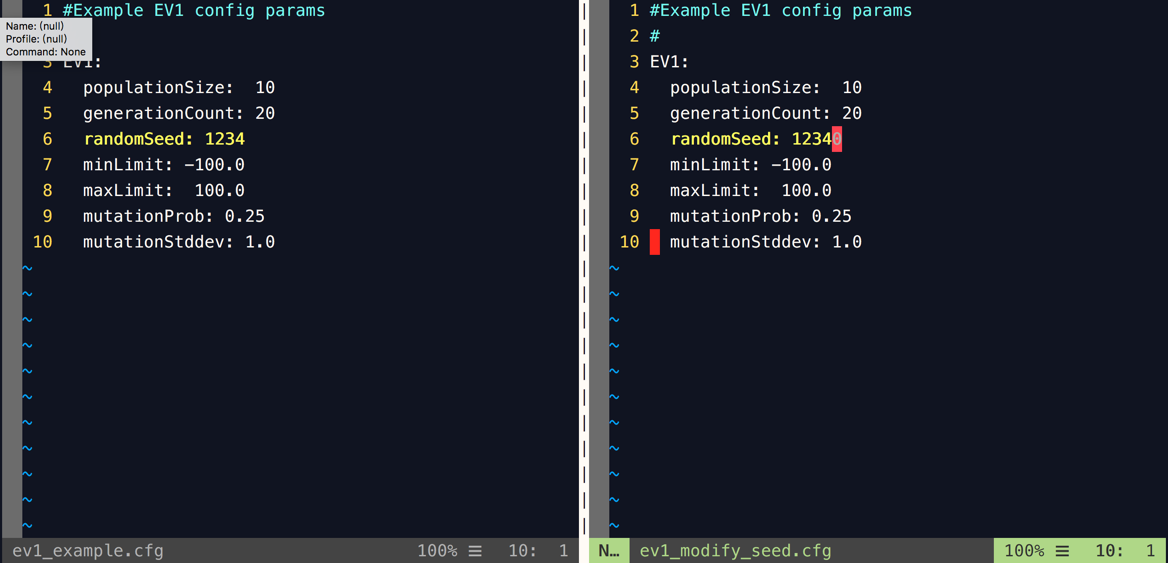
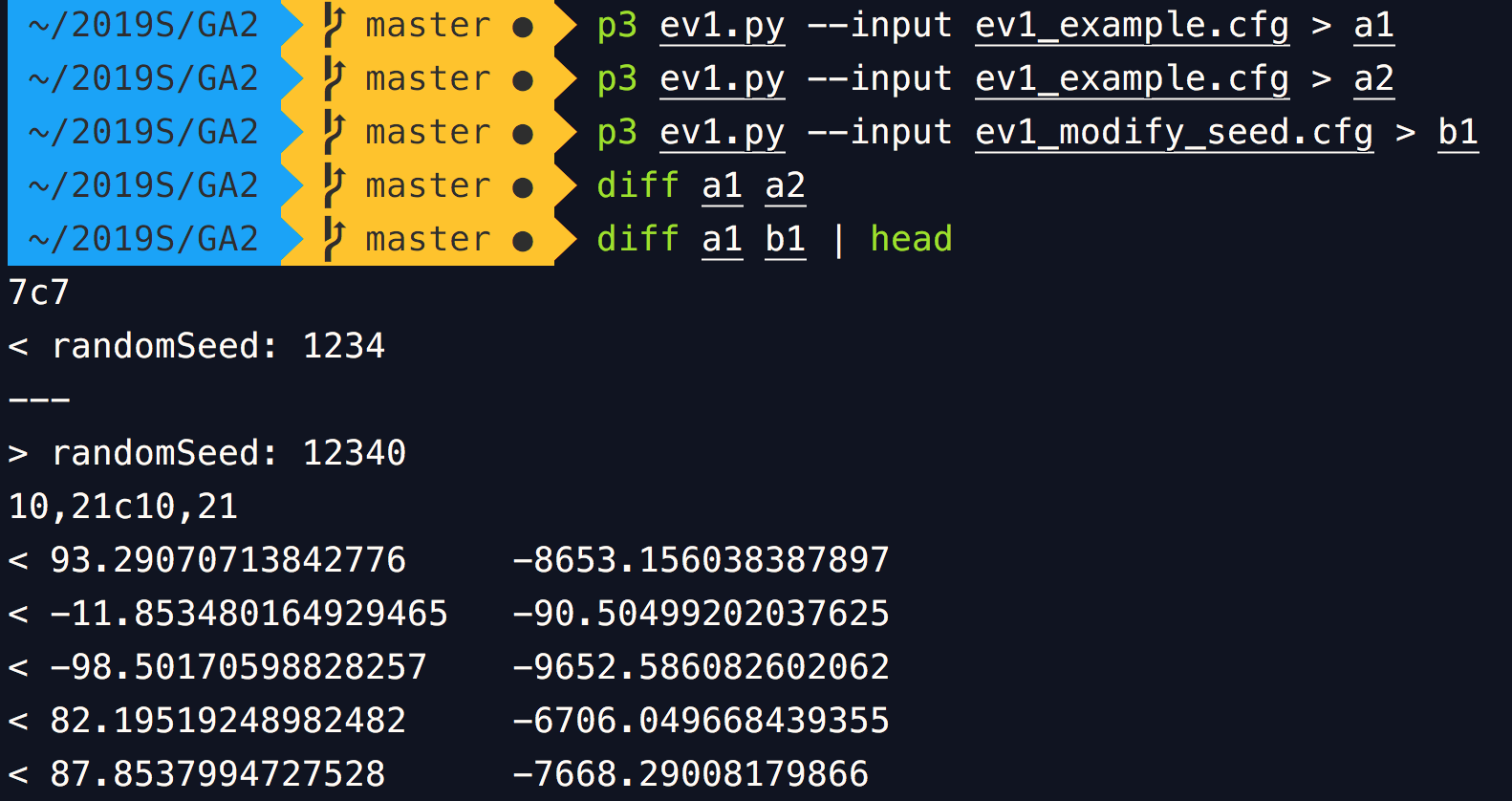
Part1: experiment with parameters

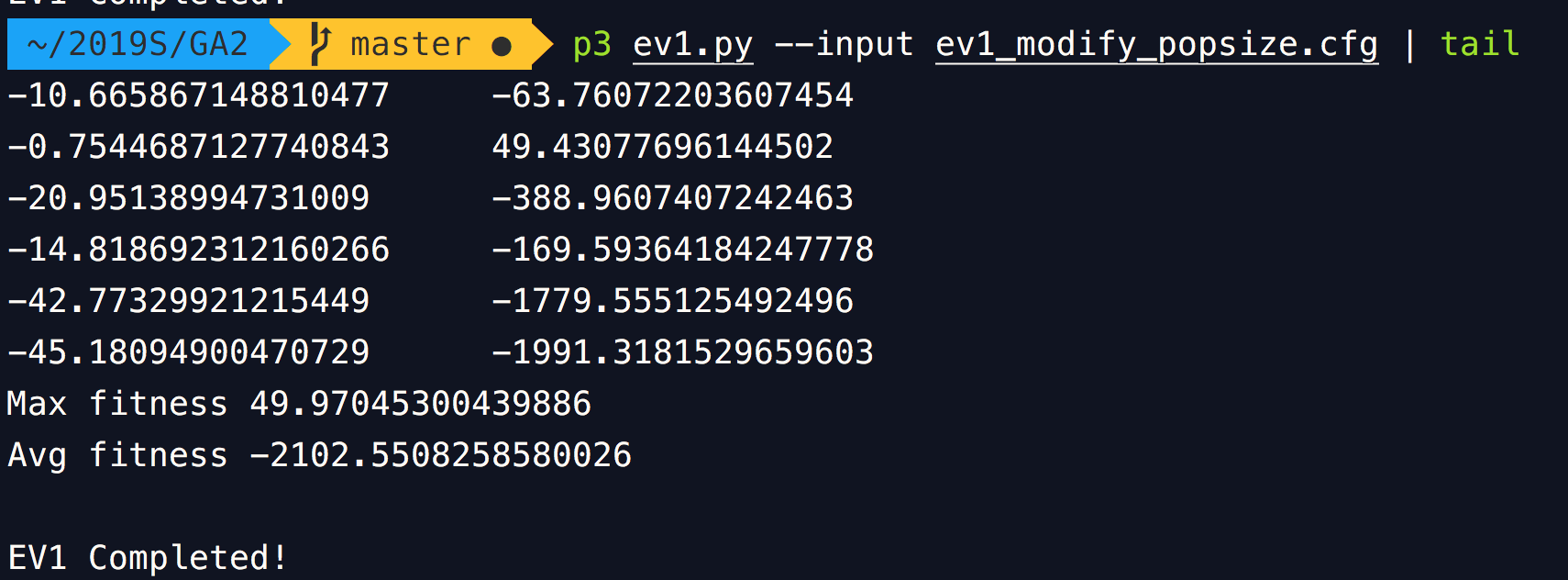
1. Random seed   
   modify seed from 1234 to 12340.
   1. Run ev1.py twice with same random seed (=1234)  
      and output to a1 & a2 respectively.
   2. Run ev1.py with another random seed (=12340)  
      and output to b1
   3. Use **diff** to see results.  
      different random seed may cause different results.

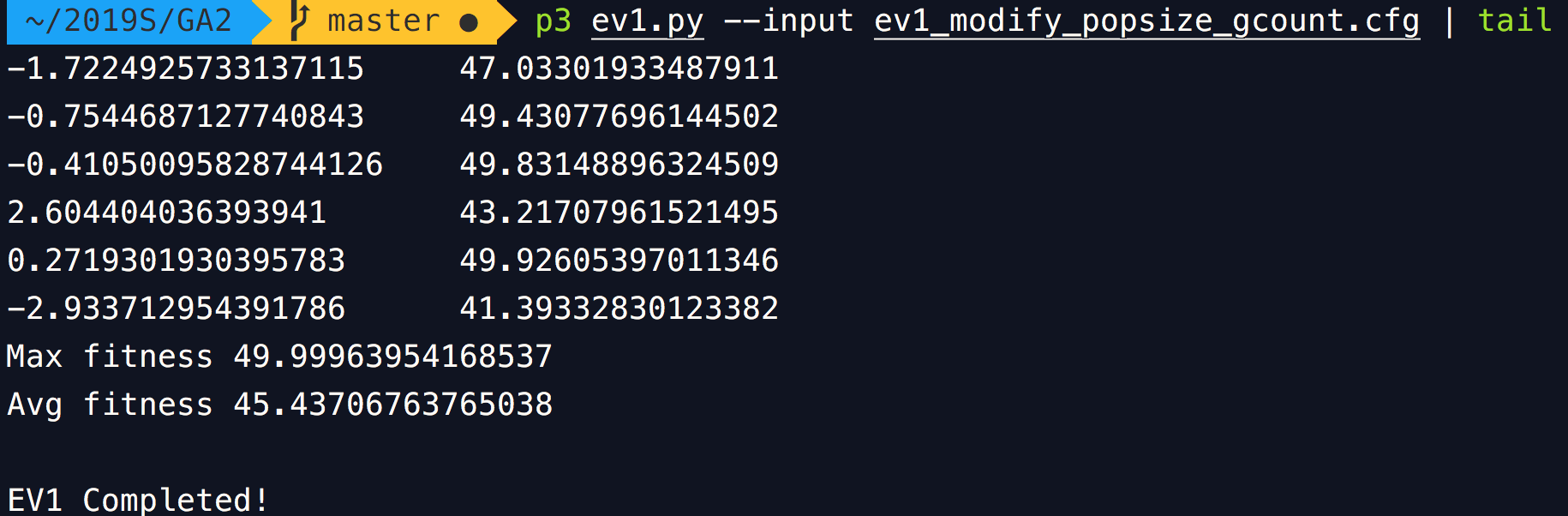
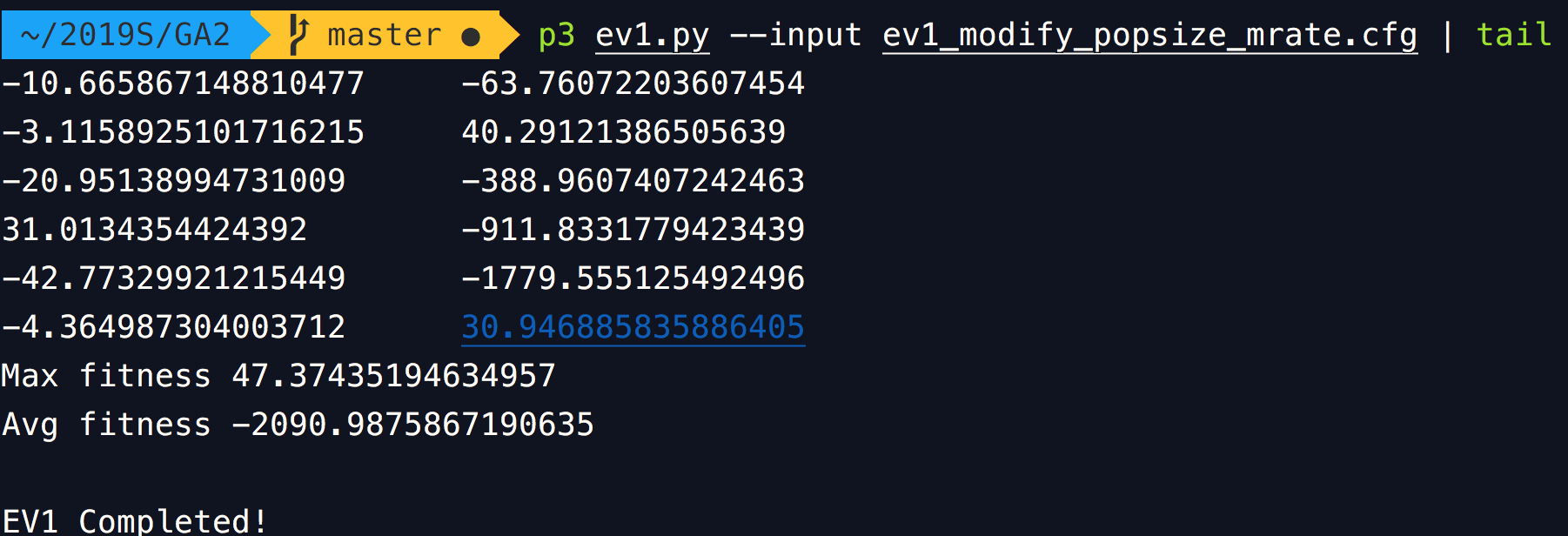


1. population size, generation count, mutation rate
   1. Modify population size to 100

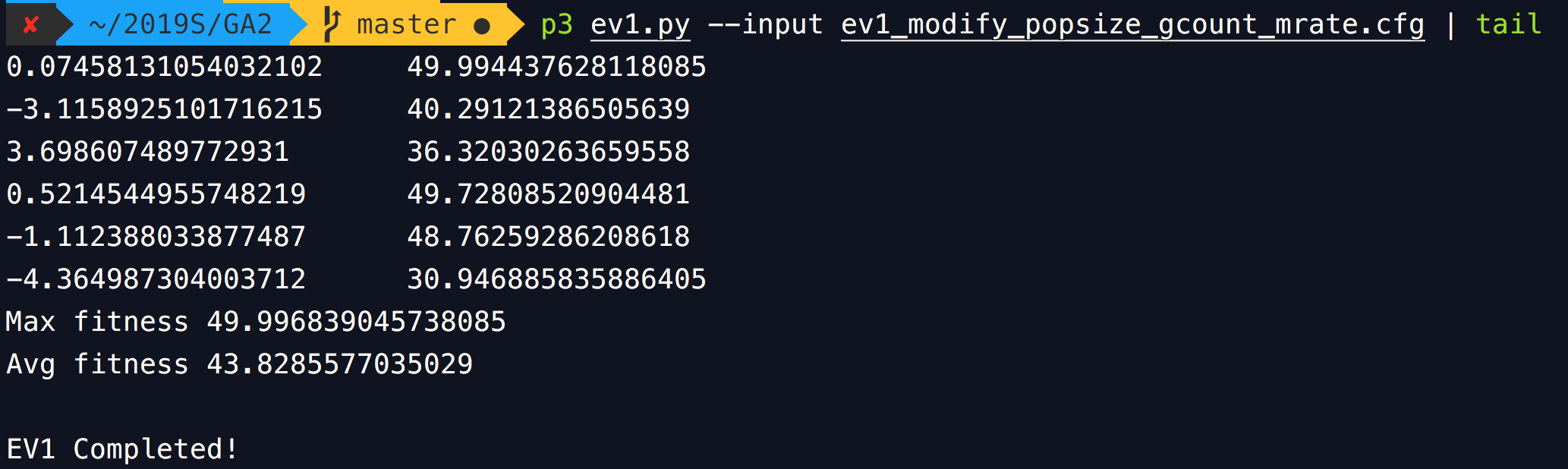
Many individuals still get poor fitness value.

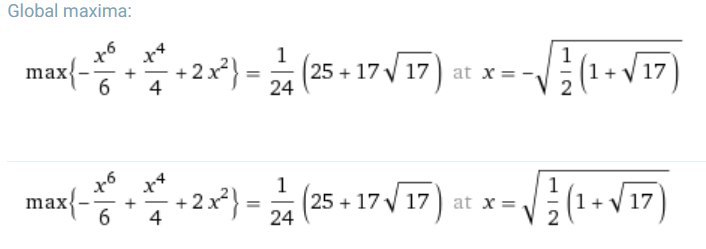
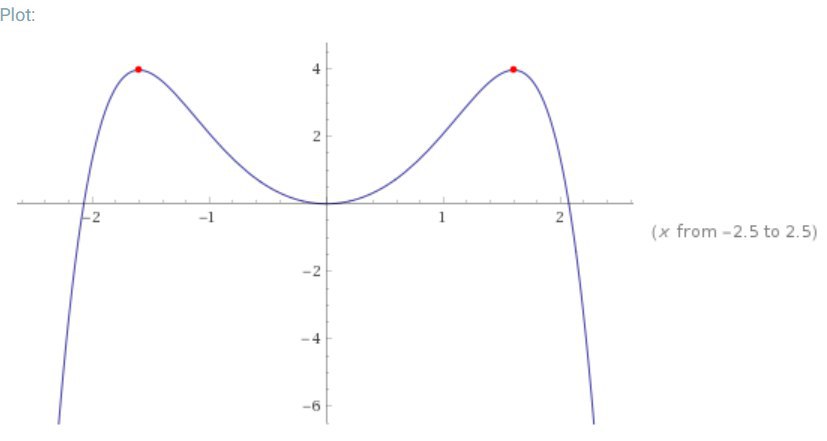
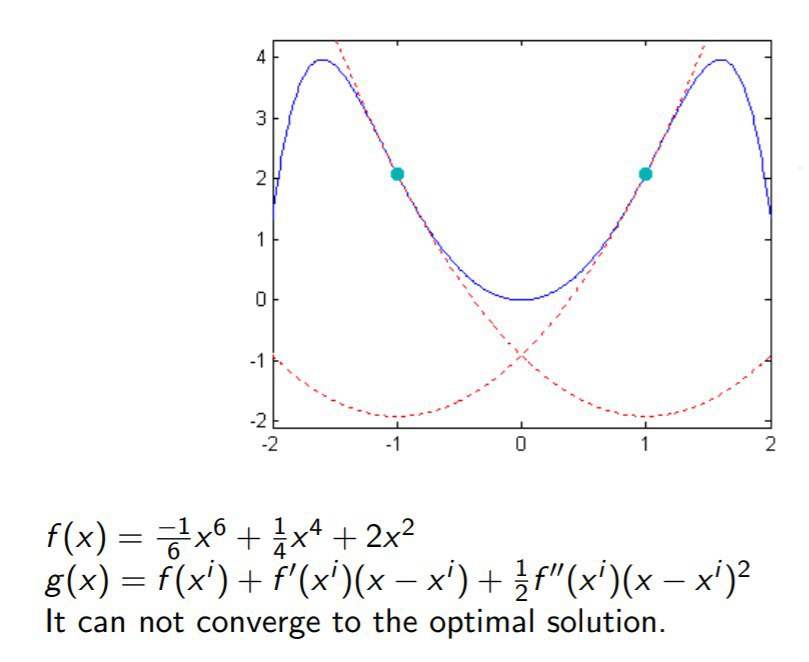
The max fitness is close to optimum because this function is too simple.

From avg fitness, it seems not converge yet, so we need to run more iterations.  


* 1. population size=100, generation count=200  
     With more generations, avg fitness converges more obviously.  
     
  2. population size=100, generation count=20, mutation rate=0.9  
     Increasing mutation rate could increase search space(or diversity), but this result shows we need more generations to constraint the search space. 
  3. population size=100, generation count=200, mutation rate=0.9  
     This max fitness value is higher than result of 2.2.

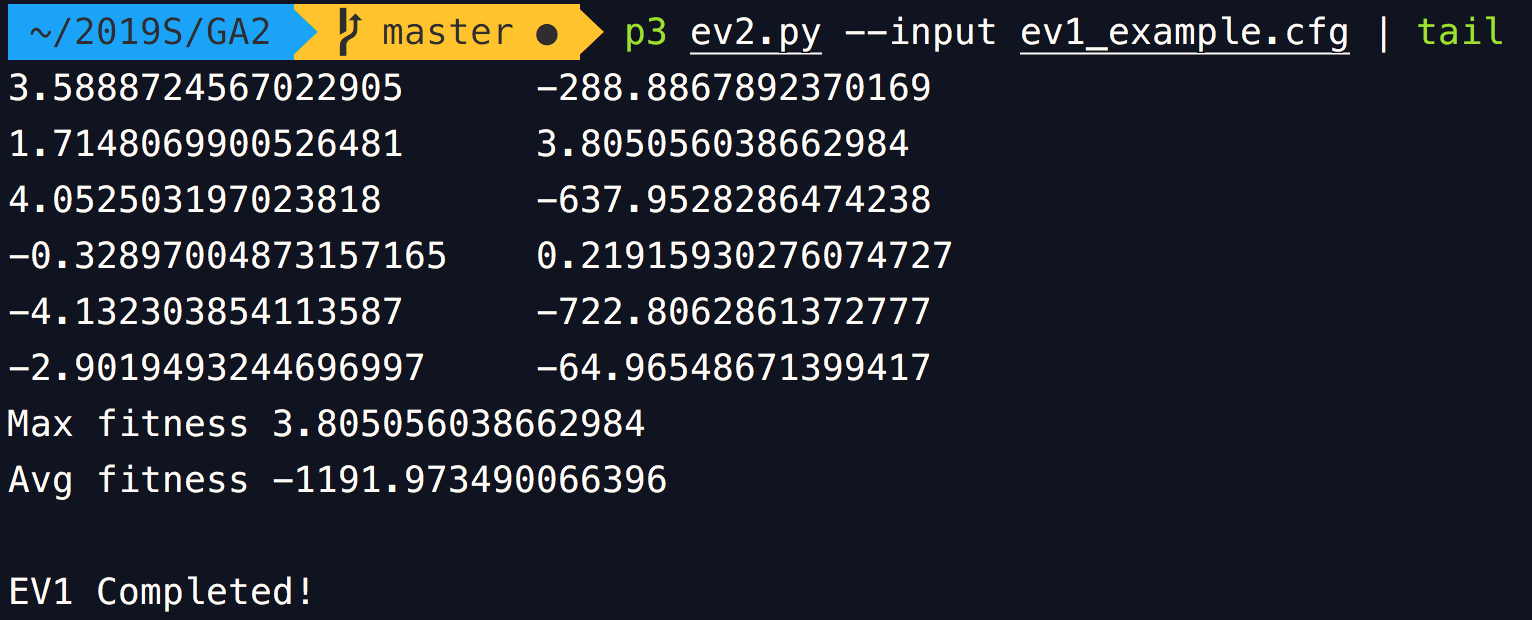
Increase mutation rate gives the best individual more chances to get better (or get worse, like result of 2.3).

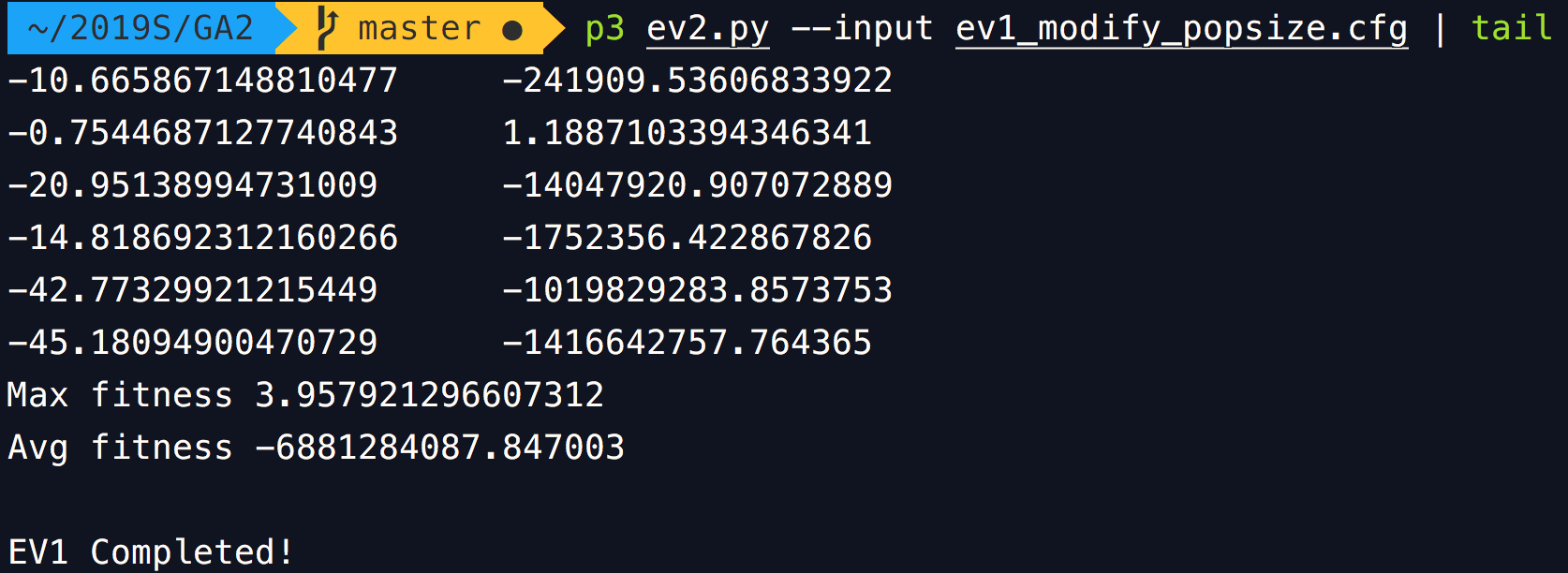
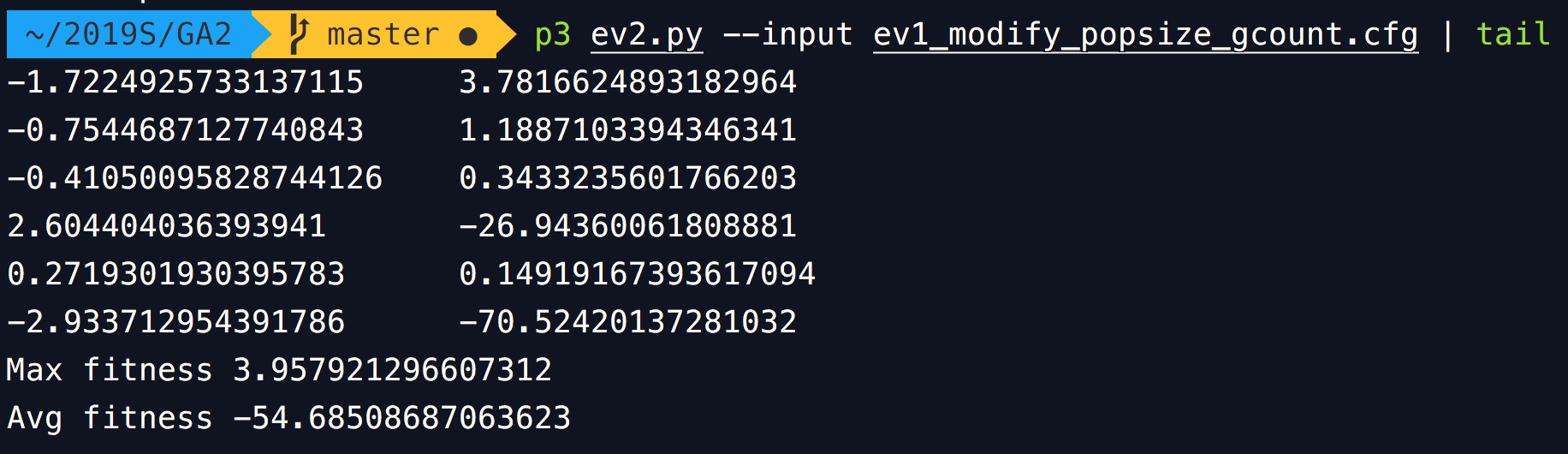
* 1. use more generations than 2.3, so that better individuals could get close to optimum more.  
     

1. Change fitness function to   
   which has global maxima ≈ 3.9622:  
   It looks like   
     
   This function is interesting.   
   When we constraint its domain to this small interval , we can try to find its minimum. (Due to this constraint, we know there is a global minimum f(0)=0 )  
   This image shows the result of using quadratic Newton’s method to approximate the minimum, starting point is x=1 or x=-1.  
   x will get stuck in 1, -1, 1, … , as an alternating sequence, and converge to   
    

The following I directly use this function as fitness function to get maximum without constraints.

* 1. Population size=10, generation count=20



* 1. Population size=100, generation count=20
  2. Population size=100, generation count=200

Part2: Modification

