

PRÁCTICA 2

TÉCNICAS DE BÚSQUEDA BASADAS EN POBLACIONES

**PARA EL PROBLEMA DE AGRUPAMIENTO
CON RESTRICCIONES (PAR)**

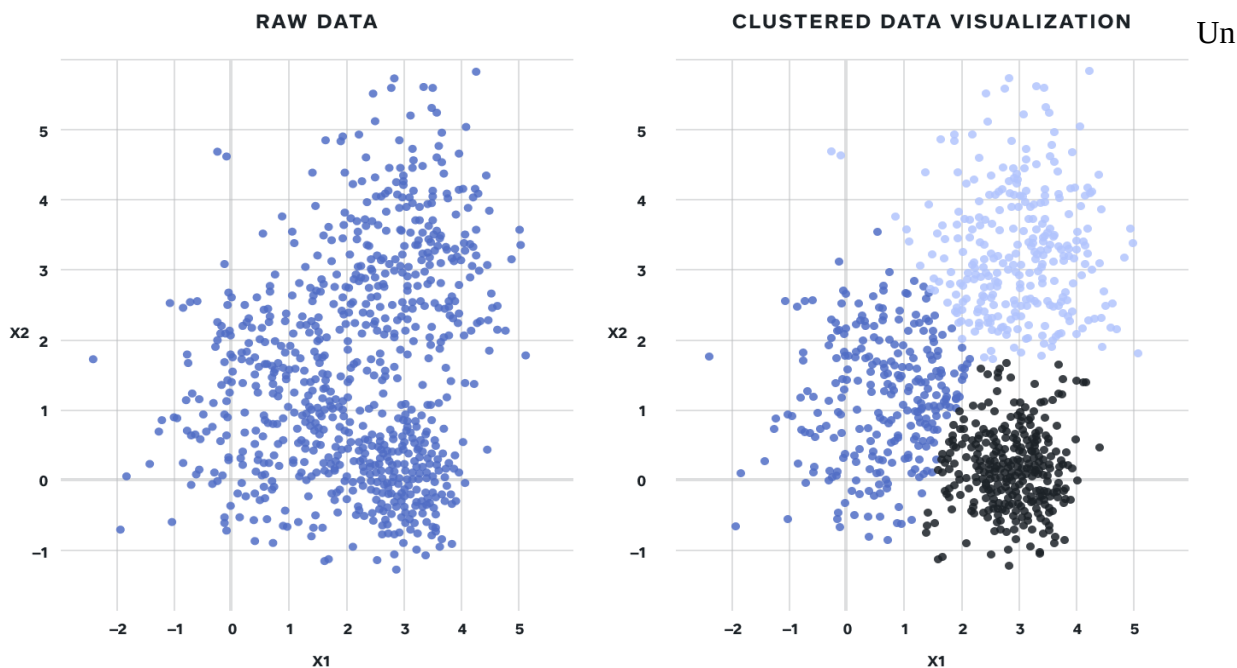
**Antonio José Blázquez Pérez
3ºCSI METAHEURÍSTICAS
45926869D**

ÍNDICE

1. DESCRIPCIÓN DEL PROBLEMA.....	3
2. FORMALIZACIÓN Y DESCRIPCIÓN DE LA APLICACIÓN DE LOS ALGORITMOS.....	4
2.1 FORMALIZACIÓN DE CONCEPTOS.....	4
2.2 REPRESENTACIÓN DE LA INFORMACIÓN.....	5
2.3 FUNCIÓN OBJETIVO.....	6
2.4 DATAFRAMES.....	8
3. ALGORITMOS GENÉTICOS.....	8
3.1. DESCRIPCIÓN.....	8
3.2 OPERADORES DE SELECCIÓN.....	9
3.3 OPERADORES DE CRUCE.....	9
3.4 OPERADOR DE MUTACIÓN.....	10
3.5 OPERADORES DE REEMPLAZAMIENTO.....	10
3.6 AGG Y AGE.....	11
4. ALGORITMO MEMÉTICO.....	12
4.1 ESTUDIO PRELIMINAR DE LOS AG.....	12
4.2 BÚSQUEDA LOCAL SUAVE.....	13
4.3 VARIANTES DE AM.....	13
5. PROCEDIMIENTO CONSIDERADO PARA EL DESARROLLO DE LA PRÁCTICA.....	14
5.1 CÓDIGO Y LIBRERÍAS USADAS.....	14
5.2 MANUAL DE USUARIO.....	14
6. ANÁLISIS DE RENDIMIENTO.....	15
6.1 DESCRIPCIÓN DE LOS CASOS DEL PROBLEMA.....	15
6.2 CUESTIONES SOBRE BL Y COPKM.....	16
6.3 PROBLEMÁTICA.....	17
6.4 RESULTADOS OBTENIDOS.....	17
6.5 ANÁLISIS DE RESULTADOS.....	21
7. BIBLIOGRAFÍA.....	23

1. DESCRIPCIÓN DEL PROBLEMA

El agrupamiento o clustering busca clasificar objetos de un conjunto en subconjuntos o clusters a través de sus posibles similitudes. Es una técnica de aprendizaje no supervisado que permite descubrir grupos inicialmente desconocidos o agrupar objetos similares, de manera que podemos encontrar patrones en grandes grupos de datos que serían imposibles(o extremadamente difíciles) de encontrar sin esta técnica. Una muestra de agrupamiento sería la siguiente:



ejemplo cotidiano del problema del agrupamiento sería dividir una cantidad de frutas en verdes, poco maduras y muy maduras que, aunque trivial, nos permite entender el concepto que hay detrás de la técnica que nos ocupa. Con ésto podemos comprender otras aplicaciones del clustering, como pueden ser el análisis de caracteres escritos a mano, muestras de diálogo, huellas dactilares o imágenes, la clasificación de especies en subespecies o el agrupamiento para moléculas o proteínas.

Para poder aplicar esta técnica debemos medir d características de nuestro grupo de n objetos, por ejemplo el color o la textura en el caso de las frutas o la simetría o la intensidad en el caso de caracteres escritos a mano, obteniendo una conjunto de datos de longitud n con d dimensiones.

En nuestro caso concreto abordaremos una variación del clustering clásico, el Problema de Agrupamiento con Restricciones(PAR), es decir, además de nuestro conjunto de datos tenemos cierta información a la que llamaremos restricciones o, más concretamente, restricciones de instancia. Ésto quiere decir que tenemos alguna información sobre objetos que tienen que pertenecer al mismo cluster(ML, Must-Link) y sobre objetos que no pertenecen al mismo cluster(CL, Cannot-Link), siendo éstas restricciones débiles, o lo que es lo mismo, podemos incumplir restricciones pero debemos minimizar el número de incumplidas al máximo.

El planteamiento combinatorio de este problema es NP-Completo, por tanto a continuación propondremos algunas alternativas de algoritmos para intentar resolver el problema de forma aproximada en un tiempo razonable. El problema se abordará con k (nº de clusters) conocida, ya que su búsqueda es otro problema complejo.

2. FORMALIZACIÓN Y DESCRIPCIÓN DE LA APLICACIÓN DE LOS ALGORITMOS

2.1 FORMALIZACIÓN DE CONCEPTOS

Vamos a definir distintos conceptos que se nombrarán a lo largo del documento y que son necesarios para entender los procedimientos en cada algoritmo.

Primeramente podemos formalizar la definición de nuestro conjunto de datos como una matriz X de $n \cdot d$, n objetos en un espacio de d dimensiones (el número de medidas que tenemos sobre cada objeto). Lo podemos notar matemáticamente como:

$$\vec{x}_i = \{x_{[i,1]}, \dots, x_{[i,d]}\} / x_{[i,d]} \in \mathbb{R} \forall j \in \{1, \dots, d\}$$

Llamaremos $C = \{c_1, \dots, c_k\}$ al conjunto de los k clusters, de manera que cada c_i será un subconjunto de X y podrá asociada una etiqueta l_i que lo nombre. Para cada cluster es posible calcular su centroide asociado $\vec{\mu}_i$, siendo el vector promedio de sus instancias, de la siguiente manera:

$$\vec{\mu}_i = \frac{1}{|c_i|} \sum_{\vec{x}_j \in c_i} \vec{x}_j$$

También debemos definir la distancia media intra-cluster, \bar{c}_i , como la media de las distancias entre cada instancia del cluster y su centroide asociado, usando en este caso la distancia euclídea, aunque es equivalente a usar, por ejemplo, la distancia Manhattan. Es calculable con la siguiente expresión:

$$\bar{c}_i = \frac{1}{|c_i|} \sum_{\vec{x}_j \in c_i} \|\vec{x}_j - \vec{\mu}_i\|_2$$

A través de estas, podemos llegar a la desviación general de C , la media de las desviaciones intra-cluster, que nos será de gran ayuda para minimizar la solución y para el posterior análisis de la solución. La expresión sería la siguiente:

$$\bar{C} = \frac{1}{k} \sum_{c_i \in C} \bar{c}_i$$

Por otro lado tenemos las restricciones, $R = ML \cup CL$, donde $ML(\vec{x}_i, \vec{x}_j)$ indica que las instancias \vec{x}_i y \vec{x}_j deben estar asignadas al mismo cluster y $CL(\vec{x}_i, \vec{x}_j)$ que las instancias (\vec{x}_i, \vec{x}_j) no pueden ser asignadas al mismo cluster. Con ello, definimos la infactibilidad o infeasibility, que es un indicador de las restricciones que incumple nuestra solución, como:

$$infeasibility = \sum_{i=0}^{|ML|} 1(h_c(\vec{ML}_{[i,1]}) \neq h_c(\vec{ML}_{[i,2]})) + \sum_{i=0}^{|CL|} 1(h_c(\vec{CL}_{[i,1]}) = h_c(\vec{CL}_{[i,2]}))$$

Siendo 1 la función booleana que devolverá 1 si la expresión que toma como argumento es verdadera y 0 en otro caso.

2.2 REPRESENTACIÓN DE LA INFORMACIÓN

Ahora que tenemos algunos conceptos claros y formalizados, pasamos a ver como representaremos la información necesaria en el proceso de resolución del problema.

Primero aclarar que los datos de entrada, nuestro dataframe, estará organizado en una matriz implementada como un vector<vector<float>> (ambos vectores de la librería STL), es decir, un vector de longitud n de vectores de reales en coma flotante de longitud d.

Para las restricciones usaremos dos formas de representación, una para facilitar el acceso cuando conocemos la restricción que queremos comprobar y otra para cuando queramos recorrer todas las restricciones. Para el primer caso se usará la misma representación que para el dataframe, mientras que para el segundo se construirá una lista con elementos del tipo [x,y,{1,-1}], siendo x e y los dos elementos que tiene la restricción y el último elemento 1 si conforman una restricción ML y -1 si es CL. Esta lista se ha implementado usando un vector<vector<int>>, también de la STL. De ésta manera, al recorrer esa lista evadiremos todos los pares que no tienen una restricción y nuestra búsqueda será mucho más eficiente que en una matriz.

Por último, representaremos la solución con un vector<int> de longitud n, de manera que la posición i contendrá el número del cluster asignado al objeto i.

2.3 FUNCIÓN OBJETIVO

La función objetivo es la función que debemos minimizar, es decir, la función que dice como de buena es la solución que le pasamos como parámetro.

De todas las definiciones que hemos hecho, desviación general(\bar{C}) e infactibilidad(infeasibility) nos dan información acerca de la bondad de la solución, la primera a través de la distancia promedio de cada dato a su centroide correspondiente y la segunda a través de la medida del incumplimiento de las restricciones de la solución dada. Ambas han de ser minimizadas, pero hay que darle más o menos importancia a una y a otra, de manera que debemos introducir un parámetro con el que controlar ésta importancia que debatiremos posteriormente. Así, nuestra función objetivo queda definida como:

$$f = \bar{C} + (infeasibility) * \lambda$$

Dicho ésto queda definir el parámetro λ , el cual se ha decidido proponer como el entero superior a la distancia máxima D dividida por el número de restricciones totales R.

$$\lambda = \left\lceil \frac{D}{R} \right\rceil$$

A continuación se expone la descripción en pseudocódigo de todos los operadores y de la función objetivo.

calcular_infeasibility

```
begin
  Para cada instancia de lista_restricciones
  begin
    Si ( solucion[lista_restricciones[i][0]] = solucion[lista_restricciones[i][1]] ) and
    ( lista_restricciones[2] = -1 )
      entonces infactibilidad  $\leftarrow$  infactibilidad +1

    Si ( solucion[lista_restricciones[i][0]] != solucion[lista_restricciones[i][1]] ) and
    ( lista_restricciones[2] = 1 )
      entonces infactibilidad  $\leftarrow$  infactibilidad +1
  end

  return infactibilidad
end
```

calcular_centroides

```
begin
  Para cada instancia i de solucion
    centroide[i]  $\leftarrow$  centroide[i] + dataframe[i.posicion]

  Para cada centroide
    centroide  $\leftarrow$  centroide / cluster.size

  return centroides
end
```

calcular_distancia_media_intracluster

```
begin
  Para cada instancia i de solucion
  begin
    distancia_intracluster[i]  $\leftarrow$  distancia_intracluster[i] +
    distancia_euclidea(centroide[i],dataframe[i.posicion])
  end

  Para cada cluster
    distancia_intracluster[cluster]  $\leftarrow$  distancia_intracluster[cluster] / cluster.size
  return distancia_intracluster
end
```

calcular_desv_general

```
begin
  Para cada cluster
    desv_general ← desv_general + distancia_intracluster[cluster]

  desv_general ← desv_general / #clusters

  return desv_general
end
```

calcular_lamda

```
begin
  dist_max ← calcular_máxima_distancia(dataframe)
  lambda ← dist_max * rel / lista_restricciones.size

  return lambda
end
```

calcular_f_objetivo

```
begin
  inf ← calcular_infeasibility()
  dist_intra ← calcular_distancia_intracluster()
  desv ← calcular_desv_general()
  lamb ← calcular_lambda()

  return desv + inf * lamb
end
```

2.4 DATAFRAMES

Los conjuntos de datos usados para testear la bondad de los algoritmos son los siguientes:

- Iris: Contiene información sobre las características de tres tipos de flores de Iris. Tiene 3 clases($k=3$) y 6 dimensiones.
- Ecoli: Contiene medidas sobre las ciertas características de diferentes tipos de células que pueden ser empleadas para predecir la localización de ciertas proteínas. Tiene 8 clases($k=8$) y 7 dimensiones.
- Rand: Conjunto de datos artificial formado por tres agrupamientos bien diferenciados generados en base a distribuciones normales. Tiene 3 clases($k=3$) y 2 dimensiones.
- Newthyroid: Contiene medidas cuantitativas tomadas sobre la glándula tiroides de 215 pacientes. Tiene 3 clases($k=3$) y 5 dimensiones.

3. ALGORITMOS GENÉTICOS

3.1. DESCRIPCIÓN

Los Algoritmos Genéticos(AG) son algoritmos basados en poblaciones que funcionan de la siguiente manera: se generan un número determinado de soluciones, en principio, aleatorias(en nuestro caso 50) que llamaremos cromosomas conformadas como se ha explicado en el apartado 2.2(a cada componente del vector la llamaremos gen), se seleccionan un número determinado de cromosomas que serán los padres que posteriormente se cruzarán para formar hijos, que mutarán y podrán pasar a ser parte de la población para la siguiente iteración. Por tanto necesitaremos los siguientes operadores para conformar el algoritmo:

- Selección
- Cruce
- Mutación
- Reemplazamiento

Los cuales tendrán varias versiones con las que conformaremos cuatro variantes de AG: AG Generacional(AGG) y AG Estacionario(AGE), cada uno de ellos con cruce uniforme(UN) o con cruce por segmento finito(SF), es decir, AGG-UN, AGG-SF, AGE-UN y AGE-SF.

3.2 OPERADORES DE SELECCIÓN

Operadores de selección necesitaremos dos, uno para los AGG y otro para los AGE, a continuación detallaremos ambos.

seleccion_AGG

```
begin
  repetir 50 veces
    begin
      escoger dos padres al azar de poblacion
      introducir en la poblacion al mejor de ellos
    end
  end
```

seleccion_AGE

```
begin
  repetir 2 veces
    begin
      escoger dos padres al azar de poblacion
      introducir en la poblacion al mejor de ellos
    end
  end
```

Como vemos ambos se basan en lo mismo, se seleccionan los padres de la población a través de un torneo binario. La única diferencia es que para AGG seleccionaremos un número igual al de la población y para AGE tan solo dos.

3.3 OPERADORES DE CRUCE

Para el operador de cruce usaremos el mismo para AGG y para AGE, pero uno distinto para cada uno de sus variables. Pasemos a verlos:

cruce_uniforme

```
begin
  para cada dos padres
    begin
      introducir dos veces padre1 en hijos
      repetir para cada uno de los dos hijos creados
        cambiar aleatoriamente la mitad de los genes por los de padre2
    end
  end
end
```

cruce_segmento_finito

```
begin
  para cada dos padres
    begin
      introducir dos veces padre1 en hijos
      repetir para cada uno de los dos hijos creados
        introducir un segmento de tamaño y localización aleatoria de padre2
        cambiar aleatoriamente la mitad de los genes restantes por los de padre2
    end
  end
end
```

3.4 OPERADOR DE MUTACIÓN

El operador de mutación es común a todas las versiones y es el siguiente:

mutacion

```
begin
  para 1 de cada 1000 genes escogido de manera aleatoria
    se le suma una cantidad aleatoria entre 1 y k-1 de manera modular
  end
end
```

3.5 OPERADORES DE REEMPLAZAMIENTO

El operador de reemplazamiento es distinto para AGG y para AGE, dadas las naturalezas diferentes de sus poblaciones y que además el AGG requiere de elitismo, o lo que es lo mismo, mantener el mejor miembro en la siguiente generación.

reemplazamiento_AGG

```
begin
    introducir el mejor cromosoma de la población anterior en la nueva
    eliminar el peor hijo generados
    introducir los hijos en la población nueva
    introducir los padres no seleccionados a la nueva población
end
```

reemplazamiento_AGE

```
being
    si ambos hijos son mejores que sus padres
        intercambiarlos en la población
    si solo uno de ellos es mejor
        intercambiarlo por el peor padre
end
```

3.6 AGG Y AGE

A través de los anteriores operadores ya podemos conformar las cuatro versiones ya mencionadas.

AGG_UN

```
begin
    Para los 50 cromosomas de la poblacion
        generar_sol_aleatoria(3.1 de la práctica anterior)
    Hasta alcanzar 1000 evaluaciones de f
        seleccion_AGG
        cruce_uniforme
        mutacion
        reemplazamiento_AGG
    Devolver mejor solución de la población
end
```

AGG_SF

```
begin
    Para los 50 cromosomas de la poblacion
        generar_sol_aleatoria(3.1 de la práctica anterior)
    Hasta alcanzar 1000 evaluaciones de f
        seleccion_AGG
        cruce_segmento_finito
        mutacion
        reemplazamiento_AGG
    Devolver mejor solución de la población
end
```

AGE_UN

begin

Para los 50 cromosomas de la poblacion

generar_sol_aleatoria(3.1 de la práctica anterior)

Hasta alcanzar 1000 evaluaciones de f

seleccion_AGE

cruce_uniforme

mutacion

reemplazamiento_AGE

Devolver mejor solución de la población

end

AGE_SF

begin

Para los 50 cromosomas de la poblacion

generar_sol_aleatoria(3.1 de la práctica anterior)

Hasta alcanzar 1000 evaluaciones de f

seleccion_AGE

cruce_segmento_finito

mutacion

reemplazamiento_AGE

Devolver mejor solución de la población

end

4. ALGORITMO MEMÉTICO

El algoritmo memético(AM) será una hibridación entre el mejor AGG y una búsqueda local suave(BLS), de manera que aprovechemos lo mejor de la exploración de AGG y la explotación de BLS. Esto debería mejorar los resultados de los AGG.

4.1 ESTUDIO PRELIMINAR DE LOS AG

Ahora vamos a estudiar cual es el AGG que mejores resultados ha dado en nuestros conjuntos de datos. Para ello observaremos la media de las medidas obtenidas en las ejecuciones realizadas, las cuales serán estudiadas en profundidad en el punto 6.

10%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
AGG-UN	1,83	333,20	4,45	4668750	13,41	886,60	68,92	6746884	336,31	1221,78	84,53	17909375	2,48	338,20	5,18	4362500
AGG-SF	1,80	336,80	4,45	4515625	13,48	898,20	69,69	8384375	44,61	1532,40	86,09	17778125	2,52	333,00	5,23	4287500
AGE-UN	1,64	302,00	4,01	4262500	13,42	793,80	63,07	8134375	38,37	1356,20	74,91	17734375	1,98	389,60	4,61	4068750
AGE-SF	1,75	307,40	4,17	4234375	13,42	799,00	63,43	8175000	37,05	1398,20	74,84	17734375	2,18	320,00	4,74	4071875

20%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
AGG-UN	1,49	775,00	4,60	7706250	13,41	886,60	68,92	8481250	41,65	3327,40	86,05	33559375	1,99	817,20	5,24	7640625
AGG-SF	1,52	771,80	4,62	7584375	13,44	1928,00	74,24	14843750	43,54	3310,20	87,65	33243750	2,03	834,20	5,35	7512500
AGE-UN	1,12	753,60	4,15	7300000	13,12	1694,80	66,46	14612500	33,68	3197,80	76,27	33106250	1,56	810,00	4,79	7328125
AGE-SF	1,16	753,40	4,19	7406250	12,94	1700,60	66,60	14709375	36,05	3163,80	78,12	33203125	1,54	826,60	4,83	7275000

Como se puede apreciar, los resultados no son demasiado buenos, esto se comentará también en el punto 6, ahora nos ceñiremos solo a elegir el mejor AGG que, como se puede apreciar fácilmente observando los agregados(f), es AGG-UN, ya que mejora a AGG-SF en casi todos los casos. Por lo tanto nuestro algoritmo memético quedaría formado por AGG-UN hibridado con BLS.

4.2 BÚSQUEDA LOCAL SUAVE

La búsqueda local suave(BLS) es una variante de BL que busca vecinos en orden aleatorio para un cromosoma y se queda con el mejor, teniendo en cuenta las condiciones de parada que son exceder el número de fallos predeterminado($0.1 * \text{cromosoma.size}$) y no se realiza mejora o exceder un número máximo de iteraciones, en este caso $\text{cromosoma.size}()$, que coincide con la generación de todos los vecinos posibles variando una vez cada gen de la mejor manera.

BLS

```
begin
  mezclar indices de cromosomas
  mejora = true, fallos = 0, iters = 0, mejor = evaluar cromosoma de entrada
  Mientras (mejora o fallos < max_fallos) y iters < max_iters
  begin
    mejora = false
    Para cada i > 0, i < (k-1)
      Si vecino[(gen=iters)+i] es mejor que cromosoma
        cromosoma = vecino
        mejora = true
    Si !mejora
      fallos++
    iters++
  end
end
```

4.3 VARIANTES DE AM

Como se pide, se relizarán tres versiones de AM a partir de AGG-UN, ya descrito anteriormente, aunque sobre el mismo código ya que se puede controlar fácilmente mediante parámetros. Las versiones serían:

- AM(10,1): cada 10 cromosomas se aplica BLS a toda la población, se llama a `memetico(1,false)`
- AM(10,0.1): cada 10 cromosomas se aplica BLS al 10% de la población, se llama a `memetico(0,1,false)`
- AM(10,0.1mej): cada 10 cromosomas se aplica BLS al 10% de la población, se llama a `memetico(0.1,true)`

memetico(ratio, mejor)

begin

Para los 10 cromosomas de la poblacion

generar_sol_aleatoria(3.1 de la práctica anterior)

Hasta alcanzar 1000 evaluaciones de f

seleccion_AGG

cruce_uniforme

mutacion

reemplazamiento_AGG

Si mejor

lanzar BLS para los mejores cromosomas.size*ratio cromosomas

Si no

lanzar BLS para cromosomas.size*ratio cromosomas

Devolver mejor solución de la población

end

5. PROCEDIMIENTO CONSIDERADO PARA EL DESARROLLO DE LA PRÁCTICA

5.1 CÓDIGO Y LIBRERÍAS USADAS

La práctica ha sido desarrollada en C++, siendo todo el código implementado a mano, con ayuda de las explicaciones dadas en clase y en el guión, y usando las siguientes librerías:

- ◆ iostream: para entrada y salida por teclado y pantalla respectivamente.
- ◆ fstream: para entrada y salida por ficheros.
- ◆ STL: todos los vectores usados provienen de ésta librería.
- ◆ cmath: para algunos cálculos necesarios los algoritmos.
- ◆ algorithm: para usar la función shuffle()
- ◆ ctime: para medir tiempos de ejecución
- ◆ cstdlib: para la generación de números aleatorios

5.2 MANUAL DE USUARIO

Compilación:

Se incluye un makefile, por lo que se compila tan solo ejecutando la orden ‘make’ desde la terminal.

Ejecución completa:

La ejecución completa planteada para el ejercicio(6 conjuntos(3*2 % de restricciones)*5 ejecuciones distintas) se pueden hacer con ayuda del script incluido. Tan solo es necesario darle permisos de ejecución y ejecutarlo con el algoritmo como parámetro, 'copkm' para el algoritmo de comparación COPKM, 'bl' para el algoritmo de búsqueda local primero el mejor, 'aggun' para AGG-UN, 'aggsf' para AGG-SF, 'ageun' para AGE-UN, 'agesf' para AGE-SF. 'am1' para AM(10,1), 'am0.1' para AM(10,0.1) o 'am0.1mej' para AM(10,0.1mej)

Ejecución manual:

Si se desea ejecutar a mano un sólo conjunto, se puede hacer llamando a 'clustering' de la siguiente forma:

```
./clustering <dataframe> <restricciones> <k> <ALG> <seed>
```

ALG = {0:COPKM, 1:BL, 2:AGG-UN, 3:AGG-SF, 4:AGE-UN, 5:AGE-SF, 6:AM(10,1), 7:AM(10,0.1), 8:AM(10,0.1mej)}

También es posible llamarlo sin argumentos, entonces pedirá por teclado la información necesaria.

6. ANÁLISIS DE RENDIMIENTO

6.1 DESCRIPCIÓN DE LOS CASOS DEL PROBLEMA

Se van utilizar todos los dataframes anteriormente mencionados para analizar como de buenos son los algoritmos planteados para resolver el problema del agrupamiento con restricciones, con el 10 y el 20% de restricciones en cada conjunto de datos. Los datos organizan en tablas y se analizarán posteriormente.

Para ello se han realizado con cada uno de los cuatro dataframes, Iris, Ecoli, Newthyroid y Rand, tanto con el 10% como con el 20% de restricciones, un total de 5 veces, lo que hace un total de 40 ejecuciones(4 conjuntos x 2 conjuntos de restricciones x 5 veces).

Para poder replicar el total de ejecuciones todas ellas se han realizado utilizando semillas para la generación de números aleatorios, por lo que conociendo estas semillas se puede realizar el estudio exactamente de la misma manera y con los mismos resultados. Las semillas usadas en este estudio son las siguientes:

Ejecución 1	Ejecución 2	Ejecución 3	Ejecución 4	Ejecución 5
798245613	123456789	25022020	17042026	459268694

6.2 CUESTIONES SOBRE BL Y COPKM

Como se ha introducido un dataframe más para el estudio, ha sido necesario probarlo con los algoritmos de la práctica anterior, por que se adjunta dicha información para su posterior estudio.

BL 10%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	0,669	0	0,669	0,151	13,91	764,00	30,90	0,45	29,336	57	36,142	5,950	0,757	0	0,757	0,116
Ejecución 2	0,669	0	0,669	0,120	14,54	748,00	29,52	0,63	30,591	135	47,286	1,241	0,757	0	0,757	0,254
Ejecución 3	0,669	0	0,669	0,218	13,77	759,00	29,89	0,50	34,316	136	51,280	4,410	0,757	0	0,757	0,127
Ejecución 4	0,669	0	0,669	0,222	13,79	752,00	29,63	0,36	30,610	143	47,380	4,137	0,757	0	0,757	0,196
Ejecución 5	0,669	0	0,669	0,148	13,56	770,00	30,28	0,38	32,187	149	50,770	4,354	0,757	0	0,757	0,103
Media	0,67	0,00	0,67	0,17	13,91	758,60	30,04	0,46	31,41	124,00	46,57	4,02	0,76	0,00	0,76	0,16

BL 20%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	0,669	0	0,669	0,162	15,182	1550	30,865	1,141	28,967	204	41,146	1,916	0,757	0	0,757	0,152
Ejecución 2	0,669	0	0,669	0,171	14,745	1638	32,488	1,250	32,345	316	51,885	1,962	0,757	0	0,757	0,430
Ejecución 3	0,669	0	0,669	0,422	13,621	1746	34,420	0,922	29,404	190	41,253	7,133	0,757	0	0,757	0,206
Ejecución 4	0,669	0	0,669	0,351	15,325	1597	31,770	1,094	30,643	210	42,957	7,329	0,757	0	0,757	0,244
Ejecución 5	0,669	0	0,669	0,192	15,271	1556	30,988	1,031	32,005	169	45,545	8,000	0,757	0	0,757	0,130
Media	0,67	0,00	0,67	0,26	14,83	1617,40	32,11	1,09	30,67	217,80	44,56	5,27	0,76	0,00	0,76	0,23

COPKM 10%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	0,669	0	0,669	0,020	16,136	732	35,732	0,016	36,919	2	36,959	0,268	0,757	0	0,757	0,004
Ejecución 2	0,669	0	0,669	0,004	16,707	715	37,082	0,031	39,523	64	40,740	0,234	0,757	0	0,757	0,004
Ejecución 3	0,669	0	0,669	0,009	17,107	721,250	37,500	0,023	39,584	89	41,450	0,265	0,757	0	0,757	0,004
Ejecución 4	0,669	0	0,669	0,004	18,245	716	38,958	0,016	33,933	1	33,954	0,203	0,757	0	0,757	0,003
Ejecución 5	0,669	0	0,669	0,003	17,340	722	38,226	0,031	39,410	3	39,466	0,273	0,757	0	0,757	0,004
Media	0,67	0,00	0,67	0,008	17,11	721,25	37,50	0,02	37,87	31,80	38,51	0,25	0,76	0,00	0,76	0,00

COPKM 20%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	0,669	0	0,669	0,007	16,283	1554	36,236	0,078	35,730	8	35,821	0,327	0,757	0	0,757	0,008
Ejecución 2	0,669	0	0,669	0,008	16,040	1555	38,713	0,047	36,721	9	36,803	0,265	0,757	0	0,757	0,006
Ejecución 3	0,669	0	0,669	0,006	16,500	1548	39,072	0,094	35,360	0	35,360	0,328	0,757	0	0,757	0,006
Ejecución 4	0,669	0	0,669	0,006	16,059	1542	38,543	0,047	35,360	0	35,360	0,329	0,757	0	0,757	0,007
Ejecución 5	0,669	0	0,669	0,007	38,403	1556	38,403	0,063	37,584	12	37,706	0,264	0,757	0	0,757	0,009
Media	0,67	0,00	0,67	0,007	25,08	1551,40	38,63	0,06	36,15	5,80	36,21	0,30	0,76	0,00	0,76	0,01

***En rojo:** Cicla, se han introducido valores medios para que no afecte al resto

Como se puede ver, para COPKM se obtiene un comportamiento peor que en Ecoli, a pesar de que este último tiene más dimensiones y una k bastante más grande, mientras que para BL los resultados son también malos pero destaca que los resultados son parecidos a COPKM.

De esto se puede empezar a observar que Newthyroid es un conjunto de datos más complejo que el resto y no en términos de dimensiones o de conjuntos, si no en la distribución de estos conjuntos, es decir, en la dificultad de separar los clusters.

6.3 PROBLEMÁTICA

Importante: Antes de comenzar a ver los resultados, es necesario comentar que los resultados no son buenas, no por la calidad de los algoritmos si no por la propia implementación. Teóricamente los AG deberían de ser mejores que BL y que COPKM, al menos en la mayoría de casos; sin embargo tras mucho tiempo dedicado al problema, el cual se presenta en las cuatro versiones de AG, me ha sido imposible encontrar el error en la implementación por lo que ante la falta de tiempo y teniendo en cuenta que esta práctica no debería de haber consumido tanto de este se ha tenido que dejar así. Esto implica que no se van analizar las mejoras respecto a la práctica anterior, sino que se hará un estudio de donde y cómo funcionan mejor este tipo de algoritmos teniendo en cuenta que existe un error derivado de la implementación.

6.4 RESULTADOS OBTENIDOS

A continuación se detallan en el formato anteriormente comentado los resultados obtenidos en las pruebas realizadas:

AGG-UN 10%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	1,785	335	4,410	4937500	13,421	886	68,891	8453125,000	43,757	1487	84,038	17843750,000	2,659	327	5,278	4421875,000
Ejecución 2	1,892	319	4,406	4687500,000	13,237	891	69,021	8359375,000	44,265	1507,000	85	17875000,000	2,183	356	5,034	4312500,000
Ejecución 3	1,624	361	4,469	4609375,000	13,402	891	69,186	8406250,000	44,293	1504	85,036	18125000,000	2,426	330	5,068	4296875,000
Ejecución 4	1,882	329	4,475	4500000,000	13,496	874	68,215	8453125,000	43,347	1532	84,849	17718750,000	2,444	359	5,319	4421875,000
Ejecución 5	1,961	322	4,483	4609375,000	13,497	891	69,280	8421875,000	43,134	1501	83,796	17984375,000	2,679	319	5,203	4359375,000
Media	1,83	333,20	4,45	4668750,00	13,41	886,60	68,92	6746883,85	336,31	1221,78	84,53	17909375,00	2,48	338,20	5,18	4362500,00

AGG-UN 20%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	1,709	779	4,843	7750000,000	13,421	886	68,891	8515625,000	41,117	3329	85,492	34031250,000	1,808	837	5,143	7593750,000
Ejecución 2	1,593	767	4,679	7640625,000	13,237	891	69,021	8656250,000	41,510	3328	85,871	33578125,000	1,966	822	5,237	7812500,000
Ejecución 3	1,410	792	4,583	7640625,000	13,402	891	69,186	8343750,000	41,155	3310	85,590	33218750,000	1,927	825	5,216	7578125,000
Ejecución 4	1,405	760	4,452	7781250,000	13,496	874	68,215	8500000,000	43,237	3323	87,468	33484375,000	1,873	824	5,148	7703125,000
Ejecución 5	1,350	777	4,447	7718750,000	13,497	891	69,280	8390625,000	41,244	3347	85,852	33484375,000	2,368	778	5,469	7515625,000
Media	1,49	775,00	4,60	7706250,00	13,41	886,60	68,92	8481250,00	41,65	3327,40	86,05	33559375,00	1,99	817,20	5,24	7640625,00

AGG-SF 10%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	1,639	353	4,421	4609375,000	13,463	890	69,116	8484375,000	44,684	1561	86,906	17796875,000	2,550	335	5,233	4250000,000
Ejecución 2	1,948	323	4,488	4578125,000	13,619	892	69,432	8296875,000	44,879	1534	86,391	17750000,000	2,741	314	5,255	4312500,000
Ejecución 3	1,816	335	4,456	4500000,000	13,809	878	68,779	8406250,000	43,903	1508	84,754	17796875,000	2,300	338	5,201	4234375,000
Ejecución 4	1,783	341	4,456	4453125,000	13,065	920	70,664	8406250,000	45,383	1529	86,803	17812500,000	2,713	329	5,348	4359375,000
Ejecución 5	1,804	332	4,420	4437500,000	13,453	911	70,451	8328125,000	44,207	1530	85,613	17734375,000	2,310	349	5,103	4281250,000
Media	1,80	336,80	4,45	4515625,00	13,48	898,20	69,69	8384375,00	44,61	1532,40	86,09	17778125,00	2,52	333,00	5,23	4287500,00

AGG-SF 20%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	1,215	798	4,422	7484375,000	13,549	1916	73,924	14828125,000	43,468	3333	87,863	33375000,000	2,410	772	5,487	7562500,000
Ejecución 2	1,731	755	4,769	7421875,000	13,262	1930	74,145	14843750,000	43,569	3294	87,477	33187500,000	1,748	880	5,256	7640625,000
Ejecución 3	1,377	787	4,542	7578125,000	13,407	1938	74,562	14796875,000	43,258	3334	87,659	33218750,000	2,265	807	5,482	7453125,000
Ejecución 4	1,398	773	4,492	7843750,000	13,394	1912	73,674	14859375,000	44,096	3288	87,924	33093750,000	1,891	851	5,283	7468750,000
Ejecución 5	1,880	746	4,881	7593750,000	13,565	1944	74,910	14890625,000	43,305	3302	87,320	33343750,000	1,831	861	5,255	7437500,000
Media	1,52	771,80	4,62	7584375,00	13,44	1928,00	74,24	14843750,00	43,54	3310,20	87,65	33243750,00	2,03	834,20	5,35	7512500,00

AGE-UN 10%	Iris				Newthyroid				Ecoli				Rand			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
Ejecución 1	1,931	290	4,217	4328125,000	13,323	776	61,894	8218750,000	38,302	1337	74,431	17656250,000	1,675	347	4,451	4015625,000
Ejecución 2	1,324	325	3,862	4187500,000	13,321	807	63,845	8093750,000	38,066	1358	74,840	17812500,000	2,376	302	4,753	4093750,000
Ejecución 3	1,445	311	3,891	4281250,000	13,349	824	64,938	8187500,000	38,098	1384	75,545	17640625,000	1,336	376	4,347	4109375,000
Ejecución 4	1,815	288	4,084	4265625,000	13,597	766	61,555	8093750,000	40,199	1336	76,365	17843750,000	1,757	645	4,519	4046875,000
Ejecución 5	1,665	296	3,997	4250000,000	13,522	796	63,105	8078125,000	37,172	1366	73,358	17718750,000	2,744	278	4,970	4078125,000
Media	1,64	302,00	4,01	4262500,00	13,42	793,80	63,07	8134375,00	38,37	1356,20	74,91	17734375,00	1,98	389,60	4,61	4068750,00

AGE-UN 20%	Iris				Newthyroid				Ecoli				Rand			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
Ejecución 1	1,329	751	4,339	7265625,000	13,173	1665	65,474	14734375,000	31,524	3223	74,449	33234375,000	1,250	857	4,659	7406250,000
Ejecución 2	1,038	766	4,119	7343750,000	13,252	1745	68,129	14593750,000	33,506	3209	76,234	33078125,000	1,675	790	4,824	7296875,000
Ejecución 3	1,106	757	4,149	7265625,000	12,728	1682	65,640	14578125,000	35,187	3151	77,167	33125000,000	1,154	875	4,641	7359375,000
Ejecución 4	1,082	742	4,067	7281250,000	13,282	1767	69,003	14531250,000	34,073	3197	76,618	33078125,000	1,823	786	4,952	7296875,000
Ejecución 5	1,058	752	4,084	7343750,000	13,150	1615	64,039	14625000,000	34,118	3209	76,865	33015625,000	1,922	742	4,879	7281250,000
Media	1,12	753,60	4,15	7300000,00	13,12	1694,80	66,46	14612500,00	33,68	3197,80	76,27	33106250,00	1,56	810,00	4,79	7328125,00

AGE-SF 10%	Iris				Newthyroid				Ecoli				Rand			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
Ejecución 1	1,298	340	3,978	4328125,000	13,045	801	63,186	8390625,000	37,278	1418	75,439	17859375,000	1,868	329	4,503	4093750,000
Ejecución 2	1,813	297	4,154	4203125,000	13,473	797	63,371	8125000,000	37,909	1410	76,033	17718750,000	1,946	330	4,589	4078125,000
Ejecución 3	1,921	300	4,285	4265625,000	13,685	789	63,083	8140625,000	37,916	1364	74,830	17796875,000	2,688	303	5,096	4062500,000
Ejecución 4	1,880	301	4,252	4156250,000	13,428	805	63,812	8078125,000	34,933	1391	72,582	17671875,000	2,283	314	4,798	4078125,000
Ejecución 5	1,826	299	4,183	4218750,000	13,478	803	63,686	8140625,000	37,220	1408	75,298	17625000,000	2,138	324	4,693	4046875,000
Media	1,75	307,40	4,17	4234375,00	13,42	799,00	63,43	8175000,00	37,05	1398,20	74,84	17734375,00	2,18	320,00	4,74	4071875,00

AGE-SF 20%	Iris				Newthyroid				Ecoli				Rand			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
Ejecución 1	1,089	787	4,253	7406250,000	12,830	1715	66,929	14718750,000	36,312	3123	77,922	33687500,000	1,153	866	4,604	7312500,000
Ejecución 2	1,016	761	4,078	7375000,000	12,158	1621	63,310	14609375,000	36,610	3155	78,530	33187500,000	2,562	713	5,404	7312500,000
Ejecución 3	1,532	718	4,421	7343750,000	13,012	1715	67,130	14671875,000	35,259	3203	77,910	33093750,000	1,267	853	4,650	7187500,000
Ejecución 4	1,071	751	4,092	7359375,000	13,178	1773	69,115	14828125,000	37,086	3164	79,117	32921875,000	1,458	849	4,831	7265625,000
Ejecución 5	1,093	750	4,104	7546875,000	13,541	1679	66,495	14718750,000	34,969	3174	77,145	33125000,000	1,246	852	4,642	7296875,000
Media	1,16	753,40	4,19	7406250,00	12,94	1700,60	66,60	14709375,00	36,05	3163,80	78,12	33203125,00	1,54	826,60	4,83	7275000,00

AM (10,1) 10%	Iris				Newthyroid				Ecoli				Rand			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
Ejecución 1	1,181	348	3,923	4593750,000	12,989	815	64,015	8312500,000	31,272	1339	66,763	18171875,000	1,894	335	4,577	4218750,000
Ejecución 2	1,927	299	4,283	4531250,000	13,602	821	65,009	8562500,000	29,137	1326	64,865	17656250,000	1,825	345	4,588	4187500,000
Ejecución 3	1,245	344	3,956	4359375,000	13,458	803	63,732	8234375,000	27,953	1380	65,331	17765625,000	1,497	375	4,499	4250000,000
Ejecución 4	1,639	321	4,169	4328125,000	13,363	780	62,197	8218750,000	31,129	1325	67,023	17875000,000	1,931	324	4,525	4156250,000
Ejecución 5	1,765	297	4,106	4328125,000	13,131	788	62,466	8234375,000	33,090	1341	69,073	17812500,000	1,767	354	4,602	4250000,000
Media	1,55	321,80	4,09	4428125,00	13,31	801,40	63,48	8312500,00	30,52	1342,20	66,61	17856250,00	1,78	346,60	4,56	4212500,00

AM (10,1) 20%	Iris				Newthyroid				Ecoli				Rand			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
Ejecución 1	1,059	772	4,163	7593750,000	11,422	1781	67,624	15515625,000	28,570	3148	70,493	34609375,000	1,679	815	4,928	8125000,000
Ejecución 2	1,126	766	4,208	7515625,000	13,870	1816	71,176	14968750,000	28,603	3127	70,285	33625000,000	1,796	775	4,885	7468750,000
Ejecución 3	1,145	745	4,142	7578125,000	13,261	1709	67,190	15000000,000	29,687	3087	70,795	32984375,000	1,488	852	4,884	7500000,000
Ejecución 4	1,130	751	4,152	7625000,000	13,510	1784	69,781	14953125,000	29,594	3144	71,466	33093750,000	1,837	789	4,981	7500000,000
Ejecución 5	1,106	763	4,175	7593750,000	13,077	1648	65,081	14781250,000	26,296	3164	68,245	32984375,000	1,360	842	4,716	7531250,000
Media	1,11	759,40	4,17	7581250,00	13,03	1747,60	68,17	15043750,00	28,55	3134,00	70,26	33459375,00	1,63	814,60	4,88	7625000,00

AM (10,0,1) 10%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	1,732	301	4,097	4218750,000	13,353	778	62,062	8015625,000	38,527	1426	77,158	17468750,000	1,984	341	4,714	4093750,000
Ejecución 2	1,697	301	4,069	4265625,000	13,248	801	63,397	8031250,000	36,441	1428	75,126	17500000,000	1,860	346	4,631	4109375,000
Ejecución 3	1,717	320	4,239	4218750,000	13,372	794	63,083	8093750,000	35,026	1452	74,361	17769875,000	2,562	300	4,964	4109375,000
Ejecución 4	1,239	347	3,973	4218750,000	13,394	816	64,482	8218750,000	38,396	1384	75,889	17578125,000	1,544	356	4,395	4046875,000
Ejecución 5	1,624	319	4,138	4234375,000	13,641	800	63,728	8046875,000	37,245	1432	76,038	17468750,000	1,671	358	4,537	4125000,000
Media	1,60	317,60	4,10	4231250,00	13,40	797,80	63,35	8081250,00	37,13	1424,40	75,71	17557100,00	1,92	340,20	4,65	4096875,00

AM (10,0,1) 20%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	1,076	742	4,061	7203125,000	13,099	1682	66,176	14578125,000	37,043	3161	79,178	33078125,000	1,524	818	4,784	7203125,000
Ejecución 2	1,075	753	4,105	7375000,000	13,599	1744	68,632	14687500,000	36,548	3226	79,550	32812500,000	1,231	876	4,745	7218750,000
Ejecución 3	1,484	714	4,357	7265625,000	12,536	1756	67,948	14515625,000	33,252	3223	76,192	32609375,000	1,125	882	4,641	7203125,000
Ejecución 4	1,031	770	4,129	7296875,000	12,954	1641	64,738	14703125,000	36,031	3216	78,899	32984375,000	1,195	858	4,615	7234375,000
Ejecución 5	1,103	748	4,112	7218750,000	13,113	1713	67,168	14656250,000	35,467	3230	78,522	32890625,000	1,169	868	4,629	7187500,000
Media	1,15	745,40	4,15	7271875,00	13,06	1707,20	66,93	14628125,00	35,67	3211,20	78,47	32875000,00	1,25	860,40	4,68	7209375,00

AM (10,0,1mej) 10%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	1,793	310	4,236	4312500,000	13,684	785	62,831	8093750,000	24,847	1322	606599,000	17578125,000	1,538	354	4,373	4140625,000
Ejecución 2	1,456	319	3,970	4265625,000	13,257	761	60,902	8156250,000	26,632	1319	62,364	17484375,000	1,802	340	4,522	4125000,000
Ejecución 3	1,418	340	4,097	4281250,000	13,878	761	61,523	8125000,000	26,075	1324	61,942	17687500,000	1,520	372	4,499	4156250,000
Ejecución 4	1,670	314	4,144	4265625,000	13,063	798	63,024	8125000,000	29,224	1273	63,710	17500000,000	2,576	290	4,899	4109375,000
Ejecución 5	1,204	342	3,900	4281250,000	13,728	789	63,126	8093750,000	27,859	1377	65,162	17593750,000	1,838	328	4,464	4156250,000
Media	1,51	325,00	4,07	4281250,00	13,52	778,80	62,28	8118750,00	26,93	1323,00	121370,44	17568750,00	1,85	336,80	4,55	4137500,00

AM (10,0,1mej) 20%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
Ejecución 1	1,079	753	4,109	7390625,000	13,209	1682	66,286	15078125,000	27,947	3058	68,710	33328125,000	1,301	845	4,669	7312500,000
Ejecución 2	1,040	760	4,097	7343750,000	13,211	1705	67,014	14750000,000	25,266	3121	66,869	32953125,000	1,834	791	4,987	7343750,000
Ejecución 3	1,107	731	4,048	7359375,000	11,433	1613	62,332	14250000,000	22,374	3292	66,255	33109375,000	1,184	856	4,596	7312500,000
Ejecución 4	1,046	760	4,104	7375000,000	13,725	1693	67,150	14734375,000	25,887	3182	68,303	33203125,000	1,213	869	4,676	7296875,000
Ejecución 5	0,993	757	4,038	7343750,000	13,070	1739	67,946	14765625,000	22,975	3219,000	66	33125000,000	1,758	804	4,962	7328125,000
Media	1,05	752,20	4,08	7362500,00	12,93	1686,40	66,15	11762504,60	664,09	2543,78	6625054,03	26518750,35	1,46	833,00	4,78	7318750,00

Con todo esto, junto con los resultados de la práctica anterior, obtenemos unos valores medios presentados en la siguiente tabla con se podrá estudiar el comportamiento de cada algoritmo más fácilmente.

10%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
COPKM	0,67	0,00	0,67	0,01	17,11	721,25	37,50	0,02	37,87	31,80	38,51	0,25	0,76	0,00	0,76	0,00
BL	0,67	0,00	0,67	0,17	13,91	758,60	30,04	0,46	31,41	124,00	46,57	4,02	0,76	0,00	0,76	0,16
AGG-UN	1,83	333,20	4,45	4668750	13,41	886,60	68,92	6746884	336,31	1221,78	84,53	17909375	2,48	338,20	5,18	4362500
AGG-SF	1,80	336,80	4,45	4515625	13,48	898,20	69,69	8384375	44,61	1532,40	86,09	17778125	2,52	333,00	5,23	4287500
AGE-UN	1,64	302,00	4,01	4262500	13,42	793,80	63,07	8134375	38,37	1356,20	74,91	17734375	1,98	389,60	4,61	4068750
AGE-SF	1,75	307,40	4,17	4234375	13,42	799,00	63,43	8175000	37,05	1398,20	74,84	17734375	2,18	320,00	4,74	4071875
AM(10,1)	1,55	321,80	4,09	4428125,00	13,31	801,40	63,48	8312500,00	30,52	1342,20	66,61	17856250,00	1,78	346,60	4,56	4212500,00
AM(10,0,1)	1,60	317,60	4,10	4231250,00	13,40	797,80	63,35	8081250,00	37,13	1424,40	75,71	17557100,00	1,92	340,20	4,65	4096875,00
AM(10,0,1mej)	1,51	325,00	4,07	4281250,00	13,52	778,80	62,28	8118750,00	26,93	1323,00	121370,44	17568750,00	1,85	336,80	4,55	4137500,00

20%	Iris				Newthyroid				Ecoli				Rand			
	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T	Tasa_C	Tasa_inf	Agr.	T
COPKM	0,67	0,00	0,67	0,01	25,08	1551,40	38,63	0,06	36,15	5,80	36,21	0,30	0,76	0,00	0,76	0,01
BL	0,67	0,00	0,67	0,26	14,83	1617,40	32,11	1,09	30,67	217,80	44,56	5,27	0,76	0,00	0,76	0,23
AGG-UN	1,49	775,00	4,60	7706250	13,41	886,60	68,92	8481250	41,65	3327,40	86,05	33559375	1,99	817,20	5,24	7640625
AGG-SF	1,52	771,80	4,62	7584375	13,44	1928,00	74,24	14843750	43,54	3310,20	87,65	33243750	2,03	834,20	5,35	7512500
AGE-UN	1,12	753,60	4,15	7300000	13,12	1694,80	66,46	14612500	33,68	3197,80	76,27	33106250	1,56	810,00	4,79	7328125
AGE-SF	1,16	753,40	4,19	7406250	12,94	1700,60	66,60	14709375	36,05	3163,80	78,12	33203125	1,54	826,60	4,83	7275000
AM(10,1)	1,11	759,40	4,17	7581250,00	13,03	1747,60	68,17	15043750,00	28,55	3134,00	70,26	33459375,00	1,63	814,60	4,88	7625000,00
AM(10,0,1)	1,15	745,40	4,15	7271875,00	13,06	1707,20	66,93	14628125,00	35,67	3211,20	78,47	32875000,00	1,25	860,40	4,68	7209375,00
AM(10,0,1mej)	1,05	752,20	4,08	7362500,00	12,93	1686,40	66,15	11762504,60	664,09	2543,78	67,42	26518750,35	1,46	833,00	4,78	7318750,00

6.5 ANÁLISIS DE RESULTADOS

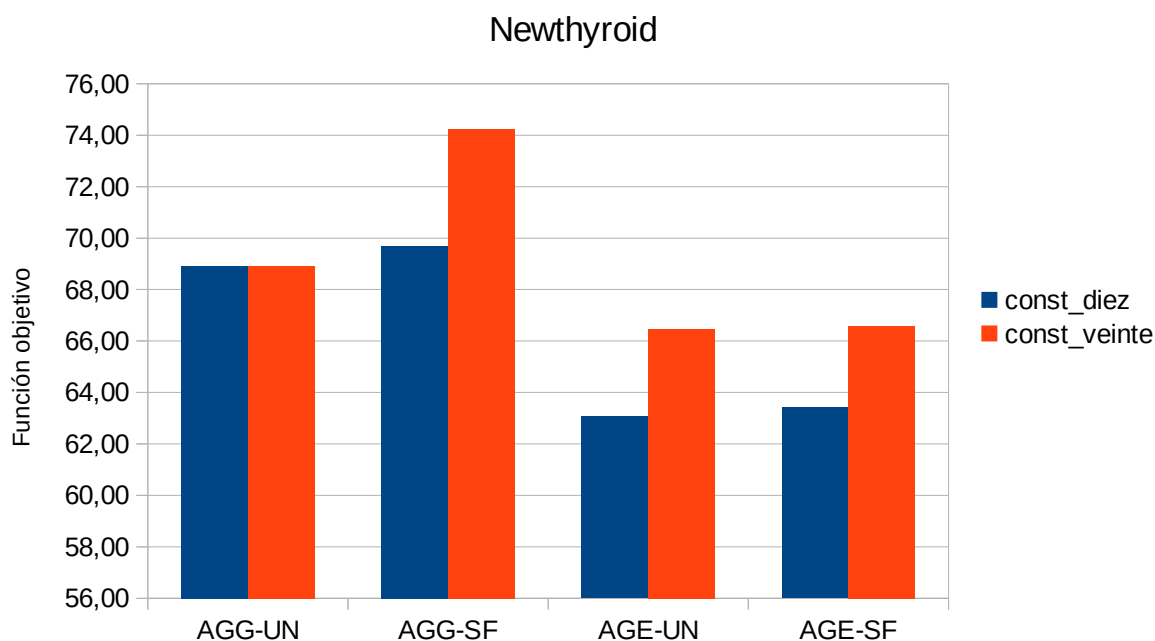
Como resumen de lo dicho en la primera práctica, sabemos que COPKM funciona muy bien para conjuntos de datos sencillos, obteniendo además los resultados en poco tiempo, mientras que BL aunque consume más tiempo, sabemos que va a ser mejor en otro conjunto más complejos sabiendo que ésta sí que tiene en cuenta más directamente la desviación general.

Añadiendo el nuevo conjunto de datos, que como ya hemos visto es más complejo que el resto, confirmamos lo dicho, BL se comporta mejor en esas circunstancias.

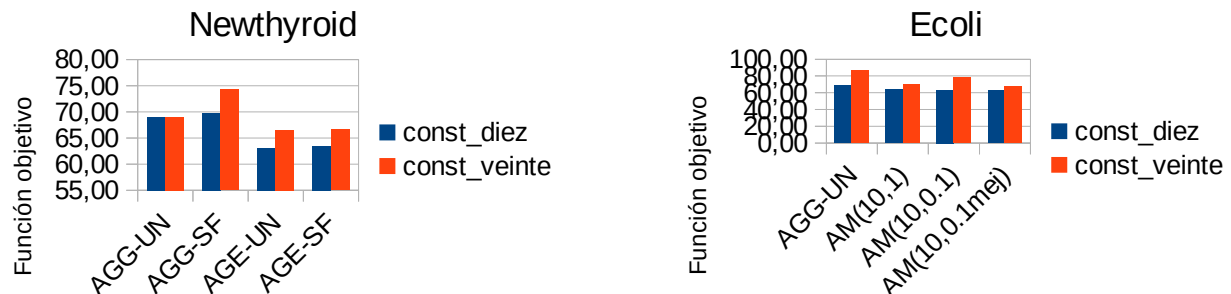
Como para esta práctica tenemos más variedad que comparar, aclaremos un concepto sobre BL: se basa en la explotación, es decir, tiene una gran capacidad para encontrar mejores soluciones y llegar al óptimo más cercano. Sin embargo hay conjuntos de datos que tienen mucho óptimos locales y que necesitan de capacidad de exploración para encontrar óptimos globales.

Habrà que tener en cuenta también que los AG y los AM son mucho más costosos en tiempo que BL y COPKM, por lo que para conjuntos donde funcionen bien estos últimos no sería interesante usarlos. Un ejemplo son Iris y Rand, donde BL y COPKM ya encuentran óptimos y no son necesarios los AG(recaltar que estos también deberían dar el óptimo sin el error antes mencionado).

Centrémonos entonces en los conjuntos de datos más complejos, Ecoli y Newthyroid, para realizar el análisis, destacando primero que los AGE funcionan mejor que los AGG, probablemente porque estos pueden realizar más iteraciones ya que cada una de ellas requiere un menor número de evaluaciones de la función objetivo.



Ahora bien, comparando los comportamientos de AGG-UN, el seleccionado para hibridar, y todos los AM, podemos deducir algunas propiedades de los conjuntos de datos.



Es claro que Newthyroid obtiene mejores resultados tras hibridar, sin embargo Ecoli apenas nota la BLS. Es por ello por lo que podemos deducir que Newthyroid se beneficia tanto de la exploración, para salir de algunos óptimos locales, como de la explotación para llegar a esos óptimos, por lo que es un buen candidato para usar en él algoritmos meméticos. Ecoli parece ni inmutarse ante el cambio de AG a AM, por lo que parece que prima la exploración y no tanto la explotación de manera que el uso de los AG que funcionan mejor (como son en este caso cualquiera de los AGE) parece la mejor solución para este caso, prueba de ello que BL no funcione demasiado bien como vimos en la primera práctica.

A través de este estudio hemos podido ver que, dependiendo de la naturaleza del conjunto de datos, será recomendable usar un tipo de metaheurística u otra, ya que algunos se beneficiarán más de la exploración y otros de la explotación.

7. BIBLIOGRAFÍA

- <http://www.cplusplus.com/reference/vector/vector/>
- <http://www.cplusplus.com/reference/cstdlib/>
- <http://www.cplusplus.com/reference/fstream/>
- <http://www.cplusplus.com/reference/ctime/>
- <http://www.cplusplus.com/reference/algorithm/count/>