



Universidad de Granada

E.T.S Ingeniería Informática y de Telecomunicación

Grado en Ingeniería Informática

Departamento de Ciencias de la Computación e Inteligencia
Artificial

Metaheurísticas - MH

Grupo A1 (Miércoles 17:30 - 19:30)

Curso 2019/2020

Práctica 2.b: Técnicas de Búsqueda basadas en Poblaciones para el Problema del Agrupamiento con Restricciones

Javier Rodríguez Rodríguez

78306251Z

e.doblerodriguez@go.ugr.es

1. Descripción del problema

El problema propuesto consiste en el agrupamiento (clustering) de instancias de datos no etiquetadas que están sujetos a un conjunto de restricciones de instancia de tipo Must-Link (ML) o Cannot-Link (CL). Estas restricciones se consideran débiles, es decir, no son limitantes a una solución, pero el cumplimiento de estas es un factor importante de calidad de la solución.

Este problema, perteneciente al campo de aprendizaje semi-supervisado, es NP-Completo, y por tanto, resulta imposible obtener un algoritmo que halle la solución óptima en un tiempo razonable. Entendiendo esta limitación, y razonando dentro del campo de la metaheurística, se proponen varios algoritmos basados en poblaciones. En particular, algoritmos genéticos, basados en el cruce genético en poblaciones de seres vivos, y algoritmos meméticos, que combinan el poder explorativo de los algoritmos genéticos con la explotación de los algoritmos de Búsqueda Local. A su vez, cambiaremos parámetros y mecanismos internos de cada algoritmo: en el caso de los genéticos se implementarán dos modelos distintos, el generacional (AGG), donde en cada iteración se reemplaza toda la población por una nueva (pero con elitismo para conservar a un potencial óptimo), y estacional (AGE), donde en cada iteración competirán las dos peores soluciones contra dos nuevos hijos para pertenecer en la población. A su vez, en cada caso se utilizarán dos operadores de cruce distintos: Operador Uniforme (UN), donde en el cruce se mezclan uniformemente la información de ambos padres, y Segmento Fijo (SF), donde cierto trozo de información de un padre tiene mayor importancia y se conserva en el cruce. Si bien hay variaciones suficientes para determinar un orden de calidad entre ellos, también consideraremos las soluciones obtenidas por los algoritmos de Búsqueda Local (BL) y K-medias con restricciones (COPKM), realizados en la práctica I, con el fin de determinar qué técnicas de metaheurística resultan mejores.

La comparación entre los algoritmos se establecerá en dos niveles distintos: la calidad de la solución y la eficiencia temporal. La calidad debe medir adecuadamente qué tan parecidos son realmente los datos que se agrupan en un mismo cluster y cuántas restricciones incumple, mientras que la eficiencia temporal debe medir el tiempo requerido para dar una solución que el criterio del algoritmo considere suficiente.

2. Aplicación de los algoritmos:

Para aplicar los algoritmos descritos, primero que nada, es necesario describir las estructuras de datos que contienen toda la información. Los datos de entrada, las instancias y las restricciones, son provistas en forma de archivos en texto plano cuyos nombres identifican el conjunto de datos de donde se extrajeron, y en el caso de ser restricciones, el porcentaje de relaciones restringidas. Estos ficheros se leen y almacenan en una estructura de matriz: para las instancias, cada fila corresponde a una instancia, mientras que las columnas corresponden a las distintas dimensiones de cada instancia; para las restricciones, inicialmente es una matriz cuadrada simétrica donde cada valor $[i,j] \in \{-1,0,1\}$ corresponde a la restricción entre la instancia i y la instancia j , donde cada valor representa un tipo de restricción, siendo -1 una restricción CL, 1 una restricción ML y 0 ninguna restricción. Sin embargo, una vez leída se registran los pares que tienen restricciones ML y CL respectivamente en dos listas de pares, con el fin de optimizar ciertas operaciones.

La solución del problema, para un conjunto de instancias y restricciones dado, vendrá dado en un vector de tantos elementos como instancias de datos haya, y que en cada posición contendrá el índice del cluster al que pertenece la instancia en dicha fila en la matriz de datos: es decir, Solución $[i]$ contendrá el número de cluster al que pertenece Datos $[i]$.

Para la medición de la calidad se emplea una función objetivo, cuyo valor ha de expresar la tasa de diferencia entre los datos de un mismo cluster y la cantidad de restricciones incumplidas, y por tanto, se ha de minimizar mientras de mayor calidad sea la solución. Utilizaremos la siguiente función

$$\text{Objetivo} = C_{\text{General}} + \text{Infeasibility} * \lambda$$

donde:

- C_{General} corresponde a la desviación general de la solución, es decir, el valor medio de desviación de cada cluster. La desviación de un cluster corresponde al valor medio de las distancias entre su centroide asociado y cada una de las instancias contenidas en el cluster. Es decir:

$$(1) C_{\text{General}} = \sum_{i=1}^k C_i / k$$

$$(2) C_i = \sum_{j=1}^{n_i} |\mu_i - x_j| / j$$

donde k sea la cantidad de clústeres, μ_i el centroide del cluster i , x_j el dato j en el cluster y n_i el número de datos en el cluster i .

- Infeasibility es un número entero que corresponde a la cantidad de restricciones incumplidas
- λ es un factor de escala, con el propósito de darle peso suficiente a la infeasibility y que se valore correctamente respecto al valor de C_{General} . Un

valor general que se considera correcto es un valor mayor a la distancia máxima entre un par de instancias, seleccionando nosotros el techo de dicha distancia, entre la cantidad de restricciones. Es decir:

$$(3) \lambda = \text{techo}(D_{\max}) / |R|$$

siendo $|R|$ el número de restricciones y D_{\max} la distancia máxima entre dos instancias. Importante destacar, de cara a la implementación, que en la contabilización de restricciones debemos evitar contabilizar cada restricción dos veces (debido a la simetría de la matriz)

Para la medición de la eficiencia temporal, consideraremos como métrica el tiempo de ejecución de ambos algoritmos en condiciones similares (mismo equipo bajo carga similar).

Por último, es necesario establecer, dado que todos los algoritmos contienen instrucciones estocásticas, es necesario especificarle a los algoritmos un generador de números aleatorios que, para una semilla dada, sea capaz de replicar los valores generados.

Para este nuevo grupo de algoritmos, basados en poblaciones, el manejo de soluciones (también llamadas cromosomas), si bien obedecen a la representación establecida, no se manejan de forma individual. Por el contrario, el algoritmo maneja conjuntos (poblaciones) de éstas, combinándolas entre ellas para formar nuevas posibles soluciones, y haciéndolas competir, según la calidad de su función objetivo asociada, para su preservación en la población. Debido a esto, las métricas de la práctica 1 (infeasibility, desviación general, función objetivo, vector solución) son manipuladas mediante matrices de éstas, donde cada fila representa la métrica del cromosoma asociado. Finalmente, se retornan las métricas de aquel cromosoma cuyo valor de la función objetivo sea el mínimo de la población.

El pseudocódigo de evaluación de una población es el siguiente:

```

EVALUATION(datos, RML, RCL, k, poblacion,  $\lambda$ )
para cada fila i en poblacion:
    para cada j en [0..k-1]:
        centroide[i,j] = Media(datos en cluster j en cromosoma i)
    Cgeneral = 0
    para cada j en [0..k-1]:
        Cgeneral[i] += Media(Distancia(centroide[i,j], datos en cluster j en cromosoma i))/k
    infeasibility[i] = |parejas ML en clusteres separados en cromosoma i| + |parejas CL en el mismo cluster en cromosoma i|
    Objetivo[i] = CGeneral[i] + infeasibility[i] *  $\lambda$ 
retornar CGeneral, infeasibility, Objetivo

```

Como vemos, es bastante intuitivo, no es más que replicar los mecanismos establecidos pero tantas veces como cromosomas haya en la población, y retornar la matriz de estos resultados.

Para generar la solución inicial usamos el siguiente mecanismo:

```

INITIAL_SOLUTION( $T_{Poblacion}$ , k, datos)
Poblacion[ $T_{Poblacion}$ , |datos|] = Matriz(Aleatorio en [0,k-1])
Para cada i en 0.. $T_{Poblacion}$ :
    Si Poblacion[i] tiene un cluster vacío:
        Seleccionar aleatoriamente elementos en un cluster con más de un
        elemento por cada cluster vacío
        Mover los elementos seleccionados a los clústeres vacíos

```

Este código asegura que todas las soluciones creadas son aleatorias pero que no violan la restricción fuerte de no dejar clústeres vacíos.

Los algoritmos genéticos comparten en sus modalidades ciertas operaciones. Consideremos la operación de selección, que es realizada mediante torneo binario

```

TOURNAMENT(|poblacion|)
    Torneo[2] = 2 números aleatorios en [0,|poblacion|-1]
    Ganador = Elemento i de torneo tal que Objetivo[i] es mínimo entre los
    elementos de Torneo

```

Un torneo binario clásico, donde se seleccionan aleatoriamente dos elementos del cromosoma (determinados por su índice) y se conserva aquel que sea mejor, en este caso medido por la minimalidad de su valor de la función objetivo.

El operador de cruce varía a depender de la modalidad (UN o SF), pero es posible contemplar ambos casos desde un mismo fragmento del programa, que está descrito así

```

CROSSOVER(poblacion, padres, |cromosoma|, #hijos, modalidad):
poblacion' = poblacion[padres]
para i en [0..#hijos-1]:
     $T_{Segmento}$  = Número aleatorio en [0,|cromosoma|-1] si modalidad=SF, o si
    modalidad=UN
    InicioSegmento = Número aleatorio en [0,|cromosoma|-1]
    Segmento = [InicioSegmento, InicioSegmento +  $T_{Segmento}$ ] mod |cromosoma|
    Válidos = [0,|cromosoma|-1] que no estén en Segmento
    Cruce = |Válidos|/2 números aleatorios en Válidos
    poblacion'[i, Cruce] = Padre[i', Cruce] donde i' es tal que si i es par, i' es i+1,
    y si i es par, i' es i-1
    Si Cruce tiene un cluster vacío:
        Seleccionar aleatoriamente elementos en un cluster con más de un
        elemento por cada cluster vacío
        Mover los elementos seleccionados a los clústeres vacíos

```

Como vemos, la creación del segmento fijo solo ocurre si se indica como modalidad, en caso contrario queda vacío y el cruce se hace con toda la información

de ambos padres de forma uniforme, pero en caso de que sí se desee, restringe las posibilidades del cruce tras la copia del segmento fijo.

Para la mutación el proceso es sumamente sencillo, un reemplazo con un mecanismo similar a los utilizados previamente para asegurar la validez de una solución.

MUTATION(cromosoma)

clusters_singleton = Clusters tales que tengan un solo elemento en el cromosoma

gen = Número aleatorio en $[0, k-1]$ que no esté en clusters_singleton

mutacion = Número aleatorio en $[0, k-1]$

Cromosoma[gen] = mutacion

3. Procedimiento para desarrollar la práctica:

Todo el proceso de procesamiento de datos, implementación de algoritmos y obtención, cálculo y almacenamiento de los resultados finales fue hecho en el lenguaje de programación Python 3.7.4, a través de su distribución especializada para ciencia de datos, Anaconda. La razón de la elección de este lenguaje es por ser uno de los más utilizados y destacados en problemas de aprendizaje, por lo que se consideró una buena oportunidad para practicar su uso y sintaxis, y en particular, el uso relacionado al paquete de cómputo científico NumPy, una de las principales ventajas del lenguaje contra sus competidores y la principal herramienta detrás de todas las implementaciones.

Las ejecuciones fueron realizadas sobre una distribución Ubuntu 18.04 contenida en Windows Subsystem for Linux (WSL), en un equipo con sistema operativo Windows 10 1909.

El código proporcionado se decidió dejar de lado, pues no se consideró necesario implementar el código en Python cuando NumPy provee las herramientas de randomización necesarias.

El proceso de desarrollo de la práctica se hizo de forma lineal: primero se construyeron las estructuras de datos y mecanismos de lectura de los datos, después se implementó el algoritmo Greedy k-medias con restricciones débiles y por último el algoritmo de Búsqueda Local. Tras esto se depuraron exhaustivamente ambos y se construyó el mecanismo de salida de los datos.

- Manual de ejecución:

Es requisito necesario para la ejecución de la práctica que el equipo donde se ejecuten contenga Python 3.4 en adelante, así como el paquete NumPy 1.17. Además, el fichero P02b.py debe estar en el mismo directorio que un directorio llamado “Instancias y Tablas PAR 2019-20”, que contenga los ficheros de datos y restricciones. Estos ficheros deben ser los contenidos en el fichero comprimido “nuevos conjuntos de datos PAR 2019-20.zip” disponible en la plataforma PRADO.

Una vez se tengan todos los requisitos, tan solo es necesario indicar por terminal la ejecución del script, mediante la sintaxis

```
> <path a Python> P02b.py
```

Durante la ejecución no imprimirá información por pantalla, pues todos los resultados estarán contenidos en el fichero “solutions_P01b.txt” que se creará en el mismo directorio que contiene el script.

4. Experimentos y análisis de resultados:

Una ejecución del programa siguiendo las instrucciones anteriores genera la siguientes tablas de resultados. En todos los casos las semillas correspondientes a cada ejecución son 1, 112, 241, 27, 472, y el tiempo viene expresado en segundos.

Tabla de datos para Algoritmo Genético Generacional (AGG)

Algoritmo	Dataset	%Const	Semilla	N° clústeres	C _{General}	Infeasibility	Función objetivo	Tiempo de ejecución (s)
AGG-UN	iris	10	1	3	0.6693049493711261	0.0	0.6693049493711261	26.4873
AGG-SF	iris	10	1	3	0.6693049493711261	0.0	0.6693049493711261	29.8117
AGG-UN	iris	10	112	3	0.6693049493711261	0.0	0.6693049493711261	27.5531
AGG-SF	iris	10	112	3	0.6693049493711261	0.0	0.6693049493711261	28.8057
AGG-UN	iris	10	241	3	0.6693049493711261	0.0	0.6693049493711261	26.2595
AGG-SF	iris	10	241	3	0.6693049493711261	0.0	0.6693049493711261	29.8325
AGG-UN	iris	10	27	3	0.6693049493711261	0.0	0.6693049493711261	26.6877
AGG-SF	iris	10	27	3	0.6693049493711261	0.0	0.6693049493711261	28.9584
AGG-UN	iris	10	472	3	0.6693049493711261	0.0	0.6693049493711261	27.5440
AGG-SF	iris	10	472	3	0.6693049493711261	0.0	0.6693049493711261	29.5475
AGG-UN	iris	20	1	3	0.6693049493711261	0.0	0.6693049493711261	27.7366
AGG-SF	iris	20	1	3	0.6693049493711261	0.0	0.6693049493711261	30.7933
AGG-UN	iris	20	112	3	0.6693049493711261	0.0	0.6693049493711261	27.1863
AGG-SF	iris	20	112	3	0.6693049493711261	0.0	0.6693049493711261	29.6605
AGG-UN	iris	20	241	3	0.6693049493711261	0.0	0.6693049493711261	27.6508
AGG-SF	iris	20	241	3	0.6693049493711261	0.0	0.6693049493711261	31.3331
AGG-UN	iris	20	27	3	0.6693049493711261	0.0	0.6693049493711261	27.7836
AGG-SF	iris	20	27	3	0.6693049493711261	0.0	0.6693049493711261	30.0764
AGG-UN	iris	20	472	3	0.6693049493711261	0.0	0.6693049493711261	26.6660
AGG-SF	iris	20	472	3	0.6693049493711261	0.0	0.6693049493711261	30.3324
AGG-UN	rand	10	1	3	0.7155992315238897	0.0	0.7155992315238897	26.7371
AGG-SF	rand	10	1	3	0.7155992315238897	0.0	0.7155992315238897	28.5928
AGG-UN	rand	10	112	3	0.7155992315238897	0.0	0.7155992315238897	25.5258
AGG-SF	rand	10	112	3	0.7155992315238897	0.0	0.7155992315238897	29.3260
AGG-UN	rand	10	241	3	0.7155992315238897	0.0	0.7155992315238897	29.0782
AGG-SF	rand	10	241	3	0.7155992315238897	0.0	0.7155992315238897	29.8036
AGG-UN	rand	10	27	3	0.7155992315238897	0.0	0.7155992315238897	27.3353
AGG-SF	rand	10	27	3	0.7155992315238897	0.0	0.7155992315238897	30.3488
AGG-UN	rand	10	472	3	0.7155992315238897	0.0	0.7155992315238897	27.4145
AGG-SF	rand	10	472	3	0.7155992315238897	0.0	0.7155992315238897	31.7049
AGG-UN	rand	20	1	3	0.7155992315238897	0.0	0.7155992315238897	29.6062
AGG-SF	rand	20	1	3	0.7155992315238897	0.0	0.7155992315238897	30.7579
AGG-UN	rand	20	112	3	0.7155992315238897	0.0	0.7155992315238897	26.2390
AGG-SF	rand	20	112	3	0.7155992315238897	0.0	0.7155992315238897	29.2560
AGG-UN	rand	20	241	3	0.7155992315238897	0.0	0.7155992315238897	27.0578
AGG-SF	rand	20	241	3	0.7155992315238897	0.0	0.7155992315238897	28.9683
AGG-UN	rand	20	27	3	0.7155992315238897	0.0	0.7155992315238897	26.5301
AGG-SF	rand	20	27	3	0.7155992315238897	0.0	0.7155992315238897	28.1890
AGG-UN	rand	20	472	3	0.7155992315238897	0.0	0.7155992315238897	27.4146
AGG-SF	rand	20	472	3	0.7155992315238897	0.0	0.7155992315238897	29.1029
AGG-UN	ecoli	10	1	8	21.30642966928693	176.0	26.028533436166818	48.2237
AGG-SF	ecoli	10	1	8	23.967641078835946	134.0	27.56287917407404	51.8983
AGG-UN	ecoli	10	112	8	21.827659885884877	141.0	25.610708926396605	52.4731
AGG-SF	ecoli	10	112	8	24.245404024849677	234.0	30.5236556239968	55.2268
AGG-UN	ecoli	10	241	8	21.807785084171606	95.0	24.35664791288518	52.2524
AGG-SF	ecoli	10	241	8	22.150958777404004	57.0	23.680276474632148	55.0723
AGG-UN	ecoli	10	27	8	19.627436607616275	151.0	23.678786998518905	53.4466
AGG-SF	ecoli	10	27	8	22.53643991017489	97.0	25.13896300896665	55.7135
AGG-UN	ecoli	10	472	8	21.37960007567799	105.0	24.196764254782465	48.9078

AGG-SF	ecoli	10	472	8	27.09873678270365	230.0	33.269667841694414	53.0115
AGG-UN	ecoli	20	1	8	21.88724531708506	164.0	24.08731639029046	53.7236
AGG-SF	ecoli	20	1	8	21.447148249895676	204.0	24.183822023882882	60.6020
AGG-UN	ecoli	20	112	8	24.142095716321297	209.0	26.945844827906228	56.7791
AGG-SF	ecoli	20	112	8	25.45704219605543	475.0	31.829199267839368	59.1981
AGG-UN	ecoli	20	241	8	21.86321320211902	352.0	26.585316968998907	56.7544
AGG-SF	ecoli	20	241	8	25.18775164184043	457.0	31.31843749827255	55.9616
AGG-UN	ecoli	20	27	8	25.098917915452382	173.0	27.41972459633369	53.8915
AGG-SF	ecoli	20	27	8	20.094227044277112	308.0	24.226067840297013	54.1857
AGG-UN	ecoli	20	472	8	21.706724313018114	250.0	25.060491192904397	51.8348
AGG-SF	ecoli	20	472	8	26.506960423905852	227.0	29.552180750842595	54.8179
AGG-UN	newthyroid	10	1	3	10.734480369144938	95.0	14.24534993436233	27.4707
AGG-SF	newthyroid	10	1	3	13.834982364359718	6.0	14.0567214947945	31.7359
AGG-UN	newthyroid	10	112	3	13.834982364359718	6.0	14.0567214947945	27.9428
AGG-SF	newthyroid	10	112	3	13.834982364359716	6.0	14.056721494794498	31.0533
AGG-UN	newthyroid	10	241	3	13.834982364359716	6.0	14.056721494794498	28.0420
AGG-SF	newthyroid	10	241	3	13.834982364359716	6.0	14.056721494794498	31.3651
AGG-UN	newthyroid	10	27	3	13.834982364359716	6.0	14.056721494794498	27.9921
AGG-SF	newthyroid	10	27	3	13.834982364359718	6.0	14.0567214947945	30.9754
AGG-UN	newthyroid	10	472	3	13.834982364359718	6.0	14.0567214947945	29.8609
AGG-SF	newthyroid	10	472	3	10.81574531175799	105.0	14.696180094366685	31.0007
AGG-UN	newthyroid	20	1	3	14.287070556514761	0.0	14.287070556514761	29.7059
AGG-SF	newthyroid	20	1	3	10.886442401267823	233.0	15.190941423219574	33.0093
AGG-UN	newthyroid	20	112	3	14.287070556514763	0.0	14.287070556514763	28.8907
AGG-SF	newthyroid	20	112	3	14.287070556514763	0.0	14.287070556514763	31.1636
AGG-UN	newthyroid	20	241	3	14.287070556514761	0.0	14.287070556514761	28.8539
AGG-SF	newthyroid	20	241	3	10.81533659309185	262.0	15.655588712196392	31.3267
AGG-UN	newthyroid	20	27	3	14.287070556514761	0.0	14.287070556514761	28.7225
AGG-SF	newthyroid	20	27	3	14.287070556514763	0.0	14.287070556514763	32.4039
AGG-UN	newthyroid	20	472	3	10.895306083824044	235.0	15.236753595234607	30.8342
AGG-SF	newthyroid	20	472	3	14.287070556514761	0.0	14.287070556514761	31.2086

Tabla de datos para Algoritmo Genético Estacionario (AGE)

Algoritmo	Dataset	%Const	Semilla	Nº clústeres	C _{General}	Infeasibility	Función objetivo	Tiempo de ejecución (s)
AGE-UN	iris	10	1	3	0.6693049493711261	0.0	0.6693049493711261	36.5475
AGE-SF	iris	10	1	3	0.6693049493711261	0.0	0.6693049493711261	39.9933
AGE-UN	iris	10	112	3	0.6693049493711261	0.0	0.6693049493711261	37.0933
AGE-SF	iris	10	112	3	0.6693049493711261	0.0	0.6693049493711261	40.3560
AGE-UN	iris	10	241	3	0.6693049493711261	0.0	0.6693049493711261	36.0660
AGE-SF	iris	10	241	3	0.6693049493711261	0.0	0.6693049493711261	39.4688
AGE-UN	iris	10	27	3	0.6693049493711261	0.0	0.6693049493711261	36.0359
AGE-SF	iris	10	27	3	0.6693049493711261	0.0	0.6693049493711261	39.6101
AGE-UN	iris	10	472	3	0.6693049493711261	0.0	0.6693049493711261	36.2983
AGE-SF	iris	10	472	3	0.6693049493711261	0.0	0.6693049493711261	39.8627
AGE-UN	iris	20	1	3	0.6693049493711261	0.0	0.6693049493711261	36.4991
AGE-SF	iris	20	1	3	0.6693049493711261	0.0	0.6693049493711261	40.5745
AGE-UN	iris	20	112	3	0.6693049493711261	0.0	0.6693049493711261	37.5989
AGE-SF	iris	20	112	3	0.6693049493711261	0.0	0.6693049493711261	40.9212
AGE-UN	iris	20	241	3	0.6693049493711261	0.0	0.6693049493711261	36.8417
AGE-SF	iris	20	241	3	0.6693049493711261	0.0	0.6693049493711261	40.5481
AGE-UN	iris	20	27	3	0.6693049493711261	0.0	0.6693049493711261	36.6126
AGE-SF	iris	20	27	3	0.6693049493711261	0.0	0.6693049493711261	40.4402
AGE-UN	iris	20	472	3	0.6693049493711261	0.0	0.6693049493711261	36.7935
AGE-SF	iris	20	472	3	0.6693049493711261	0.0	0.6693049493711261	40.3117
AGE-UN	rand	10	1	3	0.7155992315238897	0.0	0.7155992315238897	36.2596
AGE-SF	rand	10	1	3	0.7155992315238897	0.0	0.7155992315238897	39.4286
AGE-UN	rand	10	112	3	0.7155992315238897	0.0	0.7155992315238897	35.5717
AGE-SF	rand	10	112	3	0.7155992315238897	0.0	0.7155992315238897	40.2121
AGE-UN	rand	10	241	3	0.7155992315238897	0.0	0.7155992315238897	35.9463
AGE-SF	rand	10	241	3	0.7155992315238897	0.0	0.7155992315238897	40.0011

AGE-UN	rand	10	27	3	0.7155992315238897	0.0	0.7155992315238897	37.1766
AGE-SF	rand	10	27	3	0.7155992315238897	0.0	0.7155992315238897	39.9619
AGE-UN	rand	10	472	3	0.7155992315238897	0.0	0.7155992315238897	36.2106
AGE-SF	rand	10	472	3	0.7155992315238897	0.0	0.7155992315238897	40.4062
AGE-UN	rand	20	1	3	0.7155992315238897	0.0	0.7155992315238897	37.0659
AGE-SF	rand	20	1	3	0.7155992315238897	0.0	0.7155992315238897	40.4492
AGE-UN	rand	20	112	3	0.7155992315238897	0.0	0.7155992315238897	37.0653
AGE-SF	rand	20	112	3	0.7155992315238897	0.0	0.7155992315238897	41.0772
AGE-UN	rand	20	241	3	0.7155992315238897	0.0	0.7155992315238897	36.3815
AGE-SF	rand	20	241	3	0.7155992315238897	0.0	0.7155992315238897	40.2693
AGE-UN	rand	20	27	3	0.7155992315238897	0.0	0.7155992315238897	36.9933
AGE-SF	rand	20	27	3	0.7155992315238897	0.0	0.7155992315238897	42.2011
AGE-UN	rand	20	472	3	0.7155992315238897	0.0	0.7155992315238897	37.0236
AGE-SF	rand	20	472	3	0.7155992315238897	0.0	0.7155992315238897	39.7159
AGE-UN	ecoli	10	1	8	21.41493766195236	88.0	23.775989545392306	60.2293
AGE-SF	ecoli	10	1	8	21.971081332941093	63.0	23.66137984040378	64.8887
AGE-UN	ecoli	10	112	8	21.992238922258156	76.0	24.031329185229016	59.3925
AGE-SF	ecoli	10	112	8	19.594976291479938	121.0	22.84142263120986	64.9409
AGE-UN	ecoli	10	241	8	22.18328525928542	68.0	24.00773444194356	61.0637
AGE-SF	ecoli	10	241	8	22.138915390210567	63.0	23.829213897673252	67.2478
AGE-UN	ecoli	10	27	8	21.47952435722371	97.0	24.082047456015466	60.2652
AGE-SF	ecoli	10	27	8	22.345375451523342	55.0	23.821032878673307	64.0493
AGE-UN	ecoli	10	472	8	21.20301840680728	117.0	24.342144206380844	60.0301
AGE-SF	ecoli	10	472	8	21.36397892647126	114.0	24.422614320927547	64.3617
AGE-UN	ecoli	20	1	8	21.943959148303467	166.0	24.17086035654796	61.7295
AGE-SF	ecoli	20	1	8	21.936726096340035	134.0	23.734345143959082	66.8659
AGE-UN	ecoli	20	112	8	21.924971966738898	170.0	24.20553344506157	64.1653
AGE-SF	ecoli	20	112	8	21.829652941752958	145.0	23.774837732087	65.9862
AGE-UN	ecoli	20	241	8	21.756584959734436	158.0	23.876165627822566	61.6066
AGE-SF	ecoli	20	241	8	21.929599473409553	165.0	24.1430856141345	65.4768
AGE-UN	ecoli	20	27	8	22.410011219074917	164.0	24.610082292280318	61.2875
AGE-SF	ecoli	20	27	8	21.45003777886043	256.0	24.884295063863984	65.8473
AGE-UN	ecoli	20	472	8	21.89522293798951	139.0	23.75991732320628	61.1079
AGE-SF	ecoli	20	472	8	21.97589901514699	146.0	23.934498873000578	67.2511
AGE-UN	newthyroid	10	1	3	10.895306083824044	99.0	14.554001735997957	38.7799
AGE-SF	newthyroid	10	1	3	10.870556121849784	98.0	14.492295252284567	42.5343
AGE-UN	newthyroid	10	112	3	10.8189377533239	108.0	14.810242101149987	38.3558
AGE-SF	newthyroid	10	112	3	13.834982364359716	6.0	14.056721494794498	42.1526
AGE-UN	newthyroid	10	241	3	13.834982364359716	6.0	14.056721494794498	38.5201
AGE-SF	newthyroid	10	241	3	10.802488557001531	108.0	14.793792904827619	42.0810
AGE-UN	newthyroid	10	27	3	10.873519998842271	96.0	14.421346085798792	38.6862
AGE-SF	newthyroid	10	27	3	13.834982364359716	6.0	14.056721494794498	42.3995
AGE-UN	newthyroid	10	472	3	13.834982364359718	6.0	14.0567214947945	38.1064
AGE-SF	newthyroid	10	472	3	13.834982364359716	6.0	14.056721494794498	42.1080
AGE-UN	newthyroid	20	1	3	10.873519998842271	231.0	15.141070531335206	39.0706
AGE-SF	newthyroid	20	1	3	14.287070556514763	0.0	14.287070556514763	43.7087
AGE-UN	newthyroid	20	112	3	10.8189377533239	268.0	15.770035340804881	38.6446
AGE-SF	newthyroid	20	112	3	14.287070556514761	0.0	14.287070556514761	42.9336
AGE-UN	newthyroid	20	241	3	14.287070556514761	0.0	14.287070556514761	39.6138
AGE-SF	newthyroid	20	241	3	14.287070556514763	0.0	14.287070556514763	43.7423
AGE-UN	newthyroid	20	27	3	10.895462136911602	239.0	15.310806627239792	38.8007
AGE-SF	newthyroid	20	27	3	10.806049363447713	252.0	15.461559035258189	42.7163
AGE-UN	newthyroid	20	472	3	14.287070556514761	0.0	14.287070556514761	39.3199
AGE-SF	newthyroid	20	472	3	14.287070556514763	0.0	14.287070556514763	43.4332

Una visión rápida a los resultados indica que el AGE es más poderoso, pero tarda más en converger. Por esa razón, resulta relevante conocer cuándo el conjunto de datos es de dificultad necesaria para usarlos. En ambos casos, son ligeramente mejores que el Búsqueda Local implementado previamente pero toman mucho más en hallar una solución (~30 segundos)

Resultados Práctica I
Resultados globales en el PAR con 10% de restricciones

	Iris				Ecoli			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
COPKM	0.67	4.40	0.70	0.02	37.63	291.20	45.44	1.13
BL	0.67	0.00	0.67	0.33	21.26	99.60	23.93	5.07

	Rand				Newthyroid			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
COPKM	0.71	1.80	0.73	0.02	14.98	78.40	17.88	0.04
BL	0.72	0.00	0.72	0.28	12.02	69.60	14.60	0.59

Resultados globales en el PAR con 20% de restricciones

	Iris				Ecoli			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
COPKM	0.67	6.20	0.69	0.02	37.31	222.80	40.30	0.30
BL	0.67	0.00	0.67	0.34	21.85	180.40	24.27	4.13

	Rand				Newthyroid			
	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T	<i>Tasa_C</i>	<i>Tasa_inf</i>	<i>Agr.</i>	T
COPKM	0.72	0.00	0.72	0.02	14.51	170.20	17.66	0.03
BL	0.72	0.00	0.72	0.28	12.22	144.60	14.89	0.54