

Our model has three classes and Setup.java is the main class.

1. If you want to run the model under the default values of Netlogo model (Figure 1), you can just run Setup.java.



Figure 1. Default values of Netlogo model

2. If you want to change the parameters before running the model, you can make use of command line to input the corresponding values. The sequence of the arguments is the same as Netlogo model. For example, if you want to set the number of people as 1000, the max-vision of people as 10 and keep other default values, you can execute the program as figure 2 shows. In addition, our model will output four kinds of data. They are the number of poor, middle, rich and the Gini Index in each time tick. They are separated by a space.

```
[ouotsukous-MacBook-Pro:modelling-assignment Venn$ javac Setup.java People.java Land.java
[ouotsukous-MacBook-Pro:modelling-assignment Venn$ java Setup 1000 10 15 1 83 10 1 4
279 415 306 0.26349989311998023
319 507 174 0.2640020351969352
502 471 27 0.2834767025089606
556 416 28 0.2985430027233598
578 379 43 0.3254229266692511
576 379 45 0.333591675774673
596 350 54 0.3462061855670111
610 331 59 0.35581738779510524
646 302 52 0.37121312872975254
673 280 47 0.37343477533260744
716 241 43 0.38005920167028395
738 221 41 0.3866694066412632
778 191 31 0.39558156216790613
801 177 22 0.40407908739325704
```

Figure 2. Parameter setting and the results

As we can see from Figure 2, the first output line is “279 415 306 0.26...”, it means that after setup the environment (people and land), there are 279, 415 and 306 people belong to poor, middle and rich respectively. At the same time, the Gini Index is about 0.26.

3. Because of huge data size, our program can output the results into a txt file. If you want to do it, you just need to cancel the comments from the 70th line to 83th line of Setup.java (Figure 3).

As we can see from Figure 3, the 79th line determines the specific output path. You may need to change it in accordance with your own willing.

```

70 //      try {
71 //          /*output the data which we need to a txt file.
72 //             * By inputing the txt file into the Matlab, we can analyze
73 //             * the data conveniently.
74 //             */
75 //          PrintStream data=new PrintStream
76 //              /*the following address should be changed for different
77 //                 * computers.
78 //                 */
79 //              ("/Users/Venn/Desktop/modelling-experiments/data.txt");
80 //          System.setOut(data);
81 //      } catch (FileNotFoundException e) {
82 //          e.printStackTrace();
83 //      }
84

```

Figure 3. Function of output the data