela politécnica superior. Dpto. de Ingeniería Informática. 2009. Pág. 5. [28]

La segmentación de imágenes es un paso fundamental para la comprensión de la estructura y la identificación de los objetos en una escena. La diferenciación de las áreas de una imagen, que se corresponderán con zonas homogéneas y/o regiones importantes conocidas como objetos semánticos, es a menudo establecida como el primer paso en muchas aplicaciones basadas en objetos, como por ejemplo, la indexación y recuperación de imágenes (por ejemplo, mediante descriptores del estándar MPEG-7) o la codificación de la región con mayor o menor calidad en función de su interés (por ejemplo, mediante las funcionalidades propuestas en el estándar JPEG2000) [28].

5.1.4.1 Definición de segmentación de imágenes

El campo del Procesamiento Digital de Imágenes se refiere al estudio de técnicas que permitan de alguna manera mejorar una imagen, de modo que pueda ser utilizada en etapas posteriores como por ejemplo análisis de imágenes. Por otro lado, el análisis de imágenes emplea técnicas que extraen información de las imágenes. Obviamente, para poder extraer

información de las imágenes, estas deben tener una buena calidad, y es justamente allí donde encaja el Procesamiento Digital de Imágenes para mejorar dichas imágenes.

Desde este punto de vista, el procesamiento de imágenes involucra tareas como eliminación de ruido, mejoramiento del contraste, segmentación (y la binarización, la cual es una segmentación particular), detección de bordes, etc. y el análisis de imágenes involucra tareas como conteo de elementos, extracción de descriptores de objetos, etc [25].

En realidad, no existe un acuerdo común entre los autores sobre los límites del procesamiento de imágenes y otras áreas relacionadas, tales como análisis de imágenes y visión computacional. Algunas veces se hace la distinción del procesamiento de imágenes como una disciplina en la cual tanto la entrada como la salida de un proceso son imágenes. Existen otros campos tales como la visión computacional cuya meta es usar el computador para emular la visión humana, incluyendo aprendizaje y la capacidad de hacer inferencias y tomar acciones basadas en entradas visuales. Es notorio que este campo es un subcampo de la Inteligencia Artificial, el cual intenta emular la inteligencia humana [25].

Sin embargo, un paradigma muy útil es considerar tres tipos de procesos: procesos de bajo, medio y alto nivel. Los procesos de bajo nivel involucran operaciones primitivas tales como reprocesamiento de imágenes para reducir ruido, mejorar el contraste y hacer más pronunciadas las imágenes. Un proceso de bajo nivel es caracterizado porque tanto la entrada y salidas son imágenes. Procesamiento de medio nivel sobre imágenes involucra tareas como segmentación (partir una imagen en regiones u objetos), la descripción de esos objetos para reducirlos a una forma manejable por computadora, y clasificación (o reconocimiento) de objetos individuales. Un proceso de medio nivel es caracterizado por el hecho de que las entradas generalmente son imágenes, pero las salidas son atributos extraídos de esas imágenes (por ejemplo: la identidad de un objeto). Finalmente, los procesos de alto nivel involucran realizar funciones cognitivas normalmente asociadas con visión [25].

El término de segmentación (a menudo referido como agrupación o partición), abarca una amplia serie de procesos a través de los cuales se obtiene la división de la imagen en distintas regiones espaciales disjuntas atendiendo a una cierta homogeneidad de éstas. Dicha homogeneidad se establece según una serie de criterios tales como [25]:

- Características de bajo nivel (por ejemplo, el color o la textura).
- Conocimiento previo sobre la escena o imagen (por ejemplo, reconocimiento de campo).
- Conocimientos de alto nivel (estos son los denominados, modelos semánticos)
- O incluso la interacción de usuarios.

Una segmentación perfecta puede ser concebida como la asignación de todos los pixeles al objeto correcto. Obviamente esta es una tarea muy complicada debido a que para realizarla, muchas veces es necesario contar con información a priori de los objetos, además de la información local. Después de realizada una segmentación, se conocen las regiones y las discontinuidades entre regiones. Luego esas regiones son empleadas para extraer información relevante sobre los objetos contenidos en la imagen [25].

5.1.4.2 Representación de Imágenes

La información contenida en las imágenes puede ser representada de diferentes maneras. Nosotros veremos la representación espacial, dejando de lado representaciones también muy útiles como la representación en número de ondas (se obtiene al aplicar la Transformada de Fourier a una representación espacial).

Una imagen constituye una distribución espacial de la irradiación en un plano. Matemáticamente hablando, la distribución de irradiación espacial puede ser descrita como una función continua de dos variables espaciales. De manera general, la imagen se define como:

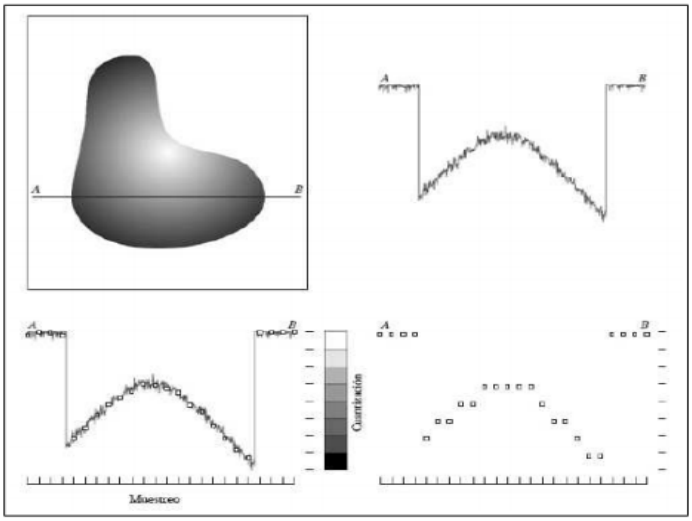
$f: \Omega \rightarrow C$, Donde $\Omega \subset R^2$ y C es llamado el espacio característico. El Espacio Característico puede ser:

- Un intervalo, por ejemplo [0, 255] o [0, ∞], para imágenes en escala de grises.
- Un subconjunto de R^2, por ejemplo $[0, 1]^2$. Para imágenes en escala de grises.

Para imágenes a color, la función *f* puede ser vista como una superficie tridimensional.

Obviamente las computadoras no pueden manejar imágenes continuas, sino solo números o arreglos de ellos. Es por eso que se necesita representar las imágenes como arreglos bidimensionales de puntos. El proceso de convertir una imagen continua en una digital se conoce como: muestreo y cuantización.

Ilustración 7 Proceso de muestreo y cuantización

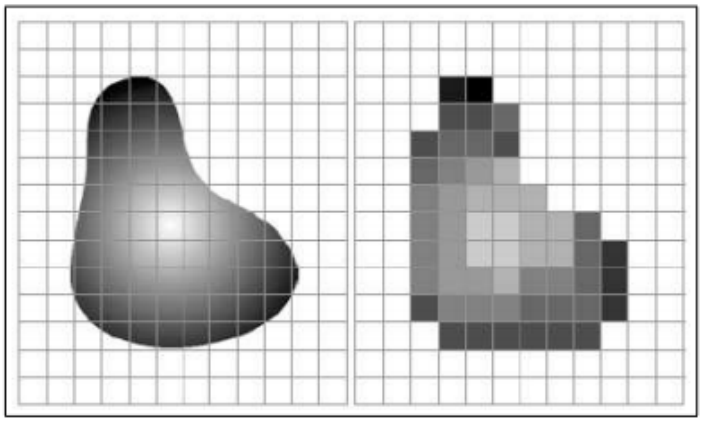


Fuente A. J. Alfaro; I. A. Sipirán, Diseño de un Algoritmo de Segmentación de Imágenes aplicando el Funcional de Mumford-Shah para mejorar el desempeño de los Algoritmos Clásicos de Segmentación [25]

El muestreo se refiere a digitalizar los valores de las coordenadas, y la cuantización a digitalizar los valores de la amplitud, respectivamente. En la ilustración 7, se observa una imagen continua, la cual ha sido cortada a lo largo del segmento AB, y se ha extraído una función unidimensional (arriba a la derecha). Luego se realiza el proceso de muestreo (discretización de las coordenadas) y cuantización (digitalizar las amplitudes que definirán

los tonos de gris, en el caso de imágenes en escala de grises). En la ilustración 8, se observa como resulta una imagen digital después de aplicar el proceso de muestreo y cuantización. [25]

Ilustración 8A la izquierda imagen continua. A la derecha resultado de la imagen después del proceso de muestreo y cuantización.



Fuente A. J. Alfaro; I. A. Sipirán, Diseño de un Algoritmo de Segmentación de Imágenes aplicando el Funcional de Mumford-Shah para mejorar el desempeño de los Algoritmos Clásicos de Segmentación [25]

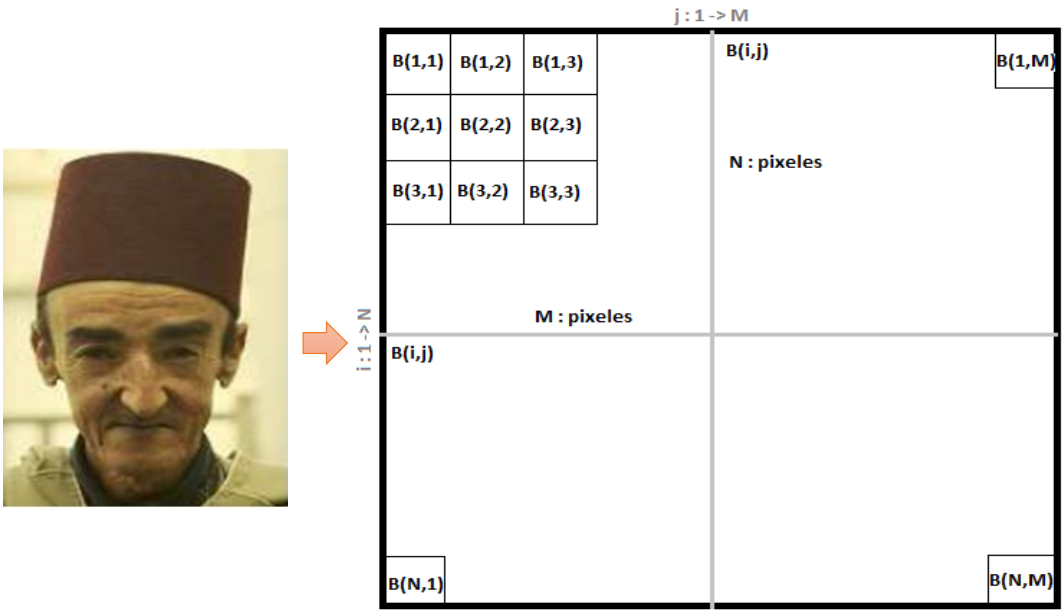
$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \dots & \dots & & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix} \tag{12}$$

Es claro que la imagen digital es una aproximación de la imagen continua. Esta aproximación será mejor dependiendo de la cantidad de unidades de muestreo y cuantización que se tomaron en cuenta. Se puede observar que las unidades de muestreo definen la resolución de una imagen y las unidades de cuantización definen el tamaño del espacio característico. A cada punto sobre la malla bidimensional resultante se le conoce como pixel (abreviatura del término en inglés picure element). A toda la malla de pixeles se le puede tratar como una matriz numérica de N filas y M columnas (ecuación 28). Evidentemente los índices de la matriz dependen del ambiente de programación en el que se trabaje, pero eso escapa al

propósito de este marco teórico, por lo convenimos en usar la notación de la Segmentación de Imágenes [25].

Así una imagen digital consiste en sí, en un arreglo matricial de puntos o pixeles. En el caso de una imagen monocromática (escala de grises) la imagen digital estará conformada por una sola matriz y en el caso de una imagen a color la imagen digital estará compuesta por la superposición de matrices, cada una con un componente diferente [29]. En el caso de las imágenes a color convencionales están conformadas por tres matrices de cada una relacionada con los colores del espectro visible: rojo *(R = RED),* verde *(G = GREEN)* y azul *(B = BLUE).* Para las imágenes de satélite se pueden tener más de tres matrices, dependiendo la resolución espectral del sensor con que fueron adquiridas se pueden contar con matrices que contengan la reflectancia por ejemplo en el espectro del infrarrojo.

Ilustración 9. Estructura de una imagen digital.



Fuente Adaptado de: N. Bahamón Cortes, Restauración de imágenes mediante un modelo matemático basado en las técnicas de detección de bordes y propagación de texturas [29]

En la ilustración 9 se muestra la composición de una imagen digital por pixeles, en resumen una imagen digital es una función bidimensional *B(x,y)* dada por sus coordenadas espaciales

y su intensidad, Así los índices de las filas y las columnas indican un punto (pixel) específico de la imagen y cada pixel en la matriz tiene un determinado valor de intensidad o brillo . Por lo general, para hacer referencia al tamaño de la imagen digital se dice que es de *MxN* pixeles, es decir, *M* denota el número de columnas y *N* el de filas [29].

En la siguiente imagen se ilustra la superposición de matrices o bandas en el espacio de color RGB para dar lugar a una imagen digital en color.

Ilustración 10 Composición de una imagen digital por bandas RGB

(a) (b) (c) (d)

(a). Banda correspondiente al Rojo; (b). Banda correspondiente al Verde; (c). Banda correspondiente al Azul; (d). Composición de la imagen digital por la superposición de las bandas RGB.

Fuente Adaptado de: N. Bahamón Cortes, Restauración de imágenes mediante un modelo matemático basado en las técnicas de detección de bordes y propagación de texturas [29]

5.1.5 ALGORITMO DE BÚSQUEDA GRAVITACIONAL GSA

Existe un nuevo algoritmo de búsqueda llamado Algoritmo de Búsqueda Gravitacional – GSA, que parece ser mejor en su funcionamiento, dicho algoritmo esta "basado en el comportamiento de enjambre de una población de agentes y la ley de gravitación" [1] y desde

[1] Ibid. Pág. 2233.

su publicación en 2009 se han generado múltiples investigaciones que han demostrado su superioridad en varios campos de aplicación. En las siguientes secciones, se presentará una breve descripción de la fuerza de gravitación y demás fundamentos teóricos de este algoritmo para proporcionar un fondo adecuado, finalmente se procederá la explicación del algoritmo:

5.1.5.1 La ley de Gravitación Universal

La ley de gravitación universal es la ley que enuncia una relación cuantitativa de la interacción gravitatoria entre distintos cuerpos con masa, fue publicada en 1687 por Isaac Newton en su libro *Philosophiae Naturalis Principia Mathematica*. La ley de gravitación universal define que la fuerza con que se atraen dos cuerpos de diferente masa únicamente depende del valor de sus masas y del cuadrado de la distancia que las separa y que dicha fuerza actúa de tal forma que es como si toda la masa de cada uno de los cuerpos estuviese concentrada únicamente en su centro. Es decir, cuanto más masivos sean los cuerpos y más cercanos se encuentren, con mayor fuerza se atraerán.

La fuerza de Gravitación Universal se calcula como:

$$F(t) = \frac{G m_x m_y}{d(x(t),y(t))^2} \tag{13}$$

Dónde:

- m_x: Es la masa del primer cuerpo
- m_y: Es la masa del segundo objeto
- d: es la distancia entre los cuerpos
- $F(t)$: es el módulo de la fuerza ejercida entre ambos cuerpos, y su dirección se encuentra en el eje que une ambos cuerpos.

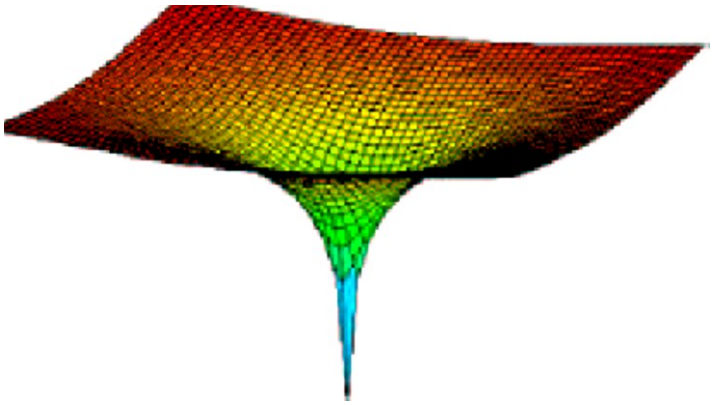
- G: Es la constante de la Gravitación Universal, igual a:

$$G = (6.67428 \pm 0.00067) \times 10^{-11} Nm^2 kg^{-2} \quad (14)$$

m_x y m_y son las masas de dos objetos d es la distancia euclidiana entre los dos objetos [30].

La fuerza gravitacional ejercida por un objeto sobre otros objetos define un campo gravitatorio alrededor del objeto. Como se muestra en la siguiente figura.

Ilustración 11 Campo de gravitación.



Fuente J. Gómez; D. Dasgupta; O. Nasraoui, New Algorithm for Gravitational Clustering [31]

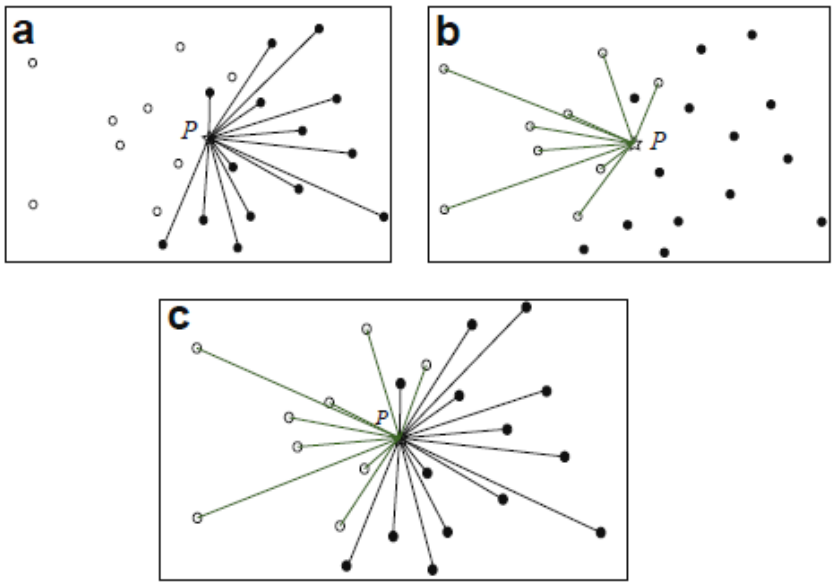
A. Principio de agrupación basado en el campo de gravitación

Suponiendo que se tienen dos clases c_1 y c_2 en un conjunto de datos de entrenamiento[2] para un dato de entrenamiento P (partícula atómica de datos), la gravitación que las partículas de datos en c_1 ejercen sobre P es F_1 (entiéndase F_1 como la sumatoria de las fuerzas de gravedad ejercidas sobre P por las partículas pertenecientes a c_1, fuerzas calculadas a partir de la Ec. 13) y F_2 es la gravitación que las partículas de datos en c_2 ejercen sobre P, si $F_1 > F_2$ entonces

[2] Datos de entrenamiento: Subconjunto de datos utilizados para el entrenamiento del modelo, para algoritmos supervisados, este subconjunto debe poseer etiquetas con la clase a la que pertenece cada objeto.

el grado de pertenencia de *P* a c_1 es más fuerte que a c_2 [4]. La fuerza de gravitación determina así a que clase pertenece el dato de entrenamiento.

Ilustración 12 Agrupación Basada en la ley de gravitación Universal.



Fuente , L. Peng; B. Yang; Y. Chen; A. Abraham, Data gravitation based classification *[32]*

La agrupación es una tarea fundamental de aprendizaje que toma un conjunto de datos y los clasifica en diferentes grupos, esto en base a la similitud calculada entre ellos de tal manera que en un mismo grupo se encuentran los objetos (registro de datos) más similares entre sí, en la literatura pueden encontrarse diferentes propuestas para la realización de esta área dentro de las que se encuentra el uso de la ley de gravitación universal en los que la similitud es simulada por la atracción que cada objeto ejerce en otro (entre mayor sea la similitud es mayor la fuerza ejercida de un objeto sobre el otro).

5.1.5.2 Algoritmos Metaheurísticos

Metaheurístico deriva del verbo griego "heuristikein" que significa "encontrar" y el prefijo "meta" que significa "en alto nivel". En términos generales un algoritmo metaheurístico puede ser visto como una estrategia de alto nivel que desarrolla búsquedas aleatorias dirigidas dentro posibles soluciones para la mejor solución (cercana a la óptima) de un problema,

originalmente fueron definidos como métodos que orquestan una interacción entre procedimientos de mejora locales y estrategias de nivel superior para crear un proceso capaz de escapar de los óptimos locales y realizar una búsqueda robusta de un espacio de soluciones y posteriormente incluyeron procedimientos que emplean estrategias para superar la trampa de los óptimos locales en espacios complejos de soluciones [33], se caracterizan principalmente por [34]:

1) Son estrategias que "guían" el proceso de búsqueda.
2) Los conceptos básicos de un algoritmo metaheurístico se puede describir en un nivel abstracto no atado a un problema específico.
3) Los algoritmos más avanzados utilizan la experiencia de búsqueda (simulan algún tipo de memoria) para guiar la búsqueda.
4) Incorporan mecanismos para evitar caer en óptimos locales.

Los algoritmos metaheurísticos puede clasificarse considerando diferentes aspectos, por ejemplo: basados o no basados en la naturaleza, con memoria o sin memoria, con función objetiva estática o dinámica, sin embargo la clasificación más apropiada es aquella que considera la manipulación en cada iteración de un solo punto del espacio de búsqueda "trayectoria" o de un conjunto "población", El término "trayectoria" es utilizado ya que la búsqueda genera una trayectoria en el espacio de búsqueda, en otras palabras la búsqueda parte de un punto y mediante la exploración del vecindario va variando la solución actual formando así una trayectoria, generalmente surgen a partir de la mejora de métodos de búsqueda local al incorporarse técnicas que les permitan escapar de óptimos locales, los algoritmo basados en "trayectoria" incorporan criterios de terminación como un número máximo de iteraciones, identificación de un estancamiento o al hallarse una solución lo suficiente aceptable.

Los algoritmos basados en "población" trabajan paralelamente con un conjunto de puntos (soluciones) en cada iteración perdiendo así una manera natural e intrínseca de explorar el espacio de búsqueda [34], por ejemplo, los algoritmos inspirados en el comportamiento de enjambres utilizan una colección de agentes (soluciones) similar a una bandada natural de

aves o peces, donde cada miembro ejecuta una serie de operaciones particulares y comparte su información con los otros, estas operaciones son generalmente simples, sin embargo, su efecto colectivo, conocido como inteligencia de enjambre, produce un resultado sorprendente. Las interacciones locales entre agentes proporcionan un resultado global que permiten al sistema resolver el problema sin utilizar ningún controlador central. En este caso, las operaciones de los miembros, incluyendo la búsqueda al azar, la retroalimentación positiva, la retroalimentación negativa y múltiples interacciones, conducen a una situación de auto-organización [35].

Se puede reconocer dos tareas comunes en los algoritmos metaheurísticos basados en la población: la *exploración* y *explotación*. La *exploración* es la capacidad de sondear el espacio de búsqueda y la *explotación* es la capacidad de encontrar el óptimo alrededor de una buena solución. En las primeras iteraciones un algoritmo de búsqueda metaheurístico explora el espacio de búsqueda para encontrar nuevas soluciones esto le evita caer en un óptimo local, otra razón de la importancia de esta tarea, con el paso de las iteraciones, la *exploración* se desvanece y se pasa a la *explotación*, así el algoritmo va afinándose en puntos semi-óptimos. La clave esencial para tener una búsqueda de alto rendimiento es un adecuado equilibrio entre *exploración* y *explotación*, por un lado para identificar rápidamente regiones en el espacio de búsqueda con soluciones de alta calidad y por el otro para no perder demasiado tiempo en las regiones del espacio de búsqueda que ya se exploran o que no proporcionan soluciones de alta calidad. Todos los algoritmos metaheurísticos basados en la población emplean la *exploración* y la *explotación* pero utilizando diferentes enfoques y operadores.

Por otro lado los agentes de un algoritmo de búsqueda basado en población, pasan por tres pasos en cada iteración para realizar la *exploración* y *explotación*: auto-adaptación, cooperación y competición, en el paso de auto-adaptación cada miembro (agente) mejora su desempeño, en el paso de cooperación, los miembros colaboran con cada otro por transferencia de información y finalmente en el paso de competición, los miembros compiten

por supervivir. Lo anterior nos lleva a concluir que todos los algoritmos metaheurísticos de búsqueda tienen un marco común.

Al revisar la literatura puede evidenciarse que por lo general los algoritmos metaheurísticos son inspirados en la naturaleza e imitan procesos físicos o biológicos, entre los más populares encontramos: el algoritmo de Optimización por Enjambre de Partículas (Particle Swarm Optimization - PSO), que simula el comportamiento de una bandada de aves; algoritmo Genético (Genetic Algorithm - GA), inspirado en la teoría de la evolución de Darwin; el algoritmo de Simulación de Cocción (Simulated Annealing - SA), inspirado en los efectos de la termodinámica; algoritmo de colonia de hormigas (Ant Colony Optimization - ACO), que simula el comportamiento de una colonia de hormigas en busca de comida [35].

5.1.5.3 Descripción Algoritmo de búsqueda gravitacional - GSA

En esta sección, se describe el algoritmo de optimización basado en la ley de la gravedad llamado Algoritmo de Búsqueda Gravitacional - GSA. En este algoritmo, los agentes se consideran como objetos y su rendimiento se mide por sus masas. Todos los objetos se atraen entre sí por la fuerza de la gravedad, y esta fuerza provoca un movimiento global de todos los objetos hacia los objetos con masas más pesadas, por lo tanto las masas cooperan usando una comunicación directa a través de la fuerza de gravedad, las masas más pesadas que corresponden a las mejores soluciones, se mueven más lentamente que las masas ligeras, esto garantiza la etapa de *explotación* en el algoritmo.

En el algoritmo GSA, cada masa (agente) tiene cuatro especificaciones: posición, masa de inercia, masa gravitacional activa y masa gravitacional pasiva. La posición de la masa corresponde a una solución del problema y las masas de inercia y gravitacional son determinadas utilizando una **función fitness *(función de adecuación)***, en otras palabras, cada masa representa una solución y el algoritmo navega ajustando correctamente las masas gravitacionales y de inercia, con el transcurso del tiempo, se espera que las masas sean

atraídas por las más pesadas, estas masas representan una solución óptima en el espacio de búsqueda.

La GSA podría ser considerada como un sistema aislado de masas. Es como un pequeño mundo artificial de masas que obedecen la ley de la gravedad y del movimiento de Newton. Más precisamente, las masas obedecen a las siguientes leyes [35]:

- **Ley de la gravedad:** Cada partícula atrae toda otra partícula y la fuerza gravitacional entre dos partículas es directamente proporcional al producto de sus masas e inversamente proporcional a la distancia entre ellos.

- **Ley de movimiento:** la velocidad actual de cualquier masa es igual a la suma de la fracción de su velocidad anterior y la variación en la velocidad. La variación en la velocidad o aceleración de cualquier masa es igual a la fuerza actuado en el sistema de dividido por la masa de inercia.

5.1.5.4 Funcionamiento del Algoritmo de búsqueda gravitacional - GSA

Se explicara a continuación de manera detallada el modelo matemático seguido por el algoritmo GSA:

Considerando, un sistema con N agentes (masas). Se define la posición del agente *i-ésimo* por:

$$X_i = \left(x_i^1, \dots, x_i^d, \dots x_i^n\right) para\ i = 1,2, \dots, N \tag{15}$$

Donde x^d_i presenta la posición del agente *i–esimo* en la dimensión *d-esima*.

En un momento determinado *'t'*, la fuerza que actúa sobre la masa *'i'* a partir de la masa *'j'* es la siguiente:

$$F_{ij}^{d}(t) = G(t)\frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon}(x_j^d(t) - x_i^d(t))$$

(16)

Donde M_{aj} es la masa gravitacional activa relacionada con agente *j*, M_{pi} es la masa gravitatoria pasiva relacionada con el agente *i*, *G(t)* es constante gravitacional en el tiempo *t,* ε es una constante pequeña y R_{ij} (t) es la distancia euclidiana entre los agentes *i* y *j*:

$$R_{ij}(t) = \left\|X_1(t), X_j(t)\right\|_2$$

(17)

Para dar características estocásticas al algoritmo, se supone que la fuerza total que actúa sobre el agente *i* en una dimensión *d* corresponde a una suma ponderada al azar de los componentes de los *d-ésimos* componentes de las fuerzas ejercidas por los otros agentes:

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} rand_j F_{ij}^d(t)$$

(18)

Donde $rand_j$ es un número aleatorio entre el intervalo [0, 1].

Por lo tanto, por la ley del movimiento, la aceleración del agente *i* en el momento *t*, y en la dirección *d-esima*, $a^d{}_i\ (t)$, corresponde a:

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)}$$

(19)

Donde M_{ii} es la masa inercial del agente *i.* Además, la siguiente velocidad de un agente se considera como una fracción de su velocidad actual más su aceleración. Por lo tanto, su posición y su velocidad corresponde a:

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t)$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1)$$

(20)

Donde $rand_i$ es una variable aleatoria entre el intervalo [0, 1]. Utilizando este número aleatorio se agrega una característica al azar a la búsqueda [35].

La constante gravitacional, *G*, se inicializa al principio y se reduce con el tiempo para controlar la precisión de la búsqueda. En otras palabras, *G* es una función del valor inicial (G_0) y el tiempo (*t*):

$$G\,(t) = G\,(G_0, t)$$

(21)

Las masas gravitatorias y de inercia son simplemente calculadas por la evaluación de la función fitness (función de adecuación). Una masa más pesada significa un agente más eficiente. Esto que decir que los mejores agentes tienen mayores atracciones y se mueven más lentamente. Suponiendo la igualdad de las masas gravitacionales y de inercia, los valores de masas se calculan utilizando el mapa de adecuación. De esta manera las ecuaciones para las masas gravitacionales y de inercial son las siguientes:

$$M_{ai} = M_{pi} = M_{ii} = M_i\,, \qquad i = 1,2,\ldots,N,$$

$$M_i(t) = \frac{fit_i(t) - worst\,(t)}{best(t) - worst\,(t)}$$

$$M_i\,(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_i(t)}$$

(22)

Donde $fit_i\,(t)$ representa el valor de la adecuación del agente *i* en el momento *t*, y *worst(t)* y *best(t)* se definen de la siguiente manera (para problemas de minimización):

$$best(t) = \min_{j \in \{1, \ldots, N\}} fit_j(t)$$

$$worst(t) = \max_{j \in \{1, \ldots, N\}} fit_j(t)$$

(23)

Una forma de realizar un buen compromiso entre la *exploración* y *explotación* es reducir el número de agentes con el transcurso del tiempo en la ecuación (18). Por lo tanto, se puede usar sólo el conjunto de agentes con mayor masa. Sin embargo, se debe tener cuidado al utilizar esta política, ya que puede reducir la potencia de la *exploración* y aumentar la capacidad de la *explotación*.

Es importante resaltar que con el fin de evitar la captura en un **óptimo local** el algoritmo debe utilizar el principio de *exploración*. En el transcurso de las iteraciones, la exploración debe desaparecer y explotación debe aparecer gradualmente. Para mejorar el rendimiento de GSA controlando la *exploración* y *explotación* sólo los agentes *Kbest* atraerán a los demás. *Kbest* varía en función del tiempo, el valor inicial K_0 va disminuyendo con el tiempo iteración tras iteración. De esta manera, al principio, a todos los agentes ejercen fuerza, y como el tiempo, *Kbest* disminuye linealmente y al final habrá un solo agente que ejerza a la fuerza a los demás. Por lo tanto, la ecuación. (18) podría ser modificado como:

$$F_i^d(t) = \sum_{j \in Kbest, j \neq i} rand_j F_{ij}^d(t)$$

(24)

Donde *Kbest* es el conjunto de los primeros *K* agentes con el mejor valor de fitness y mayor masa. Los diferentes pasos del algoritmo propuesto son los siguientes:

a) Identificación del espacio de búsqueda.
b) Inicialización aleatoria.
c) Evaluación de la aptitud de los agentes (función fitness).
d) Actualización de *G(t)*, *best(t)*, *worst(t)* y $M_i(t)$ para $i = 1,2 ,..., N.$
e) Cálculo de la fuerza total en diferentes direcciones.

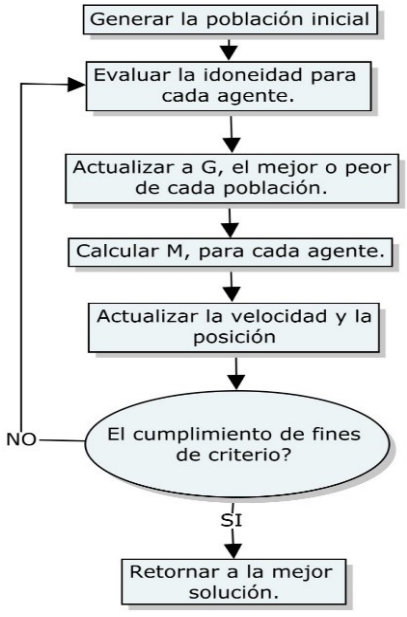
f) Cálculo de la aceleración y la velocidad.
g) Actualización de la posición de los agentes.
h) Repetir pasos c a g hasta los criterios de parada sea alcanzado.
i) Fin.

5.1.5.5 Principios del Algoritmo GSA

Para resaltar cómo el algoritmo es eficiente algunos comentarios son señalados [35]:

- Puesto que cada agente puede observar el desempeño de los otros, la fuerza gravitatoria es una herramienta de transferencia de información.
- Debido a la fuerza que actúa sobre un agente por sus agentes vecinos, puede ver el espacio alrededor de sí mismo.
- Una masa pesada tiene una gran radio de atracción efectivo y por lo tanto, una gran intensidad de atracción. De esta manera, los agentes con un mayor rendimiento tienen una masa gravitatoria mayor. Como resultado, los agentes tienden a moverse hacia el mejor agente.

Ilustración 13 principio de GSA



Fuente. E. Rashedi; H. Nezamabadi-pour; S. Saryazdi, GSA: A Gravitational Search Algorithm. [35]

- La masa de inercia está en contra del movimiento y hace que el movimiento de masas sea lento. Por lo tanto, los agentes con masa de inercia pesada se mueven lentamente y por lo tanto, buscan el espacio más localmente. Por lo tanto, puede ser considerado como una tasa de aprendizaje adaptativo.

- Constante gravitacional ajusta la precisión de la búsqueda, por lo que disminuye con el tiempo (similar a la temperatura en un algoritmo de simulación de cocción).
- GSA es un algoritmo sin memoria. Sin embargo, funciona de manera eficiente como los algoritmos con memoria (algoritmos metaheurísticos que utilizan estructuras de memoria). Los resultados experimentales han demostrado la buena tasa de convergencia del GSA.

- Se asumió que las masas gravitatorias y de inercia son los mismos. Sin embargo, para algunas aplicaciones se pueden utilizar valores diferentes para ellas. Una masa de inercia más grande proporciona un movimiento más lento de los agentes en el espacio de búsqueda y por lo tanto una búsqueda más precisa. Por el contrario, una masa gravitacional más grande provoca una mayor atracción de los agentes. Esto permite una convergencia más rápida [35].

5.1.5.6 Algoritmo de Búsqueda Gravitacional para la agrupación de Datos - GGSA

El algoritmo de Búsqueda Gravitacional para la agrupación de Datos *(Grouping Gravitational Search Algorithm - GGSA)*, es una adaptación del algoritmo de Búsqueda Gravitacional – GSA, para resolver el problema de agrupación de datos, los cambios realizados para lograr el anterior objetivo fueron principalmente dos, el primero consistió en la integración de un esquema de codificación necesario para posibilitar al algoritmo trabajar con las posibles soluciones de un problema de agrupación y el segundo consistió en la modificación algunas de las ecuaciones del GSA para adoptarlo al mencionado esquema de codificación [13].

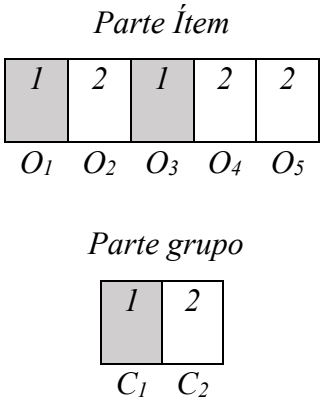
A. **Esquema de Codificación**

Como se mencionó el algoritmo GSA tuvo que ser modificado para poder trabajar con las soluciones de un problema de agrupación, para una metaheurística iterativa es necesario un esquema de representación que permita codificar la solución, la representación de una solución debe ser adecuada y relevante para adelantar un problema de optimización y fácilmente manipulable por los operadores de búsqueda de tal manera que la complejidad de estos operadores sea reducida.

La representación usada por el algoritmo GGSA es la representación de agrupamiento que es construida por dos partes: *parte ítem y parte grupo,* la *parte ítem* consiste en un arreglo de tamaño *n* (n es el número de objetos). La *parte grupo* consiste en una permutación de *D* etiquetas de grupos, Cada miembro en la *parte ítem* puede tener cualquiera de la etiquetas de grupo, indicando que el objeto pertenece al grupo indicado por la etiqueta, en la siguiente ilustración se muestra la codificación utilizada por el algoritmo GGSA para una solución C = $\{C_1\{ O_1, O_3\}, C_2\{ O_2, O_4, O_5\}\}$ de un problema de datos con $O_= \{ O_1, O_2 O_3, O_4, O_5\}$.

Cuando se optimiza una función por el algoritmo GSA, por lo general cada solución es representada por un vector de longitud *D* de números reales (*D* es la dimensión del campo de búsqueda), donde cada valor corresponde a una variable, de manera similar cuando se resuelve un problema de agrupación con el algoritmo GGSA, se puede representar una solución compuesta de *D* grupos como una estructura cuyo largo es igual al número de grupos, en otras palabras los grupos juegan un rol de variables en el estándar GSA, de manera similar la posición de cada objeto en la *d-ésima* dimensión representa el valor en la *d-ésima* variable, en el algoritmo GGSA este determina los datos que están contenidos en el *d-ésimo* grupo [13].

Ilustración 14 Codificación de una solución candidata de dos grupos para un problema de agrupación de cinco objetos de datos



Fuente Adaptado de: D. Mohammad; N. Hossein, A Grouping Gravitational Search Algorithm for data clustering [13]

B. Definición de la función objetivo (fitness)

Como es sabido el objetivo básico de un algoritmo de optimización es minimizar o maximizar una función objetivo eligiendo sistemáticamente valores tomados desde un espacio de búsqueda permitido. Por otra parte en la sección referente a la descripción del algoritmo GSA, se mencionó que este algoritmo realiza la optimización considerando las soluciones como agentes que interactúan a partir de la fuerza de gravitación universal que depende de sus masas, y que las masas son calculadas a partir una función fitness (función de adecuación), de tal manera que los agentes (soluciones) con masas más pesadas son mejores soluciones.

Para el caso del problema de agrupación, la función objetivo debe medir la calidad del resultado de la agrupación, la función más popular para esto es el *error medio cuadrático* que consideran la cohesión de los grupos en orden a evaluar la calidad de una partición dada:

$$f(O,C) = \sum_{i=1}^{D} \sum_{O_j \in C_i} \|O_j - Z_i\|^2 \quad (25)$$

Donde D es el número de grupos y $\left\|O_j - Z_i\right\|^2$ es la *distancia euclidiana* entre un objeto de datos $O_j \in C_i$ y el centro del grupo *i*, representado por el símbolo Z_i, que puede ser calculado a partir de la siguiente ecuación:

$$Z_i = \frac{1}{|C_i|} \sum_{k \in C_i} O_k \tag{26}$$

Donde $|C_i|$ es la cardinalidad del grupo C_i, es decir el número de objetos que posee el grupo *i.*

En los problemas de agrupación el objetivo es hallar los centroides de cada grupo por medio de la *minimización* de la función objetivo que puede ser la suma de las distancias entre cada objeto y el centro del grupo al que está asignado expresada por el *error medio cuadrático* calculado por la ecuación 19.

C. Mediada de Distancia

En el algoritmo GSA estándar, la medida de distancia utilizada calcula la distancia lineal entre dos escalares, esta distancia debió ser modificada para adaptar el algoritmo a los problemas de clasificación y así poder calcular la distancia entre dos grupos, Existen varias alternativas de entre las cuales fue seleccionado coeficiente de similitud denominado *Jaccard*, que dados dos grupos C_1 *y* C_2 con cardinalidad $|C_1|$ y $|C_2|$ respectivamente, el coeficiente de similitud e *Jaccard* se a partir de la siguiente ecuación:

$$Dist_J(C_1, C_2) = Dist_j(C_2, C_1) = 1 - \frac{|C_1 \cap C_2|}{C_1 \cup C_2} \tag{27}$$

Donde $Dist_J(C_1, C_2)$ es el grado de disimilada entre los dos grupos C_1 y C_2 y permite determinar cuán lejos están, en general se tiene que el $0 \leq Dist_J(C_1, C_2) \leq 1$, donde el valor será igual a 0 si $C_1 = C_2$ e igual a 1 si $C_1 \cap C_2 = \emptyset$.

La distancia Euclidiana se adaptó también para trabajar con problemas de agrupación de datos, Dado $C=\{C_{1,\dots,}C_D\}$ y $C'=\{C'_{1,\dots,}C'_D\}$ son dos grupos (clusters) candidatos de objetos de datos, se tiene que la distancia entre C y C' es:

$$Euclidean_J(C, C') = Euclidean_J(C', C) = \sqrt{Dist_J(C_1, C'_1)^2 + \cdots + Dist_J(C_D, C'_D)^2}$$

$$= \sqrt{\sum_{i=1}^{D} Dist_J(C_i, C'_i)^2}$$

(28)

Nótese, que cada operación se realiza entre un par de grupos (cluster), para que le resultado del cálculo sea apropiado el par de grupos debe ser el par más similar, para esto se usa el concepto de Pareo por Peso Máximo Bipartido (Maximum Weight bipartite Matching - MWM) aplicado en [13] que consiste en una metodología que permite organizar pares por similitud.

D. Actualización de Ecuaciones

Después de las anteriores definiciones, se introducen las adaptaciones de las ecuaciones del algoritmo GSA para trabajar en problemas de agrupación. Inicialmente la ecuación (20), correspondiente a la actualización de la posición de los agentes, se tiene que de $x_i^d(t+1) - x_i^d(t) = v_i^d(t+1)$, sustituyendo "$(x_j^d(t) - x_i^d(t))$" por $Dist_J$, la actualización de la ecuación para el algoritmo GGSA quedaría de la siguiente manera:

$$v_i^d(t+1) = \quad rand \times v_i^d(t) + \; G(t) \sum_{j \in Kbest, j \neq i} rand_j \; \frac{M_j(t)}{Euclidean_J(X_i(t), X_j(t)) + \; \varepsilon} \times Dist_J(x_j^d(t), x_i^d(t))$$

(29)

$$Dist_J(x_i^d(t+1), x_i^d(t)) \approx v_i^d(t+1)$$

(30)

Donde, *d = 1, ... , D* es el índice del grupo (cluster), $x_i^d(t)$ y $x_j^d(t)$ son los d–ésimos grupos (clusters) de datos en las soluciones X_i^d y X_j^d, respectivamente. $Dist_J(x_j^d(t), x_i^d(t))$ corresponde al grado de disimilitud entre los d–ésimos grupos (clusters) de datos en las soluciones X_i^d y X_j^d, y $Euclidean\big(X_i(t), X_j(t)\big)$ corresponde a la distancia Euclidiana entre las soluciones X_i^d y X_j^d. En la ecuación (24) el signo ≈ "al menos igual" ya que puede ser imposible construir un nuevo grupo (cluster) $x_i^d(t+1)$ de tal manera que el valor de $Dist_J(x_i^d(t+1), x_i^d(t))$ sea exactamente igual a $v_i^d(t+1)$. Cuando el valor de $v_i^d(t+1)$ es mayor que 1 en el lado derecho de la ecuación (24), se asigna $v_i^d(t+1)\,] = 1$ (ya que $Dist_J(\ldots) \leq 1$).

Para generar la solución en el algoritmo GGSA, se utilizó una metodología similar a la utilizada por el algoritmo GPSO (*Particle Swarm Optimizer For Grouping Problems*) [13], esta metodología consta de dos fases, la primera fase es denominada **fase de herencia**, que consisten en decidir que parte de la solución $X_i(t)$ será heredada por la solución $X_i(t+1)$, durante esta fase un numero de *ítems* puede quedar vacíos en la solución $X_i(t+1)$, la segunda fase es la **fase de reinserción** donde los *ítems* vacíos son insertados en los grupos (clusters) existentes.

La construcción del nuevo cluster $x_i^d(t+1)$, durante la fase de herencia, debe ser de tal manera que el grado de disimilitud con $x_i^d(t)$) este cercano al valor de $v_i^d(t+1)$, lo que significa que el grado de disimilitud entre los dos grupos debe ser al menos igual a $1 -$

$v_i^d(t+1)$, en otras palabras se busca el número de ítems compartidos entre $x_i^d(t+1)$ y $x_i^d(t)$ es igual a $n_i^d(t+1) = |x_i^d(t+1) \cap x_i^d(t)|$ del tal manera que el valor de $Dist_J(x_i^d(t+1), x_i^d(t))$ se aproxime al valor de $v_i^d(t+1)$, de hecho los números compartidos entre $x_i^d(t+1)$ y $x_i^d(t)$ son una de las partes de la solución $X_i(t+1)$ heredados desde $X_i(t)$, a partir de la ecuación (23) el número de ítems compartidos entre $x_i^d(t+1)$ y $x_i^d(t)$ es calculado de la siguiente manera:

$$Dist_J\left(x_i^d(t+1), x_i^d(t)\right) = \frac{n_i^d(t+1)}{\left|(x_i^d(t)\right|} \approx v_i^d(t+1)$$

$$\Rightarrow n_i^d(t+1) \approx (1 - v_i^d(t+1))\left|x_i^d(t)\right| \quad (31)$$

Donde, $n_i^d(t+1)$ debe ser un entero.

Tabla 4 Pseudocódigo Algoritmo GGSA

// **Inicialización**
Generar la Inicial población;
// **Iteracion**
For i = 1 to #_iteraciones

Evaluar la función fitness para cada agente;
Actualizar G, best y worst de la población;
Calcular M;
// **Ordenamiento pares - MWM**
For d = 1 to D do
Par de cluster d of $X_i(t)$ con el cluster ms similar de $X_j(t)$
(para todo $X_j(t) \in$ Kbest) por el procedimiento de pareo MWM;

// **Fase de herencia**
Calcular el valor de $Euclidean(X_i(t), X_j(t))$ (para todo $X_j(t) \in$ Kbest) usando la Eq.(27);

For d = 1 to D do
Calcular el valor de $Dist_J(x_j^d(t), x_i^d(t))$ (para todo $X_j(t) \in$ Kbest) usando la Eq.(28);
Calcular el valor de $v_i^d(t+1)$ usando la Eq.(29);
Calcular el valor de $n_i^d(t+1)$ Eq.(31);
Seleccionar aleatoriamente $n_i^d(t+1)$ desde el cluster $X_i(t)$ y asignarlos al nuevo cluster $X_i(t+1)$;
Endfor

// **Fase de inserción**

For each objeto O_j que no fue seleccionado en la fase de herencia **do**
Asignar el objeto O_j en el cluster con el centroide mas cercano;

Output: $X_i(t+1)$

// **End Iteracion**
Endfor

Fuente generación propia.

5.2 ESTADO DEL ARTE

Según los referentes consultados en el área específica del agrupamiento gravitacional, a pesar de ser limitados; los desarrollos han tenido un gran éxito en la identificación de grupos de datos similares, a partir de la identificación de los centros de los grupos de datos.

En 1977 W.E. Wright inicio la línea de los modelos de agrupación presentando un algoritmo de agrupamiento denominado “Gravitational clustering”, presentó el algoritmo para desarrollar análisis de agrupamiento en datos euclidianos, evalúo los resultados su aplicación en varios conjuntos de datos y los comparó con los obtenidos a través de algoritmos no gravitacionales con resultados exitosos [36], el algoritmo gravitacional propuesto por Wright es un algoritmo jerárquico de aglomeración, las fuerzas gravitacionales son usadas como mecanismos para unir partículas hasta que una sola partícula continua en el sistema [36], en términos generales el algoritmo funciona de la siguiente manera:

- Para determinar la nueva posición de una única partícula, todas las partículas restantes deben ser utilizadas.
- Cuando dos partículas están suficientemente cerca estas son agrupadas, una de ellas es removida de la simulación y la masa de la otra es incrementada con la masa de la partícula removida.
- Existe una distancia máxima que cada punto se puede mover con cada interacción del algoritmo, este parámetro es fijado por el usuario.
- La simulación termina cuando solo una partícula permanece en el sistema.

Las principales investigaciones desarrolladas sobre el tema se han fundamentado sobre este algoritmo.

En 1999 S. Kundu, desarrolló un método de agrupación basado en la noción de una fuerza de atracción entre cada par de puntos, con la novedad de no utilizar una medida de "similitud", Los grupos se forman al permitir que cada punto se mueva lentamente bajo el efecto resultante de todas las fuerzas que actúan sobre él y mediante la fusión de dos puntos cuando están muy cerca uno del otro. Este modelo se consideró como un refinamiento del método del vecino más cercano y el método difuso de K-medias [37].

Existen investigaciones recientes sobre agrupamiento basado en la ley de gravitación universal, de estos se han seleccionado los más relevantes para el desarrollo de este proyecto:

1) En 2003, J. Gómez, D. Dasgupta y O. Nasraoui [36], propusieron un nuevo algoritmo de agrupación gravitacional no-supervisado que se nombra para efectos de esta propuesta como "NGCA", este método determina automáticamente el número de clases, cada objeto en la base de datos es considerado como un objeto en el espacio y los objetos son movidos utilizando la ley de gravitación universal y la segunda ley de Newton, así lograron proponer un algoritmo no-supervisado que funciona correctamente en datos con ruido, la propuesta se fundamenta en la investigación desarrollada por Wright y mejora su rapidez, robustez además de un algoritmo no-supervisado.

2) En 2009 J. Goméz, D. Dasgupta y N. Olfa [31], Propusieron un algoritmo de agrupación no-supervisada de datos, fundamentado en la ley de gravitación universal y la ley de movimiento de Newton, En esta propuesta cada registro es considerado como un objeto que se mueve en el espacio usando la fuerza de gravedad y la ley de movimiento, una estructura de almacenamiento se usa para almacenar y actualizar las agrupaciones (cluster) que están conformadas por los objetos más cercanos. El

   algoritmo demostró tolerancia al ruido y la capacidad de generar agrupaciones con datos configurados en múltiples niveles de resolución.

3) En 2011, A. Hatamlou, S. Abdullah, Z. Othman [38], desarrollaron un algoritmo que se basó en la búsqueda gravitacional y un algoritmo de búsqueda heurístico "**GSA-HS**", el algoritmo de búsqueda gravitacional es usado para encontrar una solución óptima para el problema de agrupamiento y en el siguiente paso usa un algoritmo heurístico aplicado para mejorar la solución inicial.

4) En 2012, M Sánchez, O Castillo [39], desarrollaron un algoritmo para la búsqueda de grupos, también basado en la ley de la gravitación universal de Newton "FGGC", este incluyo teoría difusa al refinar los grupos de salida, integrando la lógica difusa. En términos generales incorpora al análisis de agrupamiento la granularidad difusa que tiene como principio obtener el gránulo óptimo que representa completamente el

conocimiento del conjunto de datos. La computación granular ha ido ganando mucho interés en los últimos años y este trabajo continúa esta tendencia en el área.

Por otra parte, un grupo de investigaciones recientes se han enfocado en combinar técnicas de optimización metaheuristicas con algoritmos de agrupación para mejorar el desempeño de estos últimos, el origen de esta tendencia quedo registrado en el estudio titulado metaheuristicas para Agrupación en KDD (Metaheuristics for Clustering in KDD), desarrollado por J. Rayward-Smith en 2005. En esta investigación el autor evaluó el uso de metaheurísticas en la agrupación de datos y concluyó con el alto potencial que tienen estas para resolver este tipo de problemas [40]. De la misma manera la segmentación de imágenes es un área que ha sido abordada desde hace varios años por métodos matemáticos y estadísticos [41], [42], [43], [44], mediante técnicas de la minería de datos como K-medias y Mapas Autoorganizativos – SOM [45], [46], [47], [48], [49] y recientemente confluyen las tendencias del uso de metaheuristicas con algoritmos de agrupación para la solución de la segmentación de imágenes [50], [51], [52] [53], [54], [55] , [56].

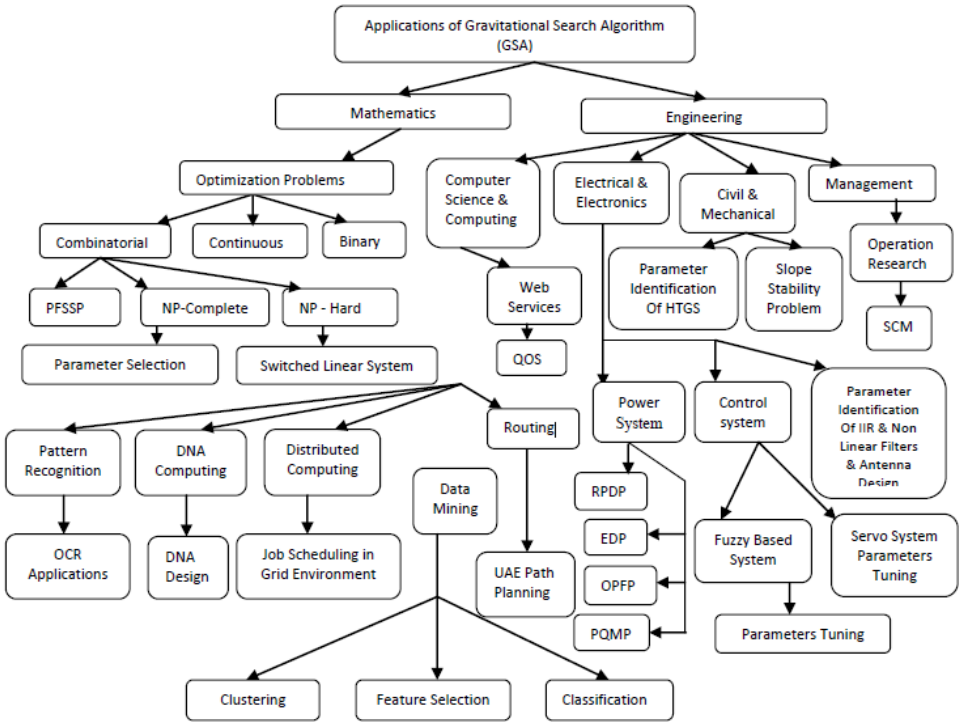
Recientemente, han sido propuestos un alto número de algoritmos inspirados en la naturaleza, por ejemplo algunas de estas propuestas se han basado en el proceso de la evolución, el comportamiento de enjambre y diferentes leyes naturales. "*Los algoritmos inspirados en la naturaleza son el último estado del arte para problemas de optimización y otros problemas que enfrentan la inflexibilidad de los métodos clásicos"*, varios investigadores han demostrado que los algoritmos inspirados en la naturaleza son convenientes para resolver problemas computacionalmente complejos como la optimización de funciones objetivo, reconocimiento de patrones, procesamiento de imágenes, agrupación, entre otros [57].

Algunas de las propuestas más populares de algoritmos basados en la naturaleza son: Optimización por Enjambre de Partículas (Particle swarm optimization -PSO), Algoritmo Genético (Genetic Algorithm - GA), Recorrido Simulado (Simulated Annealing -SA), Optimización por Colonia de Hormigas (Ant colony optimization - ACO), Optimización por Colonia Artificial de Abejas (Artificial Bee colony - ABC), Optimización por la Gran Explosión Big Bang (Optimización, Big Bang Big Crunch -BB-BC), Estos algoritmos

demostraron mejores resultados que los métodos clásicos en términos de exactitud y tiempo de convergencia computacional [57].

En 2009, fue propuesto un nuevo algoritmo metaheurístico para la optimización de problemas continuos basado en la ley de gravitación universal, fue llamado Algoritmo de Búsqueda Gravitacional *(Gravitational Search Algorithm-GSA)* [35], inicialmente fue creado para la resolver problemas de optimización en base a la ley de Gravitación Universal de Newton, que considera que cada partícula es atraída por las demás mediante la fuerza de gravitación. En este en corto tiempo este algoritmo ha ganado gran popularidad y ha sido aplicado a diferentes tipos de problemas como la agrupación y clasificación de datos.

Ilustración 15 Aplicación del Algoritmo de Búsqueda Gravitacional - GSA



Fuente Kuma, Y. ; Sahoo, G. A Review on Gravitational Search Algorithm and its Applications to Data Clustering & Classification [57]

En la Tabla 5 se encuentra un listado de las propuestas que se ha realizado a partir del algoritmo GSA para la solución de diferentes problemas:

Tabla 5 Listado de algoritmos propuestos usando GSA

Año	Autor	Algoritmo	Problema
2009	Rashedi	GSA	Optimización en espacios continuos
2010	Rashedi	BGSA	Optimización binaria
2010	Hamid Reza Hassanzadeh	MOGSA	Optimización multiobjetivo
2010	Xiangtao Li	SIGSA	Programación de flujo de trabajo
2010	A. Chatterjee	GSA	Mejora lóbulo lateral en arreglos de anillos concéntricos (Antena)
2010	Seyedali Mirjalili	PSOGSA	Para evitar la convergencia lenta debido a la característica de baja memoria de GSA
2011	Jianhua Xiao and Zhen Cheng	GSA	Problema de optimización de secuencias de ADN
2011	S. Sarafrazi	Improved GSA	Incremento de la capacidad de exploración y explotación del algoritmo
2011	M. Soleimanpour-moghadam	QGSA	Posición y velocidad de un objeto en GSA
2011	Saeid Saryazdi	GSA worked as heuristic search	Estimación de parámetros de IIR
2011	Jianhua Xiao	GCSA	Problema de selección de Partner con restricción de fecha de vencimiento
2011	Serhat Duman	GSA	Problema de despacho económico
2011	J. P. Papa	OPF-GSA	Tarea de selección de elemento
2011	M. Ghalambaz	HNNGSA	Solución de la ecuación wessinger‘s
2011	Chaoshun Li	IGSA	Identificación del parámetros para turbina hidráulica del sistema estatal
2011	Abdolreza Hatamlou	GSA	Agrupación de Datos
2011	Minghao Yin	IGSAKHM	Problema de la convergencia lenta en problemas de agrupación GSA
2011	Abdolreza Hatamlou	GSA-HS	Agrupación
2011	Soroor Sarafrazi	GSA-SVM	Mejora de la precisión en problemas de clasificación binomial
2012	Nihan Kazak	Modified GSA (MGSA).	Incremento de la tasa de búsqueda y convergencia
2012	Mohadeseh Soleimanpour	Improved QGSA	Problema de la perdida de la diversidad
2012	Lucian-Ovidiu Fedorovici	GSA with BP	Aplicación OCR
2012	Nanji H. R	multi agent based GSA	Costo computacional del algoritmo GSA original
2012	Radu-Emil Precup	Adaptive GSA	Minimizar la función Objetivo
2012	S. Duman	GSA	Problema de despacho de energía reactiva
2012	Serhat Duman	GSA	Problema de flujo de potencia óptimo
2012	Chaoshun Li	CGSA	Problema de identificación de parámetros de sistema caótico
2012	Abdolreza Hatamlou	GSA-KM	Problema Agrupación (Para mejorar capacidades de búsqueda de GSA)
2012	Chaoshun Li	GSAHCA	Identificación del modelo difuso T-S
2012	Hossein Askari	Intelligent GSA	Clasificación de datos
2012	Ahmad Asurl Ibrahim	QBGSA	Problema PQMP (Power quality monitor placement)
2012	Hamed Sadeghiet	GSA with sparse optimization	Identificación y estimación de parámetros switch linear system
2012	Mohammad Khajehzadeh	MGSA	Análisis de Estabilidad de Taludes
2012	Abbas Bahrololoum	prototype classifier based on GSA	Clasificación de datos
2012	Li Pei	GSA with PSO & DE(IGSA)	Enrutamiento (Path planning for UAE)
2013	Soroor Sarafrazi	GSA-SVM	Parámetros de SVM y aumento precisión (Clasificación)

Fuente Kuma, Y. ; Sahoo, G. A Review on Gravitational Search Algorithm and its Applications to Data Clustering & Classification [57]

Diferentes investigaciones han demostrado que el algoritmo GSA tiene un gran potencial para resolver problemas de clasificación y agrupamiento, en la siguiente tabla se encuentran una comparación de resultados obtenidos para algunas variaciones del algoritmo GSA y otras técnicas para problemas de agrupación, en esta puede observarse que el algoritmo GSA presenta mejores resultados que algunas técnicas convencionales (K-medias - KM) y otros algoritmos inspirados en la naturaleza (Optimización por enjambre de partículas - PSO). La comparación se realizó por la suma de la distancia Intra-Cluster por agrupación.

Tabla 6 Comparación de resultados agrupación algoritmo GSA, variaciones y otras técnicas

Method	Iris		Wine		Glass		CMC		Cancer	
	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.	Avg.	Std.
KM	106.05	14.63	18061.00	793.21	235.50	12.47	5893.60	47.16	3251.21	251.14
PSO	97.23	0.35	16417.47	85.49	275.71	4.55	5820.96	46.95	3050.04	110.80
GSA	**96.72**	**0.01**	**16376.61**	**31.34**	**225.70**	**3.40**	**5699.84**	**1.72**	**2973.58**	**8.17**
GSA-KM	*96.69*	**0.01**	*16294.31*	*0.04*	214.22	1.14	*5697.36*	**0.27**	*2965.21*	0.07
GSA-KMH	**96.58**	*0.002*	**16234.560**	**0**	**185.710**	*0.035*	**5685.350**	*0.310*	**2954.250**	*0.056*
GSA-HS	96.65	0.0	16292	0.0	----	----	5693.7	0.0	2964.4	0.0
PSOGSA	97.23	0.060	16331.260	0.450	*205.710*	2.090	5699.100	0.540	2969.410	0.063

Fuente Adaptado de Kuma, Y.; Sahoo, G. A Review on Gravitational Search Algorithm and its Applications to Data Clustering & Classification [57]

El algoritmo GSA tiene importantes ventajas. Inicialmente, requiere el ajuste de solo dos parámetros durante su ejecución, masa y velocidad de las partículas. Por otra parte, tiene la capacidad de encontrar una solución cercana a la **óptima global** [57]**.** La capacidad de encontrar una solución cercana a la solución óptima global hace que el algoritmo GSA se diferencie de otros algoritmos inspirados naturaleza. Sin embargo el algoritmo GSA también adolece de varios problemas como el tiempo computacional, razón por la que pueden encontrarse propuestas hibridas que buscan mejorar estos problemas [57].

En 2014, fue propuesto el algoritmo de Búsqueda Gravitacional para la agrupación de Datos *(Grouping Gravitational Search Algorithm - GGSA)*, que una adaptación del algoritmo de Búsqueda Gravitacional – GSA, para resolver el problema de agrupación de datos [13], este algoritmo presento un buen desempeño y adicionalmente cuenta con la flexibilidad para ser adaptado a problemas de segmentación de imágenes.

Aunque en la literatura se encontraron varios acercamientos de las técnicas de minería de datos en la solución de problemas relacionados con la segmentación de imágenes, no se encontró ninguna investigación que aplicara los algoritmos de agrupación basados en la ley de gravitación para este fin. Esta es una de las principales motivaciones de este proyecto ya que se aplicará de manera novedosa este tipo de algoritmos para la solución de un problema de interés para la comunidad científica.

6. MARCO EXPERIMENTAL

En el siguiente capítulo se describe el marco experimental bajo el cual se desarrolló el proceso de experimentación con los algoritmos para la segmentación de imágenes, a continuación se detalla:

- Metodología General de la investigación
- Algoritmos utilizados como referencia para realizar comparaciones.
- Conjunto de datos utilizados en la experimentación.
- Paquete de software donde se implementaron los algoritmos.
- La metodología de experimentación y los parámetros utilizados.
- Resultados numéricos de los experimentos realizados.

6.1 METODOLOGÍA GENERAL DE LA INVESTIGACIÓN

El método a aplicar en este proyecto es la investigación científica descriptiva-exploratoria con enfoque experimental. De acuerdo con el proceso formal de investigación, se utilizó un método hipotético-deductivo bajo el cual se formuló una hipótesis, que a través de un razonamiento deductivo se validó de manera empírica. Se buscó establecer, con base en la experimentación, un mecanismo de ponderación de los indicadores de evaluación de los algoritmos de forma tal que sea posible comparar dichos mecanismos.

Las etapas desarrollas siguientes etapas desarrolladas durante el desarrollo de este trabajo de investigación fueron las siguientes:

a) Análisis y selección de algoritmos basados en gravitación universal. En esta etapa el caso de estudio fue de vital importancia, y las conclusiones producto del estado del arte fueron el insumo para realizar la selección de los algoritmos.

b) Definición de dominio y contexto en el cual se realizaron las pruebas y comparación de los algoritmos.

c) Definición y aplicación de pruebas para la comparación de la efectividad de los algoritmos en la segmentación de imágenes.

d) Con base en las pruebas realizadas sobre las imágenes seleccionadas, se realizó la evaluación de cada uno de los algoritmos en cuanto a calidad de la agrupación se refiere.

6.2 SELECCIÓN DE LAS HERRAMIENTAS

La experimentación con diferentes técnicas de aprendizaje computacional es de vasto interés principalmente en el ámbito académico, debido a esto existen en el mercado varios paquetes de software que permiten la simulación de diferentes técnicas de aprendizaje computacional, sin embargo estos paquetes pueden tener un costo que pueden dificultar su acceso. Por otro lado existe la opción de iniciar una aplicación desde cero, lo que requiere de tiempo y esfuerzo que en algunos casos ya han sido realizados por proyectos que han desarrollo herramientas de software de libre acceso. Por lo tanto, lo más viable para llevar a cabo los experimentos en el entorno académico es utilizar paquetes de software que ya hayan probado su capacidad con relación al uso de las técnicas de aprendizaje computacional. Según los referentes consultados, El lenguaje y entorno para procesamiento de datos **R**, es uno de los paquetes de software que ha tenido mayor nivel aceptación en el ambiente académico, ya que posee un entorno que facilita el desarrollo de proyectos en donde se encuentran involucrados procesos de análisis de datos y su licencia tipo GNU *(General Public License).*

6.2.1 LEGUAJE DE PROGRAMACIÓN R

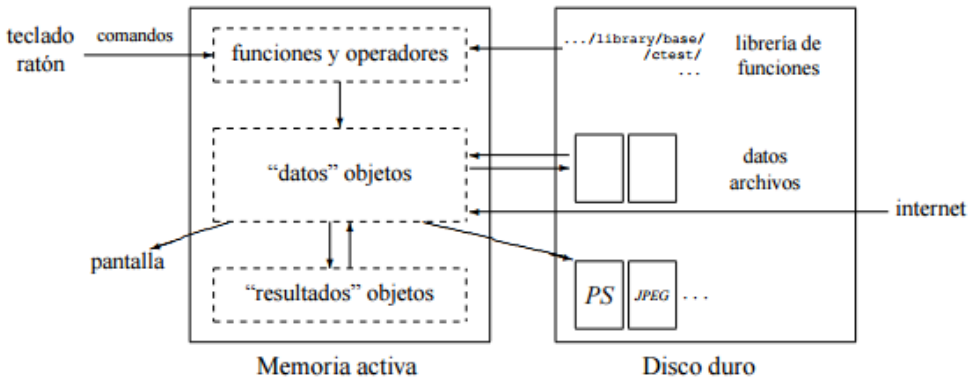
R es un sistema para análisis estadísticos y gráficos, creado por Ross Ihaka y Robert Gentleman. Tiene una naturaleza doble de programa y lenguaje de programación y se distribuye gratuitamente bajo los términos de la GNU (General Public Licence).

El lenguaje de programación R, es orientado a objetos de esta característica se deriva su simplicidad y flexibilidad. Orientado a Objetos quiere decir que las variables, datos, funciones, resultados, etc., se guardan en la memoria activa del computador en forma de objetos con un nombre específico que el usuario puede modificar o manipular con operadores (aritméticos, lógicos, y comparativos) y funciones (que a su vez son objetos).

R es un lenguaje interpretado (como Java) y no compilado (como C, C++, Fortran, Pascal), lo cual significa que los comandos escritos en el teclado son ejecutados directamente sin necesidad de construir ejecutables [58]. Contiene un amplio conjunto de herramientas estadísticas y gráficas, adicionalmente al ser un lenguaje de programación permite la creación de funcione y para algoritmos exigentes computacionalmente también es posible desarrollar bibliotecas en C, C++ o Fortran que se cargan dinámicamente.

Todas las acciones en R se realizan con objetos que son guardados en la memoria activa del computador, sin la necesidad de usar archivos temporales. La lectura y escritura de archivos solo se realiza para la entrada y salida de datos y resultados. El usuario ejecuta las funciones con la ayuda de comandos definidos. Los resultados se pueden visualizar directamente en la pantalla, guardar en un objeto o escribir directamente en el disco. Debido a que los resultados mismos son objetos, pueden ser considerados como datos y analizados como tal. Adicionalmente los archivos que contengan datos pueden ser leídos directamente desde el disco local o en un servido remoto a través de la red.

Ilustración 16 Una visión esquemática del funcionamiento de R



Fuente E, Paradis, R para Principiantes [58]

La implementación del algoritmo GGSA se realizó a partir de la definición de funciones en R. En un archivo *main* se configuran las variables de entrada y al ejecutarse invoca el conjunto de funciones creadas, para los experimentos de segmentación de imágenes se utilizaron las librerías de R "*kohonen*" y "*kmeans*". La documentación del programa del algoritmo GGSA y el listado total de librerías de R utilizados pueden ser consultados en el *Anexo 1.*

6.3 SELECCIÓN DE ALGORITMOS

El proceso de selección de algoritmos de referencia se realizó dentro del conjunto de algoritmos convencionales de minería de datos para agrupación, teniendo en cuenta la capacidad de resolver problemas relacionados con segmentación de imágenes los dos algoritmos seleccionados fueron:

- De la familia de los algoritmos basados en vecindad, se seleccionó el más popular y ampliamente implementado, algoritmo **K-medias**.

- En cuanto a los algoritmos de agrupación basados en Redes Neuronales Artificiales, se seleccionó también uno de los más populares, las **redes autoorganizativas (Self Organizing Map – SOM).**

Con relación al algoritmo de agrupación basado en la ley de agrupación universal se seleccionó al **algoritmo de Búsqueda Gravitacional para la agrupación de Datos** *(Grouping Gravitational Search Algorithm - GGSA)* una adaptación del algoritmo de optimización de búsqueda gravitacional (*Gravitational Search Algorithm* - GSA). Esta selección se realizó basada en la revisión del estado del arte ya que se encontró que este algoritmo en estudios previos ha entregado los mejores resultados en la solución de problemas similares a la segmentación de imágenes.

6.4 SELECCIÓN DE LOS PARÁMETROS POR ALGORITMO

La selección de parámetros se realizó promedio de un ejercicio piloto de que permitió afinar los diferentes parámetros por medio de la identificación de los mejores resultados, de esta manera se definió la parametrización por algoritmo de la siguiente forma:

Tabla 7 Selección de Parámetros por Algoritmo

Algoritmo	***Iteraciones***	***Parámetros***
K-medias	NA	*k= Numero de clases de la segmentación original*
SOM	100	*Topología: Hexagonal* *# Neuronas de Entrada: 3 (Número Bandas de la imagen)* *# Neuronas de Salida: Número de clases de la segmentación original*
GGSA	100	*k= Numero de clases de la segmentación original* *N= 3 (Número de soluciones)* *Alfa= 6* *G_0= 1* *Rpower= 1*

Fuente Generación Propia

6.5 SELECCIÓN DEL CONJUNTO DE DATOS

La creación del conjunto de bases para la experimentación se diseñó para considerar dos de los principales casos de aplicación de la segmentación de imágenes en la actualidad. Entre los primeros campos de aplicación de la segmentación de imágenes se encuentra el procesamiento de imágenes capturadas por sensores remotos en satélites o cámaras fotogramétricas aerotransportadas, en este proceso la imagen es segmentada según el tipo de cobertura del suelo, esto cuando no se tiene información de campo que permita realizar el

proceso de manera supervisada. Por otro lado en la actualidad con la llamada sociedad de la información, las tendencias captura y almacenamiento de información han evolucionado y ahora las imágenes gracias a su alta capacidad de representación de la realidad se han convertido para muchos casos en el formato ideal de captura de información, generándose así la necesidad de procesar un alto volumen de imágenes con el objetivo de identificar objetos, detectar bordes, contar elementos, etc. De manera sistemática y automática. Por consiguiente un segundo campo de aplicación para la segmentación de imágenes usando técnicas de aprendizaje computacional es la interpretación de imágenes convencionales (escenas).

Según lo anterior el conjunto de datos para la experimentación se construyó a partir de dos bases de datos académicas, la primera de imagen de escenas o panorámicas y la segunda de fotografías aéreas de diferentes regiones. Los detalles de cada fuente se describen son descritos a continuación.

6.5.1 BASE DE DATOS PARA LA COMPRENSIÓN DE ESCENAS DE DAGS

El grupo de investigación DAGS de la docente Daphne Koller's (*Daphne's Approximate Group of Student*) de la Universidad de Stanford se enfoca la investigación sobre el tratamiento de dominios complejos que involucran grandes cantidades de incertidumbre bajo el marco de la teoría de la probabilidad, inteligencia artificial y la informática[3].

Dentro de las contribuciones de este grupo a la comunidad científica se encuentra la publicación base de datos para la comprensión de escenas *(Scene Understanding Datasets),* esta base de datos fue introducida por Stephen Gould en 2009, para evaluar métodos para el estudio semántico y geométrico de escenas como el desarrollado por Koller y Gould en el mismo año para la detección de objetos y segmentación de regiones [59] .

[3] http://dags.stanford.edu/about.html

La base de datos está conformada por 715 imagen elegidas de bases públicas bajo los siguientes criterios de selección:

- Imágenes de escenas al aire libre.
- Tamaño aproximado de 320 x 240 píxeles.
- Presencia de al menos un objeto en primer plano.
- Horizonte presente en la imagen.

Por cada imagen existe una matriz con la clasificación semántica de cada pixel en cielo, árbol, vía, césped, agua, edificación, montaña y objeto en primer plano.

Tabla 8 Descripción archivos base de datos DAGS

Archivo	Descripción
Imagen.jpg*	Imagen digital en formato *.jpg
labels/.regions.txt*	Matriz con la clase semántica de cada pixel (cielo, árbol, vía, césped, agua, edificación, montaña, objeto en primer plano)
labels/.surfaces.tx*	Matriz con la clase geométrica de cada pixel (cielo, horizontal o vertical
labels/.layers.txt*	Matriz con las diferentes regiones de la imagen

Fuente Generación Propia

De esta base de datos fueron seleccionadas de manera aleatoria 10 imágenes para conformar el conjunto de imágenes para la experimentación. La siguiente tabla lista las características de las imágenes consideradas incluida la imagen de la segmentación original generada a partir de la matriz con la clasificación semántica de la imagen.

Tabla 9 Imágenes Seleccionadas – DAGS

#	Nombre	Ancho Pixeles	Alto Pixeles		Núm. Pixeles	Núm. Bandas	Numero de clases semánticas	Imagen
1	1001875	320	240		76800	3	3	
2	3000299	320	240		76800	3	3	
3	3000323	320	240		76800	3	4	
4	5000180	213	320		68160	3	3	
5	5000196	320	213		68160	3	4	

								
6	6000332	320	240		76800	3	4	
7	8000811	320	240		76800	3	3	
8	9000002	320	240		76800	3	4	
9	9003423	320	240		76800	3	4	

10	9004383	320	240		76800	3	4	

Fuente Generación Propia

6.5.2 BASE DE DATOS ISPRS

La Asociación Internacional de Fotogrametría y Percepción Remota – ISPRS (*International Society for Photogrammetry and Remote Sensing*), desarrollo el *Proyecto de Clasificación Urbana y Reconstrucción 3D de Edificaciones*, con el objetivo de proporcionar una base de datos, para ser utilizados por los investigadores interesados en evaluar métodos propios y algoritmos para la detección de objetos urbanos y generación de modelos 3D de edificaciones a partir de imágenes adquiridas mediante sensores remotos.

Esta base de datos está conformada por dos diferentes conjuntos de datos, el primero conformado por un juego de fotografías aéreas de la ciudad de Vaihingen en Alemania y el segundo conformado por un juego de imágenes adquiridas por un escáner laser de la ciudad de Toronto en Canadá, con miras a facilitar la ejecución de los experimentos se seleccionaron imágenes correspondientes al primer conjunto de datos ya que la resolución espacial es mucho menor con relación al segundo conjunto de datos, lo que conlleva a la manipulación de archivos de menor tamaño. Las características del conjunto de datos seleccionado son detalladas en la siguiente sección.

6.5.2.1 Conjunto de Datos - Vaihingen / Alemania

Este conjunto de datos fue originado por una prueba de captura con cámaras aéreas digitales realizada por la Asociación Alemana de Fotogrametría y Teledetección *(German Association*

of Photogrammetry and Remote Sensing - DGPF). Se compone de tres áreas de prueba localizadas en la ciudad de Vaihingen de Alemania.

Tabla 10 Descripción Áreas de Prueba ISPRS

Área de Prueba	Descripción
Área 1: "Inner City"	Área de prueba localizada en el centro de la ciudad de Vaihingen. Se caracteriza por el desarrollo denso formado por edificios históricos que tienen formas más complejas, con presencia de algunos árboles.
Área 2: "High Riser"	Esta zona se caracteriza por conjuntos de edificios residenciales de altura rodeados de árboles.
Área 3: " Residential Area"	Área residencial con presencia de casas unifamiliares pequeñas.

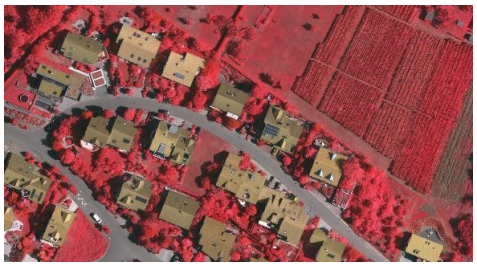
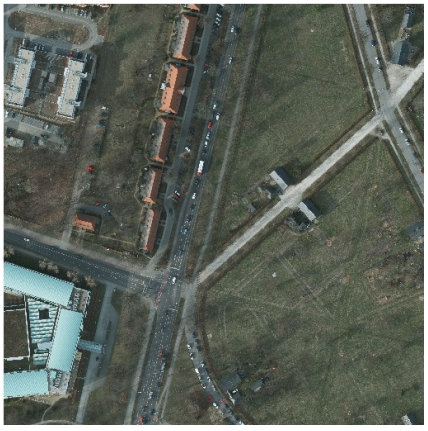
Ilustración 17 Cobertura Areas de Pureba ISPRS

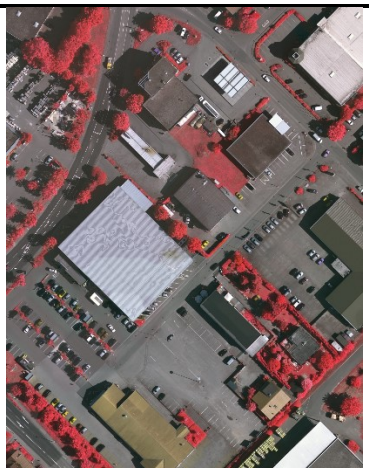
Fuente German Association of Photogrammetry and Remote Sensing

DGPF

Los detalles de las 5 imágenes seleccionadas son detallados en la siguiente tabla:

Tabla 11 Imágenes Seleccionadas ISPRS

Imagen	Detalles
	*Nombre: top_mosaic_09cm_area17* *_image.tif* *Dimensiones:* *2336 x 1281 pixeles* *Numero de Bandas:* *3 (IR – R - G)* *Numero de Clases:* *4*
	*Nombre: top_mosaic_09cm_area37* *_image.tif* *Dimensiones:* *1996 x1995 pixeles* *Numero de Bandas:* *3 (IR – R - G)* *Numero de Clases:* *4*
	*Nombre: top_potsdam_2_12_RGB.tif* *Dimensiones:* *6000 x 6000 pixeles* *Numero de Bandas:* *3 (R – G - B)* *Numero de Clases:* *4*

	*Nombre: top_mosaic_09cm_area31* *_image.tif* *Dimensiones:* *1980 x2555 pixeles* *Numero de Bandas:* *3 (IR – R - G)* *Numero de Clases:* *4*
	*Nombre: top_potsdam_2_12_RGB.tif* *Dimensiones:* *6000 x 6000 pixeles* *Numero de Bandas:* *3 (R – G - B)* *Numero de Clases:* *4*

6.6 SELECCIÓN DE LOS INDICADORES DE EVALUACIÓN DE LAS TÉCNICAS DE AGRUPAMIENTO

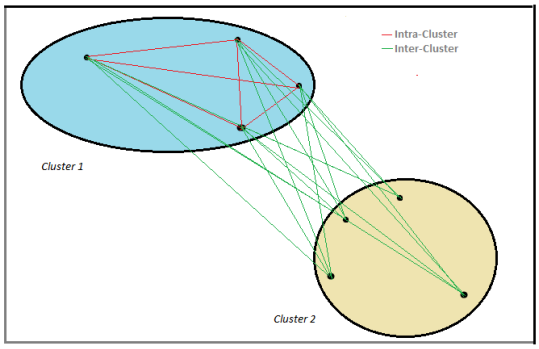
Ya que se está trabajando el problema de segmentación de imágenes que se trata de una tarea de agrupamiento no-supervisado, para la validación de resultados se requieren índices que midan la bondad de la estructura del agrupamiento (clustering) sin referirse a información externa y de esta manera en base a estos índices se puedan comparar los resultados de dos ejercicios de agrupamiento para determinar cuál de los dos es mejor.

Existe un grupo de distancias miden características relacionadas con la estructura del agrupamiento y a su vez son utilizadas para el cálculo del uno los indicadores de validación de agrupaciones más robustos conocido como el índice *Davies-Bouldin*, estas son las *distancias Inter e Intra-Cluster*.

Las distancias *Inter-Cluster*, también conocidas como de **separación**, permiten determinar qué tan distintos y separados están los grupos (clusters) con respecto a los otros, Al comparar dos agrupaciones, se puede concluir que la agrupación que posea una *mayor* magnitud en las distancias Inter-Cluster tiene una separación mejor.

Las distancias *Intra-Cluster*, también conocidas como de **cohesión**, permiten determinar qué tan cercanos están los objetos dentro de cada grupo (cluster), al comparar dos agrupaciones, se puede concluir que la agrupación que posea una *menor* magnitud en las distancias Intra-Cluster tiene una cohesión mejor.

Ilustración 18 Diagrama ilustrativo de las distancias Intra-Cluster e Inter-Cluster para la evaluación de agrupaciones no supervisadas



Fuentes Generación Propia

6.6.1 DISTANCIAS INTER-CLÚSTER

Son seis las distancias Inter-Cluster utilizadas para calcular el *Davies-Bouldin*, la distancia *Inter-Cluster Simple,* que se define como la menor distancia entre objetos dos objetos pertenecientes a diferentes grupos (clusters), la distancia *Inter-Cluster Completa,* que representa la distancia entre los objetos más lejanos pertenecientes a dos grupos diferentes, la distancia *Inter-Cluster Promedio,* que representa la distancia promedio entre todos los objetos pertenecientes a dos grupos diferentes, la distancia *Inter-Cluster Centroide,* que determina la distancia entre los centroide de dos grupos, la distancia *Inter-Cluster Promedio a Centroides,* que representa la distancia entre el centroide de un grupo a todos los objetos de diferentes grupos , y por último la *Métrica de Hausdorf* que se basa en el la máxima distancia de los objetos pertenecientes a un grupo a los más cercanos de los otros grupos.

Inter-Cluster Simple:

$$\delta_1(S,T) = Min\left\{d(x,y)_{x\in S, y\in T}\right\}$$

(32)

Donde, *S* y *T* son grupos de la agrupación *U*; *d(x,y)* es la distancia entre dos objetos cualquiera, *x* y *y* pertenecen a *S* y *T* respectivamente, |*S*| y |*T*| representan el número de objetos pertenecientes al grupo *S* y *T* respectivamente.

Inter-Cluster Completa

$$\delta_2(S,T) = max\left\{d(x,y)_{x\,\in\, S, y\,\in\, T}\right\}$$

(33)

Inter-Cluster Promedio

$$\delta_3(S,T) = \frac{1}{|S|y\,\in\,|T|}\left\{\sum_{x\in S, y\in T} d(x,y)\right\}$$

(34)

Inter-Cluster Centroide

$$\delta_4(S,T) = d(vs,vt)$$

(35)

Donde,

$$vs = \frac{1}{|S|}\sum_{x \in S} x \ \ , vt = \frac{1}{|T|}\sum_{y \in T} y$$

(36)

Inter-Cluster Promedio a Centroide

$$\delta_5(S,T) = \frac{1}{|S| + |T|} = \max\left(\sum_{x \in S} d(x, vt) + \sum_{y \in T} d(y, vs) + \right)$$

(37)

Métrica de Hausdorf

$$\delta_6(S,T) = \max\{\delta(S,T), \delta(T,S)\}$$

Donde,

$$\delta(S,T) = \ max\ _{x \in S}\left\{min\{d(x,y)_{y \in T}\}\right\},\ \delta(T,S) max_{y \in T}\{min\{d(x,y)_{x \in S}\}\}$$

(38)

6.6.2 DISTANCIAS INTRA-CLUSTER

Son tres las distancias Intra-Cluster utilizadas para calcular el *índice de Davies-Bouldin*, la distancia *Intra-Cluster Completa,* que representa la distancia entre los objetos más remotos pertenecientes al mismo grupo, la distancia *Intra-Cluster Promedio,* que representa la distancia promedio de los objetos pertenecientes al mismo grupo y la distancia *Intra-Cluster Centroide* que representa el promedio entre todos los objetos pertenecientes a un grupo y su centroide.

Intra-Cluster Completa

$$\Delta_1(S) = max\{d(x,y)_{x,y\in S}\}$$

(39)

Intra-Cluster Promedio

$$\Delta_2(S) = \frac{1}{|S| \cdot (|S|-1)} \sum_{\substack{x,y\in S \\ x\neq y}} \{d(x,y)\}$$

(40)

Intra-Cluster Centroide

$$\Delta_3(S) = 2\left(\frac{\sum_{x\in S} d(x,\bar{v})}{|S|}\right)$$

Donde,

$$\bar{v} = \frac{1}{|S|}\sum_{x\in S} x$$

(41)

6.6.3 ÍNDICE DE DAVIES-BOULDIN

El índice de Davies-Bouldin es uno de los principales índices utilizados para identificar que tan compacta y bien separada es una agrupación, este se define como:

$$DB(U) = \frac{1}{c}\sum_{i=1}^{c} max_{i\neq j}\left\{\frac{\Delta(X_i)+\Delta(X_j)}{\delta(X_i,X_j)}\right\}$$

(42)

Donde, U es cualquier agrupación de tal manera que $U \leftrightarrow X: X_1 \cup X_2 \cup \ldots X_i \cup X_c$, c es el número de grupos de la agrupación U, X_i el i-ésimo grupo de la agrupación U, $\delta(X_i, X_j)$ representa la distancia entre los grupos X_i y X_j (distancia Inter-Cluster) y $\Delta(X_i)$ representa la distancia Intra-Cluster del grupo X_i.

El valor del índice de Davies-Bouldin se minimiza mientras más compacta y bien separada sea una agrupación.

7. ESTUDIO EXPERIMENTAL DE COMPARACIÓN

En este capítulo se presentan los resultados de la experimentación realizada con los algoritmos convencionales de agrupación seleccionados y el algoritmo de agrupación GGSA para el conjunto de imágenes seleccionado.

En primer lugar se analizan los resultados de la experimentación realizada para todas las instancias analizadas y posteriormente se realiza un análisis detallado para las instancias que presentaron los resultados más interesantes.

El capítulo se concluye con una discusión sobre los resultados obtenidos por los algoritmos considerados en esta tesis para la segmentación de imágenes, especialmente enfocada sobre la comparación de los resultados obtenidos por el Algoritmo de Búsqueda Gravitacional Para Agrupación y los algoritmos convenciones seleccionados como referencia.

y ayudan como preámbulo en la evaluación, para esto se calculó en gran promedio de las distancias para cada instancia.

En términos de las distancias Intra-Cluster, se puede observar que tanto el algoritmo *GGSA* como *K-medias* presentaron un mejor comportamiento que el algoritmo *SOM*, lo anterior quiere decir que los grupos de los generados por estos algoritmos poseen una mejor estructura en cuanto a cohesión.

Por otra parte, al observar el comportamiento del gran promedio por instancia de las distancias Inter-Cluster, a diferencia de los resultados de las distancias Intra-Cluster, los mejores comportamientos son presentados por el algoritmo SOM y GGSA presenta mejores de separación que K–medias en la mayoría de los casos.

Tabla 12 Valores Distancias Intra e Inter-Cluster

Num.	Imagen	Metodo	Intracls completa	Intracls promedio	Intracls centroide	Intercls simple	Intercls completa	Intercls promedio	Intercls centroide	Intercls prom a centroide	Intercls hausdorff
1	1001875	GSA	246.5592	**53.7995**	**38.5969**	1.1381	377.8571	**137.2537**	**129.9123**	**133.9155**	182.0464
1	1001875	KMEDIAS	246.5592	53.8010	38.5982	1.1381	377.8571	137.2488	129.9069	133.9102	182.0785
1	1001875	SOM	**239.9175**	54.1962	38.7925	1.1381	**381.6646**	137.1888	129.6103	133.7069	**188.5859**
2	3000299	GSA	127.1071	15.1465	**19.5791**	**22.4011**	**263.4854**	94.1993	93.2392	93.6796	142.0131
2	3000299	KMEDIAS	127.1071	15.0832	19.5810	21.8484	263.4808	94.2388	93.2841	93.7226	**142.0179**
2	3000299	SOM	**126.6059**	**14.0701**	19.6265	21.5526	263.4802	**95.4786**	**94.5225**	**94.9583**	141.8980
3	3000323	GSA	**123.1553**	33.2418	24.7512	**52.7105**	279.5499	**176.4283**	**174.7327**	**175.5324**	**167.1163**
3	3000323	KMEDIAS	123.2766	**30.5363**	**22.6692**	44.9889	272.9459	132.2407	129.8512	130.9073	159.0403
3	3000323	SOM	139.5430	34.5773	25.7761	49.2278	**286.4162**	173.5232	171.6404	159.2174	162.4418
4	5000180	GSA	**160.3462**	44.4223	33.4156	**46.6184**	**349.5583**	201.9307	197.9416	199.7522	**197.8483**
4	5000180	KMEDIAS	**160.3462**	**44.4100**	33.4179	46.4325	**349.5583**	201.7377	197.7465	199.5579	197.7929
4	5000180	SOM	168.3143	44.4818	**32.7409**	44.7234	353.9754	**207.7790**	**203.9946**	**205.7845**	196.3177
5	5000196	GSA	120.3925	27.2936	20.3751	**66.5856**	**296.5481**	**165.5121**	**163.9630**	**164.6945**	**180.2611**
5	5000196	KMEDIAS	118.1316	**24.6607**	**17.6792**	50.8264	272.9876	139.1594	138.1876	138.6917	160.6241
5	5000196	SOM	**117.3456**	24.9142	17.9295	50.7699	272.7951	138.6308	137.6480	138.1549	161.3606
6	6000332	GSA	**191.8981**	**46.2116**	**33.8970**	23.0433	320.4716	146.6677	141.6942	144.3388	168.0383
6	6000332	KMEDIAS	199.5023	48.6936	35.7159	**43.2474**	303.5928	144.9467	135.9447	140.2380	171.3696
6	6000332	SOM	195.3491	49.2986	36.4144	40.3086	**320.5009**	**146.8308**	**138.4778**	**142.2841**	**181.6826**
7	8000811	GSA	213.2310	54.6738	40.0374	**28.3173**	334.1764	**138.0168**	**129.2013**	**133.2894**	**178.1490**
7	8000811	KMEDIAS	**160.0778**	**33.5461**	**23.9561**	21.7968	324.4316	110.8517	108.2920	109.5747	172.8110
7	8000811	SOM	**160.0778**	33.6409	24.0212	21.6587	**324.8593**	111.2608	108.7197	109.9930	173.1242
8	9000002	GSA	135.3284	**33.0900**	**24.5175**	**59.6607**	**296.8119**	**173.6955**	**171.4984**	172.6037	**177.1031**

8	9000002	KMEDIAS	140.3043	37.7342	28.1813	41.1954	286.7324	129.6134	125.8199	127.7985	161.2100
8	9000002	SOM	**134.8354**	33.5210	24.9047	58.9225	294.4502	172.9161	170.6672	**171.8160**	176.5246
9	9003423	GSA	199.1785	**44.1560**	**32.8650**	**65.5369**	323.4817	**193.8191**	**188.7698**	**191.4880**	**194.6443**
9	9003423	KMEDIAS	**197.4789**	49.2532	37.1125	20.5804	321.0914	140.4564	135.3342	137.8917	168.4053
9	9003423	SOM	200.7533	44.9793	33.5077	42.4370	**241.3869**	142.3956	138.4511	140.5009	145.4863
10	9004383	GSA	110.6085	29.6506	21.6827	57.5783	**260.8758**	**152.0341**	**150.9748**	**151.4755**	**158.0195**
10	9004383	KMEDIAS	**107.1364**	29.3916	**21.4791**	**59.0417**	258.4859	150.7627	149.7286	150.2228	157.3809
10	9004383	SOM	**107.1364**	**29.3912**	21.4792	59.0162	258.4859	150.7594	149.7258	150.2199	157.3427
11	09cm_area17	GSA	**173.3645**	38.9987	32.3350	**16.6882**	303.3438	-177.9663	**108.9720**	**112.0655**	**156.7505**
11	09cm_area17	KMEDIAS	181.4408	-62.0719	29.6412	16.0125	**304.6854**	**-9.2560**	98.3378	101.5929	155.9499
11	09cm_area17	SOM	181.4408	**-62.3549**	**29.6332**	16.0308	304.6724	-9.3099	98.3191	101.5724	155.9437
12	09cm_area31	GSA	167.3152	42.7757	30.8097	25.8619	**318.8644**	**162.1008**	**158.7241**	**160.4588**	**168.3105**
12	09cm_area31	KMEDIAS	**133.5849**	**30.7996**	**22.4345**	**41.9912**	263.7185	107.7101	106.3985	106.8331	150.4777
12	09cm_area31	SOM	165.9270	42.9447	30.9824	25.4421	318.0230	161.1921	157.8190	159.5468	167.5761
13	09cm_area37	GSA	171.5550	-269.4996	32.2207	**20.9801**	**302.2080**	**-314.1029**	**114.5931**	**117.5588**	**162.5763**
13	09cm_area37	KMEDIAS	**164.3279**	-2148.2817	31.9702	17.3089	300.2275	-445.1319	112.7488	115.7164	157.7660
13	09cm_area37	SOM	165.1836	**-1892.5870**	**31.9679**	17.2675	300.2275	-450.4774	112.7165	115.4070	157.7257
14	potsdam_2_12	GSA	**134.0424**	30.1712	21.9292	**44.5242**	269.6446	**117.7590**	**116.3974**	**116.8690**	**155.4308**
14	potsdam_2_12	KMEDIAS	**134.0424**	**30.1711**	21.9293	**44.5242**	269.6446	117.7649	116.4032	116.8747	155.4583
14	potsdam_2_12	SOM	134.0458	30.1720	**21.9287**	44.1675	**270.4407**	117.5632	116.2036	116.6751	154.8381
15	potsdam_6_17	GSA	133.5849	30.7996	22.4345	41.9912	**263.7185**	107.7101	**106.3985**	106.8331	**150.4777**
15	potsdam_6_17	KMEDIAS	132.7467	**30.7528**	**22.3981**	41.8193	262.7309	107.4878	106.1788	106.6129	150.1362
15	potsdam_6_17	SOM	**132.3735**	30.7752	22.4163	**42.1659**	262.7646	**107.7446**	106.4391	**106.8714**	150.3159

Aunque las distancias hasta acá analizadas describen las características más importantes de las agrupaciones resultantes, por si solas no permiten evaluar y comparar la bondad de la estructura de las agrupaciones generadas por cada algoritmo, razón por la cual se hace necesario el análisis de índices diseñados para tal, como ya se mencionó estos índices son calculados a partir de las distancias previamente generadas y nos entregan magnitudes que se pueden comparar para determinar si una agrupación es mejor o peor que otra.

En la Tabla 13. Se encuentran los valores resultantes para el *Índice de Davies-Bouldin*, como se mencionó este índice es calculado a partir de las distancias Inter e Intra-Cluster y entre menor sea su magnitud representa una mejor agrupación.

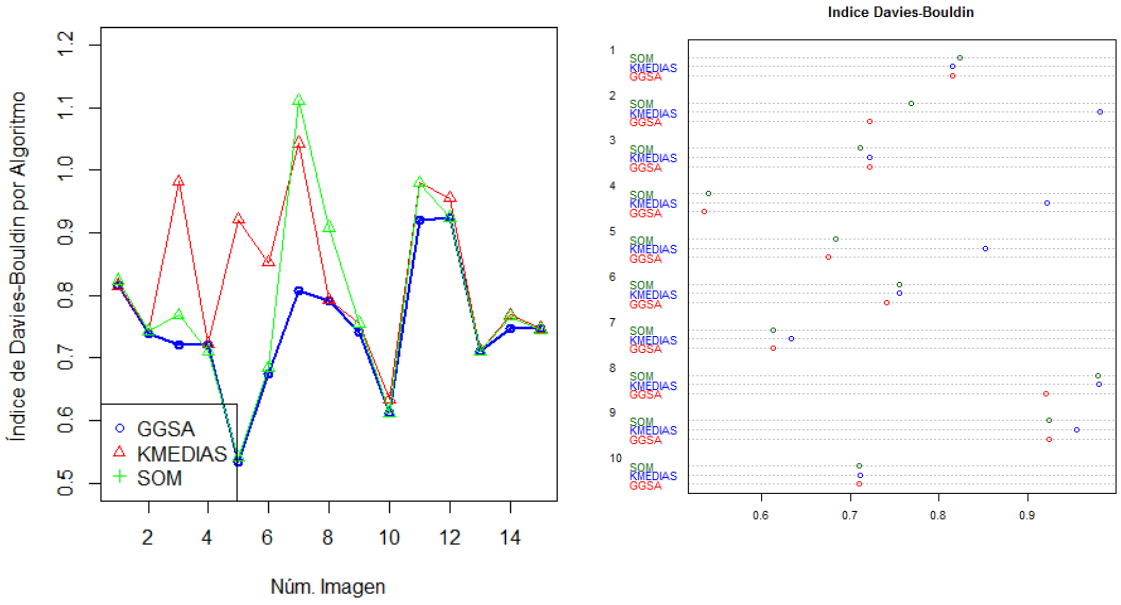
Tabla 13 Comparación de índices de validación de Davies-Bouldin

Num.	Imagen	GGSA	Kmedias	SOM
1	1001875	0.81557	**0.81557**	0.82318
2	3000299	**0.73933**	0.74290	0.74275
3	3000323	**0.73450**	0.98207	0.76885
4	5000180	0.72180	0.72203	**0.71084**
5	5000196	**0.53452**	0.92122	0.54026
6	6000332	**0.80651**	1.04336	1.11100
7	8000811	**0.79193**	0.79262	0.90804
8	9000002	**0.67476**	0.85243	0.68375
9	9003423	**0.74150**	0.75549	0.75563
10	9004383	**0.61236**	0.63359	0.61263
11	09cm_area17	**0.92115**	0.98005	0.97964
12	09cm_area31	**0.74723**	0.76881	0.76591
13	09cm_area37	**0.92459**	0.95544	0.92461
14	potsdam_2_12	**0.71000**	0.71099	0.71022
15	potsdam_6_17	**0.74560**	0.74762	0.74723

En la anterior tabla se muestran la comparación de resultados obtenidos para el *Índice de Davies-Bouldin* por los algoritmos GGSA, K-Medias y SOM, En negrilla se resaltan los menores valores que identifican a los mejores resultados. En términos generales los tres algoritmos entregaron buenos resultados. Sin embargo como puede observarse para la mayor

parte de las imágenes los valores obtenidos por el algoritmo GGSA son mejores a los obtenidos por los otros dos algoritmos. Para 13 de las 15 imágenes analizadas el algoritmo GGSA tuvo los mejores valores para el índice y para los dos casos restantes obtuvo el segundo mejor, por otro lado el algoritmo SOM obtuvo para 10 casos también el segundo mejor valor para el indicador. Lo anterior quiere decir que el mejor desempeño fue alcanzado por el algoritmo GGSA, seguido del algoritmo SOM ya que estos generaron agrupaciones más compactas y mejor separadas.

Ilustración 19 Graficas Comparación Índice Davies-Bouldin



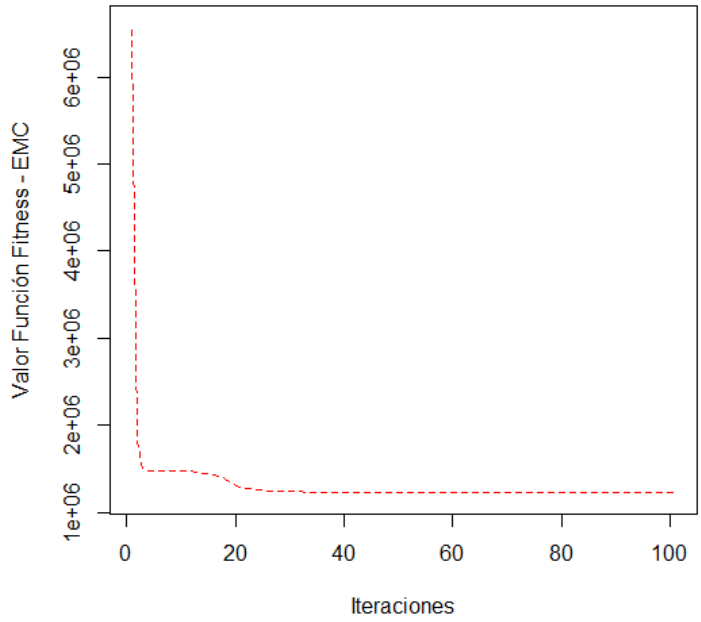
En la Tabla 15. Se consideran el número de iteraciones realizadas por cada algoritmo, el algoritmo K-medias se detiene automáticamente cuando cumple la condición de que ningún objetos se mueve de grupo, mientras que los algoritmos GGSA y SOM finalizan cuando terminan del número de iteraciones parametrizadas, para este caso se determinó que 100 eran las iteraciones necesarias para lograr buenos resultados, al observar los datos se puede ver que el número de iteraciones requerido por el algoritmo K-medias es considerablemente menor al requerido por los otros dos algoritmos, la rapidez es una de las ventajas principales de este algoritmo.

Tabla 14 Comparación de Iteraciones por Algoritmo

NÚM.	*IMAGEN*	*K-MEDIAS*	*GGSA*	*SOM*
1	1001875	100	100	3
2	3000323	100	100	3
3	5000180	100	100	2
4	5000196	100	100	3
5	9000002	100	100	3
6	9003423	100	100	3
7	9004383	100	100	2
8	09cm_area17	100	100	3
9	09cm_area37	100	100	3
10	potsdam_2_12	100	100	3

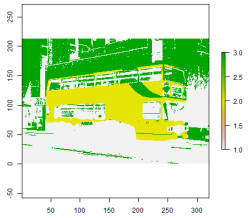
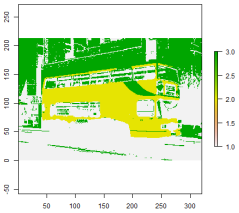
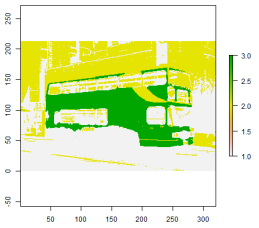
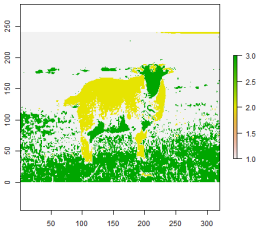
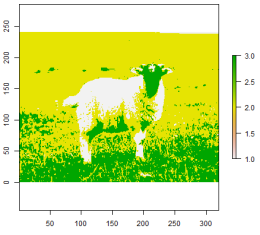
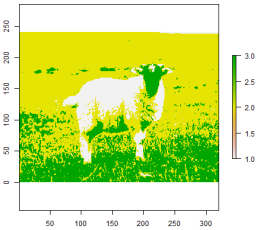
Ejemplo de evolución del Error Medio Cuadrático (EMC) – Función Fitmess minimizada por el algoritmo GGSA en función de la iteración del algoritmo.

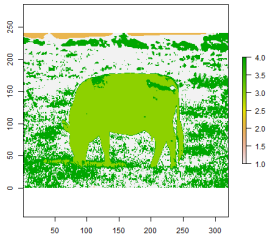
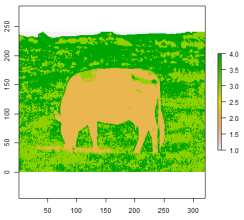
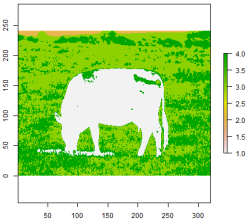
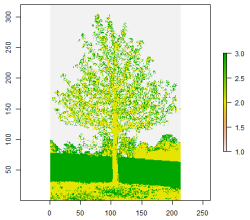
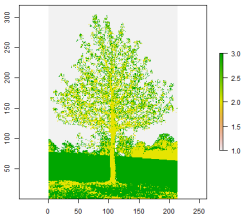
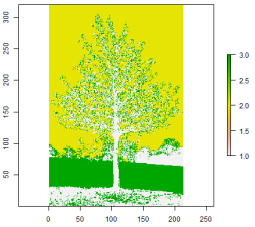
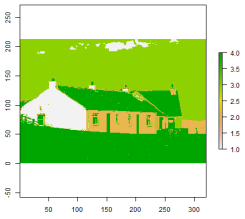
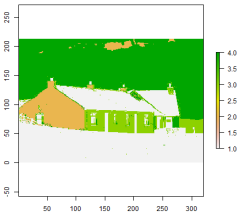
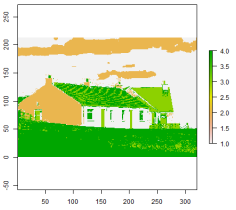
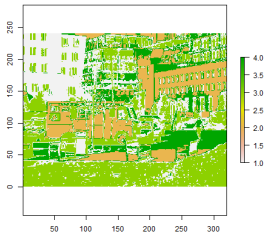
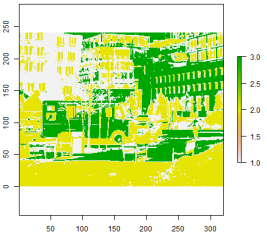
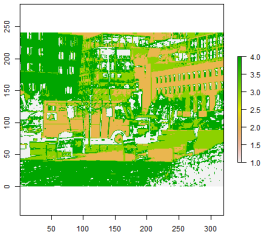
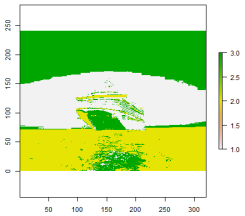
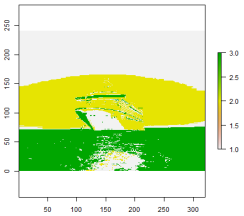
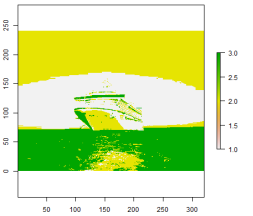
Ilustración 20 Ejemplo de evolución del valor de la función fitness - GGSA

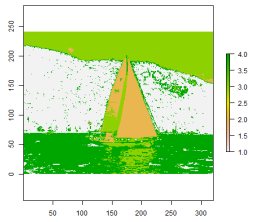
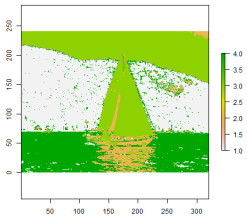
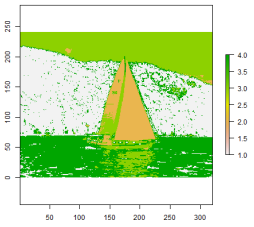
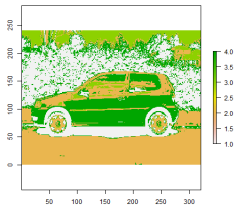
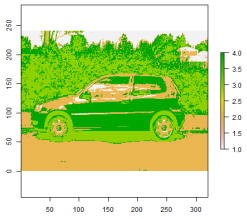
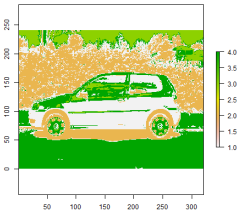
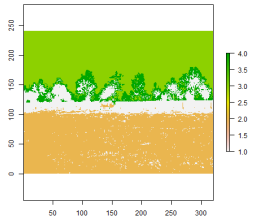
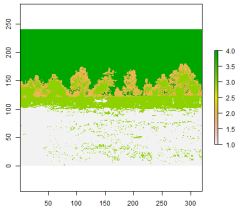
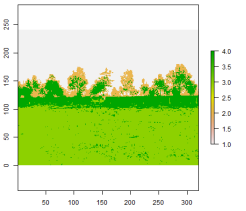
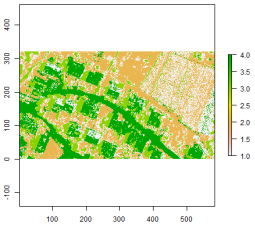
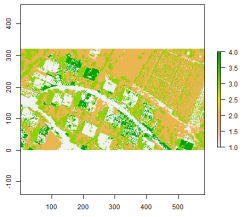
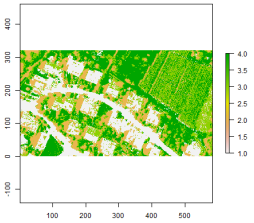
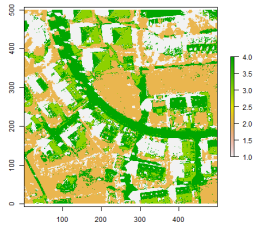
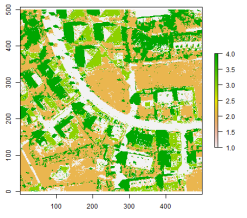
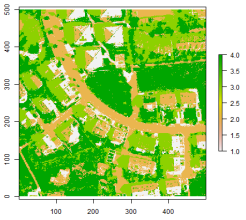


En la Ilustración 18. Se muestra un ejemplo de evolución del Error Medio Cuadrático - EMC sobre el conjunto de entrenamiento a lo largo de las iteraciones del algoritmo, hasta la iteración 100. El EMC es la función minimizada por el algoritmo GGSA En este caso, en las primeras 20 iteraciones se producen las más grandes variaciones, para posteriormente presentarse variaciones decrecientes pequeñas, lo anterior nos muestra el grado de movimiento de las partículas en torno a los mejores grupos, La atracción hacia las mejores posiciones se presenta con mayor velocidad en las primeras iteraciones, esto se debe al ajuste que van teniendo los grupos iteración tras iteración por la atracción generada por las mejores soluciones y también por el valor de la constante gravitacional en el tiempo que va disminuyendo iteración tras iteración haciendo que la fuerza atracción vaya disminuyendo en el tiempo para controlar la precisión de la búsqueda.

Tabla 15 Comparación de los Resultados por Algoritmo

#	***Original***	***GGSA***	***SOM***	***KMEDIAS***
1				
2				

3				
4				
5				
6				
7				

8				
9				
10				
11				
12				

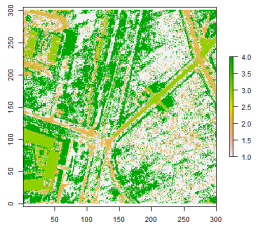
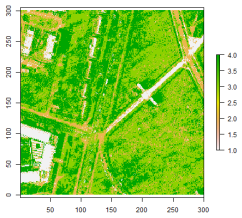
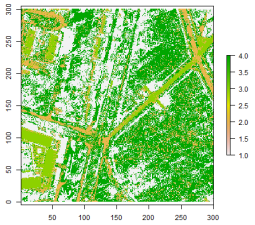
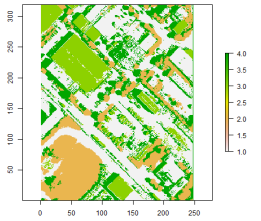
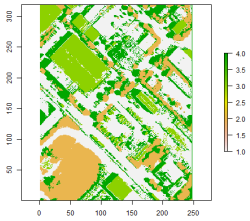
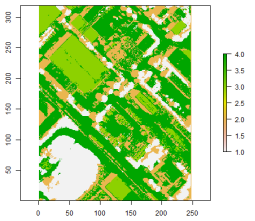
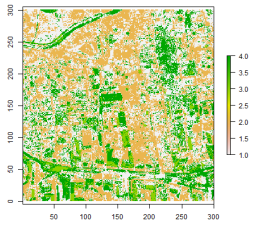
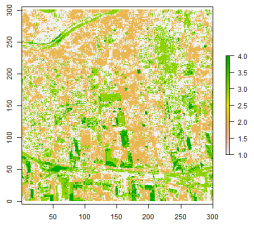
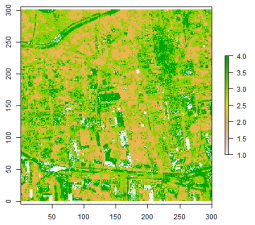
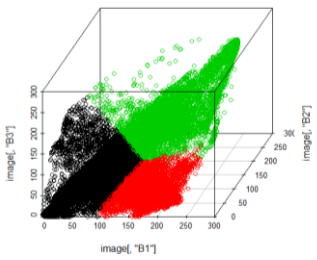
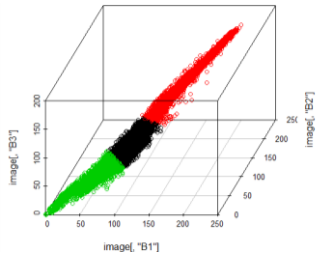
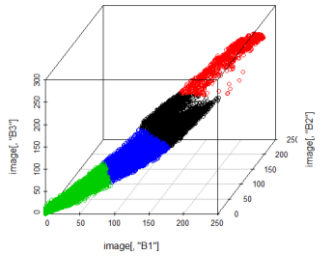
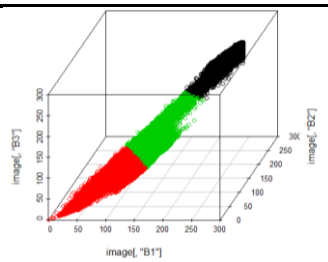
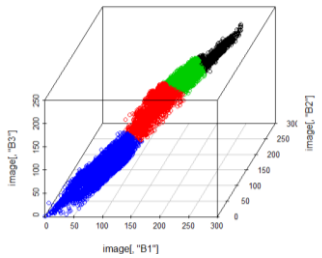
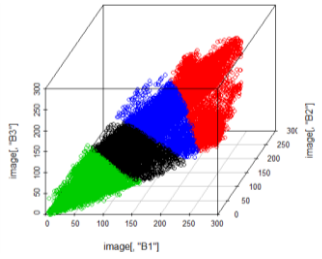
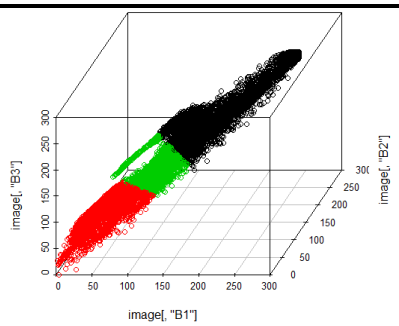
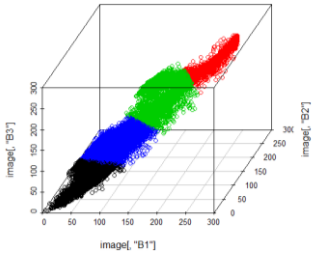
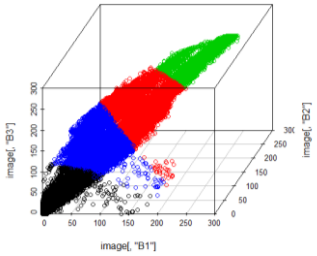
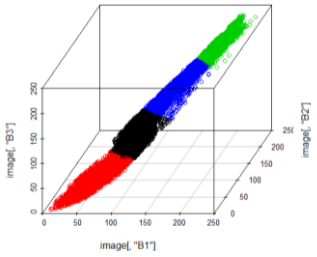
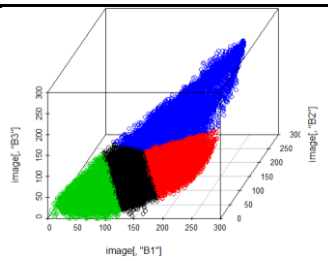
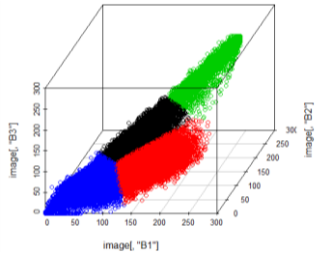
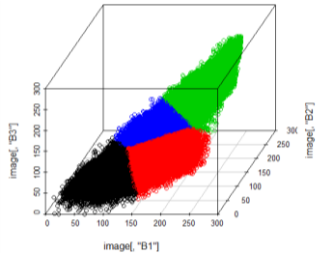
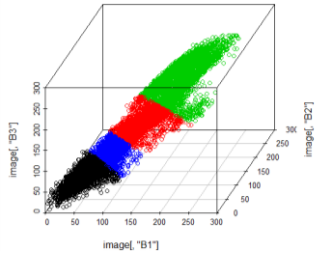
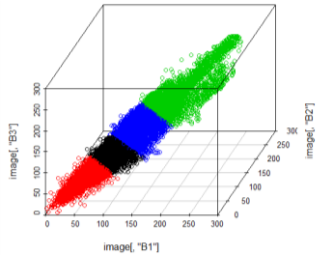
13				
14				
15				

Tabla 16 Diagramas de Dispersión Clústeres

#	Imagen	Diagrama
1	1001875	

2	3000299	
3	3000323	
4	5000180	
5	5000196	
6	6000332	

7	8000811	
8	9000002	
9	9003423	
10	9004383	
11	09cm_area17	

		
12	09cm_area31	
13	09cm_area37	
14	potsdam_2_12	
15	potsdam_6_17	

Como se observó con anterioridad, el índice de Davies-Bouldin indica que las agrupaciones generadas por el algoritmo GGSA poseen una mejor bondad en su estructura, seguido del algoritmo SOM, Se procederá ahora con el análisis de un caso en particular para validar estos resultados, para esto se eligió la imagen "5000196" la cual presento uno de las mayores discrepancias entre la magnitud del índice de Davies-Bouldin para estos dos algoritmos en comparación con el obtenido por el algoritmo K-Medias.

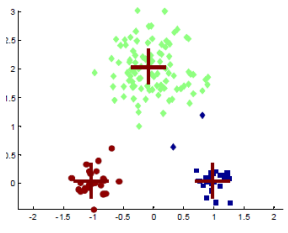
Tabla 17 Índices Davies-Bouldin para la imagen 5000196

IMAGEN	GGSA	SOM	KMEDIAS
5000196	*0.5345227*	*0.5402610*	*0.9212150*

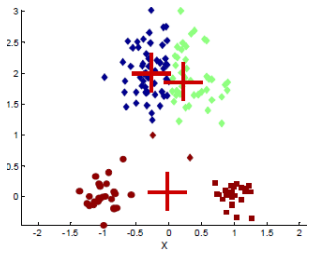
Comparando las segmentaciones obtenidas por cada uno de los algoritmos considerados, puede verse que las segmentaciones generada por el algoritmo GGSA es similar a la generada por el algoritmo SOM y que ambas difieren en gran medida de la generada por K-medias, adicionalmente al comparar estas dos con la segmentación original también puede observarse que guardan mayor relación con la segmentación de los objetos presentes en la escena, lo cual conlleva a decir que estas poseen un mayor grado de validez.

Ilustración 21 Un ejemplo típico de la convergencia del k-medias a un óptimo local[4]

a) Solución Optima



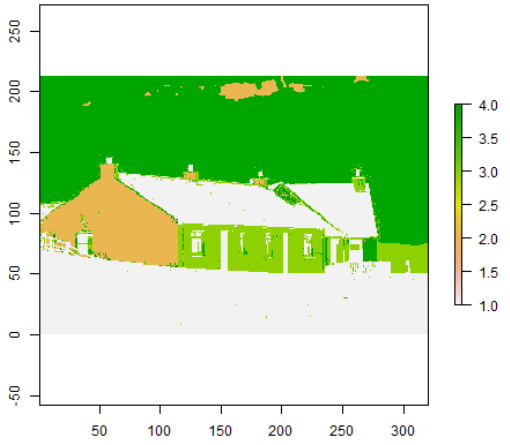
b) Solución K-medias (optimo local)

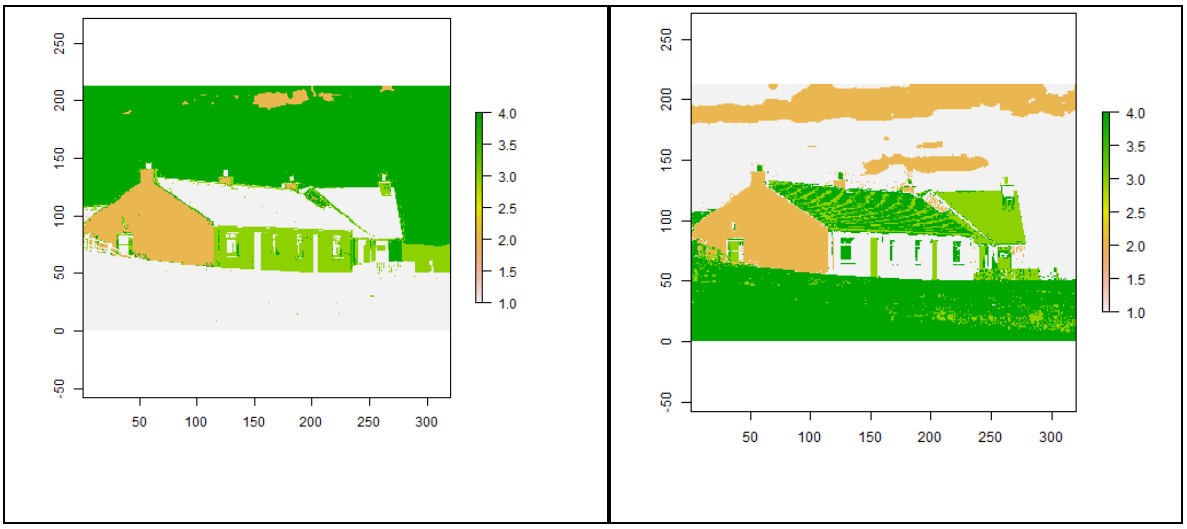


[4] Fuente http://elvex.ugr.es/decsai/intelligent/slides/dm/D3%20Clustering.pdf

Durante la investigación pudo observarse que el algoritmo K-medias converge rápidamente, sin embargo puede hacerlo a un **óptimo local** y esto puede traer malos resultados, lo anterior puede explicar el resultado deficiente obtenido para la imagen *"5000196",* el algoritmo GGSA gracias a estar fundamentado en un algoritmo de optimización y a que durante su ejecucíon se presenta movimento de objetos entre los grupos por medio de la atraccion generada por la fuerza de gravitacion puede escapar con facilidad optimos locales y entregar un resultado que se acerca el **optimo golbal.**

Tabla 18 Comparación de Resultados Imagen 5000196

<table>
<tr><td colspan="2">Comparacion Segmentaciones Resultantes para la imagen "5000196"</td></tr>
<tr><td>a) Segmentación Original
  </td><td>b) Segmentación GGSA
 
</td></tr>
<tr><td>c) Segmentación SOM</td><td>a) Segmentación K-Medias</td></tr>
</table>



Por otro lado al comparar únicamente los resultados para GGSA y SOM puede observase que los bordes de cada objeto se encuentran mejor definidos para la segmentación obtenida por el primer algoritmo, esto puede observarse visualmente en las imágenes generadas a partir de la segmentación y también por medio de la revisión de las distancias Inter-Cluster para las cuales el algoritmo GGSA obtuvo las menores magnitudes para la mayoría de los casos, indicando una cohesión en los grupos generados.

8. CONCLUSIONES

En esta investigación se desarrolló un estudio experimental comparativo entre los algoritmos convencionales K-medias y SOM y el algoritmo basado en la ley de gravitación universal GGSA para la segmentación de imágenes sobre un conjunto de imágenes reales de escenas a campo abierto y adquiridas mediante sensores remotos, para todos los casos se realizó una comparación basada en el índice Davies-Bouldin y las distancias Intra e Inter-Cluster que permiten evaluar y comparar los resultados generados por cada algoritmo, Las conclusiones que se detallan a continuación se basan en dichas comparaciones y en el análisis general de los resultados:

- En este trabajo de investigación, se realizó la adaptación del *Algoritmos de Búsqueda Gravitacional para agrupación de datos GGSA,* para su aplicación en la segmentación de imágenes. Se logró demostrar el algoritmo tiene la capacidad resolver exitosamente este tipo de problemas.

- Los resultados del algoritmo GGSA, fueron comparados con los resultados obtenidos por algoritmos convencionales K-medias y SOM. La comparación de las agrupaciones resultantes se hizo para todas las instancias por medio de las medias Intra e Inter-Cluster y el índice Davies-Bouldin que permiten validar y comparar la bondad de la estructura de las agrupaciones generadas por cada algoritmo. En términos generales el algoritmo GGSA obtuvo mejores resultados para la mayoría de imágenes segmentadas. Lo que quiere decir que el algoritmo logro encontrar agrupaciones con mayor cohesión y separación. La comparación de resultados encontró un mejor desempeño para el algoritmo GGSA, seguido por SOM y finalmente K-medias.

- Con la implementación de este algoritmo y las diferentes pruebas que se realizaron con este, se logró corroborar lo planteado en [13] referente a *"los algoritmos*

metaheurísticos cuentan con mecanismos para evitar quedar atrapado en un óptimo local", esta característica le permite afrontar al algoritmo GGSA de mejor manera una de las principales debilidades del algoritmo k-medias. Para ampliar esta afirmación consultar [33],[34].

- El algoritmo GGSA logro generar agrupaciones con una mejor cohesión que las generadas por el algoritmo SOM, esto puede evidenciarse por medio de los valores obtenidos en las distancias Inter-Cluster para las cuales el algoritmo GGSA obtuvo las menores magnitudes en la mayoría de las instancias, indicando una mejor cohesión en los grupos generados.
- En segundo lugar, Para el marco experimental diseñado en esta investigación, el algoritmo GGSA basado en la ley de gravitación logra minimizar las debilidades detectadas en los algoritmos convencionales K-medias y SOM, gracias a su mayor capacidad de manejar óptimos locales y la generación agrupaciones de mayor calidad con mejores características de cohesión y separación.

9. RECOMEDACIONES

- Se sugiere que como continuación de esta investigación, podrían realizarse pruebas de uso de técnicas híbridas con el algoritmo GGSA en busca a mejorar su lenta convergencia y disminuir el número de iteraciones necesarias para resolver los problemas de segmentación de imágenes.

- Un enfoque opcional que se podría explorar, es la posibilidad de utilizar otras adaptaciones del algoritmo GSA para la agrupación no- supervisada aplicada a la segmentación de imágenes que incorporen diferentes funciones objetivo y medidas de similitud con el fin de analizar si esto provoca una mejora en los resultados y convergencia del algoritmo.

- Este trabajo de investigación puede ser tomado como base de trabajos futuros que quisieran ampliar el contexto de aplicación de las técnicas metaheurísticas o algoritmos basados en la ley de gravitación universal aplicados a la segmentación de imágenes o de cualquier otro enfoque de aplicación que pudiera surgir.

10. BIBLIOGRAFÍA

[1] Y. Li, C. Li y X. Wu, «Novel Fuzzy C-Means Segmentation Algorithm for Image with the Spatial Neighborhoods,» de *Remote Sensing, Environment and Transportation Engineering (RSETE), 2nd International Conference*, Nanjing, 2012.

[2] M. Yeen Choong, W. Yeang Kow y Y. Kwong Chin, «Image Segmentation via Normalised Cuts and Clustering Algorithm,» de *International Conference on Control System*, 2012.

[3] S. Saraswathi y A. Allirani, «Survey on Image Segmentation via Clustering,» de *Information Communication and Embedded Systems (ICICES),International Conference*, Chennai, 2013.

[4] L. Peng, B. Yang, Y. Chen y A. Abraham, «Data gravitation based classification,» *Information Sciences -Sciences Direct,* vol. 179, nº 6, pp. 809-819, 2009.

[5] I. H. Witten y E. Frank, de *Data Mining: Practical Machine Learning Tools and Techniques*, Burlington, USA, Morgan Kaufmann, 2005.

[6] C. Krall, «Apr-Aprendaprogramar.com,» [En línea]. Available: http://www.aprenderaprogramar.com/index.php?option=com_content&id=252:mineria-de-datos-data-mining-ique-es-ipara-que-sirve-1o-parte-dv00105a&Itemid=164. [Último acceso: 13 Marzo 2015].

[7] Microsoft, «Concepto de minería de datos,» Microsoft Developer Network, Octubre 2013. [En línea]. Available: https://msdn.microsoft.com/es-es/library/ms174949.aspx. [Último acceso: 07 Marzo 2015].

[8] A. Rojas Hernandez, «Modelo Basado en la mineria de flujo de datos para el análisis de CLICS en un sitio WEB.,» Departamento de Ingenieria de Sistemas e Industrial., 2010. [En línea]. Available: http://www.bdigital.unal.edu.co/8840/1/299675.2010.pdf. [Último acceso: 07 Marzo 2015].

[9] A. Rojas, Modelo Basado en Minería de Flujos de Datos para el Análisis de CLICS en un Sitio WEB, Bogota - Colombia: Universidad Nacional de Colombia , 2010.

[10] C. Pérez López, Minería de datos : técnicas y herramientas, Madrid - España: Thomson, 2007.

[11] A. Guerra G., S. Vega P. y J. Rúiz S, «Algoritmo de agrupamiento conceptuales: un estado del arte,» CENATAV, Habana- Cuba, 2012.

[12] Elvex, «Métodos de Agrupamiento- Clustering,» [En línea]. Available: http://elvex.ugr.es/doc/proyecto/cap8.pdf. [Último acceso: 23 Abril 2015].

[13] M. BagherDowlatshahi y H. Nezamabadi-pour, «GGSA:A Grouping Gravitational Search Algorithm for Data Clustering,» *Elservier - Engineering Applications of Artificial Intelligence,* vol. 36, nº 0952-1976, pp. 114- 121, 2014.

[14] H. Barrera, J. Correa y J. & Rodríguez, «Prototipo de Software para el preprocesamiento de datos "UD-Clear",» *IV Simposio Internacional de Sistemas de Información e Ingeniería de Software en la Sociedad del Conocimiento SISOFT - Cartagena, Colombia,* pp. 167-184, 2006.

[15] M. Berry y G. S. Linoff, Data Mining Techniques, Indianapolis, Indiana, USA: Wiley, 2004.

[16] Elvex, «Clustering - Algoritmo de las K Medias,» Numerical Cruncher, [En línea]. Available: http://elvex.ugr.es/software/nc/help/spanish/nc/clustering/KMeans.html.. [Último acceso: 07 Marzo 2015].

[17] Y. Gimenez, «Clasificaci´on no supervisada: El m´etodo de k–medias,» 23 Marzo 2010. [En línea]. Available: http://cms.dm.uba.ar/academico/carreras/licenciatura/tesis/2010/Gimenez_Yanina.pdf. [Último acceso: 12 aBRIL 2015].

[18] J. Gironés Roig, Algotimos Business Intelligence, U. O. d. Catalunya, Ed., Catalunya: Business Analytics, 2014.

[19] M. Millan, «Segmentación o Clustering,» 2008. [En línea]. Available: http://ocw.univalle.edu.co/ocw/ingenieria-de-sistemas-telematica-y-afines/descubrimiento-de-conocimiento-en-bases-de-datos/material-1/Segmentacion08.pdf. [Último acceso: 23 Abril 2015].

[20] R. Salas, «Mapas Autoorganizativas de Kohonen (SOM),» 04 11 2004. [En línea]. Available: http://www.inf.utfsm.cl/~rsalas/Pagina_Investigacion/docs/Apuntes/Redes%20SOM.pdf. [Último acceso: 23 Abril 2015].

[21] J. Marin, «Mapas Auto-organizativos de Kohonen SOM,» 05 10 2008. [En línea]. Available: http://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/DM/tema5dm.pdf. [Último acceso: 23 Abril 2015].

[22] A. Diaz R., «Redes neuronales no supervoizadas con topología dinámica de la segmentacion de imagenes a color,» Depatamento de Lenguajes y Ciencias de la Computación, Malaga, 2010.

[23] J. E. R. Rodríguez, Fundamentos de la minería de datos, Bogotaá D.C, Colombia : Universidad Distrital Francisco José de Caldas, 2010.

[24] F. J. Serrano García, «Aplicación de mapas autoorganizados (SOM) a la visualización de datos,» 14 Diciembre 2009. [En línea]. Available: http://kile.stravaganza.org/projects/somvis/download/somvis_report.pdf. [Último acceso: 23 Abril 2015].

[25] A. J. Alfaro Alfaro y . I. A. Sipirán Mendoza, «Diseño de un Algoritmo de Segmentación de Imágenes aplicando el Funcional de Mumford-Shah para mejorar el desempe˜no de los Algoritmos Cl´asicos de Segmentación,» 02 Octubre 2007. [En línea]. Available: http://users.dcc.uchile.cl/~isipiran/papers/segvar.pdf. [Último acceso: 07 Marzo 2015].

[26] . I. N.A.M., S. A. Salamah y U. K. Ngah, «Adaptive fuzzy moving K-means clustering algorithm for image segmentation,» *IEEE - Consumer Electronics Society,* vol. 55, nº 4, pp. 2145 - 2153, 2009.

[27] I. Janciak y P. Brezany, «A Reference Model for Data Mining Web Services,» de *Sixth International Conference onSemantics Knowledge and Grid (SKG)*, Beijing, 2010.

[28] E. Ortiz Muñoz , «Contribuciones técnicas de la segmentación de imágenes,» Septiembre 2009. [En línea]. Available: http://arantxa.ii.uam.es/~jms/pfcsteleco/lecturas/20090930ElenaOrtiz.pdf. [Último acceso: 07 Marzo 2015].

[29] N. B. Cortés, Restauración de Imágenes Mediante un Modelo Matemático Basado en las Técnicas de Detección de Bordes y Propagación de Texturas, Bogotá, Colombia: Facultad de Ciencias - Departamento de Matemáticas, Universidad Nacional de Colombia, 2011.

[30] J. Goméz, D. Dasgupta y O. Nasraoui, Compositores, *New Algorithm for Gravitational Clustering.* [Grabación de sonido]. In Proc. of the SIAM Int. Conf. on Data Mining. 2003.

[31] J. Goméz, D. Dipankar y N. Olfa, «A New Gravitational Clustering Algorithm,» University of de Memphis - Departament of Mathematical Sciences , Memphis - USA, 2003.

[32] P. Lizhi, Y. Bo, C. Yuehui y A. Ajith, «Data gravitation based classification,» *Information Sciences,* vol. 179, nº 6, pp. 809-819, 2009.

[33] F. Glover y G. Kochenberger, «Handbook of Metaheuristics.,» *Kluwer Academic Publishers,Norwell, MA,* 2012.

[34] E. Alba, «Parallel Metaheuiristics: A new class of algorithm,» *Wiley Interscience,,* 2005.

[35] E. Rashedi, H. Nezamabadi-pour y S. Saryzdi, «GSA: A Gravitational Search Algorithm.,» *Information Sciences,* vol. 179, pp. 2232-2248, 2009.

[36] W. Wright, «Gravitational clustering, Pattern Recognition,» *Pergamon Press,* vol. 9, pp. 151-166, 1977.

[37] S. Kundu, «Gravitational clustering: a new approach based on the spatial distribution of the points,» *Pattern Recognition - ScienceDirect,* vol. 32, nº 7, pp. 1149-1160, 1999.

[38] A. Hatamlou, S. Abdullah y Z. Othman, «Gravitational Search Algorithm with Heuristic Search for Clustering Problems,» de *3rd Conference on Data Mining and Optimization*, Selangor, Malaysia, 2011.

[39] M. Sanchez, O. Castilloy, J. Castroz y A. Rodriguez, «Fuzzy Granular Gravitational Clustering Algorithm,» Autonomous University of Baja California, Tijuana, Mexico, 2012.

[40] V. J. Rayward-Smith, «Metaheuristics for Clustering in KDD,» Evolutionary Computation, 2005. The 2005 IEEE Congress, Australia, 2005.

[41] D. Panda y A. Rosenfeld, «Image Segmentation by Pixel Classification in (Gray Level, Edge Value) Space,» *Computers, IEEE Transactions,* vol. Sept. 1978, pp. 875 - 879, 1978.

[42] W. Perkins, «Area Segmentation of Images Using Edge Points,» *Area Segmentation of Images Using Edge Points,* Vols. %1 de %2PAMI-2, nº 1, pp. 8-15, 1980.

[43] D. Panda y A. Rosenfeld, «Image Segmentation by Pixel Classification in (Gray Level, Edge Value) Space,» *Computers, IEEE Transactions,* Vols. %1 de %2C-27, nº 0, pp. 875 - 879, 1978.

[44] J. Sklansky, «Image Segmentation and Feature Extraction,» *Transactions on Systems, Man, and Cybernetics,* pp. 237 - 247, 1978.

[45] «Color Clustering Using Self-Organizing Maps,» *Proceedings of the 2007 International Conference on Wavelet Analysis and Pattern Recognition, Beijing, China,2-4 Nov.,* 2007.

[46] S. Chebbout y H. F. Merouani, «Comparative Study of Clustering Based Colour Image Segmentation Techniques,» *2012 Eighth International Conference on Signal Image Technology and Internet Based Systems,* 2012.

[47] «Comparative Study ofImage Segmentation Techniques and Object Matching using Segmentation,» *International Conference on Methodsand Modelsin ComputerScience,* pp. 1 - 6, 2009.

[48] Q. Tianbai y L. Minglu, «Multispectral MR images segmentation using SOM network,» *Computer and Information Technology, 2004. CIT '04. The Fourth International Conference,* pp. 155 - 158, 2004.

[49] S. Indira y A. Ramesh, «Image Segmentation Using Artificial Neural Network and Genetic Algorithm: A Comparative Analysis,» *Process Automation, Control and Computing (PACC), 2011 International Conference,* pp. 1-6, 2011.

[50] Z. Liping, Q. Pan, G. Li y J. Liang, «Improvement of Grayscale Image Segmentation Based on PSO Algorithm,» *Computer Sciences and Convergence Information Technology, 2009. ICCIT '09. Fourth International Conference,* pp. 442 - 446, 2009.

[51] A. Bhaduri, «Color Image Segmentation Using Clonal Selection-Based Shuffled Frog Leaping Algorithm,» *Advances in Recent Technologies in Communication and Computing, 2009. ARTCom 09. International Conference,* pp. 517 - 520, 2009.

[52] L. Hongpo, S. Jun, W. Hai, T. Shuhua y T. Zhiguo, «High Resolution Sonar Image Segmentation by PSO Based Fuzzy Cluster Method,» *Genetic and Evolutionary Computing (ICGEC), 2010 Fourth International Conference,* pp. 18 - 21, 2010.

[53] T. Hongmei, W. Cuixia, H. Liying y W. Xia, «Image segmentation based on improved PSO,» *Computer and Communication Technologies in Agriculture Engineering (CCTAE), 2010 International Conference,* pp. 191 - 194, 2010.

[54] M. Cai-hong, D. Qin y L. Shi-Bin, «A Hybrid PSO-ISODATA Algorithm for Remote Sensing Image Segmentation,» *Industrial Control and Electronics Engineering (ICICEE), 2012 International Conference,* pp. 1371 - 1375, 2012.

[55] M. Sandeli y M. Batouche, «Multilevel thresholding for image segmentation based on parallel distributed optimization,» *Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference,* pp. 134 - 139, 2006.

[56] F. Hamdaoui, A. Mtibaa y A. Sakly, «Comparison between MPSO and MSFLA metaheuristics for MR brain image segmentation,» *Sciences and Techniques of Automatic Control and Computer Engineering (STA), 2014 15th International Conference,* pp. 164 - 168, 2014.

[57] Y. Kuma y G. Sahoo, «A Review on Gravitational Search Algorithm and its Applications to Data Clustering & Classification,» *Modern Education and Computer Science Press,* nº 06, pp. 79-93, 2014.

[58] E. Paradis, R para Principiantes, Paris - Fracia: Institut des Sciences de l'Evolution , 2002.

[59] S. Gould, R. Fulton y D. Koller, «Decomposing a Scene into Geometric and Semantically Consistent Regions.,» *roceedings of International Conference on Computer Vision (ICCV),* 2009.

[60] B. Sathya y R. Manavalan, «Image Segmentation by Clustering Methods: Performance Analysis,» *International Journal of Computer Applications (0975 – 8887),* p. Volume 29– No.11, Septiembre 2011.

[61] S. Abdullah, A. Hatamlou y Z. Othman, Compositores, *Gravitational Search Algorithm with Heuristic Search for Clustering Problems.* [Grabación de sonido]. Data Mining and Optimization Conference on (DMO). 2011.

[62] A. Cano, J. Luna, A. Zafra y S. Ventura , «Modelo gravitacional para clasificación,» 2011. [En línea]. Available: http://simd.albacete.org/maeb2012/papers/paper_45.pdf. [Último acceso: 10 Febrero 2015].

[63] O. Castillo, J. Castro y A. Rodríguez, «Fuzzy granular gravitational clustering Algorithm,» *Fuzzy Information Processing Society (NAFIPS), 2012 Annual Meeting of the North American,* pp. 1 - 6, 06 Agosto 2012.

[64] T. Long y L. W. Jin, «A New Simplified Gravitational Clustering Method for Multi-prototype Learning Based on Minimum Classification Error Training Advances in

Machine Vision, Image Processing, and Pattern Analysis, International Workshop on Intelligent Computing.,» *Computer Science, N. Zheng, X. Jiang, and X. Lan,* vol. 4153, pp. 168-175, 2006.

[65] C. W. Bong y M. Rajeswari, «Multi-objective nature-inspired clustering and classification techniques for image segmentation,» *Soft Computing,* vol. 4, pp. 3271-3282, 2011.

[66] D. Hand, H. Mannila y P. Smyth, Principles of Data Mining, T. Dietterich, Ed., 2004.

[67] J. Hernandez, J. Ramírez y C. Ferri, Introducción a la Minería de datos, Editorial Alhambra S. A., 2004.

[68] L. Peng Lingmin, H. Xiaobing y K. Wang, «Computational Intelligence and Design,» de *Fifth International Symposium* , Zhejiang Sci-Tech University Hangzhou, China, 2012.

[69] G. Oatley y B. Ewart, «Cluster Analysis and Data Mining Applications,» *Analysis,* vol. 1, nº 2, pp. 147-153, 2011.

[70] F. Giraldo, E. León y J. Gomez, «Caracterización de flujos de datos usando algoritmos de agrupamiento. Characterizing data stream using clustering algorithms.,» *Tecnura,* vol. 17, nº 37, 2013.

[71] J. Merelo, «Mapa autoorganizativo de Kohonen,» [En línea]. Available: http://geneura.ugr.es/~jmerelo/tutoriales/bioinfo/Kohonen.html. [Último acceso: 23 Abril 2015].

11. ANEXOS

11.1 ANEXO 1: Desarrollo algoritmo GGSA en R

La implementación del algoritmo GGSA se realizó a partir de la definición de funciones en R. En un archivo *main* se configuran las variables de entrada y al ejecutarse invoca las dos funciones principales del desarrollo:

- *RASTER*
- *GGSA*

La función principal *RASTER* lee una imagen ubicada en la dirección indicada por el usuario y la transforma en un objeto tipo *data.frame* (marco o base de datos) de dimensiones *[n_pix,d]*, es decir la imagen es transformada en una tabla compuesta por vectores correspondientes a cada pixel de la imagen raster, de esta manera el data.frame resultante tiene un numero filas igual al número de pixeles de la imagen "*n_pix*" y un numero de columnas igual al número de bandas de la imagen "*d*" (*d*: dimensionalidad del campo de búsqueda).

Esta función también entrega variables relacionadas con las características de la imagen interpretada como el número de filas y columnas que son utilizadas durante la ejecución en varias etapas por ejemplo la reconstrucción de la imagen ya segmentada.

La segunda función *GGSA*, se encarga de controlar la búsqueda dentro del espacio de búsqueda, esta invoca durante el proceso a cada una de las funciones correspondiente a cada paso del algoritmo y finalmente cuando se alcanza el número máximo de iteraciones indicado genera las salidas que son almacenadas en archivos planos e imágenes raster con la segmentación.

A continuación se presenta la documentación de cada una de las funciones implementadas bajo el desarrollo de esta investigación, posteriormente se listan todas los paquetes y las funciones de R que también fueron utilizadas dentro de la implementación del algoritmo.

Los archivos resultado entregados por el programa se listan en la Tabla 19, por cada resultado final correspondiente a cada imagen del conjunto de datos se generó un juego de archivos compuesto el siguiente listado:

Tabla 19 Listado de archivos resultado

Archivo	Descripción
<<*Nombre_imagen*>>_GGSA.png	Imagen segmentada
<<Nombre_imagen>>_GGSA.txt	Matriz con las soluciones de segmentación
<<Nombre_imagen>>_MEDIDAS_best_S	Matriz ordenada de menor a peor solución
<<Nombre_imagen>>_MEDIDAS_BestChart	Históricos de mejores Fitness
<<Nombre_imagen>>_GGSA_MEDIDAS	Medidas Intra e Inter-Cluster
<<Nombre_imagen>>_GGSA_MEDIDAS_DB	Índice DB

11.1.1. Documentación funciones implementadas en R – Algoritmo GGSA

La relación entre los diferentes pasos que componen el algoritmo GGSA y las funciones que fueron desarrolladas se muestra en la Tabla 20, adicionalmente en la ilustración 22 puede observarse el diagrama general de actividades realizadas por el algoritmo.

Ilustración 22 Diagrama de Actividades algoritmo GGSA

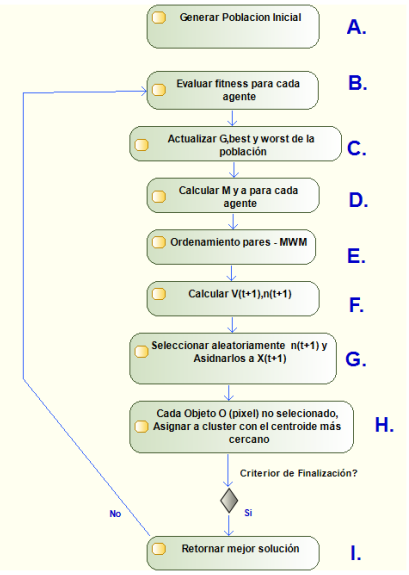
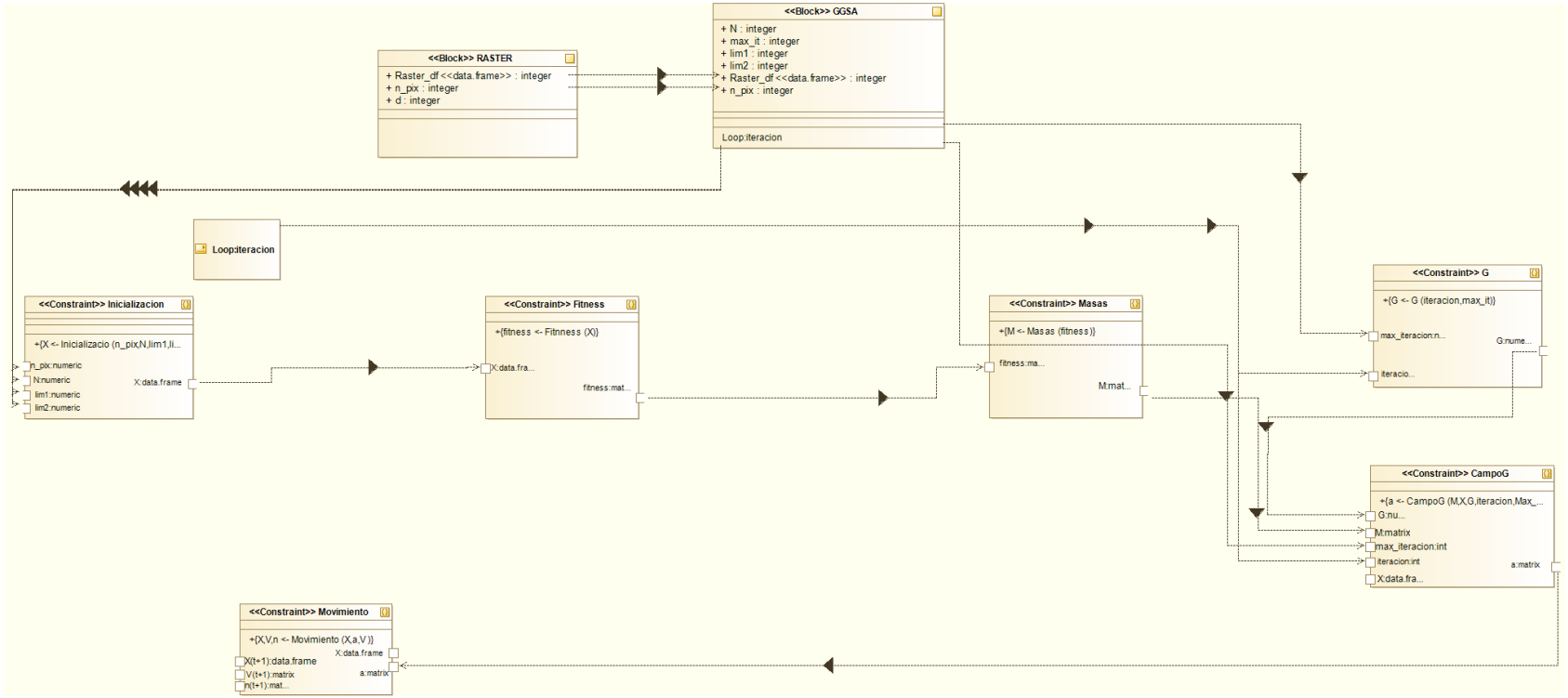


Tabla 20 Relacion Pasos GGSA y funciones creadas en R

Paso	Función
A. Generar población inicial	GGSA:Inicilizacion
B. Evaluar función fitness para cada agente	GGSA:Fitness
C. Actualizar G,best y worst de la población	GGSA GGSA: G
D. Calcular M y a para cada agente	GGSA:Masas GGSA: CampoG
E. Ordenamiento pares – MWM, calcular V(t+1).	GGSA: CampoG
F. Calcular n(t+1)	GGSA:Movimiento
G. Seleccionar aleatoriamente n(t+1) y	GGSA:Movimiento
H. Cada Objeto O (pixel) no seleccionado,	GGSA:Movimiento
I. Retornar mejor solución	GGSA

Diagrama SYSLM de parametrización implementación Algoritmo GGSA en R



RASTER *Función que maneja archivos tipo raster*

Descripción:

La función RASTER lee una imagen en diferentes formatos y la tranforma a una estrucutra de datos tipo data.frame.

Uso:

RASTER (directorio)

Argumentos:

Directorio: un *Sting* con la ruta y el nombre de la imagen.

Valores:

Rrow:	Un *integer* con el número de filas de la imagen original.
Rcol:	Un *integer* con el número de columnas de la imagen original.
d:	Un *integer* con el número de bandas de la imagen original.
n_pix:	Un *integer* con el número total de pixeles de la imagen original.
Raster_df:	*Un data.frame* de dimensiones [n_pix,d], con los valores por banda de cada pixel.

Detalles:

La función utilízala la librería *raster* para leer la imagen y almacenarlo en un objeto tipo *brick*, de este son extraídas las características de la imagen original y posteriormente es transformado en un objeto data.frame

Ejemplos:

Raster_df <- RASTER (directorio)

GGSA	*Función principal Algoritmo GGSA*

Descripción:

Función principal, controla la ejecución del proceso de búsqueda hasta que el criterio de terminación es alcanzado, centraliza la ejecución de cada paso del algoritmo invocando las demás funciones, actualiza los valores de las variables locales iteración tras iteración, genera los archivos resultado.

Uso:

GGSA(N,k, max_it , Raster_df, n_pix,d, Rrow, Rcol)

Argumentos:

N:	Un *integer* con el número de soluciones (agentes) de la población.
k:	Un *integer* con el número de grupos a segmentar.
max_it:	Un *integer* con el número máximo de iteraciones.
Raster_df:	Un *data.frame* de dimensiones *[n_pix,d],* con los valores por banda de cada pixel.
n_pix:	Un *integer* con el número total de pixeles de la imagen original.
d:	Un *integer* con el número de bandas de la imagen original.
Rrow:	Un *integer* con el número de filas de la imagen original.
Rcol:	Un *integer* con el número de columnas de la imagen original.

Valores:

BestChart:	Una *list* con el histórico de mejores valores fitness por iteración
best_S;	Una *list* con los valores fitness por agente, indica el mejor agente (solución)

Detalles:

Algoritmo de Búsqueda Gravitacional para Agrupación - GGSA

Genera los siguientes archivos de salida: <<Nombre_imagen>>_GGSA.png, <<Nombre_imagen>>_GGSA.txt, <<Nombre_imagen>>_MEDIDAS_best_S, <<Nombre_imagen>>_MEDIDAS_BestChart,<<Nombre_imagen>>_GGSA_MEDIDAS,<<Nombre_imagen>>_GGSA_MEDIDAS_DB

Ejemplos:

ggsa <- GGSA(N, max_it , Raster_df, n_pix,d, Rrow, Rcol);

Inicialización *Función que generar población inicial*

Descripción:

Esta función genera aleatoriamente una población con N (agentes) soluciones iniciales de segmentación para la imagen procesada. Asigna aleatoriamente cada pixel a un grupo.

Uso:

Inicialización (N,k, Raster_df)

Argumentos:

N:	Un *integer* con el número de soluciones (agentes).
k:	Un *integer* con el número de grupos a segmentar.
Raster_df:	Un *data.frame* de dimensiones [n_pix,d], con los valores por banda de cada pixel.

Valores:

X:	Una *matrix [n_pix,d+N],* con la información de *Raster_df y* N columnas adicionadles correspondientes a cada Solución inicial de segmentación, cada solución contiene la asignación aleatoriamente a cada pixel una clase *k*.

Detalles:

Crea una población de N agentes correspondientes a cada solución que el algoritmo optimiza por media su interacción a través de la ley de gravedad, cada solución inicial corresponde a la asignación aleatoria de un grupo k a cada pixel agregada en una columna.

Ejemplos:

X <- Inicialización (N,k, Raster_df);

Fitness	*Función que generar población inicial*

Descripción:

Esta función calcula el valor de la función objetivo (EMS) para cada agente.

Uso:

Fitness (X)

Argumentos:

X:	Una *matrix [n_pix,d+N],* con la información de *Raster_df y* N columnas adicionadles correspondientes a cada Solución inicial de segmentación, cada solución contiene la asignación aleatoriamente a cada pixel una clase *k*.

Valores:

fitness:	Una *matrix [N,1],* con el valor del resultado de la función objetivo para cada agente

Detalles:

El algoritmo GGSA utiliza como función objetivo de optimización "fitness" el error medio cuadrático Ec. 25:

$$f(O,C) = \sum_{i=1}^{D} \sum_{O_j \in C_i} \|O_j - Z_i\|^2$$

El error es calculado para cada agente y almacenado en una matriz.

Ejemplos:

fitness <- Fitness (X);

Masas	*Función que calcula la masa para cada agente*

Descripción:

Esta función calcula el valor de la función objetivo (EMS) para cada agente.

Uso:

Masas (fitness)

Argumentos:

fitness:	Una *matrix [N,1],* con el valor del resultado de la función objetivo para cada agente

Valores:

M:	Una *matrix [N,1],* con el valor del resultado de la función objetivo para cada agente

Detalles:

Los agentes con mejores valores de fitness tienen masas más pesadas, así mayores atracciones y un movimiento más lento, Ec. 22.

$$M_i(t) = \frac{fit_i(t) - worst\ (t)}{best(t) - worst\ (t)}$$

$$M_i\ (t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_i(t)}$$

Ejemplos:

M <- Masas (fitness);

G	*Función que calcula el valor de la contante gravitacional "G"*

Descripción:

Esta función calcula el valor de la contante gravitacional.

Uso:

G (iteración, max_it)

Argumentos:

Iteracion	Número de la iteracion, **loop** de la función GGSA.
max_it:	Un *integer* con el valor del número máximo de iteraciones.

Valores:

G:	Un *double* con el valor de la contante G para cada iteración.

Detalles:

La constante gravitacional, *G*, se inicializa al principio y se reduce con el tiempo para controlar la precisión de la búsqueda

Ejemplos:

G <- G (iteración, max_it);

CampoG	*Función que calcula el valor de la velocidad de cada agente en el campo de gravedad*

Descripción:

Esta función calcula el valor de la aceleración.

Uso:

Campo*G (M, X, G, iteracion, max_it)*

Argumentos:

M:	Una *matrix [N,1],* con el valor del resultado de la función objetivo para cada agente
X:	Una *matrix [n_pix,d+N],* con la información de *Raster_df y* N columnas adicionadles correspondientes a cada Solución inicial de segmentación, cada solución contiene la asignación aleatoriamente a cada pixel una clase *k.*
G:	Un *double* con el valor de la contante G para cada iteración.
Iteracion	Número de la iteracion, **loop** de la función GGSA.
max_it:	Un *integer* con el valor del número máximo de iteraciones.

Valores:

V: Una *matriz* que contiene los valores de velocidad.

Detalles:

Ordenamiento pares – MWM, Calculo de la fuerza de gravedad y actualización de la velocidad, Ec 29 y 30.

$$v_i^d(t+1) = \quad rand \times v_i^d(t) + G(t) \sum_{j \in Kbest, j \neq i} rand_j \frac{M_j(t)}{Euclidean_j(X_i(t), X_j(t)) + \varepsilon} \times Dist_j(x_j^d(t), x_i^d(t))$$

$$Dist_j(x_i^d(t+1), x_i^d(t)) \approx v_i^d(t+1)$$

Ejemplos:

V <- Campo*G (M, X, G, iteracion, max_it)*

Movimiento	*Función que calcula n (t+1) y actualiza las soluciones X con las fases de herencia e inserción*

Descripción:

Esta función que actualiza la matriz de soluciones X.

Uso:

Movimiento (V)

Argumentos:

V: Una *matriz* que contiene los valores de velocidad.

max_it: Un *integer* con el valor del número máximo de iteraciones.

Valores:

n_1: Un *double* con el valor de la contante G para cada iteración.

X "actualizada": Una *matrix [n_pix,d+N],* con la información de *Raster_df y* N columnas adicionadles correspondientes a cada Solución inicial de segmentación, cada solución contiene la asignación aleatoriamente a cada pixel una clase *k*.

Detalles:

Calcula el valor de $n_i^d(t+1)$ Eq. (31) y Selecciona aleatoriamente $n_i^d(t+1)$ desde el cluster $X_i(t)$ y asignarlos al nuevo cluster $X_i(t+1)$ en la fase de herencia, en la fase de inserción los objetos no seleccionados en la fase de herencia los asigna al en el cluster con el centroide más cercano.

$$n_i^d(t+1) \approx (1 - v_i^d(t+1))\left|x_i^d(t)\right|$$

Ejemplos:

X<- Movimiento (V);

11.1.2. Listado de librerías de R

Librería	Función
raster	Manipulación de objetos tipo raster
clv	Calculo de distancias e índices
clusterSim	Calculo de distancias e índices
clValid	Calculo de distancias e índices
kohonen	Algoritmo Kohonen- SOM
kmeans	Algoritmo K-medias