Optimization Theory and Methods

2025 Autumn

魏可佶 kejiwei@tongji.edu.cn
https://kejiwei.github.io/

https://www.youtube.com/watch?v=iHzzSao6ypE

https://www.youtube.com/watch?v=Suugn-p5C1M

- A computer simulation model is a computer representation that mimics the behavior of a real-world system.
- Simulation models can be used to obtain virtual statistical samples to estimate the performance of a system that involves uncertainty.
- Simulation models are often preferred when analytical solutions are difficult or impossible to find.
- Typical situations where simulation can help involve complex systems whose performance needs to be assessed under various decisions and scenarios being considered. http://www.traffic-simulation.de/
- In order to develop a simulation model, several relevant questions need to be answered. We will deal with some such questions in this course over the next few classes.

1. How to generate random numbers?
2. How to generate random observations from a probability distribution?
3. How to construct a model?
4. How to advance time?
5. How to prepare a simulation program?
6. How to validate a model?
7. How long should a simulation run?
8. How many simulation runs are needed?
9. How to perform statistical analysis of the results obtained from simulation runs?

- Standard approach to produce random numbers has two steps.
  1. Produce a sequence of statistically independent random numbers that are distributed uniformly within a finite range.
  2. Process and transform these uniform random numbers into a sequence of sample points from any desired distribution.
- Let us focus on the first stage.
  - Most computer systems have functions that allow easy generation of random numbers.
  - In Microsoft Excel and in MATLAB, rand() returns a random number that is uniformly distributed between 0 and 1.
  - Successive random numbers generated by ***rand()*** function can be considered to be mutually independent samples from $\boldsymbol{U}[\mathbf{0},\mathbf{1}]$. We often denote such numbers by letter $\boldsymbol{r}$

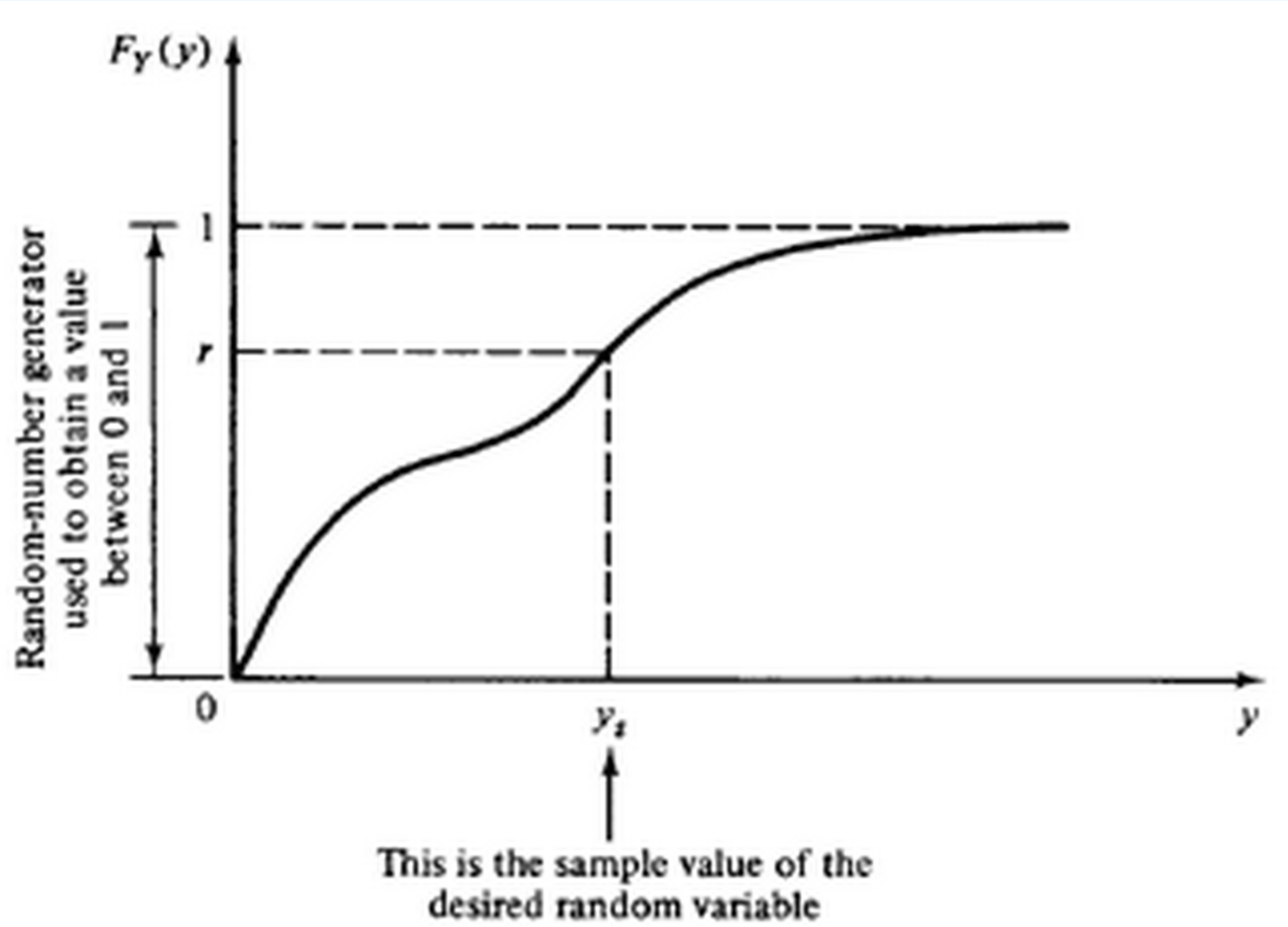
- Let us consider the simplest example: $0/1$ Bernoulli Distribution.
- We can use a function such as $rand(\quad)$ to generate a series of random numbers $r \sim U[0,1]$.
- The challenge is to use this series of numbers to generate another series that contains only 0s and 1s such that the probability of any number being 1 is p and 0 is $1-p$. In other words, we want to simulate a Bernoulli distributed random number b from each $r \sim U[0,1]$.
- A simple way is to check if the number r is in $[0,1-p]$. If so, we set b to 0, else we set it to 1. Note that the probability of r being in $[0,1-p]$ is $1-p$.
- *Inversion method* extends this idea to other distributions.

- As stated earlier, the PDF of r can be written formally as,

$$f_R(r) = \begin{cases} 1, & if\ 0 \leq r \leq 1 \\ 0, & otherwise \end{cases}.$$

- We observe that for any random variable Y, the CDF, $F_Y(y)$ is a non-decreasing function with $0 \leq F_Y(y) \leq 1$. So can we match this $F_Y(y)$ with the uniform random numbers r that we have already produced?
- Specifically, if we set $F_Y(y) = r$, then we can calculate $y = F_Y{}^{-1}(r)$. This implies that we first plot the CDF of Y. Then we draw a $U[0,1]$ random number r. Then we find the inverse image of r on the vertical axis, which will give us the corresponding value of y which has a distribution given by CDF F_Y.

↳Inversion Method Illustrated on the CDF



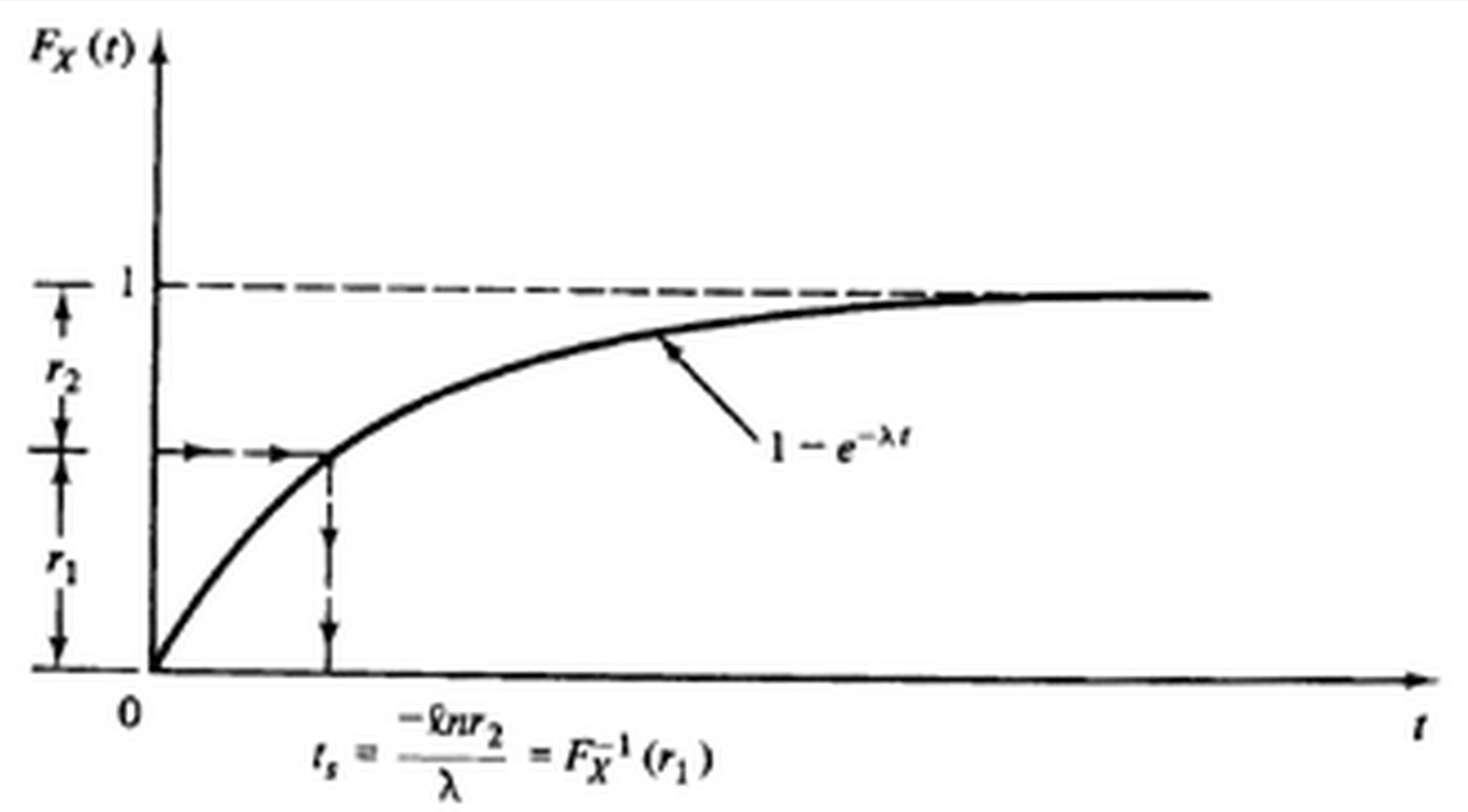
- We need to convince ourselves that the inversion method actually "works".
- Consider random variable Y, and any two real numbers a and b.
- If the method works, then for any a and b, it should give the value of $P(a \leq Y \leq b)$ correctly,
- That is, the method should be able to yield a random number between a and b with probability $P(a \leq Y \leq b)$.

- 3-Minute Activity: Spend a minute by yourself and then discuss for 2 minutes with your neighbors to check if indeed the method "works".
- Be prepared to explain your answer to the class.

↳ Inversion Method Example1

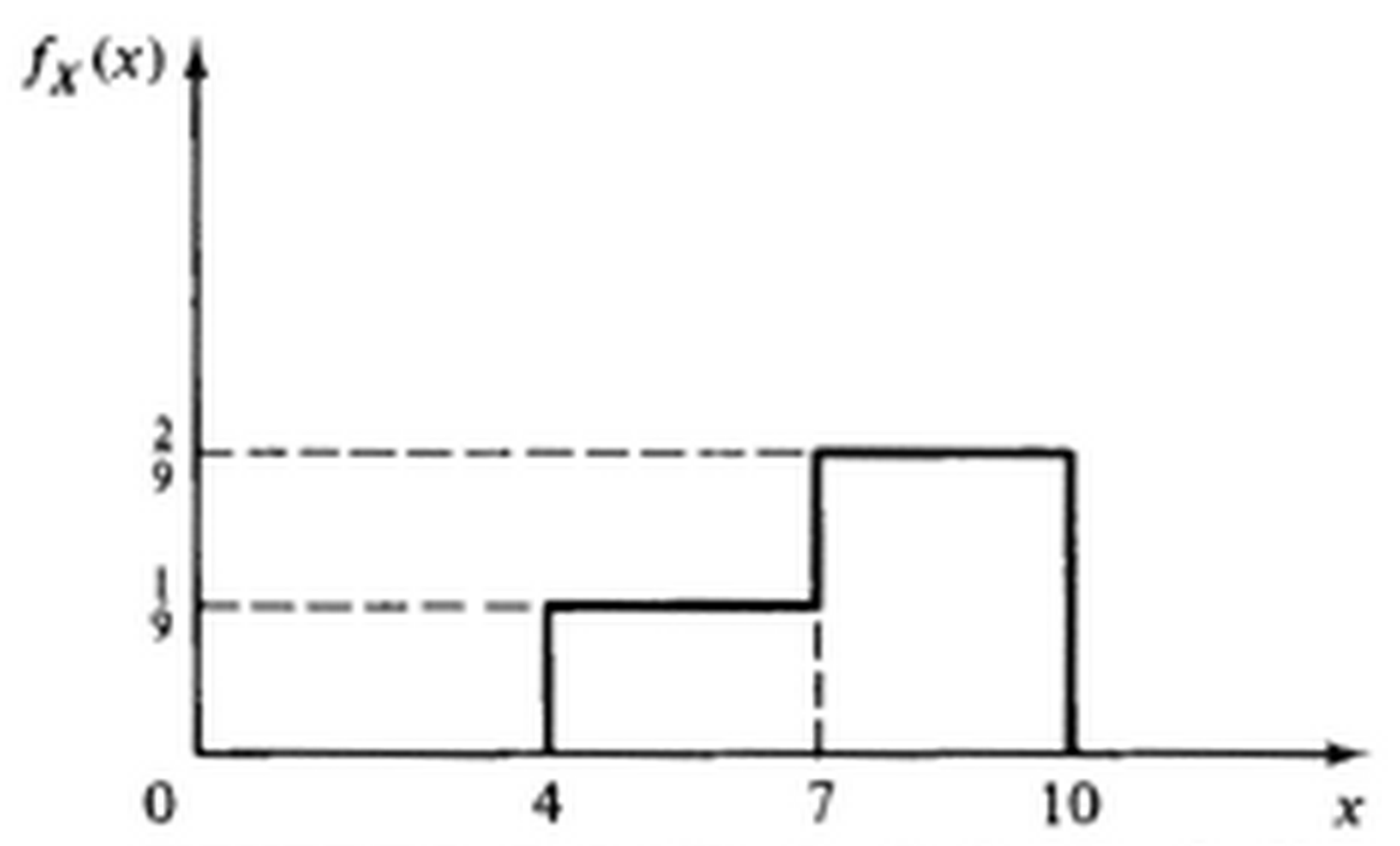
- **Problem**: Emails into your inbox follow a Poisson process with $\lambda = 3$ /hour. Generate a sequence of random numbers to simulate the arrival process.
- **Solution**: Inter-arrival times (T) in a Poisson process follow an exponential distribution. $f_T(t) = \begin{cases} \lambda \exp(-\lambda t) & for\ t \geq 0 \\ 0 & for\ t < 0 \end{cases}$.
- $r = F_T(t) = \int_0^t f_T(x)dx = \int_0^t \lambda \exp(-\lambda x)\, dx = 1 - \exp(-\lambda t)$.
- Thus, $-\lambda t = \ln(1 - r)$. So, $t = \frac{-\ln(1-r)}{\lambda} = F_T{}^{-1}(r)$.
- We can directly use this formula to calculate inter-arrival times between emails using $r \sim U[0,1]$.
- Furthermore, note that $(1 - r) = r_2 \sim U[0,1]$. So we can further simplify the formula as $t = F_T{}^{-1}(r) = \frac{-\ln(r)}{\lambda}$.

↳ Inversion Method Example1 (cont.)

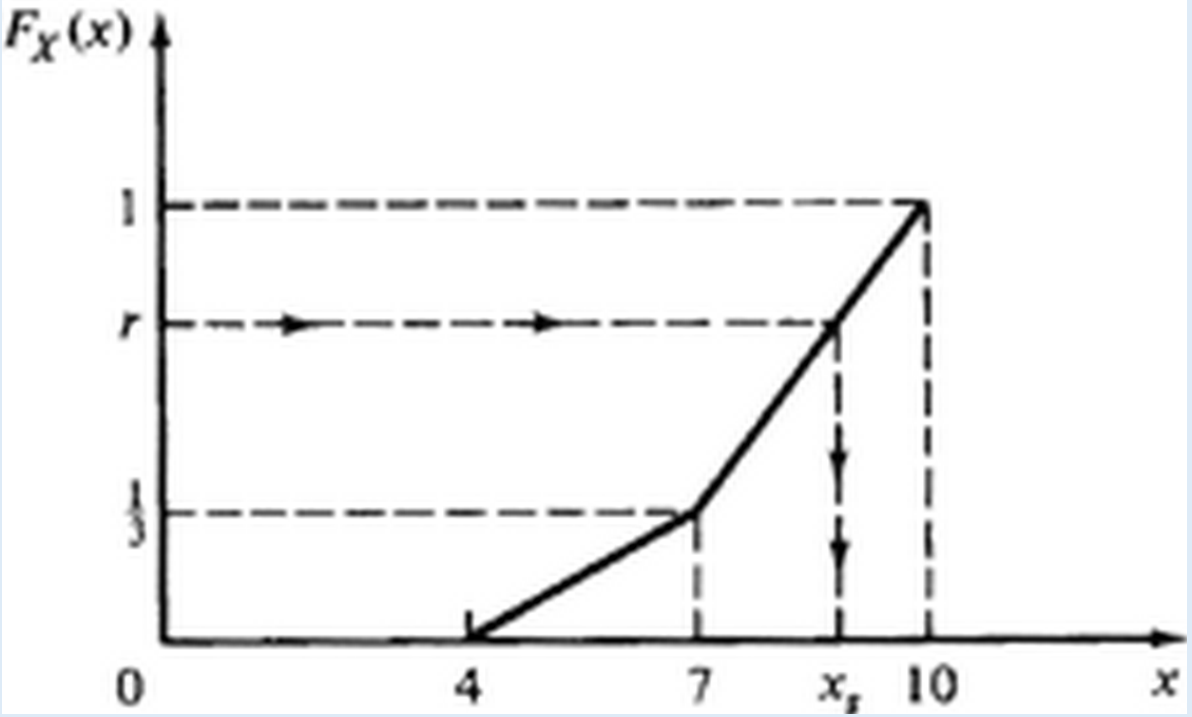


↳ Inversion Method Example 2

- Problem: Consider a 6-mile road between mile markers 4 and 10. The PDF of the location of next accident is given by the diagram below. Simulate the location X where the next accident will occur.



Solution: $F_X(x) = \begin{cases} 0 & for\ x < 4 \\ \frac{1}{9}(x-4) & for\ 4 \le x < 7 \\ \frac{1}{3} + \frac{2}{9}(x-7) & for\ 7 \le x < 10 \\ 1 & for\ 10 \le x \end{cases}.$

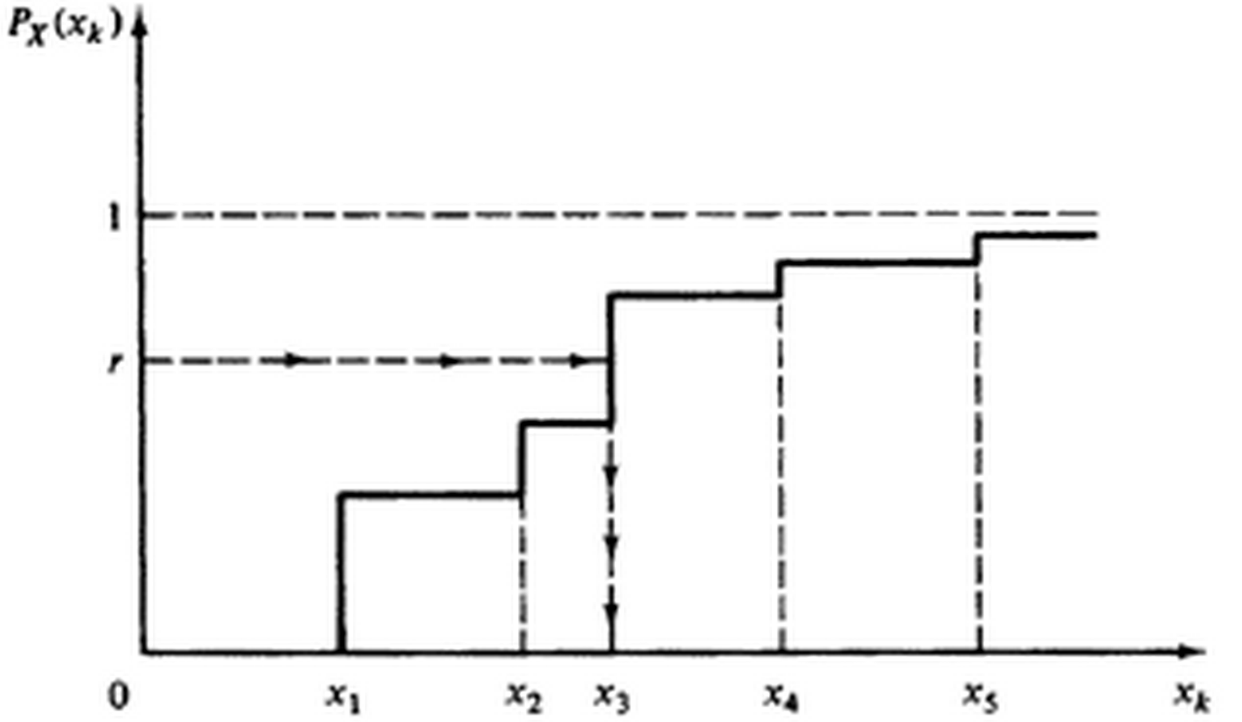


- $r \sim U[0,1]$.
- If $r \le \frac{1}{3}$, then $r = \frac{1}{9}(x-4)$, and hence $x = 9r + 4$.
- Else, $r = \frac{1}{3} + \frac{2}{9}(x-7)$, and hence $x = \frac{1}{2}(9r + 11)$.

↳ Example 2: Alternate Approach

- In general, there can be many different ways of generating a random number with the same distribution.
- Multiple random numbers from $U[0,1]$ can also be used if generating them is not too computationally intensive.
- For example 2, an alternate approach is as follows:
  - <u>Step 1</u>: Generate 2 random numbers $r_1 \sim U[0,1]$ and $r_2 \sim U[0,1]$.
  - <u>Step 2</u>: If $r_1 < \frac{1}{3}$, $x = 3r_2 + 4$, else $x = 3r_2 + 7$.

↳ Inversion Method for Discrete RVs



When simulating a discrete random variable X, with PMF and CDF given respectively by $p_X(x)$ and $F_X(x)$, from a single $r \sim U[0,1]$ variable, find the smallest value x such that $F_X(x) = \sum_{y=x_1}^{x} p_X(y) \geq r$.

- E.g. if $p_X(x) = p(1-p)^{x-1}$ for $x = 1,2,\ldots$, then $F_X(x) = 1-(1-p)^x$
- So $x = \left\lceil \frac{\ln(1-r)}{\ln(1-p)} \right\rceil$, where $\lceil a \rceil$ denotes the smallest integer greater than or equal to a.

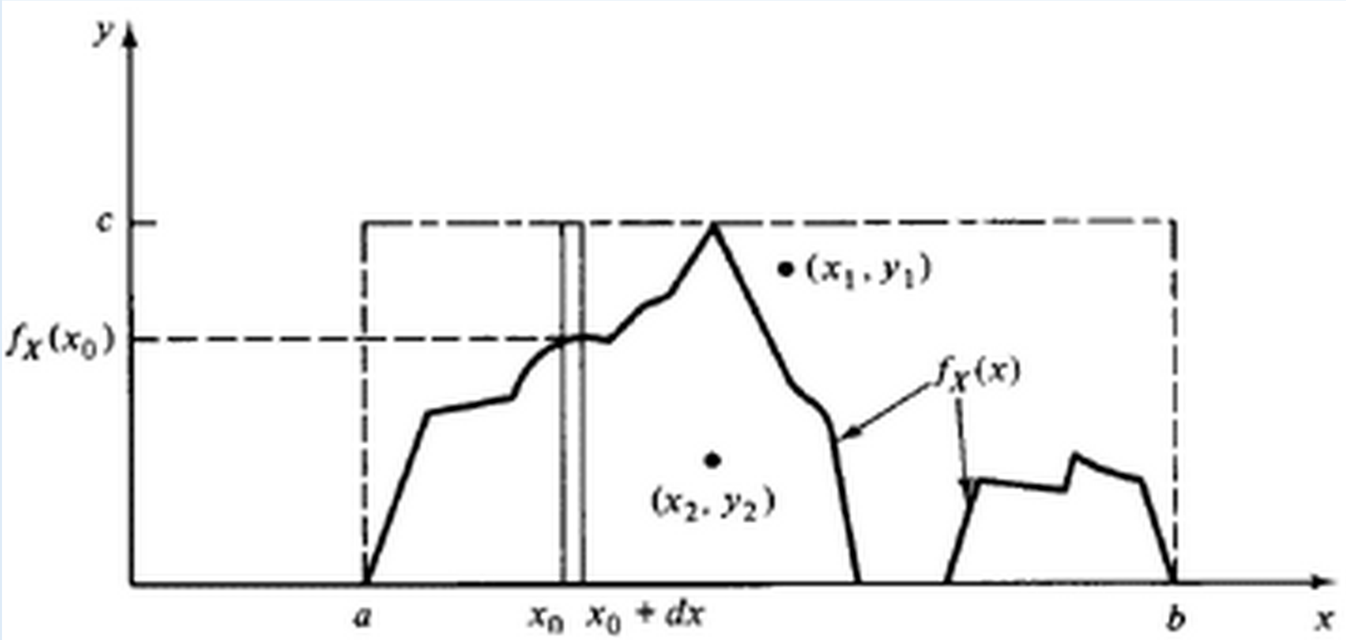
↳ Principle of the Relationship Method

- The idea is to take advantage of some known relationship between this random variable and one or more other random variables.
- <u>Example</u>: Simulating a binomial random variable.
- Binomial probability mass function gives the probability of k successes in n Bernoulli trials.
- Simulate n Bernoulli trials, i.e., obtain n values of a random variable which takes value 1 with probability p and value 0 with probability $1 - p$.
- Then count the number of successes in n trials, which gives a single value of random number drawn from binomial distribution.

↳ Principle of the Rejection Method

- This method can be used to generate random values from any distribution that (1) takes values in a finite range, and (2) has a bounded PDF/PMF (i.e., PDF/PMF does not go to infinity at any value of the random variable).
- Let X be a random variable that follows these two conditions; let c be the maximum value taken by its PDF $f_X(x)$; and let all values of X with non-zero PDF be contained in interval $[a, b]$.
- Use the following three step procedure to generate random values from distribution $f_X(x)$ using the rejection method.
  - Step 1: Enclose the PDF $f_X(x)$ in the smallest rectangle that fully contains it and whose sides are parallel to the x and y axes. This rectangle will have a width $(b - a)$ and height c.

↳ Procedure of Inversion Method



- Step 2: Use two random numbers, $r_1, r_2 \sim U[0,1]$, and transform using a, b, and c to get a point $(x_1, x_2) = (a + r_1(b - a), r_2 c)$ inside the rectangle.
- Step 3: If this point is below the PDF, the x-coordinate of this point gives you the simulated value of random variable X. Otherwise, reject this point and return to step 2.

- Let us look at the highway accident example again.
  - Step 1: This PDF can be enclosed in a rectangle whose length is $(b-a) = (10-4) = 6$ and height is $c = \frac{2}{9}$.
  - Step 2: Let $(r_1, r_2) = (0.34, 0.81)$ be the two random numbers being drawn from $U[0,1]$. The corresponding point in the rectangle will be $(x_1, y_1) = \left(4 + 6 * 0.34, 0.81 * \frac{2}{9}\right) = (6.04, 0.18)$.
  - Step 3: Since this point is above
    - $(6.04, 0.11)$ it will be rejected. Let
    - the new point be $(r_1, r_2) = (0.41, 0.15)$
    - leading to $(x_1, y_1) = (6.46, 0.033)$, so
    - the sample value $x = 6.46$ will be accepted.

↳ Rejection Method: Pros & Cons

- The rejection method is very general.
- As long as the PDF satisfies the two aforementioned conditions, this method can be used.
- The PDF does not need to have a closed form mathematical expression either.
- The downside is that sometimes many rejections are needed before simulating each random value.
- The expected number of trials until an accepted value is found equals $c(b-a)$ (Why?).

- Approximate a complicated distribution using another which is easier to simulate using one of the earlier three methods.
- Some approximations can be sophisticated.
- E.g. Simulating a normal distribution by applying Central Limit Theorem (CLT).
- Generate k random variables, each from $U[0,1]$ and take sum. $Z = r_1 + r_2 + \cdots + r_k$.
- Each r variable has mean $= \frac{1}{2}$ and standard deviation $= \frac{1}{\sqrt{12}}$.
- So mean of Z is $\frac{k}{2}$ and standard deviation is $\sqrt{\frac{k}{12}}$.

- For large values of k, $\frac{Z-\frac{k}{2}}{\sqrt{\frac{k}{12}}}$ is approximately normal with mean 0 and standard deviation 1.
- So a random variable X from any normal distribution with mean μ and standard deviation σ can be generated as $\mu + \left(\frac{Z-\frac{k}{2}}{\sqrt{\frac{k}{12}}}\right)\sigma$.
- Another common type of approximation is to approximate a complicated CDF using a simpler CDF (such as a piecewise linear CDF) and then
  - using inversion method, or
  - using a rejection method.

18.8 Method Comparison

- We have seen 4 different methods so far (Inversion, Relationships, Rejection, Approximations) for generating random numbers from a given probability distribution.
- Obviously they have some pros and cons each.
- Compare the four methods and list as many pros and cons as you can.

- 4-Minute Activity: Spend 2 minutes by yourself and then discuss for 2 minutes with your neighbors.

- Be prepared to explain your answer to the class.

18.8 Method Comparison

↳ Comparison of Random Number Generation Methods

Method	Advantages	Disadvantages	Typical Applications
Inversion Method	- Simple and intuitive principle - Exact if the inverse CDF is available - Works well for many standard distributions	- Inverse CDF often not available in closed form - Numerical inversion can be computationally expensive	- Exponential distribution - Geometric distribution - Any distribution with a closed-form inverse CDF
Relationship Metho	- Efficient by exploiting known relationships between distributions - Avoids complex mathematics - Easy to implement once relation is known	- Limited to cases with known relationships - Not a universal approach	- Generating Binomial from Bernoulli - Generating Chi-square, t, or F distributions from Normal

↳ Comparison of Random Number Generation Methods (cont.)

Method	Advantages	Disadvantages	Typical Applications
Rejection Method	- Does not require inverse CDF or closed form - Flexible for complex or nonstandard distributions	- Can be inefficient if rejection rate is high - Requires a suitable proposal distribution	- Normal distribution (via Box-Muller or rejection variants) - Gamma distribution - Complicated PDFs without closed forms
Approximation Method	- Useful when exact simulation is infeasible - Flexible, can be combined with other methods - Often faster in practice	- Introduces approximation error - Accuracy depends on approximation technique and parameters	- Approximating Normal via Central Limit Theorem (CLT) - Approximating complex CDFs with piecewise linear functions - Simulation in large-scale Monte Carlo where efficiency matters

Objective :

Key Concepts :