SIMULATION OF THE INEQUALITY PROCESS

MODELING AND SIMULATION IN SCIENCE, ENGINEERING, AND ECONOMICS

Yao Xiao *
School of Art and Science
New York University Shanghai
Shanghai, China
yx2436@nyu.edu

Ziyue Wang *
School of Art and Science
New York University Shanghai
Shanghai, China
zw2818@nyu.edu

ABSTRACT

We are all aware of the unequal distribution of wealth across the society, and the inequality keeps rising in recent years. An interesting experiment of randomly exchanging dollars within a certain population yields an exponential-like distribution of wealth rather than a uniform distribution, which sheds light on one possible way of modeling this inequality process. In this paper, we model the inequality process via simulating random exchanges of surplus within a certain population (with certain constraints). Different sets of parameters are applied to the model to simulate different types of societies, and we observe the pattern of the wealth distribution. These are fully implemented with Python, and can be found at Yao Xiao's Github repository[2].

Keywords Economics System · Inequality Process · Surplus Theory · Wealth Distribution

1 Introduction

Consider a scenario where n people sit in a room and each person possesses m dollars initially. At each clock tick, a randomly-chosen person gives one dollar to another randomly-chosen person. After a certain amount of time, how will the wealth be distributed among the n people? Intuitively, this can be considered as a random walk on an undirected graph $G = (V, E)$, where each vertex $v \in V$ is an n-tuple summing up to mn, representing some state of the wealth distribution in the room, and two vertices are connected if and only if one is reachable from the other with a single transaction. We know that the stationary probability measure for any vertex is given by

$$\mathbb{P}(v) = \frac{\deg(v)}{2|E|}, \tag{1}$$

according to László Lovász [9]. Therefore, since G is almost regular, intuitively we would get a uniform distribution. However, in the post [1], a simple program implied that the distribution will become exponential-like, which indicates extreme inequality, which is similar to an inequality process in the economic system. This sheds light on the feasibility of simulating the inequality process via (constrained) random exchange.

In this paper, we will model the economic system in the same way as described above, creating an initial population with an initial distribution of wealth. By saying *wealth*, we actually mean the only the *surplus*, and the rest of an individual's wealth will not be modeled and assumed constant. In particular, we will use equally distributed, uniformly distributed, and normally distributed initial wealth in our simulation. Then, two people are selected at random at each clock tick, exchanging their wealth based on various different types of transaction functions, which we will discuss later. We will simulate for a certain number of clock ticks (which we denote by *step* in the rest of this paper), sufficiently large so as to observe a certain better of the distribution of wealth of the population. We will then analyze the pattern to observe the inequality of the population, and perform distribution fitting to figure out the specific shape of the distribution of wealth.

*The authors would like to thank Professor Charles S. Peskin (email: peskin@cims.nyu.edu) and Assistant Mengjian Hua (email: mh5113@nyu.edu) for their helpful remarks, and are indebted to them for meticulous reading and constructive suggestion.

[2] https://github.com/Charlie-XIAO/Econ-simulation

Organization. In the rest of this section, we provide an outline of this paper. In Section 2, we will describe the equations used for our simulation, in particular the transaction functions we use and the motivations behind them (Section 2.1). In addition, we describe our method of distribution fitting on the discrete distribution of wealth across the population (Section 2.2). In Section 3, we will validate our simulation results by comparing with the existing research results on real economic systems. In Section 4, we will apply different types of transaction functions with different sets of parameters on various distributions of initial wealth, and elaborate on the results of our simulations and their further implications. We will also select a reasonable set of parameters for the current society discuss our simulation of the real-world economy. Finally, in Section 5, we will wrap up previous discussions and draw some conclusions on the inequality process and the distribution of wealth across the society.

2 Equations

2.1 Transaction Functions

In this section, we will propose four different transaction functions, each of which simulating a different type of society. The first three of them are based on the propositions in the surplus theory of social stratification [5], as we will discuss as follows.

Proposition 1 (Fugitivity of Surplus Wealth Principle [5]). Surplus is the difference between subsistence and the total production of wealth; societal net product. At the level of the individual person, where people are able to produce a surplus, some of the surplus will be fugitive and leave the possession of people who produce it. Moreover, this implies encounters in which surplus wealth changes hands fairly readily.

With Proposition 1, we are able to simulate a hunter-gatherer society, in which different people are in charge of different necessities that are exchanged later within small groups. We propose the corresponding transaction function, `win_take_partial`, which acts on the two randomly picked individuals as

$$X'_A = X_A + dU \cdot X_B - (1-d)U \cdot X_A, \tag{2}$$
$$X'_B = X_B + (1-d)U \cdot X_A - dU \cdot X_B, \tag{3}$$

where

$X_A, X'_A =$ the surplus wealth of A before (respectively, after) an encounter with B,
$X_B, X'_B =$ the surplus wealth of B before (respectively, after) an encounter with A,
$$d = \begin{cases} 1, & \text{with probability } 0.5, \\ 0, & \text{otherwise,} \end{cases} \quad \text{and } \ U \sim \text{Uniform}(0,1).$$

Proposition 2 (The Snowball [5]). Wealth confers on those who possess it the ability to extract wealth from others. So netting out each person's ability to do this in a general competition for surplus wealth, the rich tend to take surplus away from the poor.

With Proposition 2, we are able to simulate a simple ranked society, in which there is a bias towards the richer party. The bias can be simply modeled by a assigning the richer party a large probability of taking the wealth rather than giving the wealth. Therefore, we propose the corresponding transaction function, `win_take_biased`, which acts on the two randomly picked individuals as

$$X'_A = X_A + dU \cdot X_B - (1-d)U \cdot X_A, \tag{4}$$
$$X'_B = X_B + (1-d)U \cdot X_A - dU \cdot X_B, \tag{5}$$

where

X_A, X'_A, X_B, and X'_B are as previously stated,
$$d = \begin{cases} 1, & \text{with probability } \delta \text{ if } X_A > X_B \text{ and } (1-\delta) \text{ otherwise,} \\ 0, & \text{otherwise,} \end{cases}$$
$U \sim \text{Uniform}(0,1)$.

Using $\delta \in (0.5, 1)$, we can simulate the previous scenario in which the transaction is biased towards the richer party. Note that with $\delta = 0.5$, `win_take_biased` is the same as `win_take_partial`, and with $\delta \in (0, 0.5)$, it can be easily generalized to make the transaction biased towards the poorer party.

Proposition 3 (Resistance Principle [5]). Surplus should be viewed as being made up of layers and that the top layers are more fugitive, more easily lost than the bottom layers, those close to the level of subsistence.

From Proposition 3, we can see the layering of surplus and subsistence. Note that *layer* here does not mean an explicit structure of layers, but an implicit measure of the level of resistance to loss, meaning that in a society with more layers, any individual is less likely to lose to much of his wealth in a single transaction. Lenski also hypothesized that with the evolution of the industrial society, there will be more layers with increasing resistance to loss, and the total expectation of loss should thus drop [8]. This is modeled via the corresponding transaction function, `win_take_layer`, which acts on the two randomly picked individuals as

$$X'_A = X_A + dZ \cdot X_B - (1-d)Z \cdot X_A, \tag{6}$$

$$X'_B = X_B + (1-d)Z \cdot X_A - dZ \cdot X_B, \tag{7}$$

where

X_A, X'_A, X_B, X'_B, and d are as previously stated,

$$Z = \sum_{k=1}^{l} \frac{U_k^k}{l}, \quad \text{with } U_k \sim \text{Uniform}(0,1),\ k = 1, \cdots, n.$$

Note that using $l = 1$, the model becomes a single-layer economy, and thus `win_take_layer` becomes the same as `win_take_biased`. Moreover, we check that

$$\mathbb{E}(Z) = \frac{1}{l}\sum_{k=1}^{l} \mathbb{E}(U_k^k) = \frac{1}{l}\sum_{k=1}^{l} \int_0^1 x^k dx = \frac{1}{l}\sum_{k=1}^{l} \frac{1}{k+1}, \tag{8}$$

which significantly decreases as l increases (especially for small values of l), indicating the increase of resistance to loss since Z represents the proportion of wealth that the loser is likely to lose in each transaction.

Based on the propositions and observations as discussed above, we further embed a certain tax policy into the model of the economic system, which takes a portion from the amount of exchange and later distributes equally across the total population. This partially simulates the modern society, in which government policy and intervention (*e.g.*, taxing) takes microeconomic effects on the distribution of wealth across the society. The corresponding transaction function, `win_with_tax`, acts on the two randomly picked individuals as

$$X'_A = X_A + d \cdot (ZX_B - \texttt{tax}(ZX_B)) - (1-d) \cdot ZX_A, \tag{9}$$

$$X'_B = X_B + (1-d) \cdot (ZX_A - \texttt{tax}(ZX_A)) - d \cdot ZX_B, \tag{10}$$

where

X_A, X'_A, X_B, X'_B, d, and Z are as previously stated,

`tax` : the amount of the transaction $\mapsto$ the amount of tax taken for later distribution.

The `tax` function can take a simple proportion, or impose some more complicated taxing policy as we will specify when we simulate the real-world economy system later.

2.2 Distribution Fitting

To explicitly analyze the distribution of wealth as a result of the inequality process, we load all generic continuous random variables from the `scipy.stats` Python library. We take the normalized histogram (with the horizontal axis representing wealth and the vertical axis representing the number of people, normalized in the sense that the height of the histogram bars sum up to 1). For each distribution, we try to maximize its likelihood function $\mathcal{L}(\theta|x)$ with the normalized histogram where $\boldsymbol{\theta}$ denotes the set of parameters of the continuous random variable. The maximum likelihood estimation [4] can be done by

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} f(\mathbf{x}; \boldsymbol{\theta}) = \arg\min_{\boldsymbol{\theta}} \left(-\sum_{i=1}^{n} \log f(x_i, \boldsymbol{\theta}) \right), \tag{11}$$

where f is the probability density function, and $\mathbf{x}$ is the random vector as a sample of n independent random variables with this probability density function.

After fitting each continuous random variable to the normalized histogram (with the best set of parameters respectively), we compute for each of them the mean-squared error, and sort in ascending order to find the top 20 fits. We also perform the Kolmogorov-Smirnov test [3] which measures if the sample comes from a population with the specific distribution.

3 Validation

In this section, we will compare our results to the existing research results on real economic systems to validate our simulation. Note that our simulation is hard to validate in other ways, given that there are no invariants except for the total amount of wealth in the system (which is clearly constant since the gain of the winner and the loss of the loser coincide trivially by definition of the transaction functions). The simulation results will be discussed more in detail in Section 4.

Previous research on different countries has shown that the income distribution of the richest (the tail) approximates the Pareto distribution [7][10]. Moreover, the gamma distribution is a conjugate prior to the Pareto distribution, which is indistinguishable from the latter near the tail. Various simulation results of our model indicate that the gamma distribution is one of the best fits of the the normalized histogram (with different transaction functions and different sets of parameters), one of which as is shown in Figure 1. Note, however, that the Pareto distribution is not among the top 20 best fits of the normalized histogram in this graph. This is due to the implementation of the fit algorithm. We are using the maximum likelihood estimation, whereas the method of moments estimation may generate some highly accurate fits that are missed by the maximum likelihood estimation but also miss some nice fits that can be found by it. To conclude, our simulation results do agree with the previous research.

Another recent research on the economic exchange claims that the beta prime distribution is suitable for describing real-world wealth distribution, given that it can mimic various behaviors (*e.g.*, exponential behavior for both large and small values) [11]. Indeed, the beta prime distribution is also among the top 20 best fits of the normalized histogram shown in Figure 1. Moreover, we have another simulation with carefully chosen parameters that simulate the real-world economic system, the result of which is shown in Figure 2.

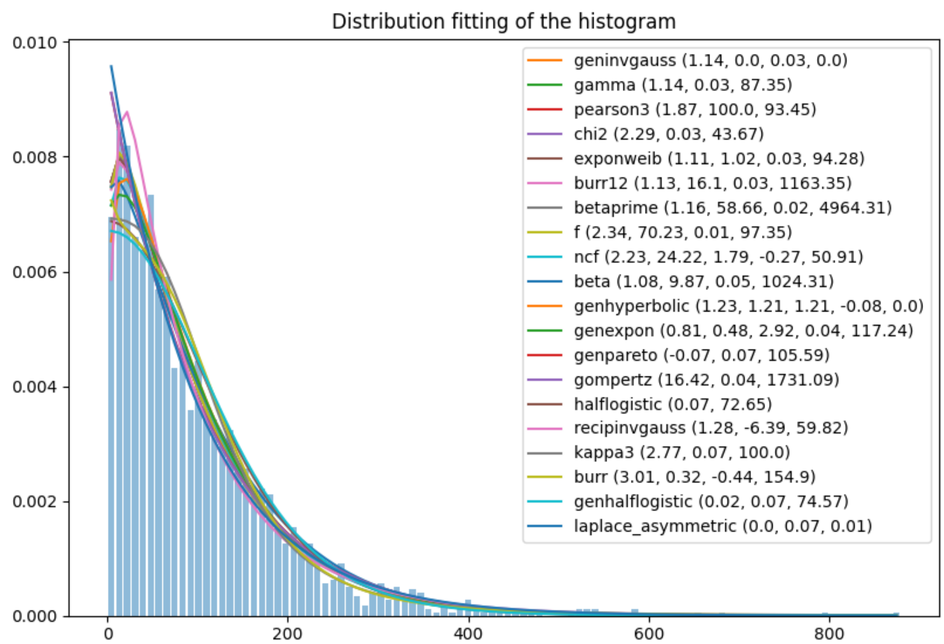


Figure 1: The gamma distribution is one of the best fits of the normalized histogram of some simulation result, which agrees to the existing research results.

4 Results and Discussion

In this section, we provide some of our simulation results. The full Python implementation of our simulation can be found at Yao Xiao's Github repository[3]. Note that due to the limitation of space, we will display only a small portion

[3]https://github.com/Charlie-XIAO/Econ-simulation

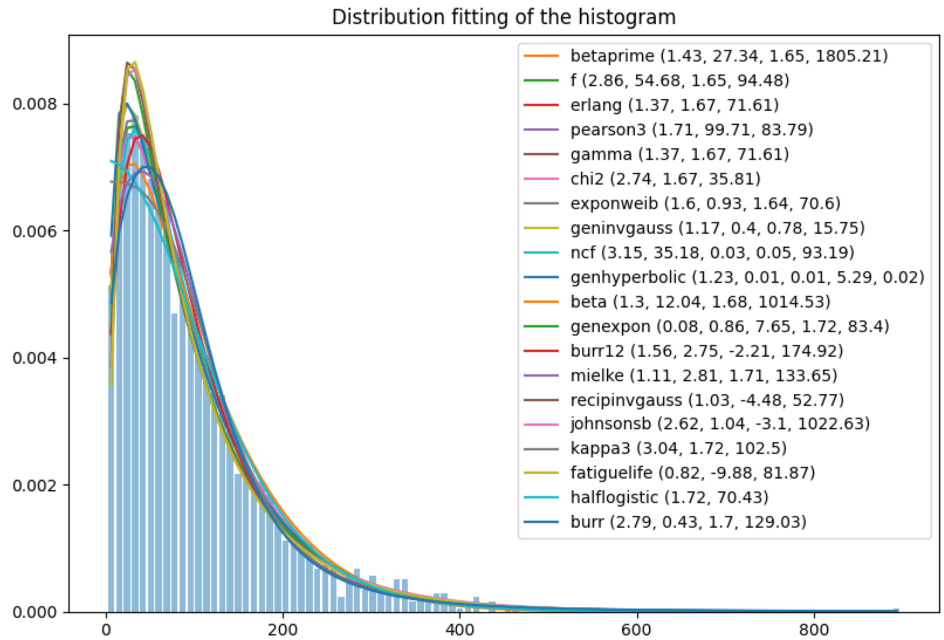


Figure 2: The beta prime distribution fits the normalized histogram of real-world economy simulation especially well, which agrees with the existing research results.

of figures and printouts of our simulation. The still figures of all our simulation results can be found here, and the printout and animations can be found here. See also the Appendices (Section A and Section B). Also note that in all our simulations, we use populations of size 2000 with mean initial wealth 100. We simulate for 2×10^4 steps for simulations in Section 4.1 and Section 4.2, and for 2×10^5 steps for simulations in Section 4.3 and Section 4.4.

For each simulation Section 4.1 and Section 4.2, we provide the following plots:

- **Gini coefficient**: the horizontal axis represents the number of exchanges, and the vertical axis represent the Gini coefficient of the economic system, measuring the inequality of the society.
- **Histogram**: the histogram of total population at the start and by the end of the simulation. The horizontal axis is the wealth, and each bar represents the number of people with a certain range of wealth. This graph vividly demonstrates the distribution of wealth across the society.
- **Ordered curves**: the ordered curves at the start and by the end of the simulation. This graph can visualize how the wealth of a person at a certain rank in the society changes during the simulation.
- **Percentiles**: the 1st, 5th, 25th, 50th, 75th, 95th, 99th percentiles are considered. The quantiles are of interest since they represent important statistical information. The other percentiles of interest represents the behavior of the wealthiest (respectively, the poorest) population.

4.1 The Evolution of the Society

In this section, we consider only the case of equally-distributed initial wealth. Together with the assumption that the mean of the initial wealth is 100, this means that each individual is initially assigned a wealth of 100.

The hunter-gatherer society. We use the transaction function `win_take_partial` for this part of simulation, and the plots are shown as in Figure 3.

From the plot of the Gini coefficient, we can see that the it converges to approximately 0.63, indicating a pretty high level of inequality in the society. From the distribution histograms, we can see more in detail that the distribution of the wealth is exponential-like. The ordered curves again confirms that most of the total wealth is concentrated in the

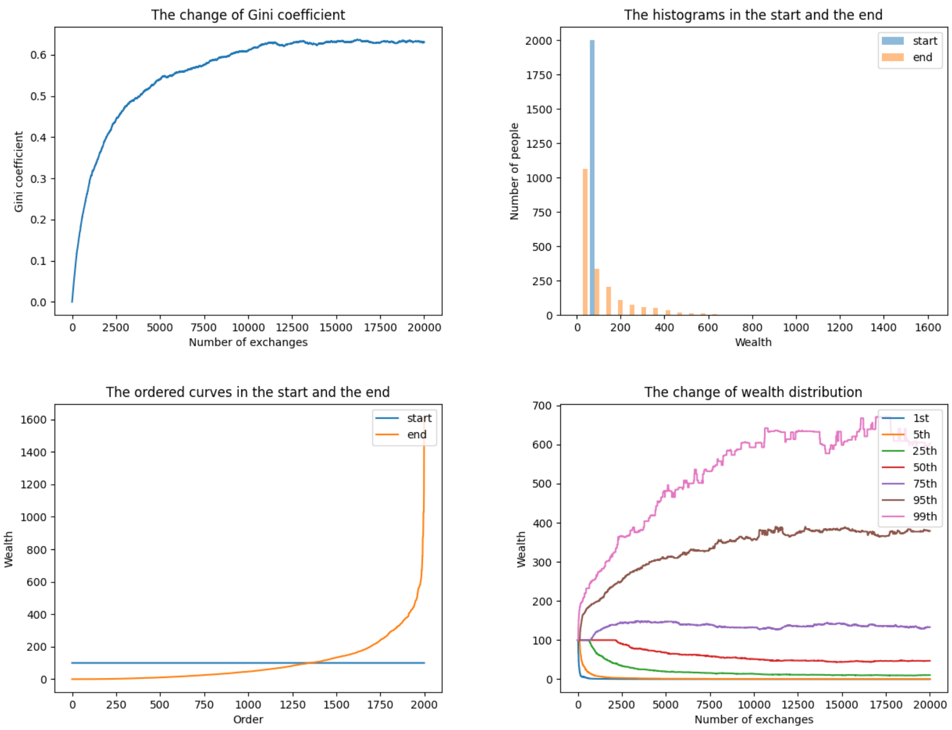


Figure 3: The simulation results of the hunter-gatherer society using the transaction function `win_take_partial`.

the very few richest individuals. Finally, from the graph of percentiles, we can see that the richest 1% is continuously becoming even richer, while the poorest 25% holds almost no wealth.

The ranked society. We use the transaction function `win_take_biased` for this part of simulation. There is a bias parameter δ involved in this transaction function, and we discuss two cases.

- $\delta \in (0.5, 1)$. In this case, the transaction is biased towards the richer party, meaning that richer individuals in the society are more likely to gain wealth than to lose wealth, and the poorer individuals in contrary. Therefore, this simulates a ranked society. We experimented on $\delta = 0.6$ and $\delta = 0.8$, and the plots for $\delta = 0.8$ are shown as in Figure 4.

  Comparing with the hunter-gatherer society, we can see that the Gini coefficient becomes even higher and keeps growing till the end of our simulation. Moreover, the graph of the percentiles shows that more than half of the (poorest) population owns almost no wealth at all, comparing to the 25% in the hunter-gatherer society. All of these results indicate a severer inequality in the ranked society, which is indeed the case by intuition.

- $\delta \in (0, 0.5)$. This case is not for simulating the ranked society (since the transaction is biased towards the poorer party), but for discovering if a large enough bias towards the poor can lead to equality in a society. We experimented on $\delta = 0.4$ and $\delta = 0.2$, but both still show a obvious inequality in the distribution of wealth across the society, though not as unequal as the previous case or as the hunter-gatherer society. Therefore, we even reduced to $\delta = 0$, where only wealthy people give money to the poor but not the converse. The simulation results of this extreme case are as shown in Figure 5.

  In this case, we can see that the Gini coefficient converges to only approximately 0.45, which is close to the Gini coefficient of the modern society nowadays. The histogram finally does not show an exponential shape, also meaning that the inequality is reduced. The ordered curves become much smoother, and from the graph of percentiles we can see that even the poorest 5% of people now possess a small amount of wealth. To conclude, using $\delta = 0$ simulate a system in which the inequality level is acceptable (compared to nowadays). However, it is unrealistic for such a society to exist.

The industrial society. We use the transaction function `win_take_layer` for this part of simulation. In addition to the bias parameter δ, we introduce a new layer parameter l, and we analyze its effects on the inequality process. Due to

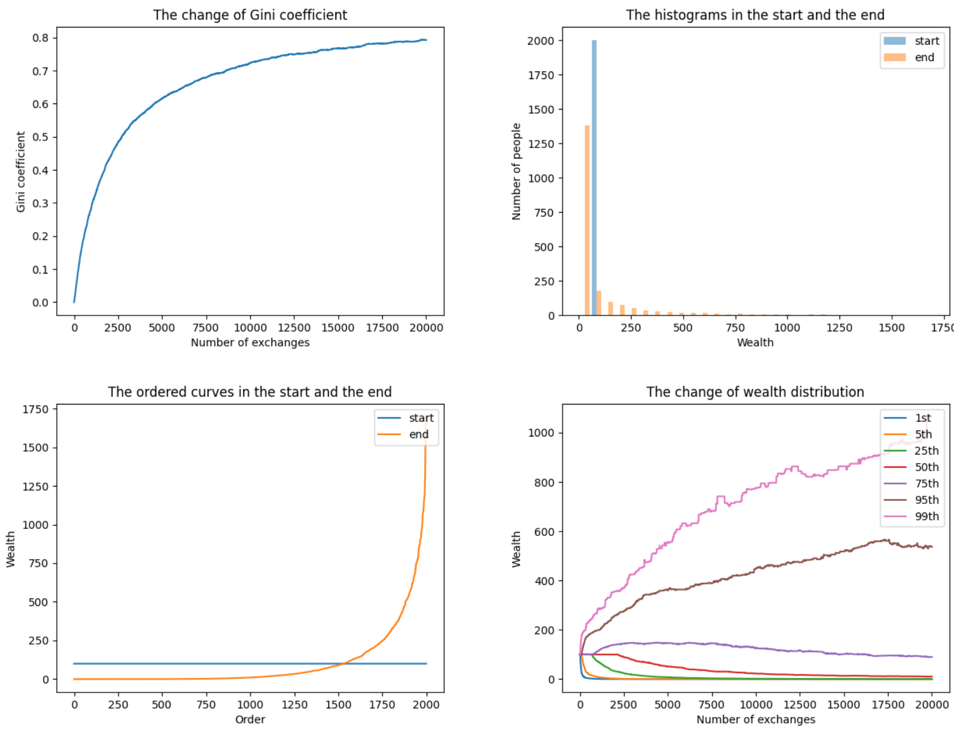


Figure 4: The simulation results of the ranked society with bias 0.8 towards the richer party in the economic system, using the transaction function `win_take_biased` with $\delta = 0.8$. This means that, the richer party in each transaction has 80% probability of gaining wealth while only 20% probability of losing wealth.

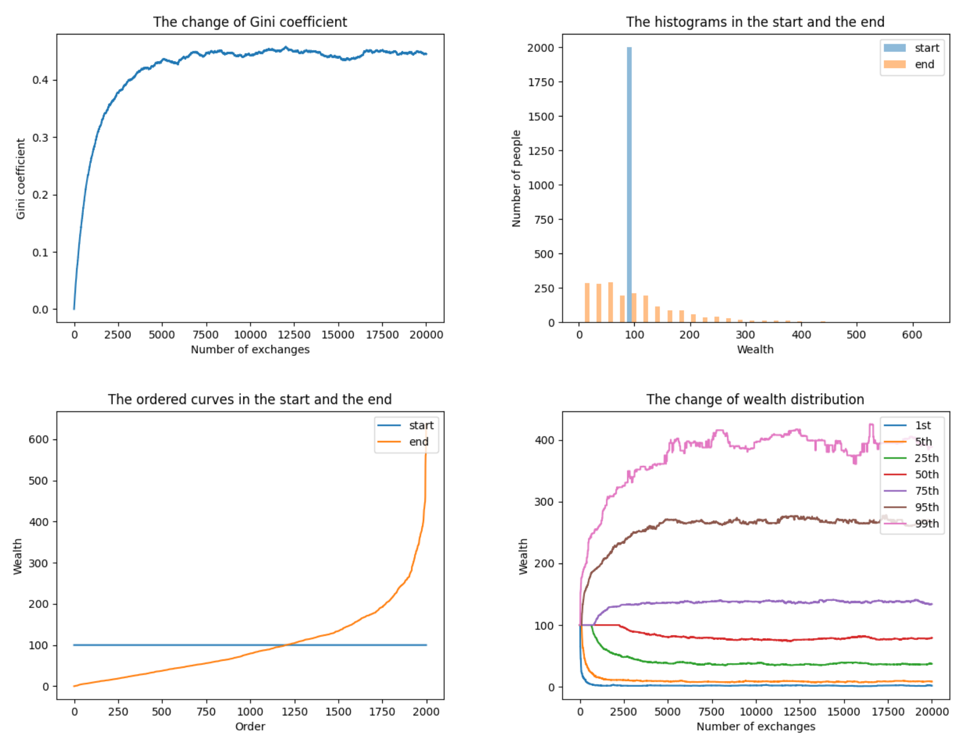


Figure 5: The simulation results of a society in which only wealthy people give money to the poor but not the converse, using the transaction function `win_take_biased` with $\delta = 0$.

the limitation of space, we will only show the results for $\delta = 0.5$ (which means there is no bias), with only the graphs of histograms as an illustration. Experiments are conducted for $l = 2$ and $l = 5$, whose results are as shown in Figure 6.

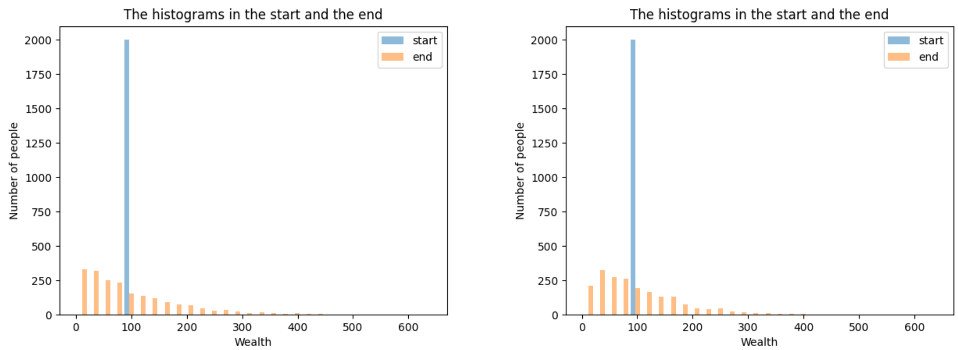


Figure 6: The simulation results of the industrial society using the transaction function `win_take_layer` with $\delta = 0.5$. The plot on the left is the result for $l = 2$, while the plot on the right is the result for $l = 5$.

The Gini coefficients, though not plotted, converge to approximately 0.47 and 0.42 for $l = 2$ and $l = 5$ respectively, which are decreasing as l increases, also comparing to the hunter-gatherer society in which $l = 1$. As for the histograms, the slope of the shape for $l = 2$ is much gentler than that for $l = 1$, and the shape of $l = 5$ is not even exponential-like but similar to a bell-shaped curve, indicating less inequality in the society. The graphs of the percentiles lead to the same conclusion. Indeed, this result agrees with the intuition that, with greater resistance to loss, one will be less likely to become poor very quickly.

The modern society. We use the transaction function `win_with_tax` for this simulation, introducing tax apart from the factors δ and l as previously discussed. In this section, we take the `tax` function to simply return a certain proportion of wealth for each exchange. Again, due to limitation of space, only the graphs of histograms will be plotted. Moreover, we select $\delta = 0.6$ and $l = 5$, which best approximates the reality. We experiment for extracting 3%, 10%, 20%, and 45% tax from each transaction, and their plots are shown in Figure 7.

We can see that 3% of tax does not make a great difference in the final wealth distribution, compared with previous results. However, as the tax rate increases, the wealth becomes more evenly distributed across the population. With just 10% of tax, the Gini coefficient comes to 0.45, and the histogram changes from the exponential-like shape to a bell shape, as compared with Figure 5 in which the rich are forced to give wealth to the poor. As the tax rate keep increasing, the bell shape becomes clearer, indicating an increasingly equal wealth distribution (indeed, the Gini coefficient for 45% tax rate is only approximately 0.36).

However, we also note that taxes, in the real world, are not just a simple proportion, but with more complicated strategies to stabilize the wealth distribution and minimize the inequality of the society. A real-world simulation will be introduced and discussed in Section 4.4.

4.2 The effect of the Initial Wealth Distribution

In this section, we will apply different distributions of initial wealth to our model.

- **Equally-distributed initial wealth**: This is as discussed in the previous section. All individuals within the simulated economic system will be assigned the same amount of initial wealth.
- **Uniformly-distributed initial wealth**: The initial wealth distribution follows a continuous uniform random variable. With the assumption that the initial mean of wealth is 100, `init` $\sim \mathrm{Uniform}(0, 100)$.
- **Normally-distributed initial wealth**: The initial wealth distribution follows a normal random variable. In addition to initial mean of 100, we further assume that initial standard deviation of wealth is 20 in our simulations. That is, `init` $\sim \mathcal{N}(100, 400)$, where $\mathcal{N}(\mu, \sigma^2)$ denotes the normal distribution.

We have repeated all the experiments done for the equally-distributed initial wealth on the other two types on initial wealth distributions, but the experiment on the transaction function `win_take_layer` is sufficient to show the results since the other experiments are similar. To illustrate the comparison more explicitly, we will use specific data instead of plots, as is shown in Table 1.

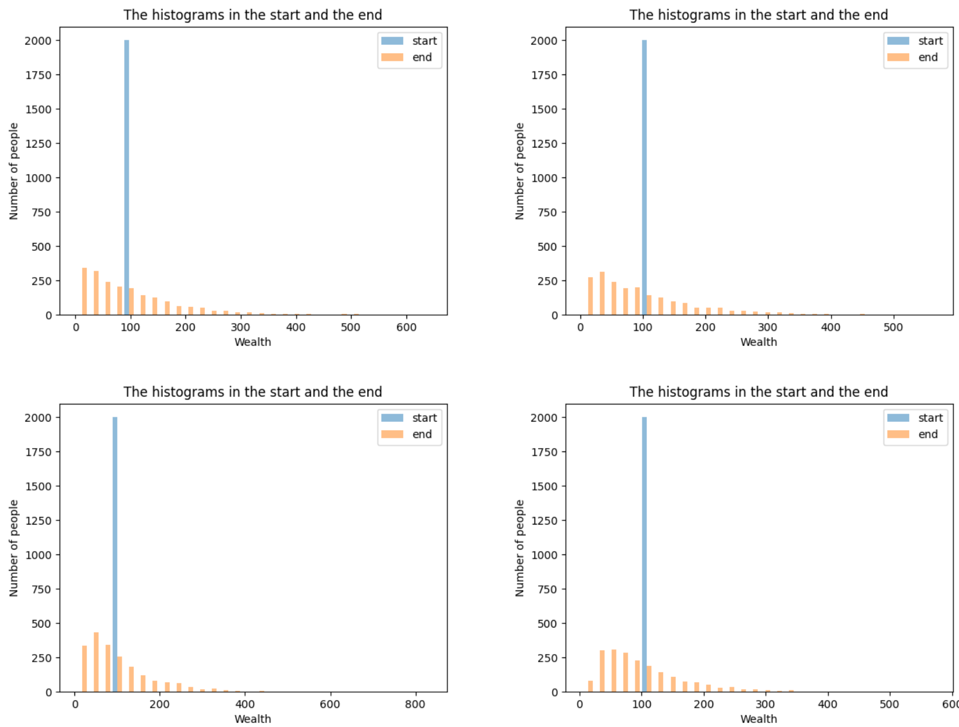


Figure 7: The simulation results of the modern society using the transaction function `win_with_tax` with $\delta = 0.6$, $l = 5$, and `tax` as the simple tax function. The plots (from left to right, and from top to bottom) are the results for `tax`$\% = 3\%$, 10%, 20%, and 45%, respectively.

Equally-distributed initial wealth:

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
20000	0.63	139.88	0	0	10	46	133	378	603

Uniformly-distributed initial wealth:

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.02	1	9	52	99	150	187	195
20000	0.63	133.99	0	0	10	46	137	374	633

Normally-distributed initial wealth:

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	20.22	53	68	86	100	114	132	145
20000	0.64	142.29	0	0	10	43	128	388	664

Table 1: The simulation of the hunter-gatherer society, given different distributions of initial wealth. The columns `gini` represent the Gini coefficient, `std` represents the standard deviation, and the percentages represent the corresponding percentiles. Step `0` and Step `20000` refer to the start and the end of the simulation, respectively.

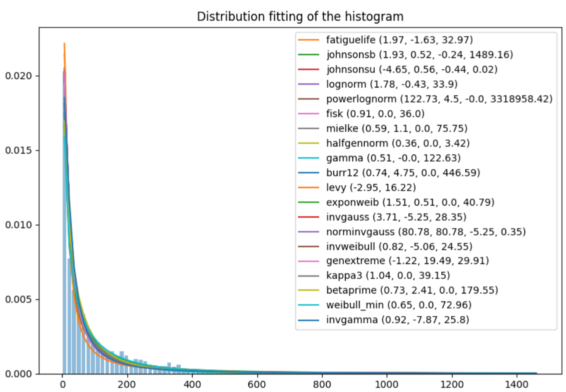
We can see that the final Gini coefficient for all three kinds of initial wealth distributions are similar, namely around 0.63. Moreover, corresponding percentiles show a similar amount of wealth. In particular, the 5% poorest population occupies no wealth, and the top 1% richest population occupies most of the wealth ($600+$). Therefore, we can conclude that different distributions of initial wealth will not contribute to significant difference in the final distribution of wealth in the long run.

4.3 Distribution Fitting

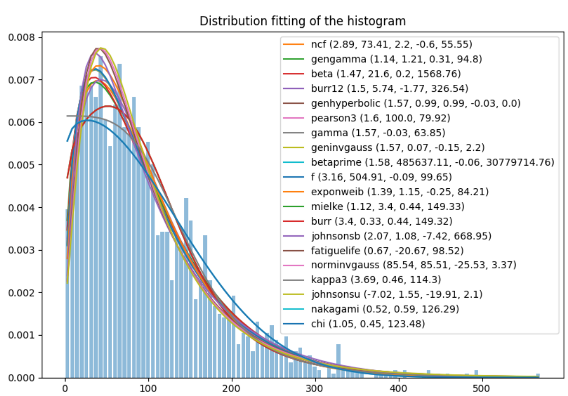
In this section, we fit then continuous probability distributions from `scipy.stats` Python library to the normalized histogram. Experiments are performed on the following sets of parameters for equally-distributed initial wealth.

- $\delta = 0.5, l = 1$: the shape of the histogram is exponential-like.
- $\delta = 0.6, l = 1$: the shape of the histogram is exponential-like, and is much steeper than the previous case.
- $\delta = 0.5, l = 5$: the histogram shows a bell shape.
- $\delta = 0.6, l = 5$: the histogram shows a bell shape, but is less obvious than the previous case, with a very short left tail (the part to the left of the top of the bell).

These four cases cover all the shapes of the histogram that we observe in the previous experiments. Therefore, it is sufficient to perform distribution fitting for these cases. Due to the limitation of space, we will show only the plots for the second and the third case as in Figure 8, with the former being "the most unequal" and the latter being "the most equal" among these four test cases.




```
steps=200000, bias=60%, layer=1
+-------------+----------+----------+
|    distr    |   MSE    | KS-stat  |
+-------------+----------+----------+
|   lognorm   | 7.87E-06 | 6.93E-02 |
| powerlognorm | 9.49E-06 | 5.02E-02 |
|    gamma    | 1.30E-05 | 1.11E-01 |
|  betaprime  | 2.06E-05 | 7.16E-02 |
|   invgamma  | 2.54E-05 | 7.73E-02 |
+-------------+----------+----------+
```




```
steps=200000, bias=50%, layer=5
+-------------+----------+----------+
|    distr    |   MSE    | KS-stat  |
+-------------+----------+----------+
|   gengamma  | 1.64E-05 | 1.41E-02 |
|     beta    | 1.67E-05 | 1.47E-02 |
|    gamma    | 1.76E-05 | 1.88E-02 |
|  betaprime  | 1.76E-05 | 1.87E-02 |
|     chi     | 2.87E-05 | 4.93E-02 |
+-------------+----------+----------+
```

Figure 8: The plots and printout for distribution fitting. In the graphs on the left, the top 20 fits are listed in order, and the data in the parentheses are their corresponding parameters. In the printout on the right, the mean-squared error `MSE` and the Kolmogorov-Smirnov test statistics `KS-stat` of some well-known distributions among the top 20 fits are listed.

We can see that in both cases shown here, the gamma distribution and the betaprime distribution fit the normalized histogram very well, and in fact they are among the top 20 in almost all previous cases. This agrees with the research results on the income distribution of various different countries [7][10], as is also specified in Section 3.

4.4 Case Study: Real-World Economy

In this section, we perform an experiment simulating the real-world economy of China. In China, residents are subject to the taxes on personal income enforced by People's Republic of China, in which different levels of income are subject to different tax rates [6]. For instance, if a resident of China has 100000 CNY of annual income, 36000 of it will be applied a tax rate of 3% and the rest a tax rate of 10%. We simulate this policy and design the piecewise constant tax rates, with the jumps at the same proportion as the original policy, but adjusted to fit the initial mean of wealth in our model. The corresponding mapping of `tax` is visualized in Figure 9.

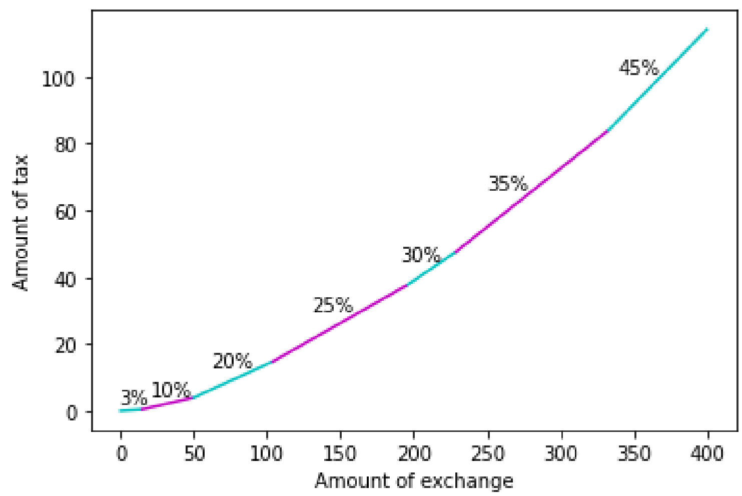


Figure 9: The function `tax` that is used for simulating the taxes on personal income enforced by People's Republic of China in our model, where the initial mean wealth is 100.

This `tax` function is applied to the transaction function `with_with_tax`. Moreover, we consider $\delta = 0.6$ and $l = 5$ as the best set of parameters to simulate the real-world economy today. We simulate for 2×10^5 step for the accuracy of our result, and we perform distribution fitting as well. Due to the limitation of space, we show only the plot of Gini coefficient and the plot of percentiles to prove that the distribution of wealth has indeed reached a steady state (Figure 10), and fitted normalized histogram to reflect the distribution of wealth in our simulated real-world economy (Section 3, Figure 2).

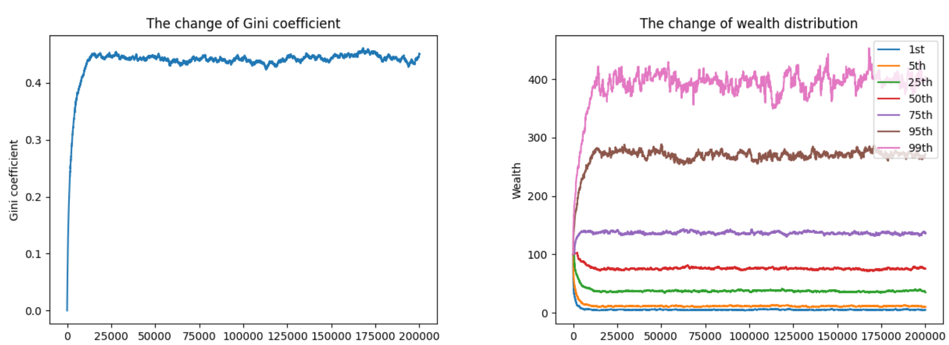


Figure 10: The simulation results of the real-world economy using the transaction function `win_with_tax` with $\delta = 0.6$, $l = 5$, and the `tax` function as described above. The distribution fitting is shown in Section 3, Figure 2, which is not shown here again due to the limitation of space.

As we can see, the Gini coefficient converges to approximately 0.45, and China's Gini coefficient has been around this value for decades [2]. The plot of percentiles also indicate a reasonable behavior, in which only the wealth of the top 10% richest population varies a lot along with the simulation while the wealth of other people are stable. From Section 3, Figure 2, we again claim that the gamma distribution, as discussed in Section 4.3, fits the normalized histogram very well, and the beta prime distribution fits it even better, with both results agreeing with the researches on the modern world economic exchange and wealth distribution [7][10][11]. Though this does not necessarily mean that we are simulating the real-world economy in a correct way and following the real-world economic activities, this does indicate that our model, to some extent, reflects the wealth distribution of a real-world society and the inequality process correctly.

5 Summary and Conclusions

In this paper, we described a way of modeling the inequality process using random transactions within a population with certain transaction functions and restrictions. We followed the evolution of society, and analyzed the effect of bias and layers in the transactions on the distribution of wealth across the society. That is, with a lower bias towards the rich and more layers (*i.e.*, a more industrialized society), the distribution of wealth tends to be more equal. However, in the real world, it is unrealistic to impose any set of parameters to reduce inequality. Therefore, tax is introduced into our model, and we also performed a real-world simulation using the policy of taxes on personal income of People's Republic of China, along with a reasonable set of bias and layers. The results turn out to agree with the real world economy, in a sense that the Gini coefficient is around 0.45, and the distributions that fit well with the normalized histogram include the gamma distribution and the beta prime distribution. This in turn validates the feasibility of our modeling in reflecting the inequality process and the distribution of wealth across the society.

References

[1] 2017. Counterintuitive Problem: Everyone in a room keeps giving dollars to random others. You'll never guess what happens next. http://www.decisionsciencenews.com/?p=6221

[2] 2022. China: Gini Coefficient from 2004 to 2021. https://www.statista.com/statistics/250400/inequality-of-income-distribution-in-china-based-on-the-gini-index/

[3] 2022. Kolmogorov–Smirnov Test. https://en.wikipedia.org/w/index.php?title=Kolmogorov%E2%80%93Smirnov_test&oldid=1125984976 Page Version ID: 1125984976.

[4] 2022. Maximum likelihood estimation. https://en.wikipedia.org/w/index.php?title=Maximum_likelihood_estimation&oldid=1128284664 Page Version ID: 1128284664.

[5] John Angle. 1986. The Surplus Theory of Social Stratification and the Size Distribution of Personal Wealth. *Social Forces* 65, 2 (Dec. 1986), 293. https://doi.org/10.2307/2578675

[6] Jane Cheung. 2022. People's Republic of China: Taxes on Personal Income. https://taxsummaries.pwc.com/peoples-republic-of-china/individual/taxes-on-personal-income

[7] Frank Cowell. 2011. *Measuring Inequality*. Oxford University Press.

[8] Gerald E. Lenski. 2013. *Power and Privilege: A theory of Social Stratification*. UNC Press Books.

[9] László Lovász. 1994. Random Walks on Graphs. 2, 1 (May 1994), 1–37.

[10] Harold Lydall. 1968. *The Structure of Earnings*. Clarendon Press.

[11] M. Dashti Moghaddam, Jeffrey Mills, and R. A. Serota. 2019. Generalized Beta Prime Distribution: Stochastic Model of Economic Exchange and Properties of Inequality Indices. (2019). https://doi.org/10.48550/ARXIV.1906.04822

Appendix

A Implementation and Python Code

A.1 Population Setup

```
class Population(ABC):                                    # The abstract class
    def simulate(self, transaction, n:int, *args):      # The simulation method
        """
        :param function: the transaction function to use in the simulation
        :param n: the total number of transactions to simulate
        """
        for _ in range(n):
            A, B = random.sample(range(self.size), 2)   # Select two people at random
            self.update(transaction(self.history[-1], A, B, *args))
                                                        # Perform a transaction

class EqualPopulation(Population):                      # Inherits the abstract class
    def __init__(self, size:int, mean:float) -> None:
        self.size = size
        self.history = [np.array([mean] * size)]
    # Inherits the ‘simulate‘ method

class UniformPopulation(Population):                    # Inherits the abstract class
    def __init__(self, size:int, mean: float) -> None:
        self.size = size
        sample = np.random.uniform(0, 200, size)        # Generate a uniform sample
        factor = size * mean / np.sum(sample)           # Fit the sample to the mean
        self.history = [sample * factor]
    # Inherits the ‘simulate‘ method

class NormalPopulation(Population):                     # Inherits the abstract class
    def __init__(self, size:int, mean:float, std:float) -> None:
        self.size = size
        self.history = [np.random.normal(mean, std, size)]
    # Inherits the ‘simulate‘ method
```

A.2 Transaction Functions

```
def win_take_partial(population:np.ndarray, A:int, B:int) -> np.ndarray:
    """
    :param population: the population in which the transaction takes place
    :param A: the index of one person in the population to make transaction
    :param B: the index of the other person in the population to make transaction
    :return: the new population after (if exists) the transaction
    """
    result = np.copy(population)
    ratio = np.random.uniform(0, 1)             # Determine the ratio to give/take
    if random.random() < 0.5:                   # Determine the winner
        result[A] = population[A] + ratio * population[B]
        result[B] = population[B] - ratio * population[B]
    else:
        result[B] = population[B] + ratio * population[A]
        result[A] = population[A] - ratio * population[A]
    return result

def win_take_biased(population:np.ndarray, A:int, B:int, bias:float) -> np.ndarray:
    """
    :param bias: the bias towards the richer party in the transaction, should be in [0, 1]
```

```
    :return: the new population after (if exists) the transaction
    Other params same as above
    """
    assert(bias >= 0 and bias <= 1)
    result = np.copy(population)
    ratio = np.random.uniform(0, 1)             # Determine the ratio to give/take
    if population[A] > population[B]:          # Determine the richer party
        richer, poorer = A, B
    else:
        richer, poorer = B, A
    if random.random() < bias:                  # Determine the winner
        result[richer] = population[richer] + ratio * population[poorer]
        result[poorer] = population[poorer] - ratio * population[poorer]
    else:
        result[poorer] = population[poorer] + ratio * population[richer]
        result[richer] = population[richer] - ratio * population[richer]
    return result

def win_take_layer(population:np.ndarray, A:int, B:int, bias:float, layers:int) -> np.ndarray:
    """
    :param layers: the number of layers that forces a resistance to loss from the loser
    :return: the new population after (if exists) the transaction
    Other params same as above
    """
    assert(bias >= 0 and bias <= 1)
    result = np.copy(population)
    ratio = np.sum(np.square(np.random.uniform(0, 1, layers)) / layers)
                                                # Determine the ratio to give/take
    if population[A] > population[B]:          # Determine the richer party
        richer, poorer = A, B
    else:
        richer, poorer = B, A
    if random.random() < bias:                  # Determine the winner
        result[richer] = population[richer] + ratio * population[poorer]
        result[poorer] = population[poorer] - ratio * population[poorer]
    else:
        result[poorer] = population[poorer] + ratio * population[richer]
        result[richer] = population[richer] - ratio * population[richer]
    return result

def win_with_tax(population:np.ndarray, A:int, B:int, bias:float, layers:int):
    """
    :return: the new population after (if exists) the transaction
    Params same as above. `tax` is a function that takes an amount of exchange,
    and returns the tax to be taken, according to some specific tax policy
    """
    result = np.copy(population)
    ratio = np.sum(np.square(np.random.uniform(0, 1, layers)) / layers)
                                                # Determine the ratio to give/take
    if population[A] > population[B]:          # Determine the richer party
        richer, poorer = A, B
    else:
        richer, poorer = B, A
    if random.random() < bias:                  # Determine the winner
        exchange_amount = ratio * population[poorer]
        tax = tax(exchange_amount, init_mean)   # Determine the amount of tax
        result += tax / len(population)         # Evenly distribute the tax
        result[richer] = population[richer] + exchange_amount - tax
        result[poorer] = population[poorer] - exchange_amount
```

```
    else:
        exchange_amount = ratio * population[richer]
        tax = tax(exchange_amount, init_mean)   # Determine the amount of tax
        result += tax / len(population)         # Evenly distribute the tax
        result[poorer] = population[poorer] + exchange_amount - tax
        result[richer] = population[richer] - exchange_amount
    return result
```

A.3 Distribution Fitting

For the each distribution fitting, we refer to the 'scipy.stats' documentation. The code is long and complicated, thus will not be explicitly shown here but implicitly represented by the `fit` method that returns the set of parameters of the fitted distribution.

```
for name in distributions:              # Iterate all distributions in 'scipy.stats'
    try:
        # Fitting distribution and computing errors
        distribution = eval("stats.{}".format(name))
        param = distribution.fit(data)                  # Get fitted parameters
        fit_pdf = distribution.pdf(hist_bins, *param)   # Get probability density
        mse = np.sum(np.square(fit_pdf - hist))         # Compute mean-squared error
        ks_stat, ks_pval = stats.kstest(data, distribution(*param).cdf)
                                                        # Perform Kolmogorov-Smirnov test
        # Storing information
        mses.append(mse)
        ks_stats.append(ks_stat)
        ks_pvals.append(ks_pval)
        pdfs[name] = fit_pdf
        params[name] = tuple([float("{0:.2f}".format(n)) for n in param])

    except:                             # Failed to fit the distribution
        # Mark errors as infinity to avoid affecting the final result
        mses.append(np.inf)
        ks_stats.append(np.inf)
        ks_pvals.append(np.inf)
        pdfs[name] = None
        params[name] = (,)

# Create dataframe for data collection
df_info = pd.DataFrame({"MSE": mses, "KS-stat": ks_stats, "KS-pval": ks_pvals})
df_info.index = distributions
# Obtain the top 20 based on mean-squared error
best_fits = df_info.sort_values(by="MSE").index[0:min(20, len(distributions))]

# Plot the fitting distributions
for name in best_fits:
    plt.plot(hist_bins, pdfs[name], label="{} {}".format(name, params[name]))
# Initialize the output table of fitting errors
table = PrettyTable()
table.field_names = ["distr", "MSE", "KS-stat", "KS-pval"]
for name in best_fits:
    table.add_row([name,
        "{:.2E}".format(df_info.loc[name, "MSE"]),
        "{:.2E}".format(df_info.loc[name, "KS-stat"]),
        "{:.2E}".format(df_info.loc[name, "KS-pval"]),
    ])
```

B Printouts of Simulation Results

In this section, we will provide the full printouts of all our experiments. Apart from this section, you may also find the still figures (with printouts) of all our simulation results here, and corresponding animations (with printouts) here.

B.1 Equally-Distributed Initial Wealth

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.41	74.62	0	3	41	100	139	237	320
4000	0.51	97.35	0	1	23	73	147	294	404
6000	0.56	113.12	0	1	17	62	140	323	490
8000	0.59	122.36	0	0	14	56	133	356	557
10000	0.61	133.63	0	0	13	51	132	357	592
12000	0.62	135.18	0	0	12	47	137	365	613
14000	0.63	134.91	0	0	11	45	140	382	597
16000	0.63	140.42	0	0	9	45	136	376	638
18000	0.63	138.91	0	0	9	46	136	372	609
20000	0.63	139.88	0	0	10	46	133	378	603

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 80% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.43	80.77	0	2	31	100	139	257	356
4000	0.57	113.32	0	0	11	64	145	342	477
6000	0.64	136.25	0	0	5	42	145	363	632
8000	0.69	153.72	0	0	2	30	140	402	741
10000	0.72	166.94	0	0	1	22	125	449	776
12000	0.75	177.94	0	0	1	18	114	469	843
14000	0.76	185.20	0	0	0	14	104	502	831
16000	0.77	191.89	0	0	0	13	101	537	891
18000	0.78	201.22	0	0	0	12	95	550	957
20000	0.79	209.72	0	0	0	10	90	536	1065

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.42	78.41	0	3	39	100	136	249	363
4000	0.54	106.00	0	1	17	72	143	308	482
6000	0.60	122.39	0	0	11	56	144	346	565
8000	0.62	133.45	0	0	9	48	142	361	636
10000	0.64	136.92	0	0	7	42	138	389	619
12000	0.65	144.07	0	0	7	41	137	390	671
14000	0.67	148.14	0	0	6	39	131	416	681
16000	0.67	147.34	0	0	5	34	134	412	670
18000	0.68	153.62	0	0	5	35	125	416	711
20000	0.69	158.94	0	0	4	34	124	403	789

Equal population, size=2000, mean=100.0, simulating 20000 steps

```
Exchange strategy: winner takes random proportion of wealth from the loser
                   however, the richer party has 40% chance of winning
+-------+------+--------+-----+-----+-----+-----+-----+-----+-----+
|  step | gini |  std   |  1% |  5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+--------+-----+-----+-----+-----+-----+-----+-----+
|   0   | 0.00 |  0.00  | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
|  2000 | 0.40 | 74.30  |  0  |  5  |  41 | 100 | 135 | 239 | 331 |
|  4000 | 0.49 | 93.68  |  0  |  3  |  27 |  77 | 145 | 277 | 422 |
|  6000 | 0.54 | 107.54 |  0  |  2  |  22 |  64 | 143 | 323 | 462 |
|  8000 | 0.55 | 110.53 |  0  |  1  |  21 |  61 | 139 | 330 | 493 |
| 10000 | 0.57 | 114.67 |  0  |  1  |  17 |  59 | 141 | 341 | 502 |
| 12000 | 0.58 | 121.11 |  0  |  1  |  17 |  56 | 137 | 342 | 546 |
| 14000 | 0.58 | 121.83 |  0  |  1  |  16 |  56 | 137 | 346 | 567 |
| 16000 | 0.58 | 120.38 |  0  |  1  |  15 |  57 | 137 | 343 | 543 |
| 18000 | 0.58 | 123.44 |  0  |  1  |  16 |  54 | 137 | 339 | 569 |
| 20000 | 0.59 | 125.50 |  0  |  1  |  15 |  55 | 140 | 339 | 590 |
+-------+------+--------+-----+-----+-----+-----+-----+-----+-----+

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
                   however, the richer party has 20% chance of winning
+-------+------+--------+-----+-----+-----+-----+-----+-----+-----+
|  step | gini |  std   |  1% |  5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+--------+-----+-----+-----+-----+-----+-----+-----+
|   0   | 0.00 |  0.00  | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
|  2000 | 0.37 | 69.05  |  1  |  8  |  48 |  98 | 131 | 235 | 299 |
|  4000 | 0.44 | 83.41  |  0  |  5  |  36 |  82 | 140 | 261 | 374 |
|  6000 | 0.48 | 92.12  |  0  |  4  |  31 |  74 | 138 | 282 | 424 |
|  8000 | 0.49 | 96.47  |  0  |  4  |  29 |  71 | 141 | 290 | 428 |
| 10000 | 0.50 | 99.56  |  0  |  4  |  27 |  71 | 140 | 291 | 441 |
| 12000 | 0.50 | 100.15 |  0  |  4  |  29 |  69 | 140 | 290 | 453 |
| 14000 | 0.51 | 101.91 |  0  |  3  |  26 |  69 | 143 | 297 | 471 |
| 16000 | 0.51 | 102.71 |  0  |  2  |  26 |  67 | 141 | 311 | 462 |
| 18000 | 0.53 | 108.02 |  0  |  3  |  24 |  64 | 138 | 313 | 512 |
| 20000 | 0.52 | 107.24 |  0  |  3  |  26 |  66 | 135 | 314 | 484 |
+-------+------+--------+-----+-----+-----+-----+-----+-----+-----+

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
                   however, the richer party has 0% chance of winning
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+
|  step | gini |  std  |  1% |  5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+
|   0   | 0.00 |  0.00 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
|  2000 | 0.36 | 66.85 |  2  |  10 |  52 | 100 | 129 | 227 | 312 |
|  4000 | 0.42 | 79.10 |  2  |  9  |  39 |  84 | 135 | 260 | 353 |
|  6000 | 0.43 | 83.19 |  1  |  8  |  38 |  80 | 137 | 264 | 394 |
|  8000 | 0.45 | 86.86 |  1  |  7  |  35 |  77 | 139 | 263 | 415 |
| 10000 | 0.44 | 84.62 |  2  |  8  |  37 |  77 | 138 | 263 | 395 |
| 12000 | 0.46 | 89.08 |  2  |  9  |  34 |  74 | 136 | 276 | 414 |
| 14000 | 0.45 | 85.46 |  2  |  8  |  35 |  77 | 139 | 269 | 391 |
| 16000 | 0.44 | 84.75 |  0  |  7  |  38 |  80 | 136 | 264 | 380 |
| 18000 | 0.45 | 86.30 |  2  |  9  |  36 |  77 | 138 | 266 | 394 |
| 20000 | 0.44 | 85.95 |  1  |  8  |  37 |  79 | 133 | 264 | 388 |
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
                   with the loser resisting the loss of wealth at Lvl. 2
                   the richer party has 80% chance of winning
+-------+------+--------+-----+-----+-----+-----+-----+-----+-----+
|  step | gini |  std   |  1% |  5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+--------+-----+-----+-----+-----+-----+-----+-----+
|   0   | 0.00 |  0.00  | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
|  2000 | 0.32 | 57.46  |  6  |  17 |  56 | 100 | 130 | 209 | 270 |
```

4000	0.43	80.06	1	7	35	82	144	258	339
6000	0.51	98.67	0	3	24	67	147	295	433
8000	0.56	111.21	0	2	18	59	145	326	486
10000	0.59	120.38	0	1	14	53	143	343	529
12000	0.62	130.25	0	0	11	49	136	384	584
14000	0.64	137.97	0	0	9	43	131	401	627
16000	0.65	143.35	0	0	8	41	135	425	656
18000	0.67	149.99	0	0	7	36	131	437	709
20000	0.67	153.83	0	0	6	35	130	408	742

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.29	52.88	7	20	62	100	126	201	253
4000	0.38	70.10	3	12	46	87	137	236	312
6000	0.43	80.46	2	7	38	79	144	253	360
8000	0.46	87.42	1	6	33	75	143	269	388
10000	0.48	94.90	1	5	29	72	139	283	428
12000	0.50	98.43	1	4	27	69	141	292	444
14000	0.51	102.68	1	3	24	68	142	294	496
16000	0.52	104.08	0	3	25	66	142	297	488
18000	0.52	106.82	0	3	25	66	141	308	481
20000	0.52	104.40	0	2	24	65	140	313	482

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 50% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.27	50.03	9	27	63	100	127	191	240
4000	0.36	64.91	4	15	50	90	138	223	296
6000	0.40	75.63	2	10	42	84	140	243	332
8000	0.43	80.98	2	8	39	79	140	261	358
10000	0.44	84.96	1	7	36	77	138	268	382
12000	0.45	86.62	1	8	36	75	137	278	392
14000	0.47	90.44	1	7	33	73	140	274	429
16000	0.47	92.88	1	8	33	73	138	281	445
18000	0.47	91.32	1	7	32	74	139	280	411
20000	0.47	90.48	1	8	32	74	140	281	403

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 40% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.27	48.21	10	26	66	100	125	189	235
4000	0.34	62.56	7	18	54	89	132	217	288
6000	0.37	70.02	6	15	48	85	135	237	318
8000	0.40	75.00	4	13	46	81	136	255	356
10000	0.41	76.69	4	14	41	81	139	247	337
12000	0.41	79.21	2	12	42	78	136	251	365

14000	0.41	78.79	2	11	42	81	136	253	365
16000	0.42	80.96	3	12	42	80	133	262	380
18000	0.42	81.22	2	11	39	80	136	261	379
20000	0.43	81.57	3	11	38	79	139	257	383

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 20% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.24	43.88	17	33	70	100	123	177	225
4000	0.29	52.91	12	27	60	93	129	195	255
6000	0.31	57.42	10	25	59	91	129	209	288
8000	0.32	59.59	9	24	56	89	130	217	278
10000	0.33	61.04	11	25	55	89	129	222	294
12000	0.33	62.03	11	24	55	86	130	220	298
14000	0.34	62.45	11	22	53	87	132	218	290
16000	0.34	62.97	8	22	54	87	131	223	301
18000	0.33	62.04	8	22	54	87	131	220	290
20000	0.34	62.00	8	22	54	89	132	218	291

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 0% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.22	39.71	20	41	71	100	121	172	211
4000	0.26	47.19	17	34	67	94	125	185	248
6000	0.26	47.69	18	34	63	94	130	181	241
8000	0.27	49.24	16	32	63	93	129	189	246
10000	0.28	50.21	18	31	62	94	129	192	248
12000	0.28	50.16	19	32	63	92	127	194	253
14000	0.28	50.69	17	32	63	92	126	194	249
16000	0.27	48.64	16	32	65	93	126	193	246
18000	0.27	49.65	15	31	64	92	128	190	241
20000	0.28	50.48	15	32	63	92	126	195	254

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 80% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.30	54.69	12	25	59	100	129	201	269
4000	0.42	78.04	4	11	40	81	140	247	344
6000	0.48	92.04	1	5	29	71	143	281	388
8000	0.52	101.45	0	3	22	68	143	308	440
10000	0.55	108.15	0	2	19	62	144	327	493
12000	0.58	117.64	0	1	17	56	139	345	546
14000	0.60	124.96	0	1	14	51	136	375	566
16000	0.61	128.01	0	0	13	49	137	382	577
18000	0.62	131.91	0	0	12	46	133	390	589
20000	0.63	134.98	0	0	11	44	136	376	595

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.27	48.46	16	31	64	100	129	188	237
4000	0.36	65.73	9	18	50	87	135	229	306
6000	0.40	74.60	4	12	43	82	140	250	346
8000	0.43	81.88	3	10	37	79	140	262	370
10000	0.45	86.73	2	8	33	77	140	280	383
12000	0.47	91.87	2	6	32	74	142	275	409
14000	0.47	92.61	1	6	33	73	139	281	424
16000	0.46	90.96	1	6	33	73	142	276	405
18000	0.46	90.94	2	7	33	73	140	270	386
20000	0.47	93.03	1	6	33	74	138	280	404

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 50% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.26	47.07	18	32	64	100	126	186	232
4000	0.33	60.28	9	23	53	90	133	219	267
6000	0.36	66.69	7	18	50	86	136	229	311
8000	0.38	70.76	6	16	46	83	138	231	338
10000	0.39	73.59	5	14	45	81	138	240	330
12000	0.40	76.55	5	14	45	81	135	252	351
14000	0.41	77.22	4	12	43	80	134	250	361
16000	0.41	78.84	4	12	42	79	137	254	361
18000	0.41	78.61	4	12	43	80	138	254	362
20000	0.42	79.90	4	11	41	79	138	251	366

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 40% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.24	43.73	20	35	68	100	126	177	216
4000	0.30	53.79	14	27	60	90	132	198	266
6000	0.32	58.69	12	25	56	88	132	208	287
8000	0.34	62.22	11	21	53	88	135	210	288
10000	0.34	63.83	11	21	54	87	131	223	305
12000	0.35	64.48	8	20	54	86	134	219	310
14000	0.36	67.77	7	19	51	85	132	233	319
16000	0.36	66.94	9	20	50	86	133	221	314
18000	0.36	66.62	9	20	51	84	133	227	320
20000	0.36	68.74	7	20	50	83	133	231	331

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 20% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.22	39.33	27	43	71	100	123	169	203
4000	0.25	45.77	22	37	67	93	127	180	238
6000	0.26	48.24	21	36	63	92	127	192	249
8000	0.26	47.96	22	35	64	92	126	185	254
10000	0.27	48.95	19	34	64	91	128	190	251
12000	0.27	49.47	20	34	64	92	127	193	251
14000	0.28	50.20	21	33	62	91	128	195	241
16000	0.27	50.50	21	34	63	90	129	193	249
18000	0.27	50.09	18	34	63	92	126	197	251
20000	0.28	51.56	18	34	62	91	127	197	257

Equal population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 0% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.19	34.22	35	49	74	100	120	163	188
4000	0.21	37.92	34	46	72	95	124	170	203
6000	0.22	38.60	32	45	71	95	122	173	209
8000	0.22	39.66	34	45	70	94	122	175	214
10000	0.21	38.65	35	46	70	95	122	170	210
12000	0.22	40.13	33	46	71	93	121	172	218
14000	0.22	39.11	32	44	71	95	125	171	205
16000	0.22	39.01	33	45	70	95	123	174	208
18000	0.22	39.02	32	45	71	94	121	173	211
20000	0.21	38.87	35	47	72	94	122	169	216

B.2 Uniformly-Distributed Initial Wealth

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.02	1	9	52	99	150	187	195
2000	0.48	90.62	0	2	26	75	153	274	383
4000	0.55	106.81	0	1	17	63	151	310	459
6000	0.57	116.22	0	0	15	57	143	339	518
8000	0.60	125.44	0	0	13	52	143	358	586
10000	0.62	132.18	0	0	11	48	136	379	605
12000	0.62	132.40	0	0	10	45	141	376	620
14000	0.62	130.03	0	0	11	48	139	370	616
16000	0.62	129.10	0	0	11	49	139	385	591
18000	0.62	131.84	0	0	11	49	137	352	643
20000	0.63	133.99	0	0	10	46	137	374	633

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 80% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	56.73	2	9	51	101	150	187	195
2000	0.52	99.82	0	0	16	73	158	294	412

4000	0.62	128.66	0	0	7	47	149	361	575
6000	0.67	144.44	0	0	3	33	148	414	634
8000	0.70	155.77	0	0	1	26	139	448	678
10000	0.73	167.29	0	0	1	21	126	454	765
12000	0.75	180.28	0	0	1	17	117	480	888
14000	0.76	187.87	0	0	0	15	112	496	885
16000	0.77	191.27	0	0	0	14	105	535	870
18000	0.78	196.59	0	0	0	12	100	539	912
20000	0.79	202.29	0	0	0	10	93	558	967

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.34	58.15	2	12	49	98	152	189	196
2000	0.51	96.07	0	1	22	76	156	279	426
4000	0.58	116.41	0	0	11	57	148	341	476
6000	0.62	127.60	0	0	9	49	145	361	559
8000	0.64	135.41	0	0	7	44	138	389	597
10000	0.66	145.35	0	0	6	40	135	393	643
12000	0.67	148.63	0	0	6	35	134	398	646
14000	0.67	151.94	0	0	6	35	129	394	719
16000	0.67	151.98	0	0	6	36	126	401	725
18000	0.68	152.78	0	0	5	33	127	421	730
20000	0.69	160.64	0	0	5	32	120	444	741

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 40% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.34	58.68	2	10	48	99	152	190	197
2000	0.47	86.49	0	2	30	82	146	277	371
4000	0.52	102.19	0	2	23	67	147	310	432
6000	0.56	115.73	0	1	18	59	144	336	546
8000	0.58	119.66	0	1	16	58	142	342	540
10000	0.58	121.64	0	1	16	58	137	343	575
12000	0.58	121.35	0	1	15	54	141	352	533
14000	0.59	124.79	0	0	16	55	138	344	572
16000	0.58	123.63	0	1	18	57	133	334	571
18000	0.58	122.38	0	0	16	58	139	345	562
20000	0.58	122.92	0	1	16	57	138	344	562

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 20% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.37	2	9	49	100	149	188	196
2000	0.44	80.05	0	5	37	83	145	256	350
4000	0.48	91.74	0	4	30	72	143	282	396
6000	0.50	97.99	0	4	28	69	143	293	421
8000	0.51	102.55	0	4	27	69	141	294	493
10000	0.51	100.57	0	3	27	70	140	296	469
12000	0.52	107.76	0	3	25	67	134	303	507
14000	0.52	107.29	0	2	25	66	134	317	505
16000	0.51	103.50	0	3	26	70	135	303	480
18000	0.52	105.48	0	3	25	67	137	311	464

20000	0.51	104.08	0	4	26	66	143	305	474

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 0% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.34	58.32	2	11	49	100	150	191	199
2000	0.41	77.71	2	9	40	83	143	242	327
4000	0.43	81.58	1	8	38	81	138	260	367
6000	0.44	83.50	0	7	38	80	137	266	387
8000	0.44	86.00	1	7	37	78	140	263	403
10000	0.45	85.91	1	8	35	78	143	267	384
12000	0.45	86.89	1	7	37	75	138	270	406
14000	0.44	84.70	1	7	37	80	138	265	384
16000	0.44	87.11	1	7	36	79	139	261	392
18000	0.45	86.68	2	8	37	77	136	269	389
20000	0.45	85.76	1	7	36	78	138	267	387

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 80% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.32	56.08	1	10	52	100	148	186	194
2000	0.44	79.74	0	4	32	83	154	250	322
4000	0.51	96.95	0	2	21	68	155	300	402
6000	0.56	109.29	0	1	16	59	151	325	458
8000	0.59	117.92	0	0	13	53	147	345	489
10000	0.61	125.40	0	0	10	47	143	363	541
12000	0.63	130.03	0	0	9	45	141	379	551
14000	0.64	136.79	0	0	8	43	139	381	648
16000	0.65	142.08	0	0	7	44	132	394	656
18000	0.66	146.22	0	0	6	39	133	397	695
20000	0.66	148.81	0	0	6	37	135	388	694

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.32	55.69	1	10	51	101	147	185	192
2000	0.40	72.22	1	7	38	88	147	231	282
4000	0.44	81.02	1	6	34	81	148	260	334
6000	0.47	88.88	0	6	31	73	146	276	371
8000	0.49	93.94	1	5	29	68	144	298	400
10000	0.50	97.77	1	4	26	69	140	309	420
12000	0.51	100.80	0	3	26	67	137	309	454
14000	0.53	104.65	0	3	22	64	142	320	461
16000	0.54	106.28	0	3	22	63	138	324	469
18000	0.54	107.62	0	2	22	64	140	334	463
20000	0.54	108.23	0	2	23	63	138	318	467

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2

the richer party has 50% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.73	1	10	50	100	151	188	198
2000	0.40	71.67	2	9	43	85	147	229	299
4000	0.42	78.49	2	9	38	81	143	246	344
6000	0.44	84.76	1	7	38	77	138	260	386
8000	0.45	86.92	1	7	36	76	137	268	399
10000	0.46	90.66	1	8	32	73	141	284	396
12000	0.47	92.66	1	7	32	73	140	282	405
14000	0.47	92.76	2	7	34	73	135	283	435
16000	0.47	94.31	1	7	33	71	138	281	463
18000	0.47	94.70	1	6	33	72	134	288	434
20000	0.47	95.29	1	6	32	73	134	287	437

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 40% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.34	58.91	1	9	48	97	151	193	201
2000	0.38	68.82	2	12	45	87	142	226	306
4000	0.40	73.88	3	11	45	83	138	248	337
6000	0.40	75.27	3	12	45	82	136	243	357
8000	0.41	77.05	2	12	43	81	137	250	359
10000	0.42	79.11	3	11	42	79	139	252	369
12000	0.42	81.18	3	11	41	77	138	254	376
14000	0.42	79.58	3	13	41	79	135	261	364
16000	0.42	78.83	3	13	41	79	136	256	356
18000	0.41	78.59	4	13	41	79	138	255	359
20000	0.43	81.37	3	11	40	76	136	265	370

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 20% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.16	2	9	50	99	150	187	195
2000	0.34	60.82	5	18	54	89	136	211	267
4000	0.34	61.85	6	20	55	87	131	220	295
6000	0.33	60.93	7	21	55	88	133	216	283
8000	0.33	61.42	8	22	54	89	131	218	287
10000	0.32	59.39	7	22	56	90	131	214	272
12000	0.33	60.02	10	24	56	88	131	211	287
14000	0.33	61.43	9	24	56	88	131	214	296
16000	0.33	62.07	9	22	56	88	131	218	285
18000	0.34	62.64	10	21	55	86	130	216	304
20000	0.33	61.61	11	24	53	89	131	221	300

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 0% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.83	2	9	48	100	150	188	197

2000	0.31	54.85	10	25	58	91	135	195	258
4000	0.29	52.65	11	30	61	92	129	193	259
6000	0.29	52.76	14	29	61	91	130	197	258
8000	0.29	52.30	14	31	61	91	128	200	257
10000	0.29	52.46	17	31	62	91	126	203	264
12000	0.28	51.27	18	31	63	90	128	193	250
14000	0.27	48.96	18	34	65	92	126	192	244
16000	0.27	49.79	18	33	62	92	127	193	243
18000	0.28	50.98	16	32	63	92	127	195	253
20000	0.27	49.11	16	34	65	91	128	191	255

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 80% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.34	58.48	2	9	49	98	150	192	200
2000	0.44	78.34	1	5	33	83	154	243	311
4000	0.50	94.95	0	3	25	71	150	291	389
6000	0.53	104.80	0	2	21	65	145	307	459
8000	0.56	111.64	0	1	17	61	146	322	503
10000	0.58	116.81	0	1	14	53	147	340	522
12000	0.60	124.67	0	1	13	50	143	354	571
14000	0.61	129.26	0	0	12	50	136	381	582
16000	0.62	132.67	0	0	11	48	137	378	614
18000	0.63	136.39	0	0	11	45	132	392	632
20000	0.64	140.59	0	0	10	43	131	383	642

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.83	1	9	48	99	150	189	197
2000	0.39	70.23	1	10	44	86	144	230	298
4000	0.41	77.49	2	9	41	82	139	242	354
6000	0.43	81.47	2	9	37	81	138	253	357
8000	0.45	85.15	2	9	36	77	138	277	379
10000	0.46	87.87	1	6	35	74	141	277	403
12000	0.46	89.76	2	7	34	72	141	283	412
14000	0.47	92.11	2	7	32	73	138	286	430
16000	0.47	91.22	1	7	31	72	143	289	411
18000	0.47	91.08	1	6	32	72	142	281	414
20000	0.48	93.56	0	6	30	73	140	283	429

Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 50% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.34	58.76	3	9	48	101	148	191	198
2000	0.38	67.90	3	12	46	89	142	222	292
4000	0.39	72.73	4	12	45	84	138	237	316
6000	0.40	74.89	5	14	44	82	136	246	331
8000	0.40	75.69	5	14	44	81	138	244	350
10000	0.40	75.87	5	13	43	81	137	245	343

12000	0.42	79.61	4	13	41	77	138	255	367
14000	0.42	80.22	4	13	39	77	140	253	379
16000	0.42	80.17	3	12	41	79	137	256	367
18000	0.41	78.83	4	12	42	80	134	256	355
20000	0.41	78.49	4	11	43	80	134	264	337

```
Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
                   with the loser resisting the loss of wealth at Lvl. 5
                   the richer party has 40% chance of winning
```

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.38	2	10	49	99	148	189	198
2000	0.34	61.42	4	16	52	90	137	211	272
4000	0.35	63.67	6	19	52	87	134	224	292
6000	0.35	66.09	7	19	51	86	132	230	307
8000	0.36	66.68	9	20	49	86	134	230	306
10000	0.36	65.91	9	19	50	84	133	225	307
12000	0.36	65.92	8	19	51	85	134	230	297
14000	0.35	65.01	7	19	53	86	132	227	303
16000	0.36	67.03	8	19	51	83	136	228	308
18000	0.36	67.35	8	18	50	84	135	228	307
20000	0.37	69.57	8	19	50	83	131	236	330

```
Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
                   with the loser resisting the loss of wealth at Lvl. 5
                   the richer party has 20% chance of winning
```

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.33	57.10	1	11	50	100	149	186	193
2000	0.30	53.94	6	24	60	92	134	193	252
4000	0.29	52.91	13	30	61	89	131	197	258
6000	0.28	52.09	19	33	61	91	127	197	262
8000	0.28	51.30	18	33	62	91	128	194	256
10000	0.28	50.05	19	34	63	91	128	190	248
12000	0.27	49.82	19	34	63	91	128	196	242
14000	0.27	49.86	18	33	64	92	128	193	242
16000	0.27	49.65	18	33	63	91	128	190	249
18000	0.28	50.44	17	33	63	91	126	195	252
20000	0.27	50.62	18	33	63	91	127	193	261

```
Uniform population, size=2000, mean=100.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
                   with the loser resisting the loss of wealth at Lvl. 5
                   the richer party has 0% chance of winning
```

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.34	59.37	1	9	47	100	152	191	201
2000	0.27	47.75	11	31	64	94	129	184	228
4000	0.24	44.07	26	41	67	93	123	183	229
6000	0.23	40.77	28	43	70	95	123	172	212
8000	0.23	40.68	31	43	70	94	124	173	218
10000	0.22	40.15	33	44	71	95	122	170	221
12000	0.22	39.80	31	44	71	96	122	173	212
14000	0.22	38.70	34	45	71	94	124	170	208
16000	0.22	38.70	30	45	70	95	123	170	207
18000	0.22	39.43	33	45	71	94	123	172	212
20000	0.22	40.09	31	45	70	94	123	175	213

```
+-------+------+-------+----+----+-----+-----+-----+-----+-----+
```

B.3 Normally-Distributed Initial Wealth

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	20.22	53	68	86	100	114	132	145
2000	0.42	78.09	0	3	40	91	138	238	356
4000	0.52	102.51	0	1	22	70	144	313	466
6000	0.56	112.76	0	1	17	62	147	325	500
8000	0.59	121.55	0	0	13	55	140	339	537
10000	0.61	126.64	0	0	11	50	141	358	538
12000	0.61	131.69	0	0	11	48	136	364	586
14000	0.63	136.24	0	0	9	46	140	384	625
16000	0.64	140.84	0	0	10	46	129	392	616
18000	0.63	137.27	0	0	10	48	132	385	606
20000	0.64	142.29	0	0	10	43	128	388	664

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 80% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	19.12	56	69	88	101	114	132	145
2000	0.46	87.25	0	1	30	89	142	266	371
4000	0.58	118.85	0	0	11	59	152	334	526
6000	0.65	138.96	0	0	4	43	143	403	619
8000	0.70	158.74	0	0	2	28	135	439	667
10000	0.73	171.98	0	0	1	24	124	478	778
12000	0.75	183.61	0	0	1	18	119	488	913
14000	0.76	192.41	0	0	0	17	110	498	881
16000	0.77	201.99	0	0	0	15	109	497	964
18000	0.78	204.40	0	0	0	11	104	519	950
20000	0.79	210.70	0	0	0	9	99	529	969

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.12	20.44	52	65	86	99	114	132	147
2000	0.43	79.06	0	3	37	90	137	255	358
4000	0.53	102.15	0	0	19	71	147	310	439
6000	0.58	119.19	0	0	13	56	145	339	520
8000	0.62	128.63	0	0	10	47	143	362	551
10000	0.64	138.40	0	0	9	43	138	381	608
12000	0.65	143.91	0	0	7	41	140	390	669
14000	0.66	144.76	0	0	6	38	136	394	690
16000	0.66	146.80	0	0	6	41	131	395	687
18000	0.67	148.08	0	0	6	37	134	411	670
20000	0.68	151.06	0	0	5	36	130	410	707

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
however, the richer party has 40% chance of winning

```
+-------+------+-------+----+----+-----+-----+-----+-----+-----+
```

```
|  step | gini |  std   | 1% | 5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+--------+----+----+-----+-----+-----+-----+-----+
|   0   | 0.11 | 20.26  | 54 | 66 |  86 |  99 | 113 | 132 | 149 |
|  2000 | 0.40 | 74.70  | 0  | 5  |  43 |  90 | 134 | 239 | 330 |
|  4000 | 0.49 | 93.45  | 0  | 2  |  27 |  75 | 146 | 291 | 399 |
|  6000 | 0.53 | 104.09 | 0  | 1  |  21 |  66 | 145 | 319 | 443 |
|  8000 | 0.56 | 111.54 | 0  | 1  |  17 |  59 | 145 | 330 | 485 |
| 10000 | 0.56 | 113.16 | 0  | 1  |  17 |  60 | 142 | 331 | 524 |
| 12000 | 0.57 | 117.09 | 0  | 1  |  17 |  56 | 140 | 330 | 527 |
| 14000 | 0.58 | 119.71 | 0  | 1  |  18 |  54 | 140 | 336 | 536 |
| 16000 | 0.58 | 124.52 | 0  | 1  |  17 |  56 | 136 | 353 | 587 |
| 18000 | 0.59 | 127.01 | 0  | 1  |  15 |  53 | 134 | 362 | 611 |
| 20000 | 0.59 | 126.67 | 0  | 1  |  16 |  56 | 135 | 344 | 613 |
+-------+------+--------+----+----+-----+-----+-----+-----+-----+

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
                   however, the richer party has 20% chance of winning
+-------+------+--------+----+----+-----+-----+-----+-----+-----+
|  step | gini |  std   | 1% | 5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+--------+----+----+-----+-----+-----+-----+-----+
|   0   | 0.11 | 19.89  | 55 | 67 |  86 | 100 | 113 | 133 | 146 |
|  2000 | 0.38 | 70.45  | 1  | 7  |  47 |  91 | 133 | 235 | 319 |
|  4000 | 0.45 | 85.91  | 0  | 5  |  34 |  80 | 139 | 264 | 380 |
|  6000 | 0.48 | 91.68  | 0  | 5  |  29 |  74 | 140 | 287 | 409 |
|  8000 | 0.49 | 94.99  | 0  | 4  |  28 |  73 | 143 | 290 | 428 |
| 10000 | 0.51 | 100.55 | 0  | 3  |  27 |  69 | 142 | 294 | 447 |
| 12000 | 0.51 | 102.41 | 0  | 4  |  28 |  69 | 139 | 300 | 465 |
| 14000 | 0.52 | 105.04 | 0  | 4  |  26 |  67 | 139 | 310 | 482 |
| 16000 | 0.52 | 106.26 | 0  | 4  |  25 |  66 | 135 | 318 | 507 |
| 18000 | 0.51 | 105.20 | 0  | 4  |  26 |  67 | 135 | 304 | 494 |
| 20000 | 0.52 | 103.71 | 0  | 3  |  24 |  66 | 141 | 322 | 474 |
+-------+------+--------+----+----+-----+-----+-----+-----+-----+

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes random proportion of wealth from the loser
                   however, the richer party has 0% chance of winning
+-------+------+-------+----+----+-----+-----+-----+-----+-----+
|  step | gini |  std  | 1% | 5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+-------+----+----+-----+-----+-----+-----+-----+
|   0   | 0.11 | 19.74 | 55 | 69 |  86 | 100 | 113 | 133 | 145 |
|  2000 | 0.35 | 64.38 | 3  | 12 |  53 |  92 | 133 | 219 | 295 |
|  4000 | 0.42 | 77.46 | 2  | 8  |  41 |  84 | 138 | 252 | 347 |
|  6000 | 0.43 | 82.70 | 1  | 8  |  39 |  82 | 141 | 254 | 362 |
|  8000 | 0.44 | 86.12 | 1  | 7  |  36 |  78 | 144 | 257 | 381 |
| 10000 | 0.45 | 86.56 | 1  | 8  |  37 |  77 | 138 | 265 | 392 |
| 12000 | 0.45 | 85.50 | 1  | 7  |  35 |  77 | 138 | 269 | 375 |
| 14000 | 0.44 | 85.01 | 1  | 8  |  37 |  77 | 140 | 266 | 388 |
| 16000 | 0.45 | 86.89 | 1  | 8  |  36 |  78 | 135 | 277 | 414 |
| 18000 | 0.45 | 86.06 | 1  | 7  |  37 |  78 | 138 | 269 | 395 |
| 20000 | 0.45 | 86.68 | 2  | 7  |  35 |  78 | 136 | 277 | 407 |
+-------+------+-------+----+----+-----+-----+-----+-----+-----+

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
                   with the loser resisting the loss of wealth at Lvl. 2
                   the richer party has 80% chance of winning
+-------+------+--------+----+----+-----+-----+-----+-----+-----+
|  step | gini |  std   | 1% | 5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+--------+----+----+-----+-----+-----+-----+-----+
|   0   | 0.11 | 20.21  | 53 | 67 |  86 |  99 | 113 | 133 | 147 |
|  2000 | 0.35 | 64.88  | 5  | 16 |  51 |  91 | 133 | 221 | 307 |
|  4000 | 0.45 | 84.26  | 1  | 7  |  34 |  78 | 142 | 259 | 381 |
|  6000 | 0.51 | 98.80  | 0  | 3  |  24 |  67 | 148 | 298 | 423 |
|  8000 | 0.56 | 110.06 | 0  | 1  |  17 |  59 | 149 | 314 | 479 |
```

10000	0.59	120.04	0	1	14	53	141	335	536
12000	0.62	128.11	0	0	11	46	139	359	590
14000	0.64	134.94	0	0	9	41	140	389	617
16000	0.65	140.36	0	0	8	40	138	398	658
18000	0.66	145.35	0	0	6	36	129	406	707
20000	0.67	148.88	0	0	6	37	130	411	718

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	20.12	53	65	86	99	113	133	146
2000	0.32	57.30	7	20	57	94	132	203	260
4000	0.40	72.79	3	11	42	85	140	239	325
6000	0.44	81.13	2	7	36	79	143	258	340
8000	0.47	88.43	1	6	32	73	144	276	379
10000	0.48	92.42	0	5	30	72	140	297	393
12000	0.50	95.92	1	4	27	70	141	299	422
14000	0.50	97.75	1	4	27	68	137	300	434
16000	0.51	100.36	0	4	25	70	138	316	438
18000	0.52	103.27	0	3	25	68	137	308	464
20000	0.52	104.11	0	3	24	65	142	304	462

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 50% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	20.27	54	66	85	99	113	133	146
2000	0.29	53.04	10	23	60	94	129	195	255
4000	0.36	66.03	4	17	51	87	136	218	315
6000	0.41	75.89	3	10	42	80	138	249	339
8000	0.43	80.55	2	8	40	79	139	258	364
10000	0.44	83.49	1	8	37	75	141	264	381
12000	0.45	86.56	1	8	34	75	139	274	371
14000	0.46	88.63	2	8	33	74	140	280	386
16000	0.46	89.33	1	7	34	74	139	279	414
18000	0.46	90.39	1	6	33	74	140	275	398
20000	0.46	90.30	1	7	34	76	133	285	412

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 40% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.12	20.42	52	65	86	99	113	133	149
2000	0.28	51.41	10	25	63	95	128	193	246
4000	0.34	63.14	7	18	52	90	133	216	291
6000	0.38	71.39	5	16	47	86	136	231	342
8000	0.40	75.42	3	13	44	81	137	242	355
10000	0.41	77.91	4	12	43	80	137	247	366
12000	0.42	79.93	4	12	40	78	138	263	341
14000	0.43	82.32	3	10	39	77	140	259	378
16000	0.43	80.83	3	11	41	76	139	258	367
18000	0.42	80.13	3	10	41	79	134	262	373

20000	0.42	82.38	3	12	41	79	132	265	399

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 20% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	20.22	52	66	86	99	113	132	146
2000	0.26	46.45	18	31	67	96	125	184	235
4000	0.30	54.12	11	27	60	92	129	202	251
6000	0.33	59.88	10	23	54	89	133	209	278
8000	0.33	61.40	10	23	55	88	132	218	297
10000	0.33	60.39	10	23	54	88	132	213	291
12000	0.33	61.03	10	23	54	87	133	217	284
14000	0.34	62.35	8	23	54	86	133	216	302
16000	0.34	62.91	9	22	54	86	132	220	296
18000	0.34	62.94	9	21	52	88	133	221	297
20000	0.34	64.07	9	20	52	88	132	220	317

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 2
the richer party has 0% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	19.94	53	67	86	99	113	132	146
2000	0.23	41.21	21	40	71	95	122	175	220
4000	0.26	46.41	18	34	66	94	125	187	230
6000	0.27	48.34	16	33	64	93	126	191	240
8000	0.27	49.23	18	33	64	91	126	194	242
10000	0.27	49.67	16	33	63	92	127	189	257
12000	0.27	48.85	17	33	63	92	126	191	246
14000	0.27	49.40	18	32	64	92	129	190	244
16000	0.27	49.98	16	32	63	91	128	193	253
18000	0.27	50.01	15	32	63	94	128	193	253
20000	0.28	50.95	18	32	61	93	127	196	246

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 80% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	19.74	55	68	86	100	114	131	146
2000	0.33	59.36	12	21	53	92	134	209	272
4000	0.43	79.73	3	11	37	80	143	256	348
6000	0.49	94.23	1	6	27	69	144	292	398
8000	0.54	106.79	0	3	21	62	144	319	468
10000	0.57	114.94	0	2	17	56	143	341	499
12000	0.59	120.87	0	1	14	51	140	357	537
14000	0.61	126.82	0	1	12	50	139	372	588
16000	0.61	131.03	0	0	12	48	139	371	604
18000	0.62	133.96	0	0	12	45	139	374	629
20000	0.64	138.94	0	0	10	43	134	391	634

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser

with the loser resisting the loss of wealth at Lvl. 5
the richer party has 60% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	20.07	54	66	86	100	114	132	143
2000	0.29	51.72	14	28	62	94	130	193	243
4000	0.37	68.28	7	18	48	85	135	229	320
6000	0.41	77.44	4	12	42	80	136	256	358
8000	0.43	82.42	2	10	39	78	136	266	378
10000	0.46	87.46	2	8	35	76	137	280	400
12000	0.46	88.06	1	7	33	74	139	277	391
14000	0.47	91.03	2	7	34	73	138	275	423
16000	0.47	91.87	1	7	32	71	141	285	412
18000	0.48	91.96	1	5	32	73	139	278	418
20000	0.48	93.86	1	7	30	73	137	286	437

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 50% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	20.16	56	67	86	101	114	133	146
2000	0.27	49.53	15	30	64	95	129	192	236
4000	0.34	61.91	11	22	53	89	134	216	294
6000	0.37	67.70	6	17	49	86	138	234	303
8000	0.38	71.30	4	14	47	84	138	232	324
10000	0.39	73.99	4	13	46	83	137	249	333
12000	0.40	76.03	4	12	45	82	137	247	341
14000	0.41	79.08	5	13	44	81	137	260	369
16000	0.40	77.96	5	15	43	81	138	259	369
18000	0.41	78.87	4	14	41	80	140	258	374
20000	0.41	77.99	4	14	43	81	136	258	359

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 40% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	19.96	53	64	85	99	112	132	145
2000	0.26	45.61	20	34	66	94	126	182	229
4000	0.30	54.92	13	25	57	91	130	201	265
6000	0.34	61.87	11	22	53	87	129	221	284
8000	0.34	63.16	10	22	52	85	132	223	294
10000	0.36	66.49	9	20	50	86	131	221	318
12000	0.36	66.85	9	19	50	86	132	225	318
14000	0.36	67.75	8	20	49	85	132	229	309
16000	0.36	68.00	7	18	50	84	131	233	314
18000	0.36	67.65	9	20	50	84	132	235	314
20000	0.36	67.11	6	19	50	84	132	229	320

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 20% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%

0	0.11	20.43	53	67	86	100	114	134	147
2000	0.23	40.95	26	43	70	96	124	173	219
4000	0.25	46.32	23	36	66	93	127	185	230
6000	0.27	49.86	20	33	64	92	127	195	252
8000	0.27	50.31	20	32	63	92	130	191	250
10000	0.27	49.61	22	35	65	91	127	194	251
12000	0.27	49.87	22	34	62	91	129	192	245
14000	0.27	50.27	19	32	64	93	127	194	252
16000	0.28	51.19	19	33	63	91	130	197	251
18000	0.27	50.74	19	34	63	93	128	189	254
20000	0.28	52.51	18	33	62	90	126	200	268

Normal population, size=2000, mean=100.0, std=20.0, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 0% chance of winning

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.11	19.59	55	67	87	101	112	132	145
2000	0.20	35.48	35	50	74	96	120	164	201
4000	0.21	38.32	34	47	72	95	121	171	208
6000	0.21	38.48	35	46	71	95	124	170	206
8000	0.21	38.70	32	47	71	95	123	171	205
10000	0.22	39.97	33	45	71	94	124	175	211
12000	0.22	39.67	33	45	70	94	124	173	211
14000	0.22	39.52	32	45	72	94	122	174	211
16000	0.22	40.16	33	45	71	94	123	176	217
18000	0.22	39.41	33	47	71	93	122	174	216
20000	0.22	40.60	35	46	70	93	123	177	217

B.4 Distribution Fitting

Fitting with transaction function win_take_layer, simulating 200000 steps
Testing on equal population of size=2000, mean=100.00, transaction bias=60%, layers=1

distr	MSE	KS-stat	KS-pval
fatiguelife	5.38E-06	8.62E-02	2.30E-13
johnsonsb	5.65E-06	6.41E-02	1.40E-07
johnsonsu	7.86E-06	6.91E-02	9.44E-09
lognorm	7.87E-06	6.93E-02	8.67E-09
powerlognorm	9.49E-06	5.02E-02	8.00E-05
fisk	9.84E-06	6.25E-02	3.03E-07
mielke	1.06E-05	8.42E-02	8.82E-13
halfgennorm	1.08E-05	4.05E-02	2.75E-03
gamma	1.30E-05	1.11E-01	7.70E-22
burr12	1.43E-05	6.01E-02	1.03E-06

distr	MSE	KS-stat	KS-pval
levy	1.59E-05	1.21E-01	8.09E-26
exponweib	1.59E-05	2.77E-02	9.05E-02
invgauss	1.70E-05	8.21E-02	3.60E-12
norminvgauss	1.71E-05	8.21E-02	3.61E-12
invweibull	1.74E-05	7.78E-02	5.52E-11
genextreme	1.74E-05	7.78E-02	5.52E-11
kappa3	1.83E-05	5.87E-02	1.97E-06
betaprime	2.06E-05	7.16E-02	2.34E-09
weibull_min	2.15E-05	2.67E-02	1.13E-01
invgamma	2.54E-05	7.73E-02	7.61E-11

Fitting with transaction function win_take_layer, simulating 200000 steps
Testing on equal population of size=2000, mean=100.00, transaction bias=50%, layers=1

distr	MSE	KS-stat	KS-pval
geninvgauss	5.67E-06	4.25E-02	1.43E-03
betaprime	6.30E-06	4.84E-02	1.66E-04
chi2	7.58E-06	2.16E-02	3.06E-01
gamma	8.13E-06	1.95E-02	4.28E-01
genhyperbolic	8.48E-06	1.00E+00	0.00E+00
recipinvgauss	8.60E-06	4.63E-02	3.65E-04
erlang	8.96E-06	1.32E-02	8.74E-01

distr	MSE	KS-stat	KS-pval
gengamma	1.14E-05	1.96E-02	4.24E-01
weibull_min	1.22E-05	2.15E-02	3.11E-01
halfgennorm	1.24E-05	3.34E-02	2.28E-02
pearson3	1.27E-05	1.88E-02	4.71E-01
burr	1.31E-05	3.27E-02	2.70E-02
exponweib	1.32E-05	2.29E-02	2.42E-01
exponpow	1.37E-05	2.98E-02	5.57E-02

beta	9.78E-06	1.32E-02	8.71E-01	johnsonsb	1.38E-05	4.58E-02	4.31E-04
mielke	1.11E-05	4.64E-02	3.46E-04	fatiguelife	1.80E-05	5.96E-02	1.31E-06
burr12	1.12E-05	6.20E-02	3.90E-07	nakagami	2.42E-05	4.88E-02	1.39E-04

Fitting with transaction function win_take_layer, simulating 200000 steps
Testing on equal population of size=2000, mean=100.00, transaction bias=60%, layers=5

distr	MSE	KS-stat	KS-pval	distr	MSE	KS-stat	KS-pval
geninvgauss	5.51E-06	1.16E-02	9.47E-01	genhyperbolic	6.28E-06	1.00E+00	0.00E+00
gamma	5.51E-06	1.16E-02	9.47E-01	genexpon	6.56E-06	1.71E-02	5.99E-01
pearson3	5.51E-06	1.16E-02	9.47E-01	genpareto	9.95E-06	2.78E-02	8.88E-02
chi2	5.51E-06	1.16E-02	9.47E-01	gompertz	1.01E-05	2.81E-02	8.26E-02
exponweib	5.51E-06	1.14E-02	9.55E-01	halflogistic	1.04E-05	4.03E-02	2.90E-03
burr12	5.55E-06	1.29E-02	8.88E-01	recipinvgauss	1.04E-05	3.14E-02	3.79E-02
betaprime	5.56E-06	1.41E-02	8.18E-01	kappa3	1.08E-05	3.00E-02	5.30E-02
f	5.62E-06	1.52E-02	7.39E-01	burr	1.12E-05	3.02E-02	5.13E-02
ncf	5.94E-06	1.31E-02	8.80E-01	genhalflogistic	1.19E-05	4.71E-02	2.66E-04
beta	6.24E-06	2.33E-02	2.24E-01	laplace_asymmetric	1.31E-05	3.89E-02	4.55E-03

Fitting with transaction function win_take_layer, simulating 200000 steps
Testing on equal population of size=2000, mean=100.00, transaction bias=50%, layers=5

distr	MSE	KS-stat	KS-pval	distr	MSE	KS-stat	KS-pval
ncf	1.63E-05	1.25E-02	9.08E-01	exponweib	1.93E-05	2.92E-02	6.41E-02
gengamma	1.64E-05	1.41E-02	8.19E-01	mielke	2.08E-05	1.80E-02	5.31E-01
beta	1.67E-05	1.47E-02	7.71E-01	burr	2.08E-05	1.81E-02	5.23E-01
burr12	1.73E-05	1.71E-02	5.95E-01	johnsonsb	2.23E-05	2.46E-02	1.74E-01
genhyperbolic	1.76E-05	1.00E+00	0.00E+00	fatiguelife	2.52E-05	2.50E-02	1.61E-01
pearson3	1.76E-05	1.88E-02	4.73E-01	norminvgauss	2.64E-05	2.62E-02	1.27E-01
gamma	1.76E-05	1.88E-02	4.73E-01	kappa3	2.75E-05	2.49E-02	1.64E-01
geninvgauss	1.76E-05	1.88E-02	4.77E-01	johnsonsu	2.86E-05	2.86E-02	7.38E-02
betaprime	1.76E-05	1.87E-02	4.83E-01	nakagami	2.86E-05	4.94E-02	1.12E-04
f	1.77E-05	1.92E-02	4.46E-01	chi	2.87E-05	4.93E-02	1.18E-04

B.5 Taxed Transactions

Equal population, size=2000, mean=100.00, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
with the loser resisting the loss of wealth at Lvl. 5
the richer party has 60% chance of winning
there is a 3% tax for each exchange, later distributed among all

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
2000	0.27	48.23	15	30	64	100	130	181	241
4000	0.35	63.57	9	19	50	90	134	219	290
6000	0.39	72.17	6	15	44	82	140	243	315
8000	0.41	77.31	4	12	41	80	138	257	341
10000	0.44	83.79	4	9	37	77	141	260	363
12000	0.44	84.06	3	9	38	76	140	268	371
14000	0.45	84.82	3	9	34	77	138	269	374
16000	0.45	87.04	2	8	35	76	140	274	397
18000	0.46	88.42	2	9	35	73	139	275	405
20000	0.47	91.21	3	7	33	74	139	280	435

Equal population, size=2000, mean=100.00, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser

```
                   with the loser resisting the loss of wealth at Lvl. 5
                   the richer party has 60% chance of winning
                   there is a 10% tax for each exchange, later distributed among all
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+
|  step | gini |  std  |  1% |  5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+
|   0   | 0.00 |  0.00 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
|  2000 | 0.26 | 46.05 |  17 |  31 |  66 | 103 | 126 | 179 | 239 |
|  4000 | 0.34 | 61.59 |  11 |  21 |  52 |  90 | 133 | 216 | 290 |
|  6000 | 0.37 | 68.69 |  8  |  17 |  46 |  85 | 137 | 233 | 304 |
|  8000 | 0.40 | 75.11 |  7  |  14 |  42 |  81 | 139 | 252 | 327 |
| 10000 | 0.42 | 79.68 |  6  |  14 |  41 |  76 | 137 | 264 | 355 |
| 12000 | 0.44 | 84.55 |  5  |  12 |  37 |  74 | 135 | 271 | 404 |
| 14000 | 0.44 | 87.26 |  6  |  12 |  37 |  73 | 136 | 273 | 410 |
| 16000 | 0.44 | 87.39 |  5  |  12 |  38 |  72 | 138 | 271 | 407 |
| 18000 | 0.45 | 88.05 |  5  |  11 |  36 |  73 | 138 | 274 | 411 |
| 20000 | 0.45 | 88.21 |  5  |  11 |  34 |  75 | 137 | 279 | 410 |
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+

Equal population, size=2000, mean=100.00, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
                   with the loser resisting the loss of wealth at Lvl. 5
                   the richer party has 60% chance of winning
                   there is a 20% tax for each exchange, later distributed among all
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+
|  step | gini |  std  |  1% |  5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+
|   0   | 0.00 |  0.00 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
|  2000 | 0.24 | 43.41 |  19 |  34 |  67 | 103 | 129 | 175 | 213 |
|  4000 | 0.32 | 57.90 |  14 |  25 |  55 |  89 | 134 | 210 | 262 |
|  6000 | 0.36 | 65.81 |  9  |  19 |  49 |  84 | 138 | 221 | 301 |
|  8000 | 0.37 | 70.14 |  8  |  18 |  47 |  84 | 135 | 240 | 333 |
| 10000 | 0.40 | 75.03 |  8  |  15 |  43 |  81 | 136 | 247 | 356 |
| 12000 | 0.41 | 77.85 |  7  |  16 |  41 |  78 | 137 | 258 | 357 |
| 14000 | 0.42 | 79.98 |  8  |  15 |  41 |  78 | 134 | 257 | 391 |
| 16000 | 0.42 | 82.76 |  9  |  15 |  40 |  76 | 136 | 260 | 413 |
| 18000 | 0.42 | 83.74 |  8  |  15 |  40 |  77 | 133 | 262 | 427 |
| 20000 | 0.42 | 83.28 |  8  |  14 |  40 |  76 | 133 | 264 | 374 |
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+

Equal population, size=2000, mean=100.00, simulating 20000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
                   with the loser resisting the loss of wealth at Lvl. 5
                   the richer party has 60% chance of winning
                   there is a 45% tax for each exchange, later distributed among all
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+
|  step | gini |  std  |  1% |  5% | 25% | 50% | 75% | 95% | 99% |
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+
|   0   | 0.00 |  0.00 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
|  2000 | 0.22 | 38.77 |  25 |  37 |  70 | 105 | 127 | 163 | 186 |
|  4000 | 0.29 | 50.73 |  19 |  29 |  59 |  93 | 134 | 188 | 234 |
|  6000 | 0.31 | 56.12 |  15 |  26 |  56 |  89 | 136 | 202 | 258 |
|  8000 | 0.33 | 61.04 |  17 |  27 |  53 |  85 | 131 | 222 | 285 |
| 10000 | 0.34 | 64.20 |  15 |  25 |  52 |  85 | 132 | 232 | 317 |
| 12000 | 0.35 | 65.78 |  16 |  24 |  50 |  84 | 133 | 228 | 318 |
| 14000 | 0.35 | 65.78 |  14 |  23 |  51 |  84 | 131 | 229 | 309 |
| 16000 | 0.36 | 68.25 |  14 |  23 |  49 |  82 | 131 | 236 | 320 |
| 18000 | 0.37 | 69.16 |  12 |  21 |  49 |  83 | 132 | 233 | 328 |
| 20000 | 0.36 | 69.61 |  13 |  24 |  49 |  81 | 130 | 240 | 330 |
+-------+------+-------+-----+-----+-----+-----+-----+-----+-----+

Equal population, size=2000, mean=100.00, simulating 200000 steps
Exchange strategy: winner takes some proportion of wealth from the loser
                   with the loser resisting the loss of wealth at Lvl. 5
                   the richer party has 60% chance of winning
```

```
Tax policy: part of the exchange is taken and equally distributed among all
            below 0.15 times initial mean    |  3%
            0.15 to 0.50 times initial mean  |  10%
            0.50 to 1.04 times initial mean  |  20%
            1.04 to 1.96 times initial mean  |  25%
            1.96 to 2.29 times initial mean  |  30%
            2.29 to 3.33 times initial mean  |  35%
            above 3.33 times initial mean    |  45%
```

step	gini	std	1%	5%	25%	50%	75%	95%	99%
0	0.00	0.00	100	100	100	100	100	100	100
20000	0.45	85.33	5	11	35	77	136	271	390
40000	0.44	84.68	5	11	36	76	138	268	396
60000	0.44	85.21	5	12	37	75	137	260	400
80000	0.44	85.50	5	12	37	74	136	271	401
100000	0.44	85.67	5	11	37	77	134	276	387
120000	0.44	85.41	5	12	39	76	133	280	407
140000	0.45	87.31	4	10	36	74	136	269	414
160000	0.45	86.74	5	11	35	74	139	275	407
180000	0.44	86.12	5	11	37	74	137	270	399
200000	0.45	87.59	4	10	35	75	136	270	404

Distribution fitting:

distr	MSE	KS-stat	KS-pval
betaprime	2.04E-06	1.01E-02	9.85E-01
f	2.04E-06	1.01E-02	9.85E-01
erlang	2.06E-06	1.16E-02	9.48E-01
pearson3	2.06E-06	1.16E-02	9.48E-01
gamma	2.06E-06	1.16E-02	9.48E-01
chi2	2.06E-06	1.16E-02	9.48E-01
exponweib	2.10E-06	1.21E-02	9.30E-01
geninvgauss	2.28E-06	1.24E-02	9.16E-01
ncf	2.78E-06	1.51E-02	7.45E-01
genhyperbolic	3.17E-06	1.00E+00	0.00E+00

distr	MSE	KS-stat	KS-pval
beta	3.87E-06	3.13E-02	3.85E-02
genexpon	4.79E-06	2.38E-02	2.04E-01
burr12	5.85E-06	2.19E-02	2.87E-01
mielke	6.41E-06	2.58E-02	1.38E-01
recipinvgauss	6.96E-06	2.98E-02	5.58E-02
johnsonsb	7.64E-06	2.94E-02	6.24E-02
kappa3	8.27E-06	2.17E-02	3.01E-01
fatiguelife	8.44E-06	3.28E-02	2.66E-02
halflogistic	8.52E-06	2.17E-02	3.00E-01
burr	8.96E-06	3.91E-02	4.34E-03