Clusterig Cosmológico: un enfoque del clustering gravitacional clásico inspidado en la estructura y dinámica del cosmos a gran escala

1º Aitor Castillo-López, 2º Javier Fumanal-Idocin, 3º Javier Fernández y 4º Humberto Bustince
Departamento de Estadística, Informática y Matemática, Universidad Pública de Navarra, (31006) Pamplona, Spain
aitor.castillo@unavarra.es

***Abstract*—En este trabajo proponemos un nuevo enfoque del algoritmo de clustering gravitacional basado en lo que Einstein consideró su "mayor error": la constante cosmológica. De manera similar al algoritmo de clustering gravitacional, nuestro enfoque está inspirado en principios y leyes del cosmos, y al igual que ocurre con la teoría de la relatividad de Einstein y la teoría de la gravedad de Newton, nuestro enfoque puede considerarse una generalización del agrupamiento gravitacional, donde, el algoritmo de clustering gravitacional se recupera como caso límite. Además, se desarrollan e implementan algunas mejoras que tienen como objetivo optimizar la cantidad de iteraciones finales, y de esta forma, se reduce el tiempo de ejecución tanto para el algoritmo original como para nuestra versión.**

***Index Terms*—cosmos, clustering, no supervisado, simulación, fuerza gravitacional**

I. Introducción

EL CLUSTERING, como parte del aprendizaje no supervisado, es de vital importancia para la minería de datos y la extracción de conocimiento a partir de estos. Conocer los datos y su posible estructura puede darnos información sobre el proceso subyacente de los datos en sí. Existe una gran cantidad de datos sin etiquetar accesibles a través de Internet y también es común encontrar información incompleta, faltante o mal recopilada.

Es por tanto de interés en la comunidad científica el análisis de los datos sin etiquetar y el por qué de que exista una amplia literatura orientada a realizar la extracción de conocimiento de los mismos. Concretamente, el clustering consiste en agrupar observaciones en diferentes grupos denominados "clusters", y estos pueden agruparse de acuerdo con algunos criterios, para maximizar la similitud dentro de dichas agrupaciones y minimizar la similitud entre estas. Dependiendo del algoritmo de agrupación, es posible que sea necesario especificar el número de clusters, así como otros parámetros relacionados con los datos. La extensa lista de algoritmos de clustering pueden dividirse en dos clases; sean algoritmos en los que se inicialmente se establece el número de clusters (o no-jerárquicos): k-means, k-medoids, hclust, pam, mona... Y en los que no es necesario establecer un número fijo de grupos de antemano (jerárquicos): agnes, diana... Además, el conocimiento y la influencia de diferentes ramas de la ciencia pueden inspirar a investigadores para modificar y basar algoritmos de clustering en reglas de la naturaleza tales como las biológicas o leyes físicas como es el algoritmo de clustering gravitacional [8].

La esencia del clustering gravitacional es imitar las leyes de la gravedad de Newton para atraer partículas (datos) entre sí y formar los grupos debido a la fuerza de atracción entre ellas. Esta fuerza de atracción hace que las partículas se acerquen unas a otras hasta que dos o más partículas estén lo suficientemente cerca como para fusionarlas todas en un único punto. Hasta este punto, casi todas las generalizaciones de este tipo de clustering jerárquico coinciden.

A partir de este punto, hay muchas formas diferentes de realizar el algoritmo. Es posible estudiar el papel de las masas en el algoritmo a través de funciones de superposición [1], así como cambiar la suma de las masas en colisión por otros valores como el maximo [4], etc. Pero también es posible estudiar cómo la naturaleza en el la fuerza subyacente entre las partículas puede hacer prosperar nuevos algoritmos de clustering como, por ejemplo, el enfoque de agrupamiento basado en dinámica molecular [3].

El modelo de clustering gravitacional original tiene ciertos problemas. Uno de ellos está relacionado con el hecho de que no siempre es correcto seleccionar la configuración más larga en tiempo como la más estable. Para los clusters en un entorno ruidoso esto no es cierto por el truncamiento del tiempo de vida de configuraciones concretas debido a la incorporación de los outliers los clusters. El segundo problema, el cual también se intenta abordar con este trabajo, es que el modelo original genera un problema de tipo "agujero negro". La formación de grupos de acuerdo con la fuerza de gravitación clásica produce objetos extremadamente masivos que impregnan el espacio de los datos con un campo tan fuerte que ignora las fuerzas entre otros objetos de baja masa. A estos puntos formados por una gran cantidad de datos bien se les podría denominar agujeros negros debido a su similitud con el término acuñado por John Wheeler [6] para referirse a los masivos cuerpos de los que ni los fotones pueden escapar. Por lo tanto, los datos tienden a ser atraídos por estos clusters más poblados o de alta densidad. Si en un conjunto de datos hay una gran diferencia entre las densidades de los clusters, un cluster densamente poblado puede convertirse en un agujero negro y atrapar otras partículas que no debiera (outliers) o incluso puede absorber otros clusters no tan masivos pero sí relevantes.

Volviendo a términos de física, el modelo gravitacional de Newton puede describir casi toda la dinámica en la escala del sistema solar. Pero, a la inversa, cuando uno intenta describir las trayectorias de los objetos en una escala mayor como el espacio interestelar o intergaláctico, las leyes clásicas fallan al intentar predecir dichos movimientos. Como analogía, este estudio pretende emular los clusters como galaxias y tratar el espacio de datos como un pequeño cosmos. Siendo así, al igual que en 1917 Einstein añadió la constante cosmológica a sus propias ecuaciones en un intento de unificar tanto la escala planetaria como la cosmológica [2], incluimos igualmente una fuerza de expansión a la fórmula clásica de Newton que puede, en cierto grado, corregir los problemas antes mencionados.

Finalmente, simplemente decir que a lo largo del trabajo y para acompañar las modificaciones teóricas propuestas, presentamos algunos ejemplos ilustrativos para mostrar la validez de nuestro enfoque. En la sección II explicamos el algoritmo original y su mecánica y en la sección III presentamos el fundamento y la formulación de nuestro nuevo enfoque. En la sección IV presentamos la experimentación y los resultados obtenidos al aplicar nuestro método en varios conjuntos de datos. Finalmente, en la sección V resumimos y concluimos con algunas observaciones.

II. Preliminares: Algoritmo de clustering gravitacional original

El algoritmo de clustering gravitacional se inspira en las leyes gravitacionales de Newton para el proceso de categorización de los datos. La mecánica se basa en mover las "partículas" que componen el dataset inicial entre sí de acuerdo con una relación gravitacional concreta, uniendo así unas partículas con otras a lo largo del tiempo para formar grupos que terminan por agruparse en un único punto (en teoría el centro de masas). Concretamente, para n partículas que denotaremos $p_1, p_2 ... p_n$ se encontrarán en las posiciones $s_1, s_2 ... s_n \in \mathbb{R}^N$ en un espacio N dimensional.

Este configuración de datos inicial se deja evolucionar siguiendo los pasos del Algoritmo1 como sistema. A lo largo de la vida de este, cada partícula del sistema corresponde a una categoría, siendo al principio tantas categorías como partículas y finalmente obteniendo una única categoría. La duración de todo el proceso se denota por T e implica lo siguiente:

- Desde $t_n = 0$ hasta t_{n-1} hay n (todas) las partículas.
- Desde t_{n-1} a t_{n-2} quedan $n-1$ partículas.
- ...
- De t_2 a $t_1 \equiv T$ quedan 2 partículas.

La vida relativa de una configuración con c clusters (categorías) se puede calcular como;

$$\frac{R_c = t_{c-1} - t_c}{T,}$$

donde la configuración más larga en el tiempo, interpretada como la más estable, determina el número de clústeres c.

Algoritmo 1: Clustering gravitacional de Wright

Entrada: Un dataset con n datos de N variable
Salida: Un dendrograma del proceso de unión
Inicialmente se asigna una masa $m_i = 1$ a cada partícula p_i donde $i \in [2, n]$;
Se escogen los parámetros positivos reales ϵ y δ;

- Se utiliza δ para determinar la longitud del paso de tiempo simulado entre iteraciones dt. Concretamente, en el tiempo transcurrido $[t, t + dt]$ la partícula más rápida se mueve δ.
- Si hay dos partículas a una distancia menor que ϵ, se unifican en una con la masa igual a la suma de las masas de ambas partículas en colisión y se establece la nueva posición en su centro de masas.

El tiempo se inicializa a $t = 0$;
mientras $n > 1$ **hacer**

1) En cada lapso temporal $[t, t + dt]$, para cada partícula p_i calculamos su función de influencia de movimiento:
$$g(i, t, dt) = \frac{1}{2} G \sum_{j \ neq i} \frac{m_i m_j}{m_i} \frac{s_j - s_i}{|s_j - s_i|^3} dt(t)^2 \quad (1)$$
donde G es una constante positiva. Nótese que el intervalo de tiempo dt no es una constante sino que depende de la iteración.
2) Para cada p_i, su nueva posición será:
$$s_i(t + dt) = s_i(t) + g(i, t, dt)$$
3) Elevamos t a $t + dt$.
4) Si dos partículas p_i y p_j donde $j \in [2, n]$, están a una distancia menor que ϵ, su unión se realiza como se explica arriba.
5) Se categorizan las partículas unídas bajo la misma etiqueta para construir el dendrograma.
6) Se reasigna n.

Una vez se detiene el bucle, se evalua el historial del sistema (dendrograma) para encontrar la configuración más estable en el tiempo.

III. Enfoque de la dinámica cosmológica

Como comentamos en la sección I, nuestro enfoque está inspirado en la constante cosmológica de Eistein. Es bien sabido por los astrónomos y cosmólogos que Einstein incluyó el parámetro Λ como un término en sus ecuaciones de campo porque sus ecuaciones no permitían, aparentemente, un universo estático: la gravedad haría que un universo que inicialmente estaba en equilibrio dinámico se contrajera [7].

De manera análoga, nosotros introducimos un factor Λ en la Eq. 1 del algoritmo de clustering gravitacional para tratar de maximizar la vida de una configuración estacionaria. No obstante, en primer lugar, se debe generalizar la notación la cual se explica brevemente en la siguiente subsección para abordar nuestra propuesta.

A. Generalización para algoritmos de simulación con propiedad de Markov basados en fuerzas isótropas

En la teoría de probabilidad es conocido como modelo o cadena de Márkov a un tipo de proceso estocástico discreto en el que la probabilidad de que ocurra un evento depende solamente del evento inmediatamente anterior. Esta característica de "falta de memoria" recibe el nombre de propiedad de Markov [5].

Si bien en nuestro caso las configuraciones que toman las partículas del sistema en cada paso no son estocásticas y están completamente determinadas por la fuerza percibida por cada una de ellas, la propiedad de Markov viene al dedo para describir todas aquellas simulaciones donde la configuración de un sistema en un tiempo concreto t está unicamente determinada por la configuración anterior en $t - dt$.

Ahora bien, es posible generalizar este "paso" gravitacional del Algoritmo 1 como paso de Markov para cualquier otro tipo de fuerza simulada que dependa de la posición, masa, carga, *flavour*, *spin*, *color* o cualquier otra propiedad relacionada con la partícula. No obstante, se debe remarcar que dicha fuerza simulada no puede depender de propiedades que requieran guardar información de estados previos del sistema dado que el cálculo del nuevo paso no se debe hacer usando información previa al estado actual en el caso que nos atañe.

Entonces, para cualquier fuerza f compatible con la propiedad de Markov, definimos f_{ij} como la magnitud de la fuerza que p_j ejerce sobre p_i. Dicho esto, el paso de Markov $\Delta s_i(t)$ para la partícula p_i en el momento t debería ser;

$$\Delta s_i(t) = \frac{1}{2}\frac{1}{m_i(t)} \sum_{j \neq i} f_{ij}(s_i(t), s_j(t), ...)\vec{n}_{ji}(t)dt(t)^2 \quad (2)$$

donde por definición

$$\Delta s_i(t) \equiv s_i(t + dt) - s_i(t)$$

$$\vec{n}_{ji}(t) \equiv \frac{s_i(t) - s_j(t)}{|s_i(t) - s_j(t)|}$$

tal que, para todas las fuerzas que tengan un factor de dependencia radial, es decir,

$$r_{ij}(t) \equiv |s_j(t) - s_i(t)|\,,$$

finalmente resulta

$$\Delta s_i(t) = \frac{1}{2}\frac{1}{m_i(t)} \sum_{j \neq i} f_{ij}(r_{ji}(t), ...)\vec{n}_{ji}(t)dt(t)^2. \quad (3)$$

B. Fuerza de expansión cosmológica para clustering

Generalizado el paso de Markov en la Eq. 2 supondremos que las únicas propiedades que afectan a la magnitud de la fuerza f en nuestra constribución son las masas m_i y m_j, las distancias entre partículas r_{ij} y la cantidad de clusters c que quedan en el sistema en el momento t.

Estas propiedades construyen la fuerza que, al ser atractiva inicialmente, comienza a ser repulsiva con el paso del tiempo. El objetivo es fusionar inicialmente los clusters, pero, cuando estos hayan crecido lo suficiente, se pretende hacer que sientan una fuerza repulsiva entre sí de tal manera que se favorezca una configuración de equilibrio donde los clusters formados nunca se encuentren.

Para alcanzar este propósito, la fuerza inicialmente debe ser la fuerza gravitacional original, pero a menor número de cúmulos es necesario que sea una fuerza más expansiva. Además, el término de expansión propuesto debe depender de las distancias de tal manera que, cuanto mayor sea la distancia entre las partículas del sistema, más fuerte será la repulsión entre estas. Entonces se propone;

$$f_{ij} = -G\frac{m_i m_j}{r_{ij}^2} + k_c r_{ij}, \quad (4)$$

donde,

$$k_c = \Lambda\left(1 - \frac{c(t)}{n}\right) m_i(t) m_j(t), \quad (5)$$

y $c(t)$ es el número de categorías que quedan en el espacio de simulación en el momento t y n es el número (inicial) de clusters (tantos como datos al inicio), es decir, $c(0) = n$.

Haciendo algunas simplificaciones, el paso de Markov se puede definir como:

$$\Delta s_i(t) = \frac{1}{2}\sum_{j \neq i} m_j(t)\left[-\frac{G}{r_{ij}^2} + \Lambda\left(1 - \frac{c(t)}{n}\right) r_{ij}\right]\vec{n}_{ji}dt(t)^2 \quad (6)$$

C. Reformulación del algoritmo: el parámetro δ

En primer lugar hay que decir que en nuestro enfoque se hace una reformulación del parámetro δ que puede ser aplicada en el original. Es decir, esta reformulación no es una reformulación *ad hoc* que necesita el enfoque cosmológico, todo lo contrario, la reformulación de este parámetro también se puede hacer en la versión original y en cualquier enfoque de algoritmo gravitacional. En resumen, es una modificación para mejorar este tipo de algoritmos generalizados en la Subsección III-A.

La reformulación es la siguiente. El paso de la distancia máxima δ se propone no ser un parámetro constante como lo es en la versión original. En nuestro enfoque, δ será la mitad de la distancia entre la partícula más rápida y su vecino más cercano. En base a este criterio, nos aseguramos de que cuando la fuerza es muy atractiva, la partícula más rápida no pueda intercambiar su posición con su vecina más cercana (ver Fig. 1). Entonces con esta reformulación podemos decir que:

(P1) Si dos partículas están muy cerca una de la otra, y la fuerza que sienten muy atractiva, el paso no puede ser demasiado grande para separarlas y por ende, se ven obligadas a fusionarse.

No obstante, en nuestro enfoque también hay fuerzas repulsivas así que expliquemos qué sucede cuando aplicamos el criterio anterior en otras situaciones.

Cuando comienzan a aparecer fuerzas repulsivas es debido a que la relación $c(t)/n$ es pequeña (ver Eq. ref eq: cosmomark), entonces, se tienen puntos masivos separados cada uno de los otros a grandes distancias y probablemente rodeados de partículas ligeras. En este caso, las partículas

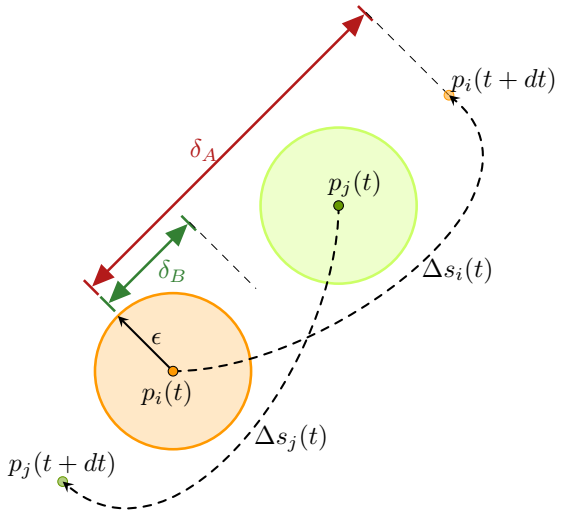


Fig. 1: En esta figura $p_i(t)$ y $p_j(t)$ son la partícula más rápida y su vecino más cercano en un tiempo arbitrario t sintiendo una gran fuerza atractiva entre sí. Si δ es constante y demasiado grande (caso δ_A), las partículas no se pueden fusionar correctamente y pueden intercambiar sus posiciones durante muchas iteraciones. Si definimos δ en base al criterio explicado (caso δ_B), nos aseguramos de que las dos partículas se fusionen.

circundantes hacen que el valor δ sea pequeño, lógicamente, porque la distancia entre estas partículas circundantes y su correspondiente grupo es menor que la distancia entre los grupos.

Entonces, estas partículas ligeras se moverán a posiciones donde estarán más cerca a sus correspondientes clusters que antes, y la fuerza que sentirán será cada vez más atractiva. Pero, no intercambiarán sus posiciones debido a los argumentos explicados anteriormente. Entonces:

(P2) Si hay partículas masivas rodeadas por otras partículas ligeras, el valor de δ será pequeño y además, en dicha situación, los pasos no pueden ser demasiado grandes para alejar la partícula ligera circundante de la masiva (o categoría) a la que pertenece.

Por lo tanto, dadas (P1) y (P2), los estados del sistema siempre van a converger a la configuración donde solo quedan partículas masivas. En este último caso, la distancia entre las partículas es grande, por lo que δ será grande, lo que hará que el sistema explote.

D. Derivación del tiempo de vida de la configuración para el instante t: el valor Δt_c

Debido al criterio anterior para δ, para cada iteración o lapso de tiempo corresponderá un valor de dt derivado de δ: ambos, dependen del tiempo y deben calcularse en cada iteración.

En el momento t, supongamos la partícula más rápida p_{fast} la i -ésima donde se cumple $p_{fast} = p_i$ donde $i = argmax(\Delta s_1(t), \ldots, \Delta s_i(t), \ldots)$, por lo que, una vez detectada esta partícula, se deben calcular tres valores:

- La distancia al vecino más cercano d_{min}, δ y dt para esa iteración.

Por definición y teniendo en cuenta la Eq. 3:

$$\delta \equiv \frac{d_{min}}{2} = \frac{1}{2}|\vec{a}_{fast}|dt^2$$

donde, a_{fast} es la suma de las contribuciones a la aceleración sobre la partícula más rápida, explícitamente;

$$a_i = \sum_{j \neq i} m_j(t) \left[-\frac{G}{r_{ij}^2} + \Lambda \left(1 - \frac{c(t)}{n} \right) r_{ij} \right] \tag{7}$$

luego, es inmediato deducir dt como

$$dt = \sqrt{\frac{d_{min}}{a_{fast}}}. \tag{8}$$

Este dt es muy fácil de calcular y no tiene costo computacional. Una vez se calcula en cada iteración, debe acumularse hasta que se absorba una partícula o, en otras palabras, hasta que cambie la cantidad de partículas $c(t)$. Entonces, definimos la cantidad de tiempo Δt_c donde el sistema ce mantiene con un número c de categorías(o partículas).

$$\Delta t_c \equiv t_{c-1} - t_c = \sum_{t_c}^{t_{c-1}} dt(t) \tag{9}$$

Como condición, si la acumulación de dt excede un valor arbitrario, el algoritmo se ve obligado a parar reconociendo que el sistema explota:

$$\Delta t_c > N \sqrt{\frac{2\left(\frac{D}{10}\right)^3}{3Gn}}, \tag{10}$$

donde, D es la medida de la diagonal del hipercubo N-dimensional que ocupan los datos al principio.

E. Reformulación del algoritmo: el parámetro ϵ

Por otro lado, hemos redefinido el parámetro ϵ. Dicho parámetro va a representar el volumen 3D de la partícula en lugar de ser un parámetro fijo. Este volumen 3D, al igual que la masa, se va a conservar en todo el sistema a lo largo del tiempo. Entonces, cuanto mayor es la masa de una partícula, mayor debe ser su valor ϵ. La regla de la transferencia de volumen es la siguiente. Si en una particula p_j es absorbida por p_i, debido a la conservación del volumen:

$$\frac{4}{3}\pi\epsilon_{new}^3 = \frac{4}{3}\pi\epsilon_a^3 + \frac{4}{3}\pi\epsilon_b^3$$

se obtiene que

$$\epsilon_i(t+dt) = \sqrt[3]{\epsilon_i^3(t) + \epsilon_j^3(t)}, \tag{11}$$

y fijando el valor ϵ_{ini} inicial como una fracción de la distancia entre una partícula aleatoria y su vecino más cercano:

$$\forall i \in [2, n], \epsilon_i(0) = Rand(d_{min})/N \tag{12}$$

finalmente podemos exponer la estructura del algoritmo que proponemos.

Pero, antes de eso, tenemos que decir que la elección de la regla de conservación de volumen 3D y no el 4D o N-D volumen no es arbitraria. Esto se debe a que si nos hemos inspirado en la ley de Gravitación Universal (GU) y las reglas de GU se basan en un universo 3D, estamos obligados a probar esta dimensionalidad al principio. La exploración de otras reglas de dimensionalidad pertenece a otro trabajo o línea futura.

F. Algoritmo de clustering cosmológico

Algoritmo 2: Clustering Cosmológico

Entrada: Un dataset con n datos de N variable
Salida: Una configuración del sistema
Inicialmente asignamos una masa $m_i = 1$ y $\epsilon_i(0)$ de acuerdo con la Eq. 12 a cada partícula p_i;
El tiempo se inicializa a $t = 0$;
Unificación Previa al Ciclo (UPC): Se comprueba qué partículas deben unificarse debido a la elección de ϵ_i inicial. Si dos partículas i y j (o más) están a una distancia menor que $\epsilon_i(0) + \epsilon_j(0)$, su unificación se realiza de acuerdo con la Eq. 11, otorgando a p_i la masa de p_j y eliminando esta última;
mientras $n > 1$ *AND Eq. 10 is fulfilled* **hacer**

1) En la iteracion correspondiente a t, es decir, el lapso $[t, t + dt]$, para cada p_i se calcula su vector de influencia de movimiento $\Delta s_i(t)$ dado por la Eq. 6, donde G y Λ son constantes positivas.
2) Para cada p_i, su nueva posición será:

$$s_i(t + dt) = s_i(t) + \Delta s_i(t)$$

   Estos dos primeros items componen el paso de Markov.
3) Se eleva t a $t + dt$.
4) Si dos partículas i y j están a una distancia menor que $epsilon_i(t) + epsilon_j(t)$, su unificación se realiza de la misma manera que en la UPC.
5) Se categorizan las partículas unídas bajo la misma etiqueta para construir el dendrograma.
6) Se reasigna n.

Una vez se detiene el bucle, evaluamos el historial del sistema (dendrograma) para encontrar la configuración más estable de clústeres. La configuración elegida es la que abarque mayor Δt_c.

Como en el algoritmo original, cada partícula del sistema corresponde a una categoría, no obstante, la duración de todo el proceso depende de si el sistema converge o explota.

Para concluir esta seción, en la Fig. 2 se muestra un esquema del algoritmo propuesto.

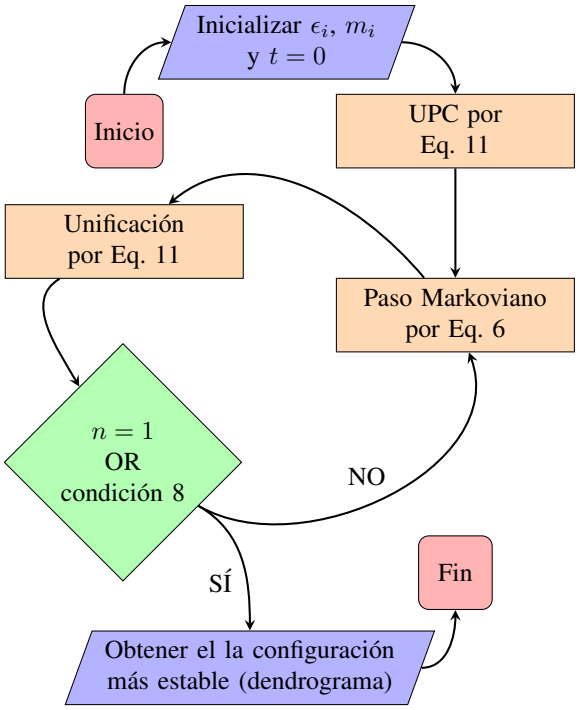
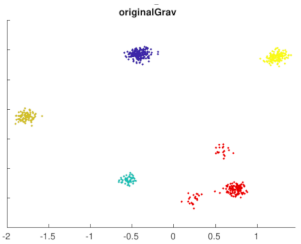
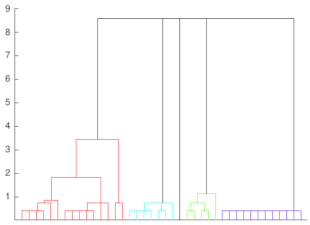


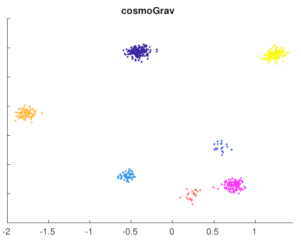
Fig. 2: Esquema del Alg. 2.



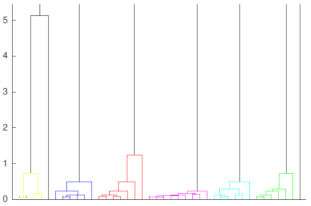
(a) Etiquetado final de clusters generados de manera aleatoria por el algoritmo gravitacional clásico de Wright.



(b) Dendrograma generado por el Alg. 1 para el dataset inicialmente compuesto por seis clusters de la izquierda.

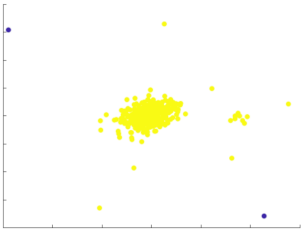


(c) Etiquetado final de los mismos clusters de Fig. 3a pero por el algoritmo cosmológico propuesto en este trabajo.

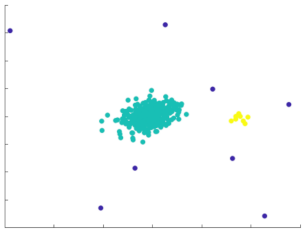


(d) Dendrograma generado por el Alg. 2 para el dataset inicialmente compuesto por seis clusters de la izquierda.

Fig. 3: Diferencias de etiquetado (izquierda) y dendrogramas (derecha) entre los Algoritmos 1 (arriba) y 2 (abajo).

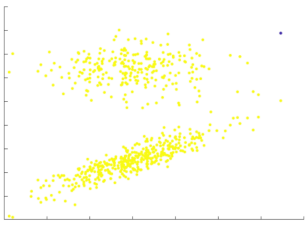


(a) El algoritmo de Wright no consigue diferenciar los clusters ni los outliers.

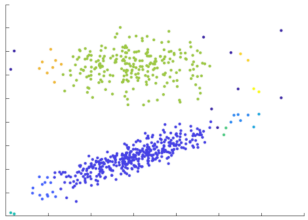


(b) Nuestro algoritmo es capaz de detectar tanto los clusters como los outliers.

Fig. 4: Comparación entre los Algoritmos 1 y 2 para un dataset bidimensional con outliers.



(a) Etiquetado de los datos según el algoritmo gravitacional clásico de Wright.



(b) Etiquetado de los datos según el algoritmo cosmológico propuesto en este trabajo.

Fig. 5: Dataset bidimensional compuesto por dos clusters gaussianos próximos entre sí.

IV. Ejemplos Ilustrativos

En esta sección se recogen tres ejemplos ilustrativos de clasificación en datasets bidimensionales para los diferentes algoritmos.

El primer conjunto de datos sobre el que se han testado los algoritmos es el grupo de clusters distinguibles que aparece en las Figuras 3a y 3c. Acompañando a estas imágenes se muestran también los dendrogramas generados por ambas simulaciones (Figuras 3b y 3d). En este claro ejemplo de la Fig. 3, la configuración más estable en el tiempo de nuestro algoritmo permite clasificar de diferente manera los clusters que se encuentran más cerca entre sí, mientras que son indistinguibles para el algoritmo original (en rojo en la Fig. 3a).

Por otro lado, se ha construido un segundo dataset para la comparación de los algoritmos en entornos con ruido. Posteriormente, se han puesto ambos algoritmos a prueba en el pequeño dataset de la Fig. 4. Es notorio como mientras nuestro algoritmo es capaz de identificar la mayoría de partículas de ruido (Fig. 4b), el original no puede (Fig. 4a).

Por último, se ha construido un tercer dataset que pudiera ser más probable encontrar en problemas de clasificación reales. Este está compuesto por dos clusters de distribución gaussiana que contienen puntos que bien podrian pertenecer a ambas categorías debido a su proximidad (Fig. 5). Como se puede apreciar en la Fig. 5a, el Alg. 1 no ha sido capaz de diferenciar entre ambas clases. Esto es debido a la continua absorción de partículas individuales por parte de los clusters más grandes. Este hecho hace que las configuraciones que pudieran parecer las más estables (más largas en el tiempo simulado), en realidad son truncadas constantemente por la integracion de nuevas partículas a la clase, y finalmente, ambas clases terminan por juntarse. Por otro lado, el Alg. 2 es capaz de separar dos clases *grosso modo* debido a la fuerza repulsiva que perciven las partículas masivas entre sí. Como resultado final se puede apreciar que los extremos de los clusteres son catalogados con diferentes etiquetas creando así pequeños sub-clusters.

V. Conclusiones y líneas futuras

Las mejoras más notorias de nuestra propuesta son;

1) Poder catalogar clusters próximos
2) Que estén inmersos entornos con ruido.

No obstante, poder aunar como una única clase clusters que tengan formas muy alargadas, ya estén bien diferenciados del resto, es un reto que no se consigue completar. Este problema se debe a la construcción del Alg. 2 *per se* ya que se basa en fuerzas radiales. Es por ello que se deberá investigar la implementación de fuerzas no isótropas o de diferentes métricas que distorsionen el espacio.

Finalmente decir que el algoritmo se ha probado en datasets bidimensionales por lo que queda pendiente analizar cómo este se adapta a datasets de mayor dimensionalidad y evaluarlo en un benchmark junto con otros algoritmos de clustering.

Agradecimientos

Este trabajo ha sido respaldado por los proyectos PID2019-108392GB-I00 (AEI/10.13039/ 501100011033) de la Agencia Estatal de Investigación.

References

[1] Armentia, J., Rodríguez, I., Idocin, J. F., Bustince, H., Minárová, M., & Jurio, A. (2019, July). Gravitational clustering algorithm generalization by using an aggregation of masses in newton law. In International Summer School on Aggregation Operators (pp. 172-182). Springer, Cham.

[2] "Kosmologische Betrachtungen zur allgemeinen Relativitätstheorie" ("Cosmological considerations in the General Theory of Relativity"). Consultado el 5 de mayo de 2021.

[3] Blekas, K., & Lagaris, I. E. (2007). Newtonian clustering: An approach based on molecular dynamics and global optimization. Pattern Recognition, 40(6), 1734-1744.

[4] Fumanal-Idocin, J., Alonso-Betanzos, A., Cordón, O., Bustince, H., & Minárová, M. (2020). Community detection and social network analysis based on the Italian wars of the 15th century. Future Generation Computer Systems, 113, 25-40.

[5] Geyer, C. J. (1992). Practical markov chain monte carlo. Statistical science, 473-483.

[6] Tyson, N. D., Strauss, M., & Gott, J. R. (2016). Welcome to the universe: an astrophysical tour. Princeton University Press. Chapter 20

[7] Rugh, S. E., & Zinkernagel, H. (2002). The quantum vacuum and the cosmological constant problem. Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics, 33(4), 663-705.

[8] Wright, W. E. (1977). Gravitational clustering. Pattern recognition, 9(3), 151-166.