HPC Questions

Rachel Bates

December 2018

Contents

1 High Performance Computing

1.1 Question 8: Neutral Model Simulation

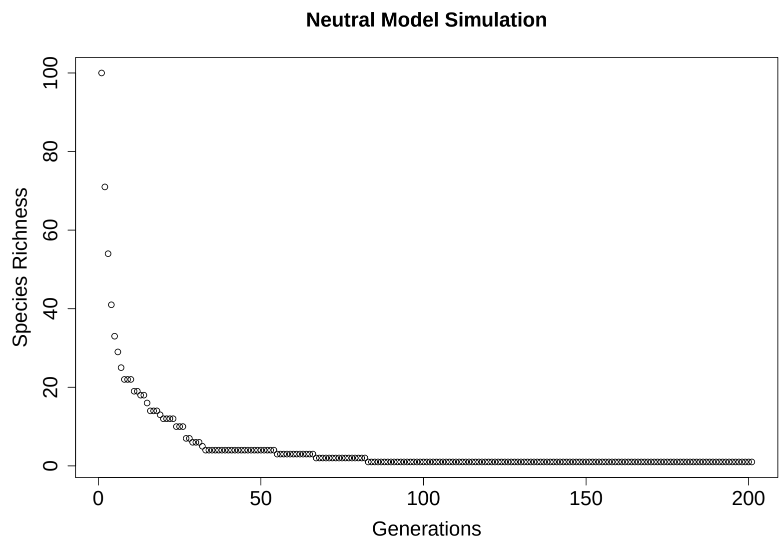


Figure 1: Neutral Model simulation of an population of 100 unique individuals (species richness = 100) over 200 generations.

The population will always converge to 1 species in this scenario given enough time. This is because without any way for new species to enter the population, when one is lost it is lost permanently. Over time, simply due to random loss of species, only one will remain.

1.2 Question 12: Min and Max Neutral Simulation

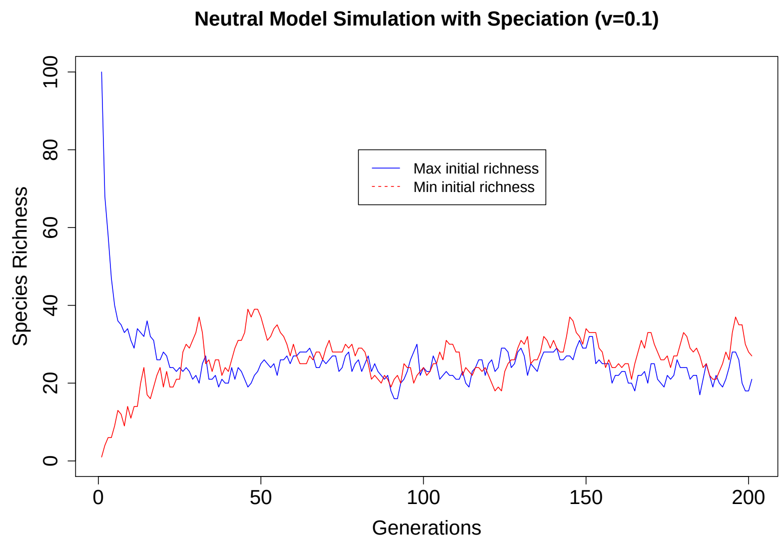


Figure 2: Neutral Model simulation of an population of 100 unique individuals (species richness = 100) over 200 generations. Individuals are replaced by a new species with a probability of v=0.1. The Min model started with 100 individuals of the same species, the Max model started with 100 individuals each from different species. The species richness converges at ∼25 for both simulations.

The neutral model equilibrates to the same value of species richness regardless of initial conditions (in this case, ∼25). This is because the species richness at equilibrium is dictated by the point at which speciation rate matches the rate of species randomly being lost, at a given population size. As such, initial species richness only affects the behaviour of the burn-in period of the model.

1.3 Question 16: Average Octaves

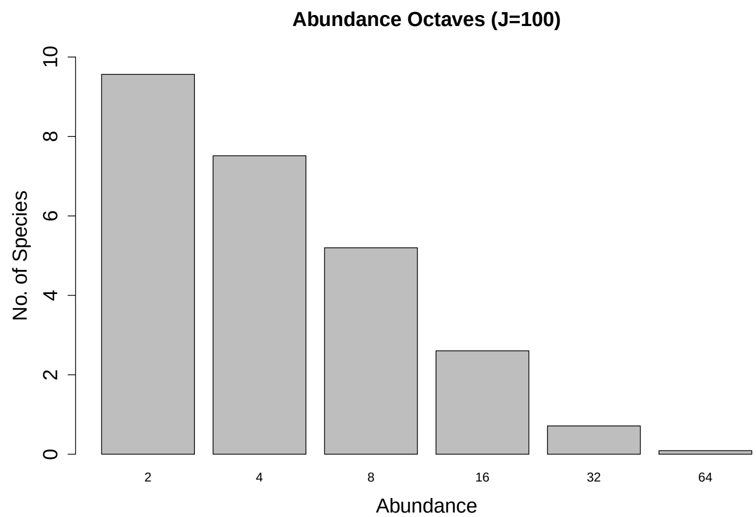


Figure 3: Number of species that have populations of a given size. For example, the first bar shows the number of species who have fewer than 2 individuals. This shows the mean distribution over 2000 generations, measured every 20 (100 measurements). Speciation rate = 0.1, burn in period of 200 generations used.

Initial condition does not matter for this, as values are only measured after the simulation reaches a dynamic equilibrium. Question 12 demonstrates why the dynamic equilibrium is not affected by initial species richness.

1.4 Challenge A

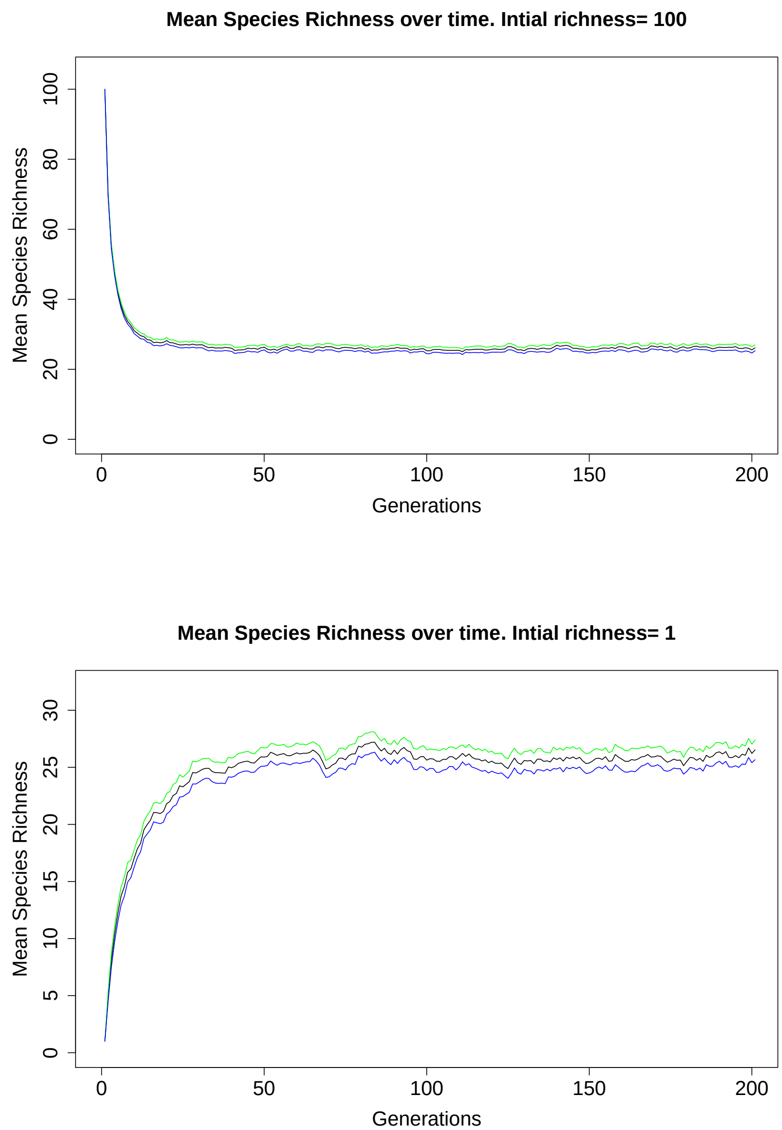
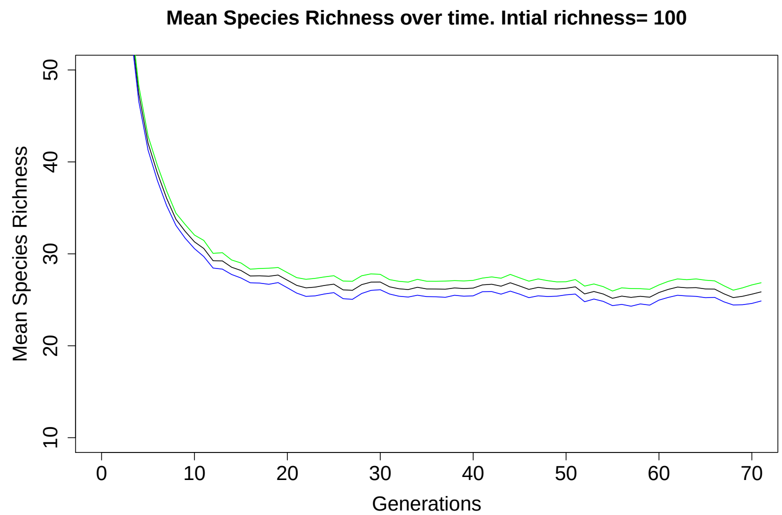


Figure 4: Mean species richness over time across 100 repeated runs. Population size = 100, speciation rate = 0.1. Coloured lines show 97.5% confidence intervals. Dynamic equilibrium is at ∼25 species, reached after ∼40 generations.



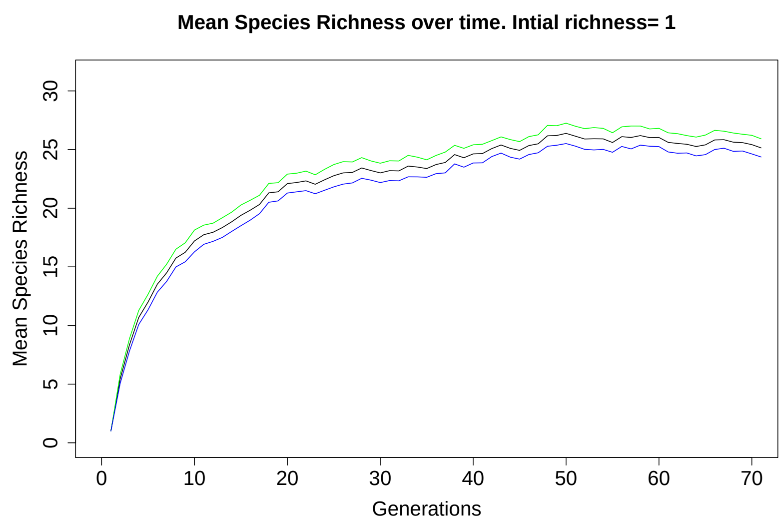


Figure 5: Mean species richness over time across 100 repeated runs. These show only the first 70 generations to better show where dynamic equilibrium is reached. Population size = 100, speciation rate = 0.1. Coloured lines show 97.5% confidence intervals. Dynamic equilibrium is at ∼25 species, reached after ∼40 generations.

1.5 Challenge B

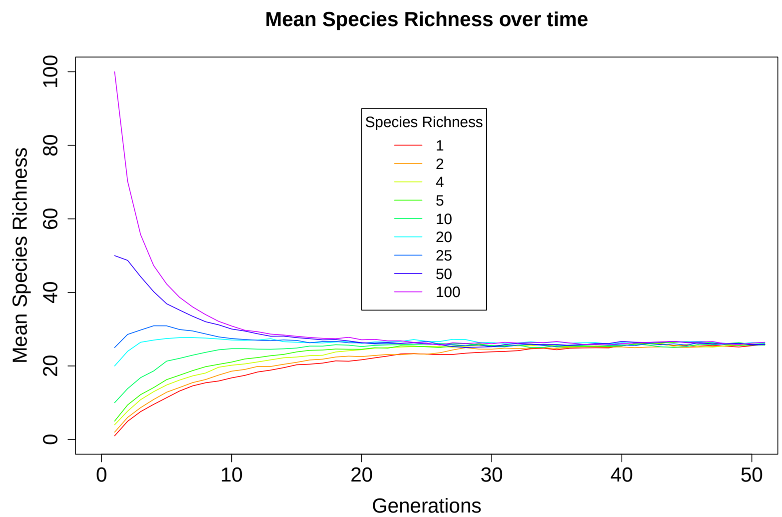


Figure 6: Mean species richness over time across 100 repeated runs, for multiple values of species richness where species evenness is the same. These show only the first 30 generations to better show where dynamic equilibrium is reached. Population size = 100, speciation rate = 0.1. Dynamic equilibrium is at ∼25 species, reached after ∼40 generations for all initial species richness conditions.

1.6 Question 20: HPC Octaves

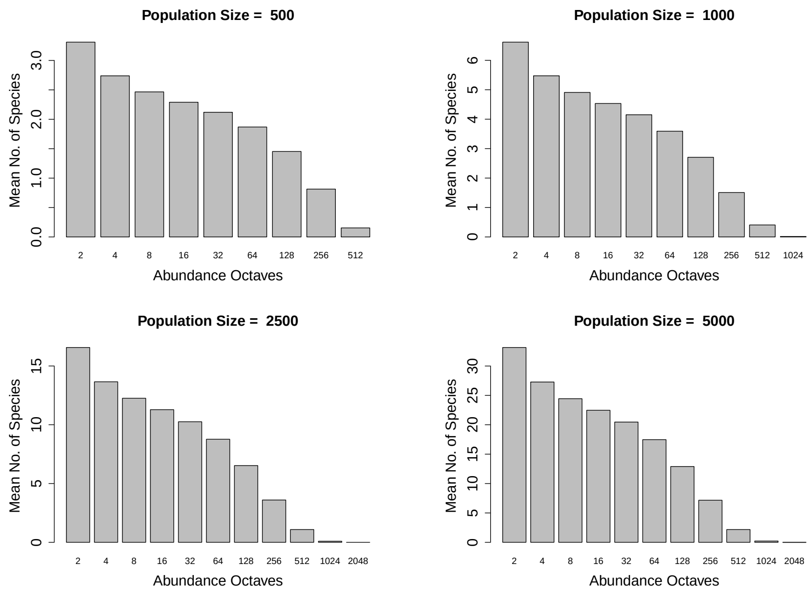


Figure 7: Average abundance octaves for each population size(J). Each population was run 25 times for 11.5 hours on the HPC, with abundance octaves collected every J/10 generations after an initial burn in period (8*J generations).

Speciation rate = 0.006621

Table 1: Final average vectors of the abundance octaves for each population size.

Bins	500	1000	2500	5000
2	3.3111176,	6.61649404	16.5712554631	33.14846824
4	2.7387795	5.47421687	13.6532874045	27.28063519
8	2.4658150	4.90836180	12.2533527578	24.43879989
16	2.2900418	4.53306943	11.2839870927	22.46690556
32	2.1196427	4.15085046	10.2549321279	20.46479764
64	1.8686462	3.59256548	8.7671382085	17.47161327
128	1.4529534	2.70596135	6.5280406279	12.89952220
256	0.8135476	1.50845399	3.5999428159	7.16006183
512	0.1536192	0.40716339	1.0875597370	2.17509837
1024		0.01547819	0.0923659237	0.21824058
2048			0.0002859205	0.00217819

1.7 Challenge D

Cluster run: 11.5 hours per simulation, for 100 simulations. Therefore 1150 total computing hours to obtain average species octaves.
Total octaves calculated: J=500 - 2252055, J=1000 - 553230, J=2500 - 73447, J=5000 - 14232.
Total = 2892964. Mean = 723241.

Coalescence: Used to obtain the same number of octaves as the cluster run.
Time spent: J=500 - 258.52 (4h18.5mins), J=1000 - 141.7 mins (2h21.7 mins), J=2500 - 62.5 mins (1h2.7 mins), J=5000 - 32.4 mins.
Total time = 7h15mins.

Therefore the coalescence simulations took 6.5% of the time that the cluster simulations did to run.

The coalescence simulation ran faster than the cluster due to always being at equilibrium, and taking a sampling based approach. This means there is no burn-in period to calculate, and not every individual has to be counted (i.e. J grows smaller as the calculation progresses).

2 Fractals

2.1 Question 21: Dimensionality of fractals

a = 1 fractal unit
d = dimension of fractal

Flat Fractal:

$$a \xrightarrow{\text{next fractal layer, 3x as big}} \text{8a}$$
$$3^1 = 3 \qquad 3^2 = 9 \qquad 3^d = 8$$

$$\begin{aligned} d.log_3 &= log_8 \\ d &= (log_8/log_3) \\ d &= 1.893 \end{aligned}$$

"3D" Fractal:

$$a \xrightarrow{\text{next fractal layer, 3x as big}} \text{20a}$$
$$3^2 = 9 \qquad 3^3 = 27 \qquad 3^d = 20$$

$$\begin{aligned} d.log_3 &= log_{20} \\ d &= (log_{20}/log_3) \\ d &= 2.729 \end{aligned}$$

2.2 Question 22: Chaos Game

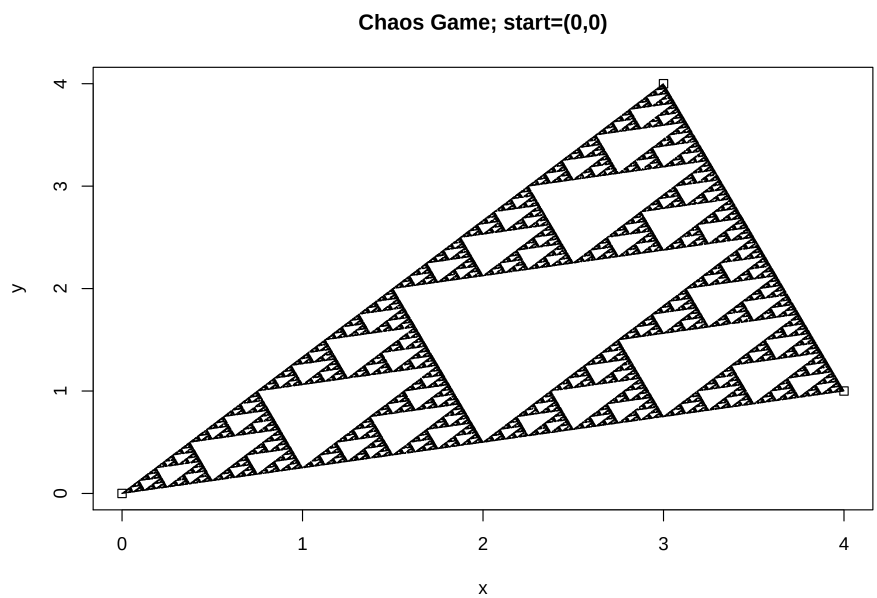


Figure 8: Shape created by moving halfway to either of three coordinates - (0,0) (3,4) (4,1) - 100000 times. The starting position is (0,0).

2.3 Challenge E

Starting from a different initial position in the chaos game still produces the same end shape, as the pattern of the stochastic movement is the same and so it will tend towards the same shape. Following are several shapes produced by changing the parameters of the chaos game.

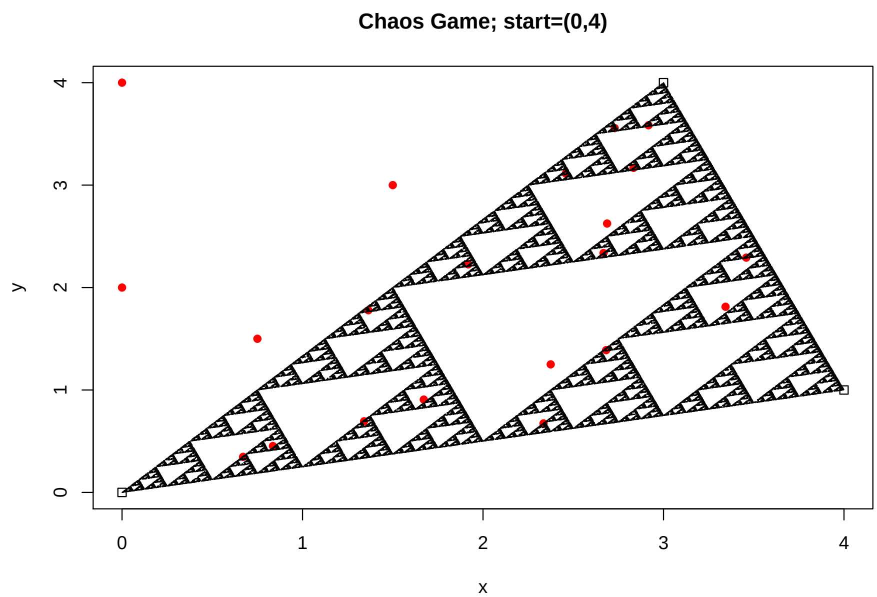


Figure 9: Shape created by moving halfway to either of three coordinates - (0,0) (3,4) (4,1) - 100000 times. The starting position is (0,4). The red dots show the first 20 plotted points.

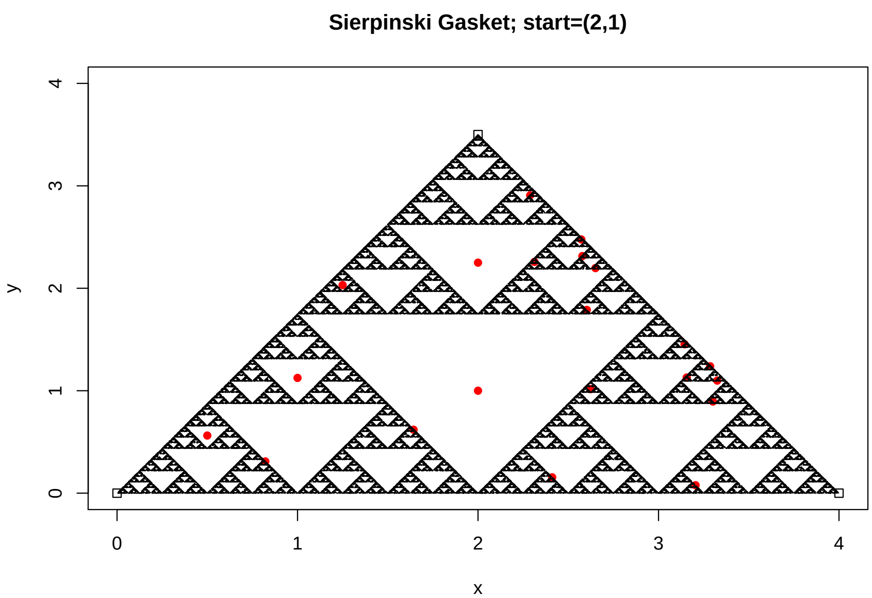


Figure 10: Shape created by moving halfway to either of three coordinates - (0,0) (2,3.5) (4,0) - 100000 times. The starting position is (2,1). The red dots show the first 20 plotted points.

The pattern is the same with an equilateral triangle (known as a Sierpinski Gasket).

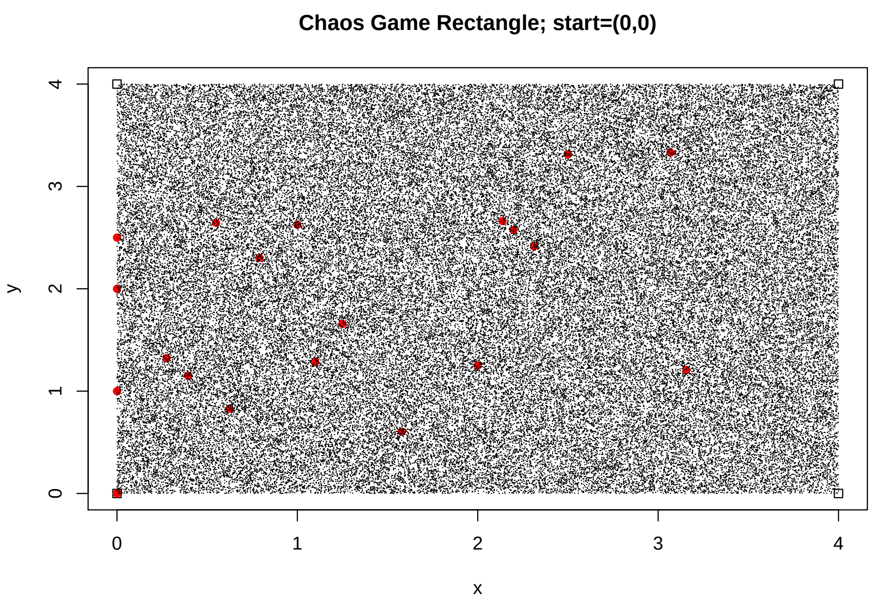


Figure 11: Shape created by moving halfway to either of four coordinates - (0,0) (0,4) (4,0), (4,4) - 100000 times. The starting position is (0,0). The red dots show the first 20 plotted points.

As you can see, the chaos game pattern no longer exists when moving between 4 points.

2.4 Question 25: Spiral

Taking this approach with spiral forces the computer to keep drawing lines to infinitely smaller lengths, with no end point. The code therefore throws the error "Error: evaluation nested too deeply: infinite recursion / options(expressions=)?" as it is essentially stuck in an infinite loop.

2.5 Question 26: Spiral 2

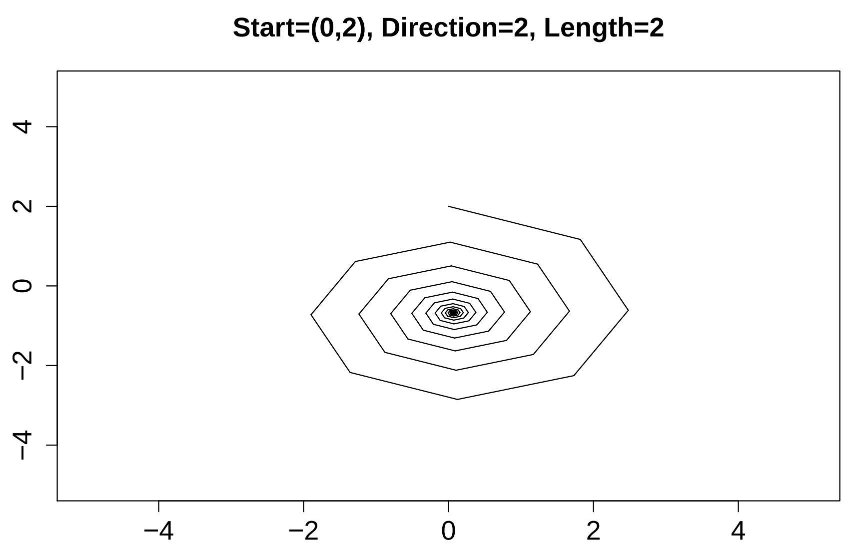


Figure 12: Spiral created by drawing lines at $\Theta + (\pi/4)$ radians to the previous line. The subsequent lines are 95% as long as the previous line.

2.6 Question 27: Tree

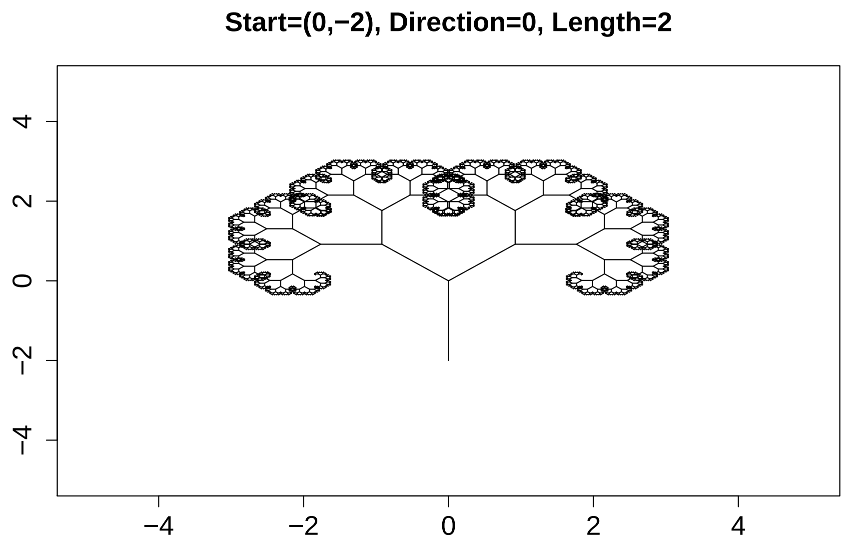


Figure 13: Tree created by drawing lines at $\Theta \pm (\pi/4)$ to the previous line. Subsequent lines were 65% the length of the previous line.

2.7 Question 28: Fern

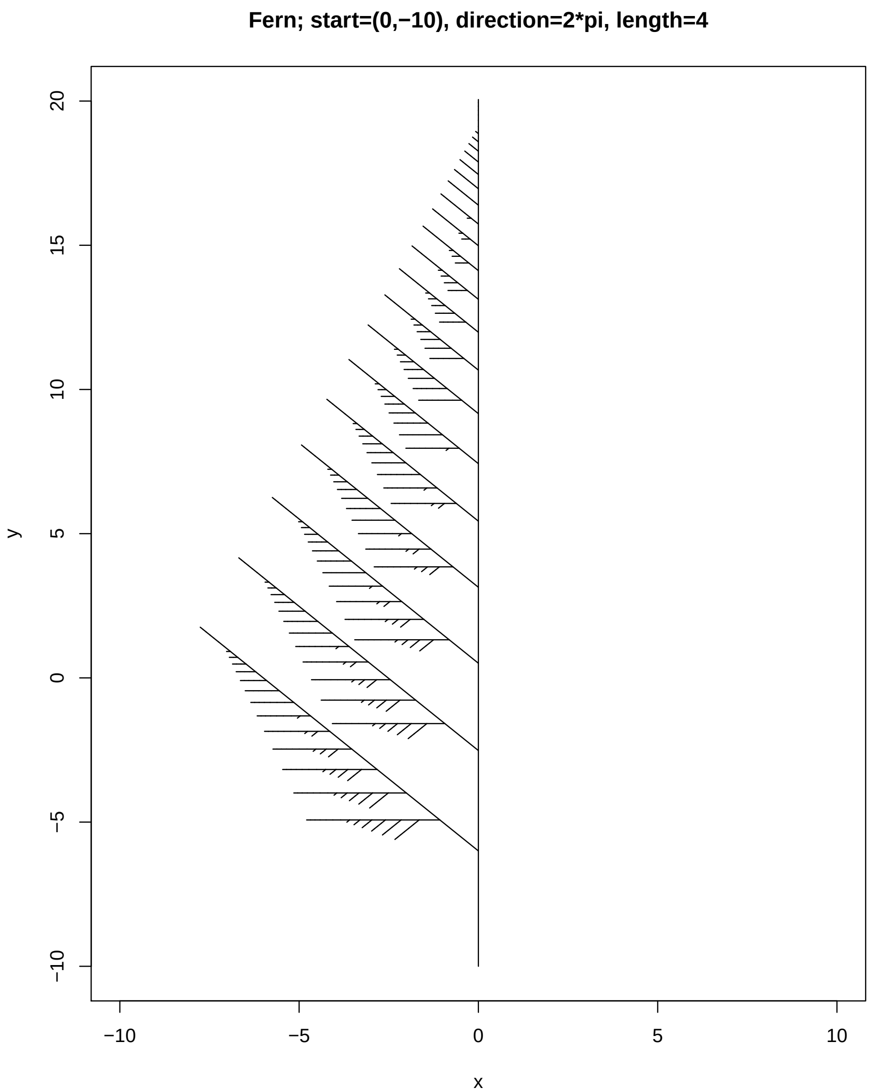


Figure 14: Fern created by drawing lines at either $\Theta - (\pi/4)$ to the previous line or carrying on in the same direction. Subsequent branching lines were 38% the length of the previous line, while lines continuing in direction were 87% of the length.

2.8 Question 29: Fern 2

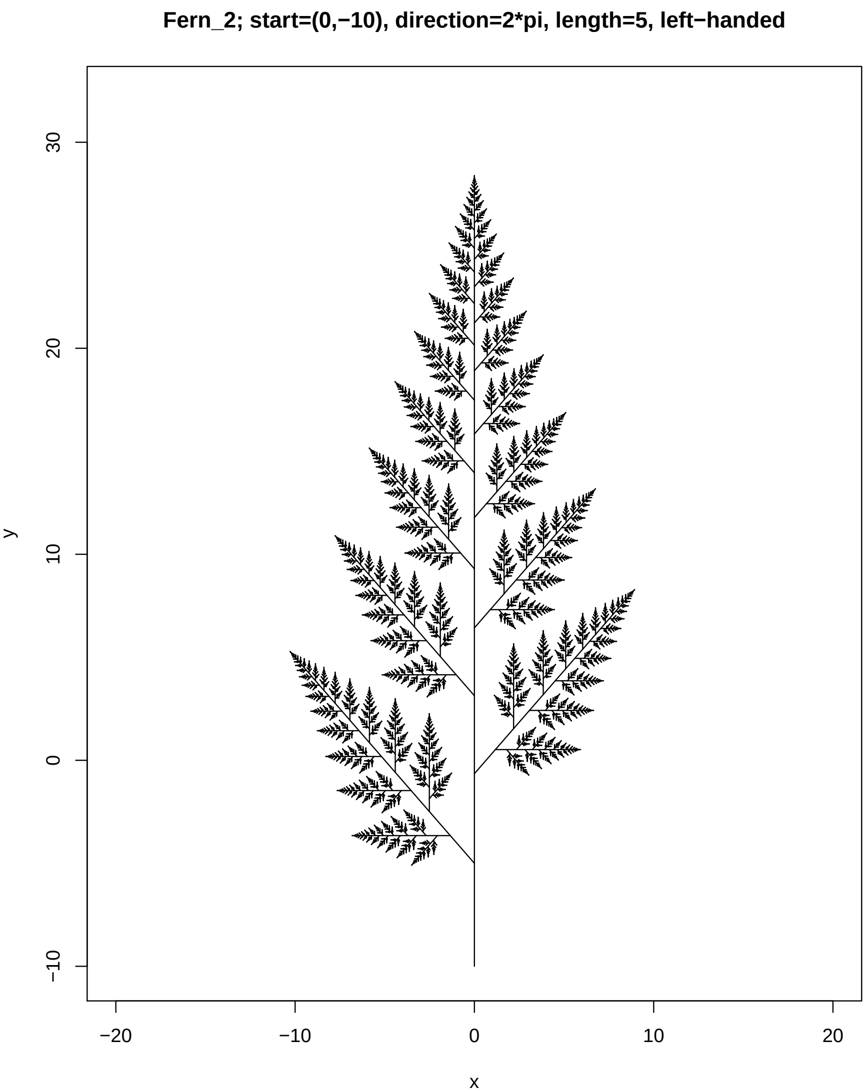


Figure 15: Fern created by branching either left, then right. Left branches draw lines at either $\Theta - (\pi/4)$ to the previous line or carrying on in the same direction. Subsequent branching lines were 38% the length of the previous line, while lines continuing in direction were 87% of the length. Right branches are the same, except the angle is $\Theta + (\pi/4)$ to the previous line.