Monte Carlo Simulation of Complex Systems

IIT MADRAS
Indian Institute of Technology Madras

Mathematics Club

Indian Institute of Technology

Simulation Methods

We carried out the Monte Carlo method by simulating random variables sampled from a distribution and evaluating the necessary quantities. We did our simulation in *Python*, *C*, and *Mathematica* languages.

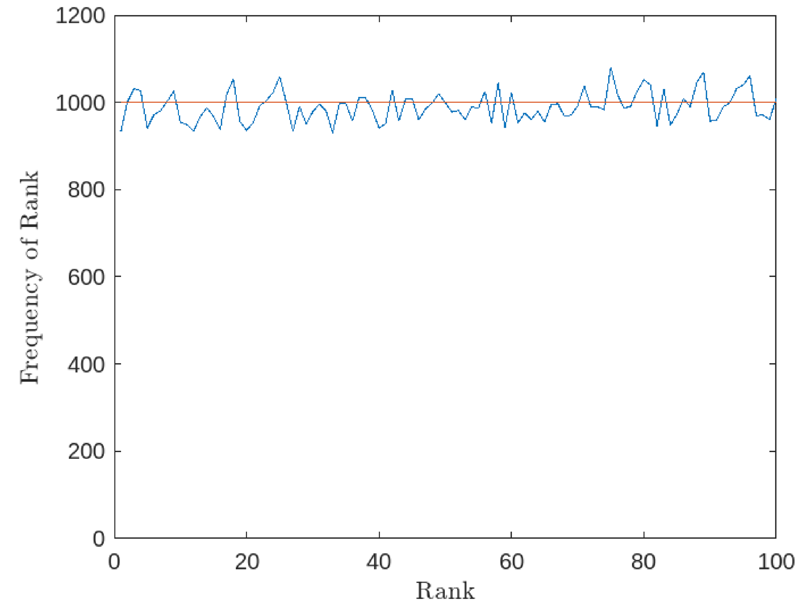
In our simulation, we used the steps outlined below:

- We used an outer *for* loop to keep track of the number of samples we take.
- We used an inner *for* loop for the required n value.
- We calculated the simulated executed value of the desired quantity at the end of the *for* loops.

We initially compared the simulated values with theoretical values, to ensure proper functioning of our simulations. Then, we extrapolated our simulated values to new values which we cannot calculate theoretically and we were able to obtain good results.

Expectation in Random Ranking

This question deals with two actions, the first choosing 100 random values from zero to one, and ordering them. The second, choosing a random variable from zero to one, and finding the rank it occupies amongst the particles obtained from the first action. The question deals with finding the probability that the particle occupies a rank k. In order to estimate this, we have used a Monte Carlo simulation modelling our experiment and we have verified our hypothesis that all ranks are equally likely.

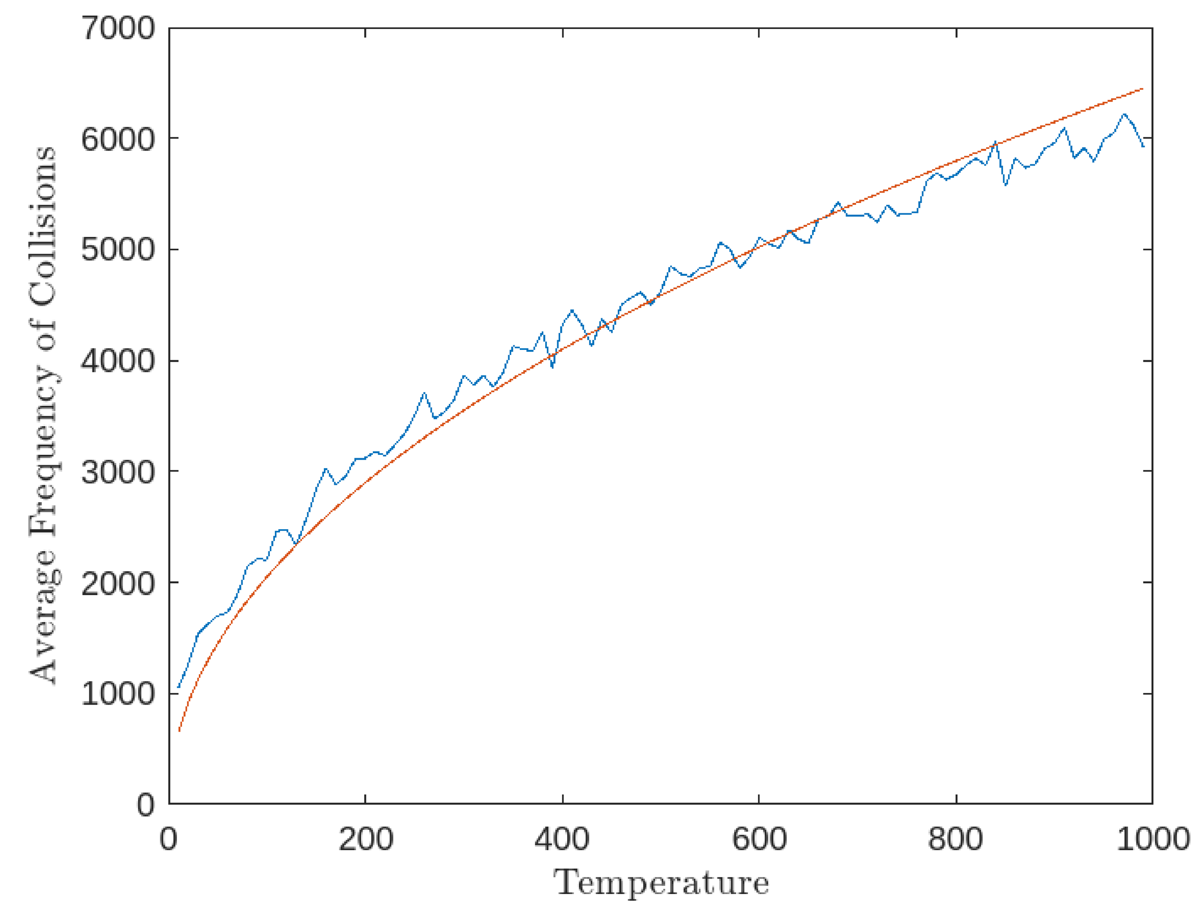


Arrhenius Factor vs Temperature

Arrhenius Equation

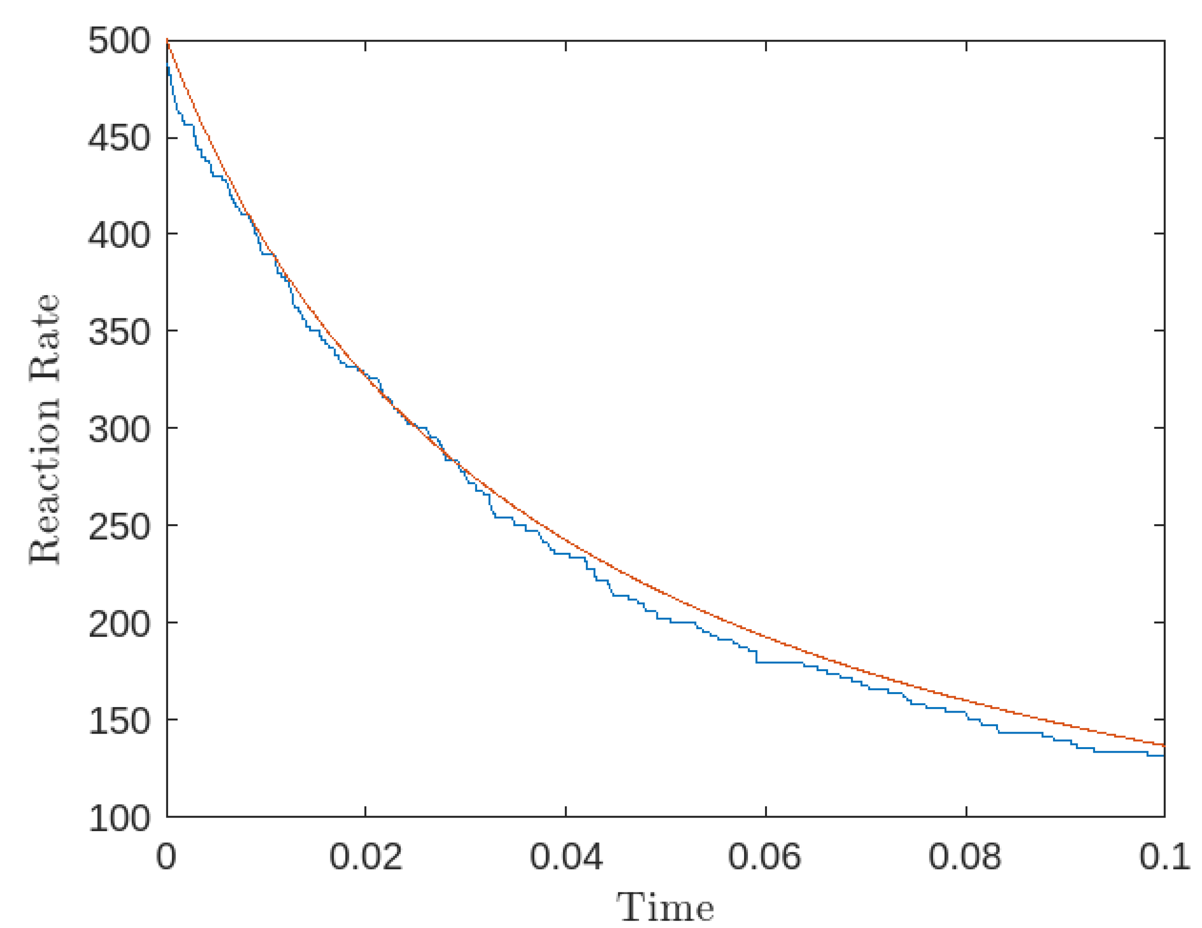
The Arrhenius rate equation relates the rate constant k to the temperature by $k = A\exp\left(\frac{-E_a}{RT}\right)$.

The pre-exponential factor A is not constant but proportional to the square root of temperature, which we obtained through the simulation.



Arrhenius Factor vs Temperature

We approached the problem by simulating the behaviour of n particles as they move with random velocities adhering to a Gaussian distribution (an approximation of the Maxwell-Boltzmann distribution).

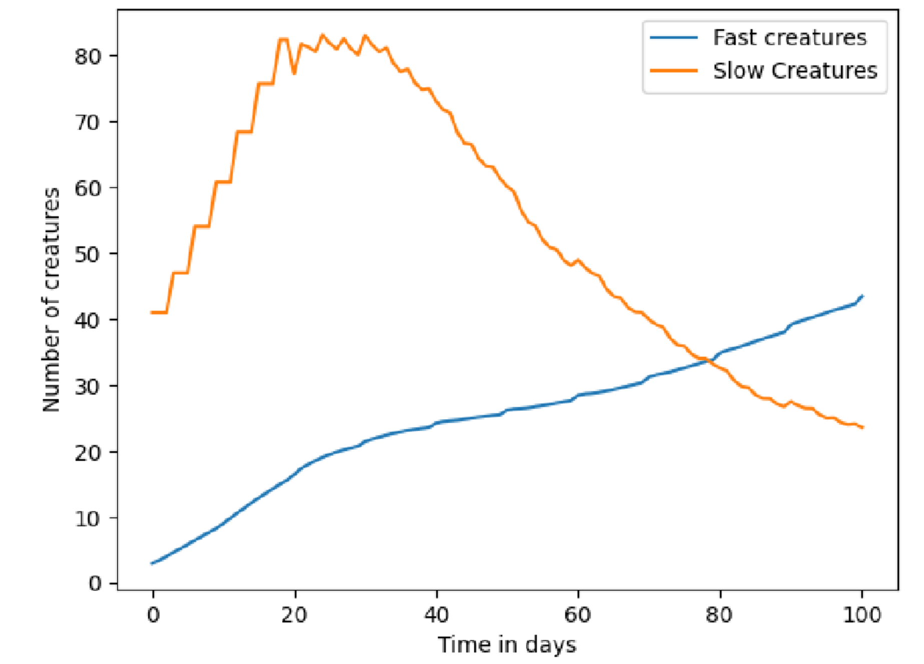


Second Order Resemblance

Natural Selection

Natural selection is the differential survival and reproduction of individuals due to differences in phenotype during evolution.

Considering speed as the phenotype, we modelled the particles' food habits, reproduction, and death (due to starvation).

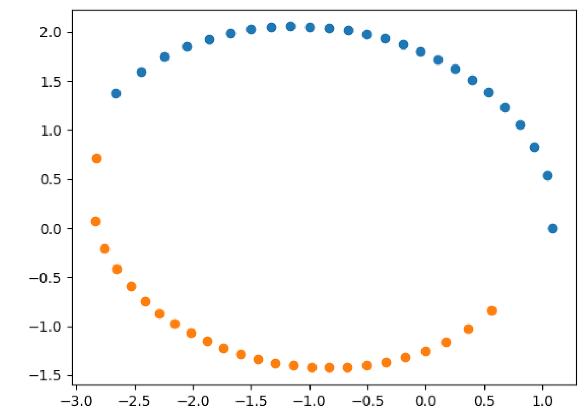


Natural Selection

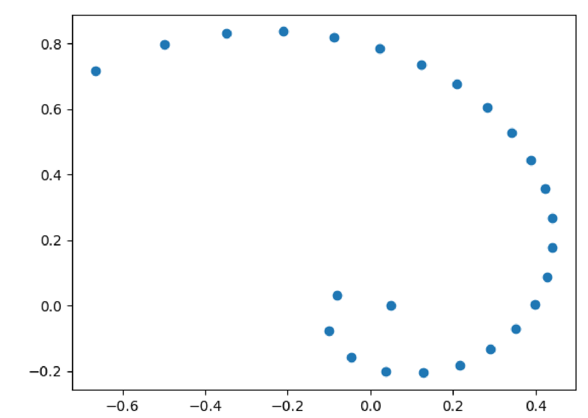
On average, the population of fast moving particles increases, so natural selection is verified by simulation!

Kepler Problem

To find the trajectory of a particle under a central force, we utilized Monte-Carlo simulation methods to solve for the desired solution.



Ellipse

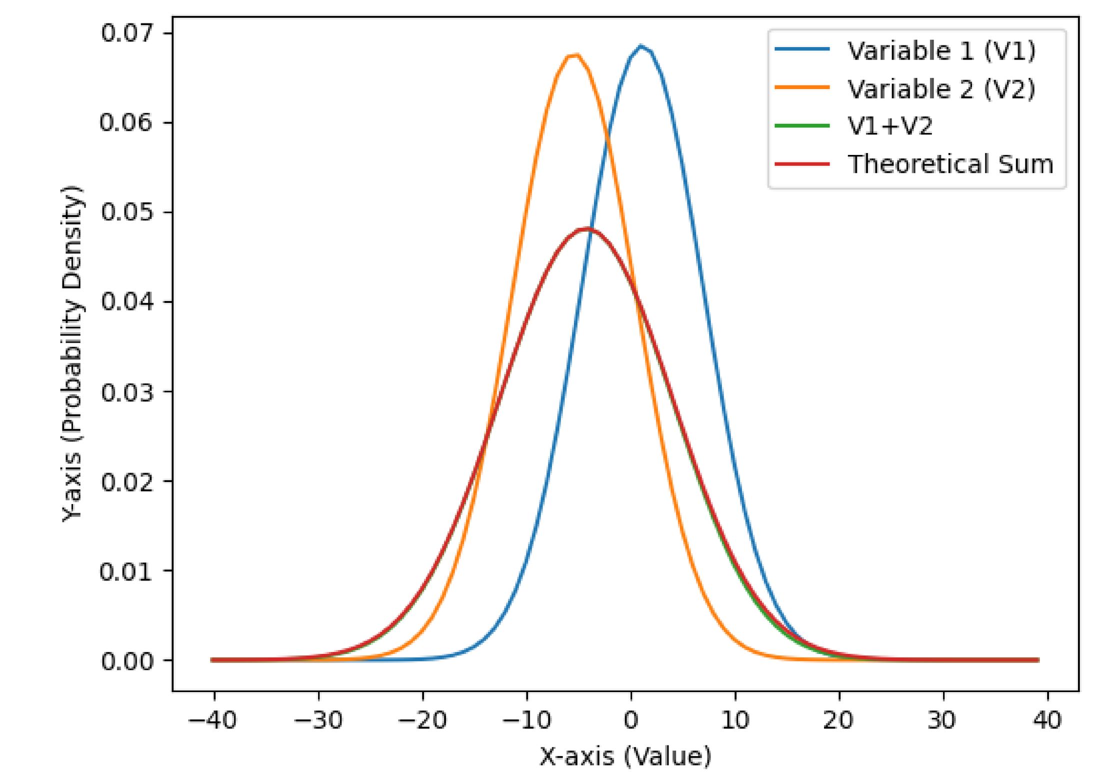


Inward Spiral

Sum of Gaussian Random Variables

The Probability Distribution Function of a Gaussian random variable is of the form:

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



Sum of two Gaussian Random Variables is Gaussian

Future Prospects

We have worked on a variety of probability and modelling problems. The Monte Carlo methods we used have been converted into a Python library, so that it can be easily *imported* and utilized for future projects. Our future plan is to work on advanced statistical modelling problems using Monte Carlo methods.

References

- Monte Carlo simulations
- Markov Chain Monte Carlo Simulations And Their Statistical Analysis: With Web-based Fortran Code, by Bernd A Berg
- Dynamic Probabilistic Systems, Volume 1: Markov Chains, by A A Markov
- Sequential Machines and Automata Theory by Taylor Booth

Haricharan, Sreejaa, Atreya, Chinmay, Himanshu, Kailash, Karthikeya, Mith, Nikhil, Pradyumnan, Praveen, Preethi, Sneha, Sudhakar, Sumedh, Venkat