Homework 2+: GA and PSO performance evaluation

Frederik Darwin

M10601836

Problem 4

Maximize $f(x,y) = 8 - \frac{\sin^2(\sqrt{(x^2+y^2)})}{(1+0.001*(x^2-y^2))^8}$ $-1 \leqq x \leqq 2;\ -1 \leqq y \leqq 1;\ x+y \geqq -1$

Maximum=8 at (x,y)=(0,0)

The problem assigned to me is already maximized using both algorithm, Genetic Algorithm and Particle Swarm Optimization. Now we should see which algorithm is superior to other using various test given.

- Statistic Performance Evaluation (Student t Test).
  By collecting the last generation/iteration population mean value data for each trial, the mean for each trial will be used again to see whether GA have better algorithm than PSO or the other way around. The data that I take is 15 for each algorithm with more details below:

Trial no.	GA MEAN	PSO MEAN
1	7.94935	7.893741445
2	7.92335	7.894756757
3	7.8618	7.916197384
4	7.8997	7.787912767
5	7.8842	7.859891394
6	7.8944	7.802619007
7	7.8824	7.926735325
8	7.9658	7.945153209
9	7.86265	7.883161526
10	7.9274	7.908277081
11	7.94065	7.954226199
12	7.914	7.909218413
13	7.91035	7.879334166
14	7.9079	7.915376361
15	7.84545	7.865541191

Review your data:

Group	GA	PSO
Mean	7.90462666667	7.88947614833
SD	0.03399445788	0.04652577095
SEM	0.00877733128	0.01201290240
N	15	15

Unpaired t test results

- P value and statistical significance:
  The two-tailed P value equals 0.3172
  By conventional criteria, this difference is considered to be not statistically significant.
- Confidence interval:
  The mean of GA minus PSO equals 0.01515051833
  95% confidence interval of this difference: From -0.01532543127 to 0.04562646794
- Intermediate values used in calculations:
  t = 1.0183 standard error of difference = 0.015
  df = 28

We can conclude from the t-test done here that my GA and PSO algorithm have no statistically significant superiority to each other

- Solution Quality

MAX	7.9658	7.954226199

Looking at the maximum value of the population mean from table before, we can see that GA have better solution quality than PSO, but not really significant since the value only differs about 0.01. And looking at the maximum value an individual has for each algorithm, both have pretty much the same value.

GA:

```
In [13]: 1 fittestnextgen[-1:]
Out[13]: [7.999]
```

PSO:

```
In [28]: 1 max(z['pBest'])
Out[28]: 7.999998192373176
```

- Consistency (variance)

Review your data:

Group	GA	PSO
Mean	7.90462666667	7.88947614833
SD	0.03399445788	0.04652577095
SEM	0.00877733128	0.01201290240
N	15	15

GA with variance about 0.008 have smaller value from PSO with 0.012 which make GA more consistent than PSO.

- Running time

Both are running on my computer with i7-7700HQ processor @2.80GHz with 16GB RAM
With (100 generation * 20 population) objective function calls at the both of the algorithm,

- GA is running at:

```
In [16]: 1 print("--- %s seconds ---" % (time.time() - start_time))
--- 1.5391771793365479 seconds ---
```

- While PSO is running at:

```
In [25]: 1 print("--- %s seconds ---" % (time.time() - start_time))
--- 2.4952707290649414 seconds ---
```

My PSO algorithm code is running slower than GA.

Looking at all the data given, I can probably conclude that my GA algorithm is better than PSO, but that doesn't mean PSO is lesser than GA as an algorithm, this can happen probably because the problem is more suited to GA algorithm, or my code at PSO algorithm are not good enough with only applying few of the so many diversities maintaining, boundary handling, and velocity damping method. But it certainly is easier to code PSO with its internal one-way information sharing rather than using cross-over and mutation.

GA with its cross-over and mutation of the offspring and restricting the reproduction of weak off springs, it eliminates not only that solution but also all of its descendants. This makes the algorithm converge towards high quality solutions within a few generations than PSO.