TP 547- Princípios de Simulação de Sistemas de Comunicação

Prof. Samuel Baraldi Mafra

Inatel
Instituto Nacional de Telecomunicações

Gerador Randu

- Algoritmo gerador de números aleatórios que foi muito usado nos mainframes da IBM das décadas de 60 e 70;
- $x_{n+1} = ((2^{16} + 3)x_n) \ mod \ 2^{31}$
- Este algoritmo apresenta uma correlação entre as amostras.

$$\begin{aligned} x_{n+2} &= ((2^{16} + 3)^2 x_n) \ mod \ 2^{31} \\ &= ((2^{32} + 6 * 2^{16} + 9)x_n) \ mod \ 2^{31} \\ &= ((6 * (2^{16} + 3) - 9)x_n) \ mod \ 2^{31} \\ &= 6 * x_{n+1} - 9x_n \ mod \ 2^{31} \end{aligned}$$

Geradores de Variáveis Aleatórias com Distribuição Uniforme

Gerador

$$x_{n+1} = ((ax_n)\ mod\ m)/m$$

Geradores de Variáveis Aleatórias com Distribuição Uniforme

- Gerador recomendado por Park e Miller e utilizado no Matlab nas primeiras versões:
  $x_{n+1} = 7^5 x_n \ mod \ (2^{31} - 1)$
- Gerador de Distribuição Uniforme:
  $x_{n+1} = (7^5 x_n \ mod \ (2^{31} - 1))/(2^{31} - 1)$

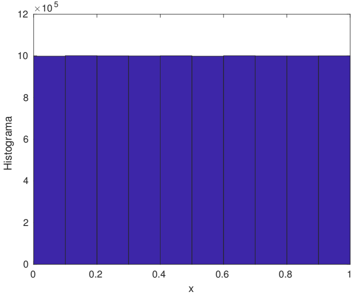
Gerador para distribuição uniforme geradoruniforme.py

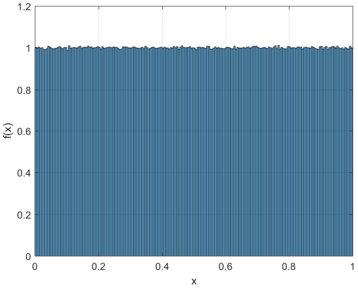
```
import numpy as np
import matplotlib.pyplot as plt
x=1
x1=np.array([x])
av=np.array([])
n=100
a=pow(7,5)
m=pow(2,31)-1
for i in range(n):

        x=(a*x)%m
        x1=np.append(x1,x)
x1=x1/m
```

Geradores de Variáveis Aleatórias com Distribuição Uniforme

$$x_{n+1} = (7^5 x_n \ mod \ (2^{31} - 1))/(2^{31} - 1)$$
$$x_0 = 1$$





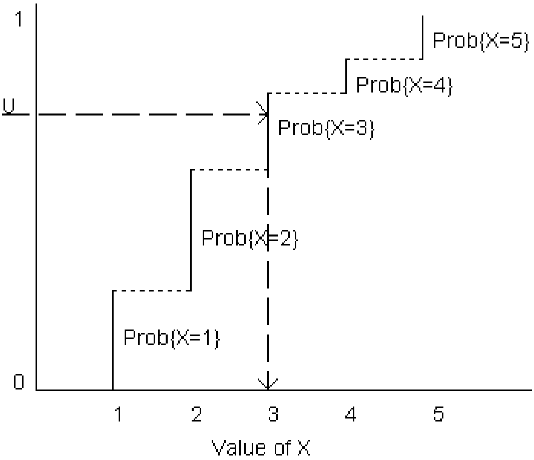
Variáveis Aleatórias (VA)

- Uma variável aleatória, X, é uma função que associa um número real com cada elemento do espaço amostral S;
- V. A. Contínuas: Uma função X definida sobre o espaço amostral e assumindo valores num intervalo de números reais, é denominada variável aleatória contínua.
  - Intervalo entre pacotes;
  - Tempo de atendimento de requisições num servidor;
  - Atraso dos pacotes na rede;
  - Variação do atraso (Jitter);
- V. A. Discretas: Uma variável aleatória é definida como sendo discreta quando o número de valores possíveis que a variável assume for finito.
  - Número de pacotes transmitidos durante um intervalo;
  - Tamanho dos pacotes recebidos;
  - Número de acessos a um determinado site, das 0h às 6h

Geradores de Variáveis Aleatórias Discretas

Geradores de Variáveis Aleatórias Discretas

- Distribuição Uniforme;
- CDF



Algoritmo de Geração

1. Gerar uma variável aleatória, u, com distribuição uniforme $u \sim U(0,1)$;
2. Se $u < p_0$, $X = x_o$ e finaliza;
3. Se $u < p_0 + p_1$, $X = x_1$ e finaliza;
4. Se $u < p_0 + p_1 + p_2$, $X = x_2$ e finaliza;

   $\vdots$
5. Se $U < p_0 + p_1 + p_2 + ... + p_n$, $X = x_n$ e finaliza

Distribuições discretas: Bernoulli(q)

- A distribuição de Bernoulli é a distribuição discreta mais simples. Uma variável de Bernoulli pode apenas assumir dois valores, que são usualmente denotados por sucesso e falha ou $x = 0$ e $x = 1$, respectivamente.
  1. $q =$ denota a probabilidade de sucesso;
  2. $p = 1 - q =$ denota a probabilidade de falha.
- Experimentos que geram uma variável de Bernoulli são denominados de experimentos de Bernoulli. Por exemplo:
  - Sistema computacional está ativo ou inativo;
  - Um pacote na rede de computadores chega ou não chega ao destino;
  - Um bit em um pacote é afetado ou não afetado pelo ruído.

Distribuições discretas: Bernoulli(q)

- q (probabilidade de sucesso) $(x = 1)$ $0 \leq q \leq 1$;
- Faixa de valores:0,1;
- pmf:$f(x) = \begin{cases} p = 1 - q, & x = 0 \\ q, & x = 1. \end{cases}$
- Média: q
- Variância: $q(1 - q)$

Geradores de Variáveis Aleatórias com Distribuição Bernoulli

Algoritmo de Geração

- Gerar uma variável aleatória, u, com distribuição uniforme $u \sim U(0,1)$;
- Se $u \leq p$ retornar 0;
- Caso contrário retornar 1.

geradorbernoulli.py
randbernoulli2.py

Distribuições discretas: Binomial(n,q)

- Distribuição binomial: conta o número de sucessos em n tentativas independentes, com probabilidade de sucesso q em cada tentativa
- A distribuição binomial é usada para modelar o número de sucessos numa seqüência de n experimentos de Bernoulli idênticos e independentes, por exemplo:
  - O número de processadores que estão ativos em um sistema com múltiplos processadores;
  - O número de pacotes que chegam ao destino sem perdas;
  - O número de bits num pacote que não são afetados pelo ruído.

Distribuições discretas: Binomial(n,q)

- Parâmetros
  - q (probabilidade de sucesso) $(x = 1)$ $0 \leq q \leq 1$;
  - $n =$ número de repetições do experimento, n deve ser um número inteiro positivo;
- Faixa de valores:0,1,...,n;
- pmf:$f(x) = \binom{n}{x} q^x (1-q)^{n-x} \quad q = 1 - p$
- cdf: $F(x) = P\{X \leq x\} = \sum_{k=0}^{x} f(k)$
- Média: $\mu = nq$
- Variância: $\sigma^2 = nq(1-q)$

Geradores de Variáveis Aleatórias com Distribuição Binomial

A distribuição binomial apresenta a seguinte pmf:

$$\Pr[X = x] = \binom{n}{x} q^x (1-q)^{n-x}$$

Podemos escrever as probabilidades de forma recursiva como

$$\Pr[X = x+1] = \frac{q}{1-q}\left[\frac{(n-x)}{x+1}\right]\Pr[X = x]$$

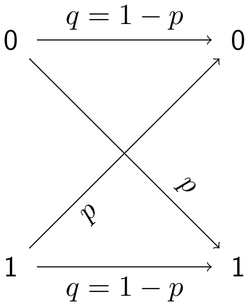
Geradores de Variáveis Aleatórias com Distribuição Binomial

Algoritmo de Geração

1. Gerar uma variável aleatória, u, com distribuição uniforme $u \sim U(0{,}1)$;
2. Calcular $c = \frac{q}{1-q}$, $pr = (1-q)^n$;
3. Iniciar $i = 0$, $F = pr$;
4. Se $u < F$, $X = i$ e finaliza;
5. Calcular $pr = c\left[\frac{(n-i)}{i+1}\right]pr$
6. $F = F + pr, i = i + 1$;
7. Ir para o passo 4.

randbinomial.py

Considerando a transmissão de pacotes com 4 bits, caracterize o número de erros no receptor em função de uma variável aleatório discreta. Qual é a probabilidade de obter 2 erros? $q = 0.8$
randbinomial1.py



Distribuições discretas: Binomial(n,q)-Exemplo

- Em um projeto de pesquisa, um sistema multiprocessado utiliza 12 processadores e está configurado de maneira a realizar todo o processamento, talvez com alguma perda de velocidade, se até 9 processadores estiverem operando;
- q=0.8 - probabilidade de cada processador operar corretamente durante o projeto;
- Determine a probabilidade do sistema operar até o final do projeto. randbinomial2.py

Distribuições discretas: Poisson(λ)

- A distribuição de Poisson é uma forma limite da distribuição binomial para n grande e p pequeno.
- Esta distribuição é aplicada para modelar o número de chegadas num intervalo de tempo:
  - Número de requisições a um servidor num dado intervalo t.
  - Número de falhas de componentes por unidade de tempo.
  - Número de consultas a um banco de dados num intervalo t segundos;
  - Número de clientes que chegam a uma fila de atendimento por unidade de tempo.

Distribuições discretas: Poisson(λ)

- Parâmetros
  - λ.
- Faixa de valores:0,1,...,∞;
- pmf:$f(x) = P(X = x) = \lambda^x e^{-\lambda}/x!$
- cdf:$F(x) = P(X \leq x) = e^{-\lambda} \sum_{i=0}^{x} \lambda^i/i!$
- Média: λ
- Variância: λ

Geradores de Variáveis Aleatórias com Distribuição de Poisson

A distribuição de Poisson apresenta a seguinte pmf:

$$\Pr[X = x] = \lambda^x e^{-\lambda}/x!$$

Podemos escrever as probabilidades de forma recursiva como

$$\Pr[X = x + 1] = \frac{\lambda}{x + 1} \Pr[X = x]$$

Geradores de Variáveis Aleatórias com Distribuição de Poisson

Algoritmo de Geração

1. Gerar uma variável aleatória, u, com distribuição uniforme $u \sim U(0{,}1)$;
2. Calcular $p = e^{-\lambda}$;
3. Iniciar $i = 0$, $F = p$;
4. Se $u < F$, $X = i$ e finaliza;
5. Calcular $p = \frac{\lambda}{i+1} p$
6. $F = F + p, i = i + 1$;
7. Ir para o passo 4.

randpoisson.py

- Um departamento de polícia recebe em média 5 solicitações por hora. Qual a probabilidade de receber 2 solicitações numa hora selecionada aleatoriamente?
- Um número médio de 6 clientes param por hora em uma bomba para colocar gasolina. Qual é a probabilidade de 3 clientes ou menos pararem em qualquer hora?