Universidad Simón Bolívar
Departamento de Computación y T.I
Inteligencia Artificial II
CI-5438
Septiembre-Diciembre 2014

INFORME PROYECTO 2 – GABIL

Profesora:
Ivette Carolina Martínez

Integrantes:
Carla Barazarte 08-10096
Ramón Márquez 10-10849
Grupo 10

Sartenejas, diciembre de 2014

INFORME

Resumen:

El presente proyecto se encarga de crear un clasificador, utilizando un algoritmo genético, que permita clasificar las tres clases de la flor Iris (Setosa, Versicolor, Virginica). El algoritmo genético empleado está basado en el sistema GABIL[1]. Además, se utiliza un librería de python para facilitar la implementación del algoritmo; este se llama “pyevolve”.

En este experimento se requiere realizar distintas codificaciones para la selección de los padres y de los sobrevivientes. Sin embargo, una de las funciones a utilizar es el de “rueda de ruleta”. De igual manera, se realizará un mecanismo de penalización de los tamaños de los clasificadores en la función fitness y compararlo con los resultados de no tener dicha penalización.

Descripción de la implementación:

Inicialmente, se utiliza una escala fijada para transformar a cada individuo en valores 0's y 1's. Para la implementación se utiliza una librería de python llamada “pyevolve” el cual permite codificar algoritmos genéticos. Cabe mencionar que esta librería no te permite modificar el tipo de selección de los padres pero si lo permite con los sobrevivientes. Por otra parte, la librería permite crear nuestros propios operadores de crossover y del fitness el cual nos es útil puesto que lo adaptamos de acuerdo al sistema GABIL. Para la operación de mutación, pyevolve te da las herramientas para usarlo. En nuestra implementación, usamos el método swap para la mutación.

El otro método empleado para selección de los sobrevivientes, además del de “rueda de ruleta”, es el “rank selector”. Este selector escoge el mejor individuo de la población cada vez. En el caso de la penalización, lo que se hace es dividir el fitness que se obtiene entre el tamaño del individuo.

Para dar un resultado, se realizan 10 corridas con cada configuración debido a que los algoritmos genéticos son estocásticos.

Descripción del algoritmo genético:

Como se dijo anteriormente, el algoritmo genético empleado está basado en el sistema GABIL. Los parámetros que se usaron para la tasa de mutación son: 0.001, 0.015 y 0,02. Para la tasa de crossover se usaron los siguientes valores: 0.6, 0.7 y 0.85.

Descripción y análisis de los experimentos realizados:

Selección por “rueda de ruleta” y sin penalización

(Tasa de mutación, Tasa de Crossover)

(0.001 , 0.6)

```
Gen. 1 (1.00%): Max/Min/Avg Fitness(Raw) [0.01(0.04)/0.01(0.00)/0.01(0.01)]
Gen. 20 (20.00%): Max/Min/Avg Fitness(Raw) [0.02(0.24)/0.02(0.00)/0.02(0.02)]
Gen. 40 (40.00%): Max/Min/Avg Fitness(Raw) [0.07(0.28)/0.06(0.00)/0.06(0.06)]
Gen. 60 (60.00%): Max/Min/Avg Fitness(Raw) [0.25(0.77)/0.20(0.02)/0.21(0.21)]
Gen. 80 (80.00%): Max/Min/Avg Fitness(Raw) [0.31(0.83)/0.24(0.05)/0.26(0.26)]
Gen. 100 (100.00%): Max/Min/Avg Fitness(Raw) [0.72(0.90)/0.43(0.18)/0.60(0.60)]
Total time elapsed: 18.385 seconds.
- GenomeBase
        Score:                  0.896178
        Fitness:                0.717421

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111110011111111111111111111111


Porcentaje correctamente clasificados:  0.946666666667
```

(0.015 , 0.6)

```
Gen. 1 (1.00%): Max/Min/Avg Fitness(Raw) [0.00(0.07)/0.00(0.00)/0.00(0.00)]
Gen. 20 (20.00%): Max/Min/Avg Fitness(Raw) [0.03(0.26)/0.02(0.00)/0.02(0.02)]
Gen. 40 (40.00%): Max/Min/Avg Fitness(Raw) [0.08(0.37)/0.07(0.00)/0.07(0.07)]
Gen. 60 (60.00%): Max/Min/Avg Fitness(Raw) [0.08(0.60)/0.06(0.00)/0.06(0.06)]
Gen. 80 (80.00%): Max/Min/Avg Fitness(Raw) [0.13(0.68)/0.11(0.00)/0.11(0.11)]
Gen. 100 (100.00%): Max/Min/Avg Fitness(Raw) [0.44(0.87)/0.32(0.02)/0.37(0.37)]
Total time elapsed: 17.858 seconds.
- GenomeBase
        Score:                  0.871111
        Fitness:                0.444575

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111111111111101001111111111111


Porcentaje correctamente clasificados:  0.933333333333
```

(0.02 , 0.6)

```
Gen. 1 (1.00%): Max/Min/Avg Fitness(Raw) [0.01(0.11)/0.01(0.00)/0.01(0.01)]
Gen. 20 (20.00%): Max/Min/Avg Fitness(Raw) [0.02(0.44)/0.02(0.00)/0.02(0.02)]
Gen. 40 (40.00%): Max/Min/Avg Fitness(Raw) [0.02(0.53)/0.02(0.00)/0.02(0.02)]
Gen. 60 (60.00%): Max/Min/Avg Fitness(Raw) [0.03(0.80)/0.03(0.00)/0.03(0.03)]
Gen. 80 (80.00%): Max/Min/Avg Fitness(Raw) [0.04(0.88)/0.03(0.00)/0.03(0.03)]
Gen. 100 (100.00%): Max/Min/Avg Fitness(Raw) [0.11(0.88)/0.09(0.00)/0.09(0.09)]
Total time elapsed: 16.794 seconds.
- GenomeBase
        Score:                  0.883600
        Fitness:                0.105537

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111111111111111101011101111111


Porcentaje correctamente clasificados:  0.94
```

(0.001 , 0.7)

```
Gen. 1 (1.00%): Max/Min/Avg Fitness(Raw) [0.01(0.11)/0.01(0.00)/0.01(0.01)]
Gen. 20 (20.00%): Max/Min/Avg Fitness(Raw) [0.03(0.58)/0.03(0.00)/0.03(0.03)]
Gen. 40 (40.00%): Max/Min/Avg Fitness(Raw) [0.08(0.58)/0.07(0.00)/0.07(0.07)]
Gen. 60 (60.00%): Max/Min/Avg Fitness(Raw) [0.09(0.91)/0.08(0.00)/0.08(0.08)]
Gen. 80 (80.00%): Max/Min/Avg Fitness(Raw) [0.20(0.93)/0.16(0.00)/0.17(0.17)]
Gen. 100 (100.00%): Max/Min/Avg Fitness(Raw) [0.54(0.95)/0.37(0.01)/0.45(0.45)]
Total time elapsed: 17.120 seconds.
- GenomeBase
        Score:                  0.947378
        Fitness:                0.537041

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111111011111011111011111111111


Porcentaje correctamente clasificados:  0.973333333333
```

(0.015 , 0.7)

```
Gen. 1 (1.00%): Max/Min/Avg Fitness(Raw) [0.01(0.12)/0.01(0.00)/0.01(0.01)]
Gen. 20 (20.00%): Max/Min/Avg Fitness(Raw) [0.03(0.44)/0.02(0.00)/0.02(0.02)]
Gen. 40 (40.00%): Max/Min/Avg Fitness(Raw) [0.10(0.87)/0.08(0.00)/0.08(0.08)]
Gen. 60 (60.00%): Max/Min/Avg Fitness(Raw) [0.10(0.87)/0.08(0.00)/0.09(0.09)]
Gen. 80 (80.00%): Max/Min/Avg Fitness(Raw) [0.16(0.93)/0.13(0.00)/0.13(0.13)]
Gen. 100 (100.00%): Max/Min/Avg Fitness(Raw) [0.25(0.93)/0.20(0.00)/0.21(0.21)]
Total time elapsed: 18.428 seconds.
- GenomeBase
        Score:                  0.934444
        Fitness:                0.250871

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111111111111011111111101111111


Porcentaje correctamente clasificados:  0.966666666667
```

(0.02 , 0.7)

```
Gen. 1 (1.00%): Max/Min/Avg Fitness(Raw) [0.00(0.04)/0.00(0.00)/0.00(0.00)]
Gen. 20 (20.00%): Max/Min/Avg Fitness(Raw) [0.01(0.23)/0.01(0.00)/0.01(0.01)]
Gen. 40 (40.00%): Max/Min/Avg Fitness(Raw) [0.03(0.49)/0.03(0.00)/0.03(0.03)]
Gen. 60 (60.00%): Max/Min/Avg Fitness(Raw) [0.02(0.49)/0.01(0.00)/0.01(0.01)]
Gen. 80 (80.00%): Max/Min/Avg Fitness(Raw) [0.10(0.69)/0.08(0.00)/0.09(0.09)]
Gen. 100 (100.00%): Max/Min/Avg Fitness(Raw) [0.20(0.85)/0.16(0.00)/0.17(0.17)]
Total time elapsed: 17.694 seconds.
- GenomeBase
        Score:                  0.846400
        Fitness:                0.199281

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111110011111111111010101111111


Porcentaje correctamente clasificados:  0.92
```

(0.001 , 0.85)

```
Gen. 1 (1.00%): Max/Min/Avg Fitness(Raw) [0.01(0.08)/0.01(0.00)/0.01(0.01)]
Gen. 20 (20.00%): Max/Min/Avg Fitness(Raw) [0.04(0.38)/0.03(0.00)/0.03(0.03)]
Gen. 40 (40.00%): Max/Min/Avg Fitness(Raw) [0.08(0.53)/0.06(0.00)/0.07(0.07)]
Gen. 60 (60.00%): Max/Min/Avg Fitness(Raw) [0.10(0.85)/0.08(0.00)/0.08(0.08)]
Gen. 80 (80.00%): Max/Min/Avg Fitness(Raw) [0.57(0.92)/0.39(0.06)/0.48(0.48)]
Gen. 100 (100.00%): Max/Min/Avg Fitness(Raw) [0.99(0.97)/0.31(0.38)/0.83(0.83)]
Total time elapsed: 17.616 seconds.
- GenomeBase
        Score:                  0.973511
        Fitness:                0.994864

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111111111111111011111111111111


Porcentaje correctamente clasificados:  0.986666666667
```

FIGURA 1

(0.015 , 0.85)

```
Gen. 1 (1.00%): Max/Min/Avg Fitness(Raw) [0.01(0.09)/0.01(0.00)/0.01(0.01)]
Gen. 20 (20.00%): Max/Min/Avg Fitness(Raw) [0.02(0.45)/0.02(0.00)/0.02(0.02)]
Gen. 40 (40.00%): Max/Min/Avg Fitness(Raw) [0.07(0.73)/0.06(0.00)/0.06(0.06)]
Gen. 60 (60.00%): Max/Min/Avg Fitness(Raw) [0.04(0.75)/0.04(0.00)/0.04(0.04)]
Gen. 80 (80.00%): Max/Min/Avg Fitness(Raw) [0.11(0.85)/0.09(0.00)/0.09(0.09)]
Gen. 100 (100.00%): Max/Min/Avg Fitness(Raw) [0.05(0.96)/0.04(0.00)/0.04(0.04)]
Total time elapsed: 17.753 seconds.
- GenomeBase
        Score:                  0.960400
        Fitness:                0.046463

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111111111111111111111111111011


Porcentaje correctamente clasificados:  0.98
```

Podemos notar que esta configuración da buenos resultados. El porcentaje de clasificados por el mejor individuo obtenido en todos los casos da mayor al 90% y muy parecidos en todos los casos.

A continuación se darán algunos resultados por el otro método de selección "**rank selector" sin penalización**:

(0.001 , 0.6)

```
Gen. 1 (0.50%): Max/Min/Avg Fitness(Raw) [0.11(0.09)/0.11(0.09)/0.11(0.09)]
Total time elapsed: 0.358 seconds.
- GenomeBase
        Score:                  0.094044
        Fitness:                0.113281

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          011111011011101011010111111110


Porcentaje correctamente clasificados:  0.893333333333
```

(0.015 , 0.6)

```
Gen. 1 (0.50%): Max/Min/Avg Fitness(Raw) [0.10(0.21)/0.08(0.03)/0.08(0.08)]
Gen. 20 (10.00%): Max/Min/Avg Fitness(Raw) [0.67(0.61)/0.00(0.16)/0.58(0.56)]
Gen. 40 (20.00%): Max/Min/Avg Fitness(Raw) [0.69(0.61)/0.00(0.28)/0.59(0.57)]
Gen. 60 (30.00%): Max/Min/Avg Fitness(Raw) [0.69(0.61)/0.00(0.18)/0.61(0.58)]
Gen. 80 (40.00%): Max/Min/Avg Fitness(Raw) [0.69(0.61)/0.00(0.26)/0.62(0.58)]
Gen. 100 (50.00%): Max/Min/Avg Fitness(Raw) [0.67(0.61)/0.00(0.06)/0.58(0.56)]
Gen. 120 (60.00%): Max/Min/Avg Fitness(Raw) [0.67(0.61)/0.00(0.24)/0.57(0.56)]
Gen. 140 (70.00%): Max/Min/Avg Fitness(Raw) [0.67(0.61)/0.00(0.19)/0.58(0.56)]
Gen. 160 (80.00%): Max/Min/Avg Fitness(Raw) [0.69(0.61)/0.00(0.24)/0.61(0.58)]
Gen. 180 (90.00%): Max/Min/Avg Fitness(Raw) [0.63(0.61)/0.10(0.17)/0.52(0.52)]
Gen. 200 (100.00%): Max/Min/Avg Fitness(Raw) [0.67(0.61)/0.00(0.19)/0.57(0.56)]
Total time elapsed: 35.904 seconds.
- GenomeBase
        Score:                  0.608400
        Fitness:                0.669347

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111110001111001001010101111011


Porcentaje correctamente clasificados:  0.78
```

(0.015 , 0.7)

```
Gen. 1 (0.50%): Max/Min/Avg Fitness(Raw) [0.22(0.21)/0.00(0.05)/0.18(0.18)]
Gen. 20 (10.00%): Max/Min/Avg Fitness(Raw) [0.49(0.44)/0.00(0.11)/0.43(0.41)]
Gen. 40 (20.00%): Max/Min/Avg Fitness(Raw) [0.50(0.44)/0.00(0.12)/0.45(0.42)]
Gen. 60 (30.00%): Max/Min/Avg Fitness(Raw) [0.50(0.44)/0.00(0.00)/0.46(0.42)]
Gen. 80 (40.00%): Max/Min/Avg Fitness(Raw) [0.49(0.44)/0.00(0.06)/0.43(0.40)]
Gen. 100 (50.00%): Max/Min/Avg Fitness(Raw) [0.49(0.44)/0.00(0.00)/0.44(0.41)]
Gen. 120 (60.00%): Max/Min/Avg Fitness(Raw) [0.50(0.44)/0.00(0.11)/0.44(0.41)]
Gen. 140 (70.00%): Max/Min/Avg Fitness(Raw) [0.49(0.44)/0.00(0.11)/0.44(0.41)]
Gen. 160 (80.00%): Max/Min/Avg Fitness(Raw) [0.50(0.44)/0.00(0.05)/0.46(0.42)]
Gen. 180 (90.00%): Max/Min/Avg Fitness(Raw) [0.48(0.44)/0.00(0.11)/0.42(0.40)]
Gen. 200 (100.00%): Max/Min/Avg Fitness(Raw) [0.50(0.44)/0.00(0.12)/0.45(0.42)]
Total time elapsed: 35.392 seconds.
- GenomeBase
        Score:                  0.435600
        Fitness:                0.499415

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111110011111101111000111110001


Porcentaje correctamente clasificados:  0.66
```

(0.015 , 0.85)

```
Gen. 1 (0.50%): Max/Min/Avg Fitness(Raw) [0.09(0.14)/0.06(0.00)/0.08(0.08)]
Gen. 20 (10.00%): Max/Min/Avg Fitness(Raw) [0.49(0.44)/0.00(0.03)/0.44(0.41)]
Gen. 40 (20.00%): Max/Min/Avg Fitness(Raw) [0.82(0.74)/0.00(0.09)/0.72(0.69)]
Gen. 60 (30.00%): Max/Min/Avg Fitness(Raw) [0.82(0.74)/0.00(0.19)/0.70(0.68)]
Gen. 80 (40.00%): Max/Min/Avg Fitness(Raw) [0.84(0.74)/0.00(0.34)/0.74(0.70)]
Gen. 100 (50.00%): Max/Min/Avg Fitness(Raw) [0.80(0.74)/0.00(0.21)/0.67(0.67)]
Gen. 120 (60.00%): Max/Min/Avg Fitness(Raw) [0.82(0.74)/0.00(0.11)/0.69(0.68)]
Gen. 140 (70.00%): Max/Min/Avg Fitness(Raw) [0.83(0.74)/0.00(0.23)/0.70(0.69)]
Gen. 160 (80.00%): Max/Min/Avg Fitness(Raw) [0.82(0.74)/0.00(0.33)/0.69(0.68)]
Gen. 180 (90.00%): Max/Min/Avg Fitness(Raw) [0.82(0.74)/0.00(0.14)/0.70(0.68)]
Gen. 200 (100.00%): Max/Min/Avg Fitness(Raw) [0.81(0.74)/0.00(0.28)/0.68(0.67)]
Total time elapsed: 34.560 seconds.
- GenomeBase
        Score:                  0.739600
        Fitness:                0.807663

        Slot [Evaluator] (Count: 1)
                Name: eval_func
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111110011111100101101010111101 1


Porcentaje correctamente clasificados:  0.86
```

Podemos notar que con este tipo de selección los resultados son relativamente buenos, pero el mejor sigue siendo el método “rueda de ruleta”.

A continuación, se presentará el mejor resultado obtenido utilizando la penalización en la mejor configuración:

Selección por “rueda de ruleta” y con penalización

(0.001 , 0.6)

```
Gen. 1 (0.50%): Max/Min/Avg Fitness(Raw) [0.00(0.00)/0.00(0.00)/0.00(0.00)]
Gen. 20 (10.00%): Max/Min/Avg Fitness(Raw) [0.00(0.02)/0.00(0.00)/0.00(0.00)]
Gen. 40 (20.00%): Max/Min/Avg Fitness(Raw) [0.00(0.02)/0.00(0.00)/0.00(0.00)]
Gen. 60 (30.00%): Max/Min/Avg Fitness(Raw) [0.00(0.02)/0.00(0.00)/0.00(0.00)]
Gen. 80 (40.00%): Max/Min/Avg Fitness(Raw) [0.00(0.02)/0.00(0.00)/0.00(0.00)]
Gen. 100 (50.00%): Max/Min/Avg Fitness(Raw) [0.00(0.02)/0.00(0.00)/0.00(0.00)]
Gen. 120 (60.00%): Max/Min/Avg Fitness(Raw) [0.00(0.02)/0.00(0.00)/0.00(0.00)]
Gen. 140 (70.00%): Max/Min/Avg Fitness(Raw) [0.01(0.02)/0.01(0.00)/0.01(0.01)]
Gen. 160 (80.00%): Max/Min/Avg Fitness(Raw) [0.02(0.03)/0.01(0.00)/0.01(0.01)]
Gen. 180 (90.00%): Max/Min/Avg Fitness(Raw) [0.03(0.03)/0.02(0.02)/0.03(0.03)]

        Evolution stopped by Termination Criteria function !

Gen. 192 (96.00%): Max/Min/Avg Fitness(Raw) [0.03(0.03)/0.03(0.03)/0.03(0.03)]
Total time elapsed: 33.995 seconds.
- GenomeBase
        Score:                  0.030295
        Fitness:                0.034180

        Slot [Evaluator] (Count: 1)
                Name: eval_func_1
        Slot [Initializator] (Count: 1)
                Name: G1DBinaryStringInitializator
                Doc:  1D Binary String initializator
        Slot [Mutator] (Count: 1)
                Name: G1DBinaryStringMutatorSwap
                Doc:  The 1D Binary String Swap Mutator
        Slot [Crossover] (Count: 1)
                Name: crossoverGabil

- G1DBinaryString
        String length:   30
        String:          111111011111101111011101111011


Porcentaje correctamente clasificados:  0.953333333333
```

Con los resultados obtenidos tenemos que la mejor configuración obtenida es la siguiente: tasa de mutación de 0.001, tasa de crossover de 0.85 y utilizando el método de selección “rueda de ruleta” el cual arroja un porcentaje de ejemplos bien clasificados de aproximadamente 98.667%. De igual manera, el mejor conjunto de reglas hallado está presente en “FIGURA 1”.

La función fitness empleada es la descrita en el sistema GABIL. Este toma un individuo y lo compara con cada ejemplo de la población dada en el archivo de entrada y que ha sido codificado en 0's y 1's. Cada ejemplo es comparado con cada regla del individuo. Si una de sus reglas hace match con el ejemplo entonces le suma 1 a una variable temporal que registra la cantidad de ejemplos con las que el individuo hace match. Luego, se divide la suma total entre el tamaño de la

población y el resultado que de se eleva al cuadrado y es lo que se retorna. En cuanto a la penalización, ayuda a reducir el tiempo de ejecución del algoritmo y no es muy necesario puesto que para el problema no hace falta tener un número de reglas grandes.

Referencias:

[1] Tom M. Mitchell, Machine Learning, McGraw-Hill, 1997..