```
>>gp=rungp(@Y8_config);
Press a key to continue


------------------------------------------------------------------------
GPTIPS 2
Symbolic data mining platform for MATLAB
Copyright (C) Dominic Searson 2009-2015

Contact: searson@gmail.com

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details: http://www.gnu.org/licenses

------------------------------------------------------------------------

Run parameters
--------------
Population size:         100
Number of generations:   100
Number of runs:          1
Parallel mode :          off
Tournament type:         regular
Tournament size:         15
Elite fraction:          0.3
Fitness cache:           enabled
Lexicographic selection: True
Max tree depth:          4
Max nodes per tree:      Inf
Using function set:      TIMES MINUS PLUS
Number of inputs:        4
Max genes:               12
Constants range:         [-10  10]
Complexity measure:      expressional
Fitness function:        regressmulti_fitfun.m

Generation 0
Best fitness:    0.62723
Mean fitness:    2.8731
Best complexity: 108
Inputs in best individual: x1 x2 x3 x4

Generation 20
Best fitness:    0.57969
```

```
Mean fitness:    0.82396
Best complexity: 126
Inputs in best individual: x1 x2 x3 x4

Generation 40
Best fitness:    0.57096
Mean fitness:    0.72005
Best complexity: 162
Inputs in best individual: x1 x2 x3 x4

Generation 60
Best fitness:    0.55916
Mean fitness:    0.64161
Best complexity: 166
Inputs in best individual: x1 x2 x3 x4

Generation 80
Best fitness:    0.55577
Mean fitness:    0.66603
Best complexity: 174
Inputs in best individual: x1 x2 x3 x4

Finalising run.
GPTIPS run complete in 0.47 min.
Best fitness acheived: 0.55269
------------------------------------------------------------

Evaluate the best individual of
the runs on the fitness function using:
>>runtree(gp,'best');
Press a key to continue


Next, use the the GPPRETTY command on the best individual:
>>gppretty(gp,'best')
Press a key to continue


Simplified genes
----------------

Gene 1 and bias term

  0.0003904 x1 (x3 + 3.0 x4) + 91.1

Gene 2

            3
  0.07807 x2

Gene 3
```

```
                    2
  - 0.003669 x1 (x2  + x4)

Gene 4

  0.07021 x1 x2

Gene 5

                 3
  - 0.0001363 x2  (x1 + x4)

Gene 6

  0.3698 x4

Gene 7

  0.001061 x4 (x2 + x3)

Gene 8

  -0.0002754 x1 (x3 - 1.0 x2 x4)

Gene 9

  - 0.3506 x1 - 0.1753 x4

Gene 10

  -13.03 x2

Gene 11

  - 0.04754 x2 - 0.02377 x3 - 0.02377 x4

Gene 12

  0.0005219 (x2 - 1.0 x4) (x2 + 2.0 x3 + x4)

Simplified overall GP expression
--------------------------------

  0.1708 x4 - 13.07 x2 - 0.02377 x3 - 0.3506 x1 + 0.001061 x4 (x2 + x3) +

     0.0005219 (x2 - 1.0 x4) (x2 + 2.0 x3 + x4) + 0.07021 x1 x2 - 0.0002754 x1 x3 -

                    2                       3                          3
     0.003669 x1 (x2  + x4) - 0.0001363 x2  (x1 + x4) + 0.07807 x2  +
```

```
     0.0003904 x1 (x3 + 3.0 x4) + 0.0002754 x1 x2 x4 + 91.1

Next, use the the DRAWTREE command:
>>drawtrees(gp,'best')
Press a key to continue

Trees drawn to trees.htm
Opening in system browser.

Finally, an HTML report listing the models on the Pareto optimal front
of model expressional complexity and performance can be generated using
the PARETOREPORT function.
>>paretoreport(gp)

Press a key to continue

100 models passed R^2 training (>= 0) and expressional complexity (<= Inf) filter ...
Computing pareto front on training data...
Removing genotype duplicates from 51 remaining models ...
6 models passed the filtering process.
Model report created in pareto.htm
Opening report in system browser.
end
>>
```