Evolution Strategies (ES)

Assignment 3 CS472-F16 Due: Fri Nov 4 at 5pm PT

Points: 200

1 The problem

Let's play with solving a real number vector problem. The problem is to place k points in a unit circle so that the the minimum distance between any two points is maximized. This is a relatively hard problem. Let's try to solve it with evolution strategies. Some example solutions are seen in Figure 1.

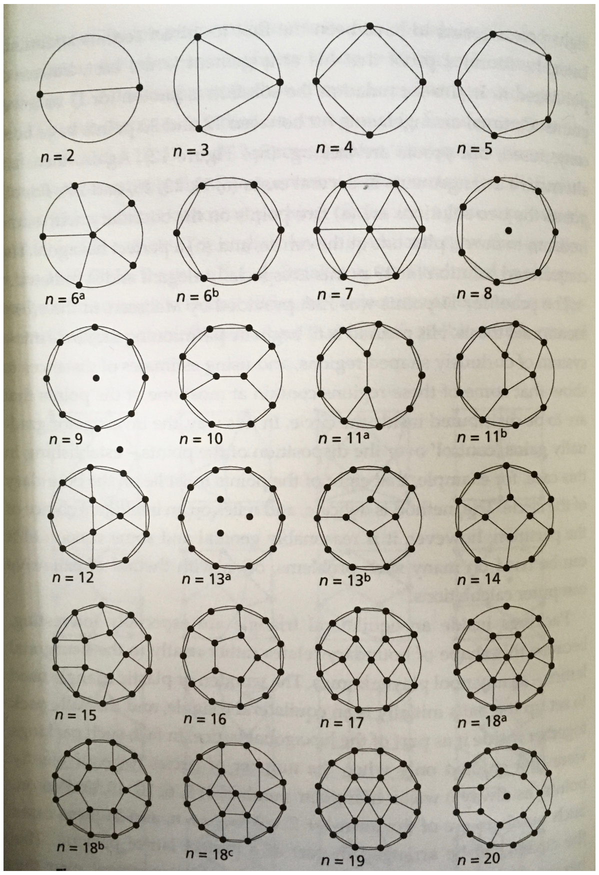

Figure 1: Dots on a circle to maximize minimum distance between points. (From *How to Cut a Cake: And Other Mathematical Conundrums* by Ian Stewart)

2 The Experiments

The output of your program should at least have the number of points followed on separate lines by the pairs of polar coordinates θ with $0 \leq \theta \leq 2\pi$ and r with $0 \leq r \leq 1$. All points are rotated so one of the points has $\theta = 0$. Either rotate the points or have your representation built that way. Include the measure of fitness. Here is the example output you should use.

```
** 15
***   0.000000   0.520035
***   0.386124   0.981772
***   0.926452   0.985252
***   1.116276   0.475057
***   1.467051   0.953166
***   2.015758   0.975875
***   2.342130   0.491621
***   2.562461   0.996566
***   3.105154   0.917787
***   3.668288   0.992983
***   4.109920   0.568308
***   4.469185   0.050688
***   4.612219   0.979806
***   5.092867   0.563311
***   5.728703   0.966497
***   Fit:   0.518519
```

Write a simple half page report named "yourSubmitName.pdf" describing what representation you used e.g. cartesian, polar, any special other encoding features. Let me know how you coded the ES (e.g id you evolve sigma?) and what parameters for your evolution worked you observe. Turn in your code and the report in a tar. The tar should include a makefile that will make the program called `points`.

3 Some Possible Results

I have a script that took the numbers in the above format and computed all the distances between the the points. Here are statistics over all the distances in configurations I found for various numbers of points. The minium distance (column 4) in the table is what is being maximized. The answers that generated these stats might not be right but they are what my algorithm got. It used a simple ES.

NumPts	Mean	Stddev	Min	Max
3	1.732	0.0	1.732	1.732
4	1.609	0.303	1.414	2.0
5	1.538	0.383	1.175	1.902
6	1.358	0.405	1.0	1.949
7	1.352	0.425	0.999	2.0
8	1.346	0.445	0.866	1.95
9	1.34	0.464	0.763	1.999
10	1.262	0.454	0.703	1.992
11	1.253	0.46	0.66	1.972
12	1.212	0.455	0.651	1.975
13	1.213	0.463	0.599	1.987
14	1.187	0.46	0.588	1.993
15	1.182	0.463	0.549	1.979
16	1.171	0.465	0.535	1.991

4 Submission

Homework will be submitted as an uncompressed tar file to the homework submission page linked from the main class page. You can submit as many times as you like. The LAST file you submit BEFORE the deadline will be the one graded. For all submissions you will receive email giving you some automated feedback on the unpacking and compiling and running of code and possibly some other things that can be autotested. I will read the results of the runs and the reports you submit.

Have fun.