TAM 598 Lecture 7:

UNCERTAINTY PROPAGATION – BASIC SAMPLING

i.e. how to generate random numbers

Announcements:

- HW 2 covers lectures 4-8; due on Feb 26

Why generate random numbers?

(1) Monte Carlo - to estimate difficult expectations, to solve high-dimensional integrals

(2) Randomized Algorithms - use random numbers to influence their behavior or decisions, to do things faster

eg) Quicksort - random choice of pivot element, reduces from $O(n^2)$ worst case to $O(n \log n)$

(3) cryptography

(4) games & graphics

Computers cannot generate true random numbers

We usually don't need true random numbers, but need sequences that have the same statistics as true random numbers:

- distributed according to the pmf or pdf
- statisically uncorrelated (no detectable patterns)

Random numbers that we generate deterministically on a computer are called pseudo random numbers

- initialize sequence with a seed, and iterate

Today: three pseudo random number generators (PRNGs)

1) Middle Square Algorithm
2) Linear Congruential Generator
3) Mersenne Twister

Algorithms for random number generation follow a typical recipe.

Middle Square Algorithm - John von Neumann

1. state space S
2. $f(s)$
3. $g(s)$

generates a sequence of numbers

$$S = \{ s_0, s_1, s_2, \dots \}$$

where s_{i+1} is entirely determined by s_i

Linear Congruential Generator

- state space S
- $f(s)$
- $g(s)$

Mersenne Twister PRNG

- used in numpy and most languages

- period length is given by a Mersenne prime $2^n - 1$

$$2^{19937} - 1$$

If $2^n - 1$ is prime, then n is prime

- maintains an internal state of 624 integers, generates random numbers by performing bitwise operations on this state to produce a new sequence of bits with each iteration

  → generate 624 numbers

  → apply a "twisting" transformation to mix bits from adjacent integers

  → range is from 0 to $2^{31} - 1 = 2,147,483,647$

  so a 31 bit integer

Say we have a PRNG that yields a uniform distribution on $\mathcal{U} = \{0, 1, 2, \ldots, m-1\}$. How do we go to something more interesting / useful?

eg) uniform on $[0, 1)$ given by u/m

SAMPLING THE CATEGORICAL

(discrete distributions)

given we can draw from $U[0,1]$

$X \sim \text{Bernoulli}(\theta)$

$X \sim \text{Categorical } K$

SAMPLING FROM CONTINUOUS DISTRIBUTIONS:

INVERSE TRANSFORM SAMPLING given draws from $U[0,1]$

Example: Poisson distribution

Rejection Sampling - use randomness on simple distributions, and make decisions about whether to reject or keep the quantity sampled. The set of things we keep is from the right distribution.

want to sample from $f(x)$ but don't have access to its CDF