

## Homework 6

Professor: Ziyu Shao

Due: 2022/11/13 10:59pm

This is the programming assignment using Python. You will submit both the source codes in the format of Jupyter Notebook (Formerly known as the IPython Notebook) and whole solution (including theory, algorithm, simulation result and analysis) in the format of PDF file. Do NOT print the file or the source codes.

1. Use the method of inverse transform sampling (or called the method of inverse CDF) to obtain samples from each of the following continuous distributions:
  - (a) Logistic distribution with CDF  $F(x) = e^x/(1 + e^x), x \in R$ .
  - (b) Rayleigh distribution with CDF  $F(x) = 1 - e^{-x^2/2}, x > 0$ .
  - (c) Exponential distribution with CDF  $F(x) = 1 - e^{-x}, x > 0$ .
2. Develop algorithms to obtain samples from each of the following discrete distributions:
  - (a) Bernoulli distribution  $\text{Bern}(0.5)$ .
  - (b) Binomial distribution  $\text{Bin}(20, 0.5)$ .
  - (c) Geometric distribution  $\text{Geom}(0.5)$ .
  - (d) Negative Binomial distribution  $\text{NBin}(10, 0.5)$ .
3. Let  $U_i \sim \text{Unif}(0, 1), i \geq 1$  be i.i.d. random variables. Define  $N$  as follows:

$$N = \max \left\{ n : \prod_{i=1}^n U_i \geq e^{-1} \right\}$$

- (a) Estimate  $E(N)$  by generating 5000 values of  $N$  and then use the sample mean.
  - (b) Estimate  $\text{Var}(N)$ .
  - (c) Estimate  $P(N = i)$ , for  $i = 0, 1, 2, 3$ .
  - (d) Can you find the exact distribution of  $N$ ?
4. (a) In the classical three-door Monty Hall problem, a contestant chooses one of three closed doors, two of which have a goat behind them and one of which has a car. Monty, who knows where the car is, then opens one of the two remaining doors. The door he opens always has a goat behind it (he never reveals the car!). If he has a choice, then he picks a door at random with equal probabilities. Monty then offers the contestant the option of switching to the other unopened door.

If the contestants goal is to get the car, should she switch doors? **Please use simulation to compare the strategy of never-switching and the strategy of switching.**

- (b) Now we consider the progressive Monty Hall problem. This time we assume there are  $n$  identical doors, where  $n$  is an integer satisfying  $n \geq 3$ . One door conceals a car, the other  $n - 1$  doors conceal goats. You choose one of the doors at random but do not open it. Monty then opens a door he knows to conceal a goat, always choosing randomly among the available doors. At this point he gives you the option either of sticking with your original door or switching to one of the remaining doors. You make your decision. Monty now eliminates another goat-concealing door (at random) and once more gives you the choice either of sticking or switching. This process continues until only two doors remain in play. What strategy should you follow to maximize your chances of winning? We consider three strategies: (1) Select a door at random and stick with it throughout. (2) Select a door at random, then switch doors at every opportunity, choosing your door randomly at each step. (3) Select a door at random, stick with your first choice until only two doors remain, and then switch. **When  $n = 4$  and  $n = 100$ , please use simulation to compare such three strategies.**

5. Estimate the value of the percolation threshold via Monte Carlo simulation.

- (a) **Percolation.** Given a composite systems comprised of randomly distributed insulating and metallic materials: what fraction of the materials need to be metallic so that the composite system is an electrical conductor? Given a porous landscape with water on the surface (or oil below), under what conditions will the water be able to drain through to the bottom (or the oil to gush through to the surface)? Scientists have defined an abstract process known as percolation to model such situations.
- (b) **The model.** We model a percolation system using an  $n$ -by- $n$  grid of sites. Each site is either open or blocked, where open means the water or other materials can flow through such site. Open sites can be further classified into two categories: full open sites and empty open sites. A full open site is an open site that can be connected to an open site in the top row via a chain of neighboring (left, right, up, down) open sites. An empty open site is an open site that is NOT full. We say the system percolates if there is a full open site in the bottom row. In other words, a system percolates if we could find a path of open sites from the top row to the bottom row. (For the insulating/metallic materials example, the open sites correspond to metallic materials, so that a system that percolates has a metallic path from top to bottom, with full sites conducting. For the porous substance example, the open sites correspond to empty space through which water might flow, so that a system that percolates lets water fill open sites, flowing from top to bottom.)

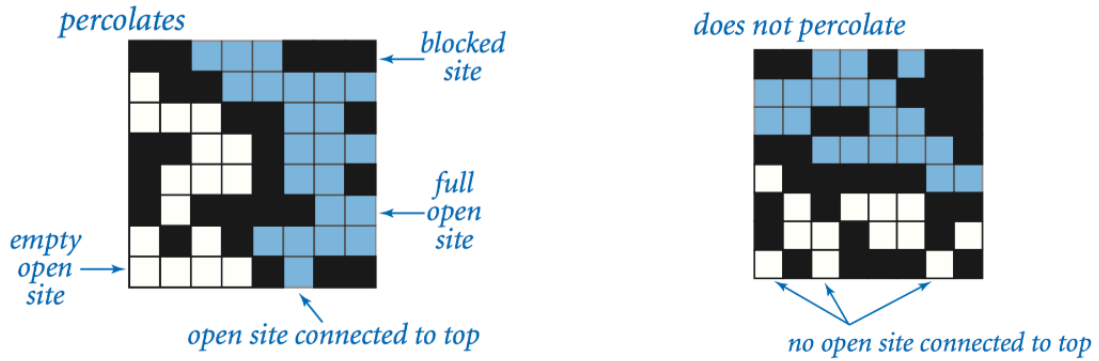
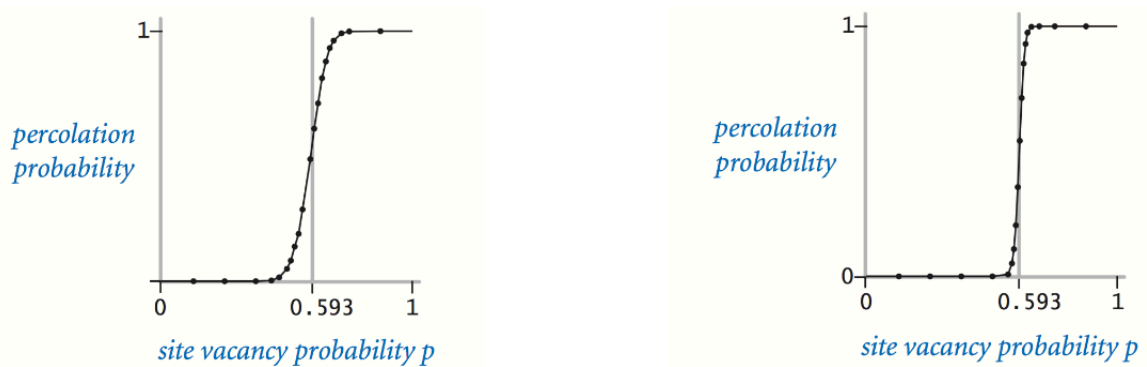


Figure 1: The blocked sites are displayed in black. The full open sites are displayed in cyan. The empty open sites are displayed in white.

- (c) **The problem.** In a famous scientific problem, researchers are interested in the following question: given an  $n$ -by- $n$  random grid, if sites are independently set to be open with probability  $p$  (and therefore blocked with probability  $1 - p$ ), what is the probability that the system percolates? When  $p = 0$ , the system does not percolate; when  $p = 1$ , the system percolates. The plots below show the site vacancy probability  $p$  versus the percolation probability for 20-by-20 random grid (left) and 100-by-100 random grid (right).

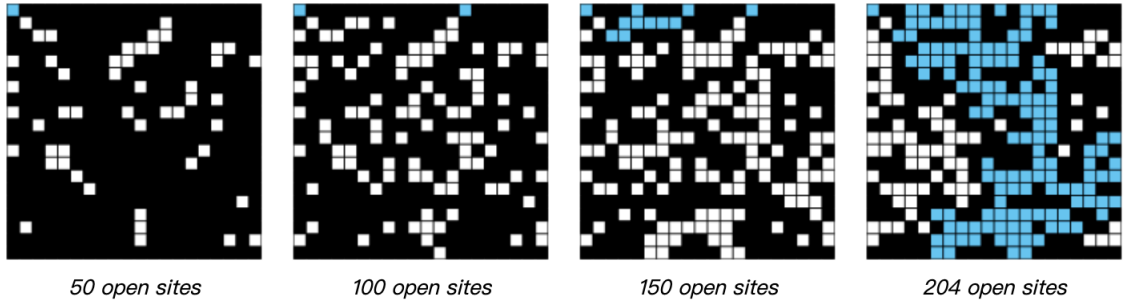


When  $n$  is sufficiently large, there is a threshold value  $p^*$  such that when  $p < p^*$  a random  $n$ -by- $n$  grid almost never percolates, and when  $p > p^*$ , a random  $n$ -by- $n$  grid almost always percolates. No mathematical solution for determining the percolation threshold  $p^*$  has yet been derived. Your task is to write a computer program to estimate  $p^*$ .

- (d) **Monte Carlo simulation.** To estimate the percolation threshold, consider the following computational experiment:
- Initialize all sites to be blocked and black.

- Repeat the following until the system percolates:
  - Choose a site uniformly at random among all blocked sites.
  - Open the site.
- The fraction of sites that are opened when the system percolates provides an estimate of the percolation threshold.

For example, if sites are opened in a 20-by-20 lattice according to the snapshots below, then our estimate of the percolation threshold is  $204/400 = 0.51$  because the system percolates when the 204th site is opened.



By repeating this computation experiment  $T$  times and averaging the results, we obtain a more accurate estimate of the percolation threshold. Let  $x_t$  be the fraction of open sites in computational experiment  $t$ . The sample mean  $\bar{x}$  provides an estimate of the percolation threshold:

$$\bar{x} = \frac{x_1 + x_2 + \cdots + x_T}{T}.$$

- (e) **The task.** Estimate the percolation threshold for 20-by-20 grid, 50-by-50 grid, and 100-by-100 grid via Monte Carlo simulation.