

## Capítulo 8

# Sistemas Aleatórios

Dizemos que um sistema é aleatório quando seu estado futuro só pode ser conhecido pela realização de uma experiência. O exemplo mais simples desse tipo de sistema é uma moeda sendo jogada para o alto. Não há como saber qual a face estará para cima quando a moeda cair, essa informação só é obtida ao realizarmos a experiência, que é jogar a moeda no caso. Porque ocorre essa indeterminação? Para poder prever o resultado deveríamos analisar a moeda com detalhe, assim como a forma com que é jogada para o alto, e resolver as complicadas equações de movimento. Se a moeda estiver sendo jogada por uma pessoa teríamos que garantir que esta seria capaz de repetir exatamente a forma de jogar a moeda. Claro que a solução desse “simples” problema é completamente inviável, sempre existirá alguma indeterminação no processo. Um sistema também aleatório é o formado por moléculas de um gás. Vamos supor que as moléculas estão separadas o suficiente para que seja razoável desprezar a interação entre elas. À temperatura ambiente essas moléculas tem um movimento que combina translação do centro de massa e rotação em torno de diversos eixos de simetria molecular. Dentro do recipiente que contém o gás as moléculas estão constantemente colidindo umas com as outras e colidindo com as paredes do reservatório. Imagine que desejemos entender o comportamento desse gás pelo conhecimento da trajetória das cerca de  $10^{23}$  partículas. Seja o caso mais simples, um gás monoatômico sem energia cinética de rotação, neste caso precisaríamos de 6 variáveis reais para cada partícula, três para definir  $\vec{r}$  e três para  $\vec{v}$ . Então, apenas para armazenar a informação de um determinado estado do gás precisaríamos de cerca de  $5 \times 10^{18}$  Mb! Supondo que temos essa quantidade de memória disponível, imagine quanto tempo levaríamos para calcular as trajetórias. O pior de tudo é que toda essa informação de nada serviria para o entendimento do comportamento do gás. É muito mais vantajoso encarar o movimento errático das moléculas como uma trajetória aleatória e usar o fato de que são muitas moléculas e que de fato observamos o resultado médio de suas trajetórias.

De uma forma geral, todos os sistemas macroscópicos se apresentam a nós pelas médias de seus componentes atômicos e moleculares, sendo a temperatura a responsável pela aleatoriedade no nível microscópico. A Mecânica Estatística é o ramo da física que estuda como calcular essas médias a partir do conhecimento das propriedades microscópicas do sistema. Existem outros sistemas aleatórios que ocorrem em uma escala muito diferente, e que não são necessariamente regidos por leis físicas microscópicas. Por exemplo a formação das espécies, a bolsa de valores, o trânsito e os terremotos e avalanches. Em todos esse caso o que se busca é uma descrição probabilística do sistema que permita o estudo de médias, correlações e outras grandezas que o caracterizem. Como sempre, a descrição puramente analítica só é possível em situações muito simples, e os cálculos numéricos são amplamente usados no estudo de sistemas aleatórios. Veremos aqui alguns elementos básicos desse tipo de problema.

### 8.1 Probabilidades: Nosso senso comum

Vamos começar explorando a noção cotidiana para o conceito de probabilidade. Em primeiro lugar, a necessidade de empregar esse conceito vem da impossibilidade de prever o resultado de um determinado experimento. Por experimento entende-se uma enorme variedade de situações, por exemplo podemos estar interessados em saber se uma jogada de moeda vai ter como resultado cara ou coroa, se vai chover ou não no dia seguinte, ou se um investimento na bolsa de valores será lucrativo. Como vimos acima, a previsão exata

desse resultados é inviável, é preferível construir um modelo que nos permita calcular a *probabilidade* de cada resultado possível. No caso da moeda, esse modelo certamente incluirá considerações sobre a distribuição de massa da moeda, sobre a forma com que é jogada e sobre o número de resultados possíveis, ou seja, devemos considerar a possibilidade da moeda cair em pé? A esse último passo chamamos de *definir o espaço de amostragem*, e é uma etapa fundamental no cálculo de probabilidades. Para a moeda, em geral, supomos dois resultados possíveis e igualmente prováveis, já que cair em pé é um evento muito raro, e, em geral, não há razão para se supor que uma face tenha prioridade sobre a outra. Note que estamos usando a qualificação *evento raro* de forma bastante qualitativa. Chamando de  $P(\star)$  a probabilidade do evento  $\star$  ocorrer, obtemos então

$$P(\text{cara}) = P(\text{coroa}) = \frac{1}{2}. \quad (8.1)$$

Como podemos comprovar experimentalmente esse modelo? Jogando a moeda. Na verdade poderíamos ter determinado  $P(\text{cara})$  e  $P(\text{coroa})$  jogando a moeda, e aqui entra em cena outro elemento importante, o número de vezes,  $N$ , que jogamos a moeda. Aplicando a previsão do modelo sem muito cuidado, podemos dizer que ao jogar a moeda  $N$  vezes teremos que o número de resultados cara ( $N_{\text{cara}}$ ) seria

$$N_{\text{cara}} = P(\text{cara})N = \frac{N}{2} = N_{\text{coroa}}. \quad (8.2)$$

É claro que se  $N$  for pequeno muitas vezes teremos resultados bem diferentes desse. À medida que formos aumentando o número de jogadas chegaremos cada vez mais perto de ter  $N_{\text{cara}} = N_{\text{coroa}} = N/2$ . Se fôssemos determinar as probabilidades experimentalmente teríamos que ter o cuidado de repetir a experiência (jogar a moeda, no caso) um grande número de vezes, e assim teríamos uma definição experimental

$$P(\text{cara}) = \lim_{N \rightarrow \infty} \frac{N_{\text{cara}}}{N}. \quad (8.3)$$

Mas, quanto grande  $N$  deve ser? O maior possível. Veremos mais tarde que a pergunta correta é: Que erro estamos cometendo ao usar a definição (8.3) com  $N$  finito? Ou melhor, quanto o valor observado para  $N_{\text{cara}}$  é diferente de  $N/2$ ? E é claro, as respostas estarão relacionada com a precisão com que estamos medindo.

## 8.2 Distribuições

Chamamos de variável estocástica ou aleatória, a variável cujo valor só pode ser determinado através de uma experiência. Usaremos a partir de agora a seguinte notação: em letras maiúsculas teremos o nome da variável (ex:  $X$  é resultado da jogada da moeda), e em minúsculas, o seu valor (ex:  $x = 1$  para cara ou  $x = 0$  para coroa). Uma variável estocástica  $X$  é uma função que associa um número real a cada ponto do espaço de amostragem.

### Variáveis estocásticas discretas

Seja  $X$  uma variável estocástica em  $\Omega$  que pode tomar um número contável (finito ou infinito) de valores, ou seja,  $X(\Omega) = \{x_1, x_2, \dots\}$ . Sabendo a probabilidade para cada valor  $x_i$  podemos definir a **distribuição de probabilidade**,  $f(x_i) = P(x_i)$  satisfazendo as seguintes condições

$$f(x_i) \geq 0, \quad (8.4)$$

e

$$\sum_i f(x_i) = 1, \quad (8.5)$$

onde a soma é sobre todos os valores possíveis da variável  $X$ .

Vejamos como exemplo o dado.  $X$  é o número tirado, e  $x_1 = 1, \dots, x_6 = 6$ . Todos os valores tem probabilidade  $1/6$  de ocorrer, portanto  $f(x_i) = 1/6$ . Essa distribuição chama-se **distribuição uniforme**.

Voltando à moeda, a probabilidade de tirar  $N_1$  caras em  $N$  jogadas é dada pela distribuição binomial

$$P(N, N_1) = \frac{N!}{N_1!(N - N_1)!} p^{N_1} q^{N - N_1}, \quad (8.6)$$

onde  $p$  é a probabilidade de tirar cara em uma jogada, e  $q$  de tirar coroa. Para uma moeda simétrica,  $p = q = 0.5$ . A Fig 8.1 mostra o gráfico de (8.6) para  $N = 10$ .

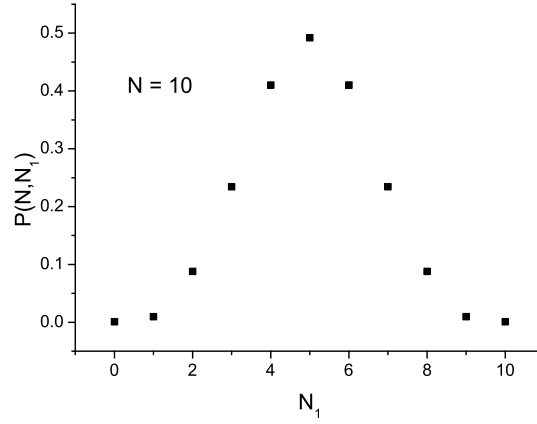


Figura 8.1: Probabilidade de se tirar  $N_1$  caras numa seqüências de 10 jogadas independentes.

A determinação de  $f(x_i)$  (que em geral não é possível), permite o conhecimento completo de um sistema. Em geral podemos apenas determinar alguns momentos da distribuição que são relacionados com observáveis que podem ser medidos. O  $n$ -ésimo momento de  $X$  é definido como

$$M_n = \langle X^n \rangle = \sum_i x_i^n f(x_i) . \quad (8.7)$$

Alguns momentos tem nomes especiais devido a sua freqüente utilização

**primeiro momento**  $\rightarrow M_1 = \langle X \rangle$  = média ou valor esperado

**segundo momento**  $\rightarrow M_2 = \langle X^2 \rangle$

O segundo momento em geral aparece combinado com o primeiro na forma  $M_2 - M_1^2 = \langle X^2 \rangle - \langle X \rangle^2 =$  variância de  $X$ .

## Variáveis estocásticas contínuas

Várias situações tem uma descrição mais adequada através de variáveis contínuas. Por exemplo, se estivermos considerando um número muito grande de jogadas, por exemplo 500, a distribuição binomial (8.6) pode ser aproximada pela função contínua [7]

$$P(N, x) = \frac{1}{\sqrt{2\pi Npq}} \exp \left[ -\frac{(x - Np)^2}{2Npq} \right] \quad (8.8)$$

A Fig. 8.2 mostra a comparação entre as descrições discreta e contínua para a distribuição binomial.

Fica sem sentido falar na probabilidade de ter  $x$  como resultado se  $X$  é uma variável contínua. Neste caso devemos definir um intervalo infinitesimal  $dx$  e definir  $dP(x)$  como a probabilidade de encontrar o resultado entre  $x$  e  $x + dx$ . Essa probabilidade depende, em princípio, de  $x$ , e também do tamanho de  $dx$ . Quanto maior for o intervalo considerado, maior será o valor numérico de  $dP(x)$  para um mesmo  $x$ . Neste caso é mais significativa a definição de **densidade de probabilidade**, da seguinte forma

$$\frac{dP(x)}{dx} = f_X(x), \quad (8.9)$$

ou seja,

$$dP(x) = P(x)dx = f_X(x)dx. \quad (8.10)$$

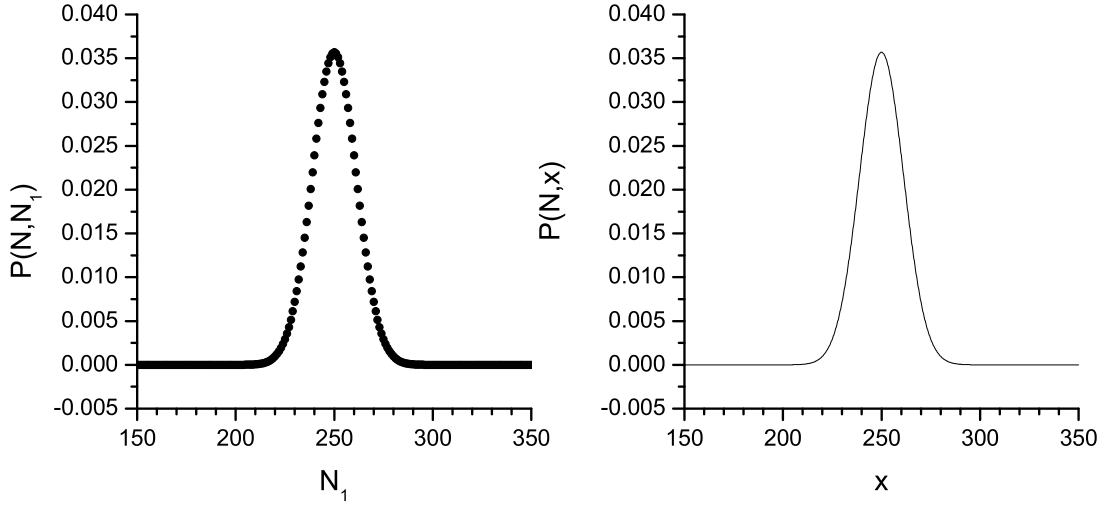


Figura 8.2: Comparação entre as distribuições dadas por (8.6) e (8.8) para  $N = 500$ ,  $p = q = 0.5$ .

A função  $f_X(x)$  define a **distribuição** da variável aleatória  $X$ .

Se queremos tratar de um intervalo não infinitesimal, por exemplo se queremos saber qual a probabilidade de ter  $x$  entre os valores  $a$  e  $b$ , temos

$$P(a \leq x \leq b) = \int_a^b f_X(x) dx. \quad (8.11)$$

A densidade de probabilidade,  $f_X(x)$ , é uma função contínua por partes satisfazendo

$$f_X(x) \geq 0 \quad (8.12)$$

e

$$\int_{\Omega} f_X(x) dx = 1, \quad (8.13)$$

onde  $\Omega$  é o espaço de amostragem.

Os momentos ficam definidos como

$$M_n = \langle X^n \rangle = \int dx x^n f_X(x). \quad (8.14)$$

## Histogramas × distribuições contínuas

Muitas vezes queremos relacionar um conjunto finito de valores de uma variável estocástica com uma determinada distribuição contínua. Isso pode ser feito através da análise do histograma obtido a partir dos valores medidos tomando-se os devidos cuidados para garantir a mesma normalização nas duas descrições.

A equação (8.10) nos diz que  $dP(x) = f_X(x)dx$  é a probabilidade de encontrar  $X$  entre  $x$  e  $x + dx$ , devidamente normalizada como em (8.13). Ao construirmos um histograma dividimos o intervalo de observação da variável em *bins* de largura  $b$  e contamos quantos dados caem em cada *bin*. A quantidade de dados classificados num determinado *bin* depende linearmente da largura deste, assim, ao finalizar a contagem teremos obtido  $\Delta n_i = n_i b$ , que é o número de dados contado no  $i$ -ésimo *bin* de largura  $b$ . A normalização deve ser tal que  $\sum_{i=1}^{N_b} \Delta n_i = N$ , onde  $N_b$  é o número de *bins* e  $N$  o número total de valores analisados. Podemos calcular a fração de dados que caíram em cada *bin* como  $\Delta F_i = \Delta n_i / N = (n_i / N)b$ . A normalização agora deve ser

$\sum_{i=1}^{N_b} \Delta F_i = 1$ . A lei dos números grandes nos garante que quando  $N \rightarrow \infty$ ,  $\Delta F_i \rightarrow P_i$ , que é a probabilidade de encontrar um valor no  $i$ -ésimo *bin*. Se desejamos obter (ou ajustar) a distribuição de probabilidade para esse conjunto de dados, devemos trabalhar com os valores de  $(\Delta F_i)/b$ , que serão a densidade de probabilidade de encontrar os valores nos *bins*.

### 8.3 Geração de números aleatórios

Num problema determinístico, o objetivo é conhecer a trajetória ou a evolução temporal de uma determinada variável. Se o sistema é aleatório, cada trajetória será diferente, mesmo partindo de uma única condição inicial. O que tem sentido é conhecer o comportamento médio, e entender como as flutuações afetam o sistema. Uma possibilidade é usar o computador para *simular* processos ou trajetórias, usando regras pré-determinadas, que em geral envolverão aleatoriedade em alguma forma. Essas regras formam o modelo, e saberemos se são adequadas *a posteriori*, ou seja, depois que calcularmos as médias e flutuações e compararmos com um sistema real. Um ingrediente fundamental é escolher a forma correta de aleatoriedade, ou seja escolher corretamente a distribuição utilizada. Na simulação computacional isso é feito sempre a partir de um **gerador de números aleatórios**. Todas as linguagens possuem um mecanismo, em geral uma função, para produzir uma sequência de números aleatórios, que na verdade são pseudo-aleatórios, com um período de repetição bastante longo. Um exemplo é o gerador linear congruente, que a partir de uma **semente**  $x_0$  gera uma sequência de números  $x_n$  de acordo com o mapa

$$x_{n+1} = (ax_n + b) \bmod 2^m, \quad (8.15)$$

A expressão  $\bmod 2^m$  significa que apenas os  $m$  primeiros bits são mantidos. Em geral  $m$  tem os valores 32 ou 64. As constantes  $a$  e  $b$  são escolhidas de forma a que a sequência  $x_1, x_2 \dots$  seja aleatória, obedecendo a uma distribuição uniforme entre 0 e  $2^m - 1$ . A razão pela qual as sequências de números geradas por (8.15) são aleatórias é o regime caótico, muito parecido com o obtido no caso do mapa logístico, adequadamente sintonizado pelos valores de  $a$  e  $b$  usados.

Veja um exemplo de uso da função disponível no C padrão ANSI no programa `unif.c` abaixo.

#### unif.c

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int s, i, N, nmax;

    printf("inicializando a sequencia...\n");
    printf("digite um numero inteiro e pressione enter\n\n");
    scanf("\n%d", &s);
    printf("informe o numero de sorteios\n\n");
    scanf("%d", &N);
    nmax = RAND_MAX; // valor maximo do numero gerado, definido em stdlib.h
    printf("\nserao sorteados %d inteiros entre 0 e %d:\n\n", N, nmax);
    srand(s); // inicializa a semente
    for(i = 1; i <= N; i++){
        printf("%d %d\n", i, rand());
    }
    return 1;
}
```

A função `rand` tem alguns problemas. Primeiro, o fato de gerar números em um intervalo que depende da máquina utilizada é desagradável, embora facilmente contornado pela divisão pelo valor máximo definido em `stdlib.h`. Sério mesmo é o fato de gerar números que se repetem cedo demais para aplicações científicas. Essa função padrão é duramente criticada pelo Numerical Recipes, que propõe várias opções de geradores

portáteis e de longo período. Vamos adotar um delas, a **ran2** listada abaixo. Trata-se de gerador de números de período maior que  $2 \times 10^{18}$ , retornando um número **double** uniformemente distribuído entre 0 e 1, excluindo os extremos. É chamada a partir de um inteiro negativo, o **idum**, usado para inicializar a sequência. **idum** não deve ser alterado entre chamadas sucessivas, para não reinicializar a geração de números.

#### ran2.c

```
/* note #undef's no fim do arquivo */
#define IM1 2147483563
#define IM2 2147483399
#define AM (1.0/IM1)
#define IMM1 (IM1-1)
#define IA1 40014
#define IA2 40692
#define IQ1 53668
#define IQ2 52774
#define IR1 12211
#define IR2 3791
#define NTAB 32
#define NDIV (1+IMM1/NTAB)
#define EPS 1.2e-7
#define RNMX (1.0-EPS)

double ran2(int *idum)
{
    int j;
    long k;
    static long idum2=123456789;
    static long iy=0;
    static long iv[NTAB];
    double temp;

    if (*idum <= 0) {
        if (-(*idum) < 1) *idum=1;
        else *idum = -(*idum);
        idum2=(*idum);
        for (j=NTAB+7;j>=0;j--) {
            k=(*idum)/IQ1;
            *idum=IA1*(*idum-k*IQ1)-k*IR1;
            if (*idum < 0) *idum += IM1;
            if (j < NTAB) iv[j] = *idum;
        }
        iy=iv[0];
    }
    k=(*idum)/IQ1;
    *idum=IA1*(*idum-k*IQ1)-k*IR1;
    if (*idum < 0) *idum += IM1;
    k=idum2/IQ2;
    idum2=IA2*(idum2-k*IQ2)-k*IR2;
    if (idum2 < 0) idum2 += IM2;
    j=iy/NDIV;
    iy=iv[j]-idum2;
    iv[j] = *idum;
    if (iy < 1) iy += IMM1;
```

```

        if ((temp=AM*iy) > RNMX) return RNMX;
        else return temp;
}
#undef IM1
#undef IM2
#undef AM
#undef IMM1
#undef IA1
#undef IA2
#undef IQ1
#undef IQ2
#undef IR1
#undef IR2
#undef NTAB
#undef NDIW
#undef EPS
#undef RNMX

```

Exemplo de uso correto :

```

#include "NR.h"
#include <stdio.h>
int main()
{
    int s, i, N;
    double x;

    printf("entre com o valor de um inteiro negativo para inicializar ran2\n\n");
    scanf("%d", &s);
    if (s>0){
        s = -s; // garante que e' negativo
        printf("\npedi um numero negativo e voce deu um positivo -> ja corrigido!\n");
    }
    printf("\nentre com o tamanho da sequencia\n\n");
    scanf("%d", &N);
    printf("\n%d doubles entre 0 e 1 serao gerados\n\n",N);

    for(i = 1; i <= N ; i++){
        x = ran2(&s);
        printf("  %d   %lf\n",i,x);
    }

    return 0;
}

```

Muitas vezes precisamos de números aleatórios que obedecem a uma distribuição que não a uniforme. Devemos obter essa distribuição a partir dos números gerados com a uniforme, o que pode ser feito de duas maneiras: pelo método da transformação, ou pelo da rejeição.

### Método da transformação

Suponha que a coleção de variáveis  $\{x_1, x_2, x_3 \dots\}$  seja distribuída de acordo com a função  $P_X(x)$ . Ou seja, a probabilidade de encontrar  $X$  com valor entre  $x$  e  $x + dx$  é  $P_X(x)dx$ . Seja  $y$  uma outra variável aleatória, tal que  $y = f(x)$ . A distribuição  $P_Y(y)$  pode ser obtida a partir de  $P_X(x)$  usando-se

$$|P_X(x)dx| = |P_Y(y)dy| \quad \text{ou} \quad P_Y(y) = P_X(x) \left| \frac{dx}{dy} \right| \quad (8.16)$$

Por exemplo, suponha que queremos gerar uma distribuição de Poisson, descrita por  $P_Y(y) = \exp(-y)$ , a partir da distribuição uniforme  $P_X(x) = C$ . Usando (8.16) temos

$$P_Y(y) = \exp(-y) = P_X(x) \left| \frac{dx}{dy} \right| = C \left| \frac{dx}{dy} \right| \Rightarrow y = -\ln x \quad (8.17)$$

Assim, usamos o gerador de números padrão para conseguir valores de  $x$  e calculamos  $y$  usando  $y = f(x) = -\ln x$ . É importante notar que em geral isso envolve manipular os números gerados de alguma forma, ajustando normalização e deslocamento.

## Método da rejeição

Nem sempre a determinação da relação entre  $y$  e  $x$  é possível. Nesses casos lançamos mão do método da rejeição. No que se segue vamos supor que  $x$  seja um número **real** aleatório, uniformemente distribuído entre 0 e 1. Queremos gerar números de acordo com uma distribuição  $P_Y(y)$ , definida no intervalo  $[y_{\min}, y_{\max}]$ , cujo valor máximo é 1, como mostra a figura 8.3. Isso significa que queremos examinar uma seqüência  $\{y_1, y_2, \dots\}$  e aceitar alguns valores, a aceitação se dando com probabilidade  $P_Y(y)$ . Assim, sorteamos um número  $y_1$  de uma seqüência uniforme entre  $y_{\min}$  e  $y_{\max}$ , usando o gerador disponível. Calculamos  $P_Y(y_1)$ . Geramos um outro número aleatório entre 0 e 1, que chamaremos de  $p_{\text{teste}}$ . O número  $y_1$  é aceito se  $P_Y(y_1) > p_{\text{teste}}$ . Continuamos o procedimento com o sorteio e teste de outro valor,  $y_2$ , etc. A seqüência dos números aceitos obedece a  $P_Y(y)$ .

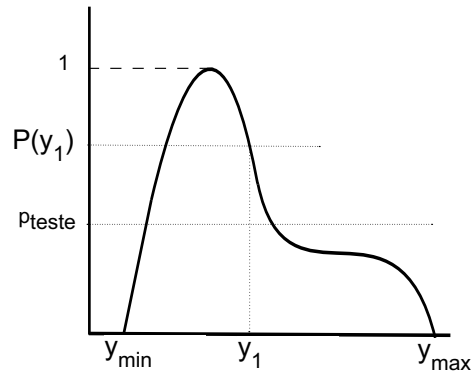


Figura 8.3: Método da rejeição para a determinação de  $P(y)$ . O número sorteado ( $y_1$ ) foi usado para o cálculo de  $P(Y_1)$ . Um segundo número aleatório,  $p_{\text{teste}}$  é sorteado e comparado a  $P(y_1)$ . Para a situação indicada,  $y_1$  foi aceito.

## Integração Monte Carlo

Suponha que desejamos calcular o valor da integral definida

$$I = \int_a^b g(x) dx. \quad (8.18)$$

Seja  $P_V(v)$  a densidade de probabilidade associada à variável aleatória  $V$  definida no intervalo  $[a, b]$ . Definimos agora a variável aleatória  $H$  como

$$H = f(x) = \frac{g(x)}{P_V(x)}. \quad (8.19)$$



O valor médio de  $H$  está relacionado com o valor da integral que desejamos calcular como

$$\langle H \rangle = \int_a^b f(x)P_V(x)dx = \int_a^b \frac{g(x)}{P_V(x)}P_V(x)dx = I \quad (8.20)$$

Na prática usaremos a média aritmética de um número finito de valores de  $H$  como valor da integral, ou seja, usaremos

$$I \approx \frac{1}{N} \sum_{j=1}^N \frac{g(v_j)}{P_V(v_j)}. \quad (8.21)$$

O erro que estaremos comentando pode ser calculado pela variância com relação ao *valor verdadeiro*  $I$ , como

$$\sigma_H^2 = \langle H^2 \rangle - I^2 = \int_a^b \frac{g^2(x)}{P_V(x)}dx - I^2. \quad (8.22)$$

Pode-se mostrar que a variância é minimizada se escolhemos  $P_V(v)$  proporcional a  $|g(x)|$ , mas pode ser bastante complicado gerar as variáveis aleatórias com essa distribuição. Em geral basta usar  $g(x)$  como um guia para a escolha de  $P_V(v)$ , lembrando que esta deve estar normalizada no intervalo  $[a, b]$ . Na verdade não vale a pena usar o método Monte Carlo para o cálculo de integrais como (8.18), mas quando se trata de uma integral múltipla é uma excelente opção.

Vejamos um exemplo. Seja  $g(x) = \sin(x)$ ,  $a = 0$  e  $b = \pi/2$ . Nesse caso o valor da integral é conhecido,  $I = 1$ . Vamos ver como calcular essa integral pelo método Monte Carlo usando duas opções para  $P_V(v)$ .

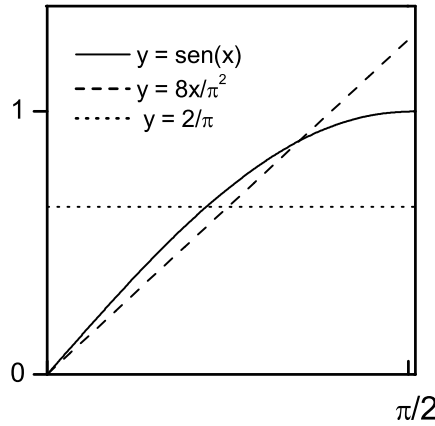


Figura 8.4: Comparação entre distribuições para o cálculo da integral  $I = \int_0^{\pi/2} \sin(x)dx$ .

Seja  $P_V(v)$  a distribuição uniforme no intervalo  $[0, \pi/2]$ :  $P_V(v) = 2/\pi$ . Se  $r$  é um número aleatório entre 0 e 1, gerado por exemplo por `ran2`,  $v$  pode ser obtido como

$$v = \frac{\pi}{2}r. \quad (8.23)$$

Sorteamos  $N = 10$  valores para  $r$ , calculamos os 10 valores de  $v$  correspondentes, e os 10 valores de  $\sin(v)$ , ou seja,

$$I_1 = \frac{\pi}{2N} \sum_{j=1}^N \sin(v_j) \quad (8.24)$$

A tabela abaixo mostra um exemplo desse cálculo.

$j$	1	2	3	4	5	6	7	8	9	10
$r_j$	0.865	0.159	0.079	0.566	0.155	0.664	0.345	0.655	0.812	0.332
$v_j$	1.359	0.250	0.124	0.889	0.243	1.043	0.542	1.029	1.275	0.521
$\text{sen}(v_j)$	0.978	0.247	0.124	0.776	0.241	0.864	0.516	0.857	0.957	0.498

O valor de  $I$  a partir dos 10 sorteios é então

$$I_1 = 0.952. \quad (8.25)$$

Podemos melhorar muito a acurácia no cálculo de  $I$  com o uso da distribuição  $P_V(v) = 8v/\pi^2$ , indicada na figura 8.4. Nosso ponto de partida é um número aleatório  $r$ , entre 0 e 1. Para gerar  $v$  obedecendo a  $P_V(v)$ , a partir de  $r$ , usamos o método da transformação.

$$\left| \frac{8x}{\pi^2} dx \right| = |dy| \quad (8.26)$$

ou seja,

$$\int_0^v \frac{8x}{\pi^2} dx = \int_0^r dy, \quad (8.27)$$

dando

$$v = \frac{\pi}{2} \sqrt{r}. \quad (8.28)$$

A variável  $v$  obedece à distribuição  $P_V(v) = 8v/\pi^2$ . Com essa escolha, o valor de  $I$  pode ser calculado com  $N$  sorteios como

$$I_2 = \frac{\pi^2}{8N} \sum_{j=1}^N \frac{\text{sen}(v_j)}{v_j}. \quad (8.29)$$

Prosseguimos então sorteando  $r$ , calculando  $v$  através de (8.28), e calculando  $\frac{\pi^2 \text{sen} v}{8v}$ . A tabela abaixo mostra esse cálculo para 10 sorteios.

$j$	1	2	3	4	5	6	7	8	9	10
$r_j$	0.865	0.159	0.079	0.566	0.155	0.664	0.345	0.655	0.812	0.332
$v_j$	1.359	0.250	0.124	0.889	0.243	1.043	0.542	1.029	1.275	0.521
$\frac{\text{sen}(v_j)}{v_j}$	0.978	0.247	0.124	0.776	0.241	0.864	0.516	0.857	0.957	0.498

O cálculo de  $I$  agora é,

$$I_2 = 1.016. \quad (8.30)$$

## 8.4 Exercícios

1. Se um determinado evento aleatório é descrito pela densidade de probabilidade  $f(x)$ , a probabilidade de que o resultado associado a esse evento esteja entre  $a$  e  $b$  é dada por

$$P(a < x < b) = \int_a^b f(x) dx.$$

Quando o intervalo de integração compreende todos os possíveis valores de  $s$ , o valor da probabilidade deve ser 1, o que define a normalização de  $f(x)$ .

Use o método de Simpson para calcular essa probabilidade nas situações indicadas abaixo, e para verificar a normalização numericamente.

(a) Distribuição Gaussiana, definida como

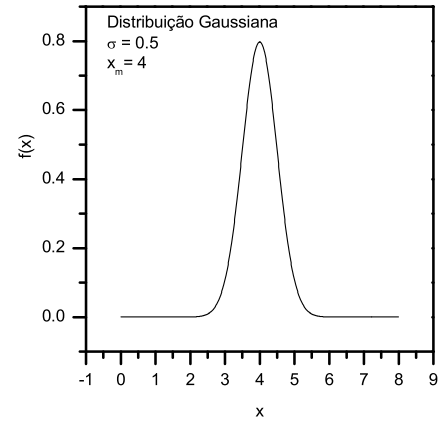
$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp \left[ -\frac{(x - x_m)^2}{2\sigma^2} \right],$$

para  $\sigma > 0$ , e  $x$  entre  $-\infty$  e  $+\infty$ .

i.  $a = x_m - \sigma$  e  $b = x_m + \sigma$

ii.  $a = x_m - 2\sigma$  e  $b = x_m + 2\sigma$

iii.  $a = x_m - 2\sigma$  e  $b = x_m + 2\sigma$



(b) Distribuição Log-normal, definida como

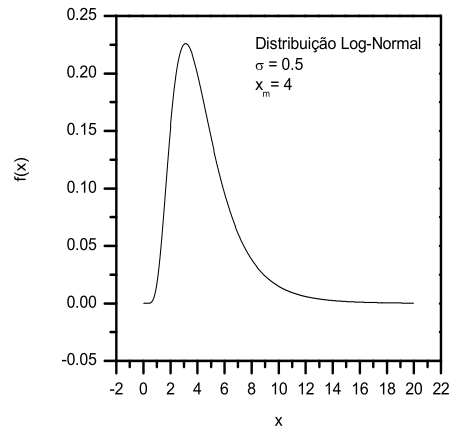
$$f(x) = \frac{1}{x\sigma\sqrt{2\pi}} \exp \left[ -\frac{[\ln(x/x_m)]^2}{2\sigma^2} \right]$$

para  $\sigma > 0$ , e  $x$  entre 0 e  $+\infty$ .

i.  $a = x_m / \exp(\sigma)$  e  $b = \infty$

ii.  $a = x_m / \exp(\sigma)$  e  $b = x_m \exp(\sigma)$

iii.  $a = x_m / [\exp(\sigma)]^2$  e  $b = x_m [\exp(\sigma)]^2$



2. Use o método da rejeição para gerar uma sequência de números distribuído de acordo com

(a) a distribuição gaussiana

$$P_Y(y) = \frac{1}{\sigma\sqrt{2\pi}} \exp \left[ -(y - y_m)^2 / 2\sigma^2 \right],$$

(b) a distribuição log-normal

$$P_X(x) = \frac{1}{x\sigma\sqrt{2\pi}} \exp \left[ -\frac{[\ln(x/x_m)]^2}{2\sigma^2} \right]$$

usando o gerador de números aleatórios **ran2** do Numerical Recipes.

Experimente diversos valores de  $\sigma$  e  $y_m$ , sempre construindo o histograma e examinando o gráfico. Responda às perguntas: O acontece com a distribuição quando  $y_m$  tem seu valor aumentado? E  $\sigma$ ? Qual o significado desses parâmetros para a forma da curva?

- Sabemos que, se  $x$  é uma variável aleatória regida pela distribuição  $P_x$ , e  $y = f(x)$ , então  $|P_X(x)dx| = |P_Y(y)dy|$ . Use esta informação para mostrar que se  $y$  obedece à distribuição gaussiana, e  $y = \ln(x)$ , então  $x$  obedece à distribuição log-normal.
- Use o método da transformação para gerar números aleatórios que obedeçam à distribuição  $P_X(x) = 8x/\pi^2$ .
- Suponha que tenhamos  $N$  partículas com velocidades de módulo  $v$  e direções aleatórias, de acordo com uma distribuição uniforme. Para um valor de  $N$  razoavelmente grande, a aleatoriedade das direções

garante que

$$\langle v \rangle = \frac{1}{N} \sum_{i=1}^N \vec{v}_i = 0 \quad (8.31)$$

e que

$$\langle v_a^2 \rangle = \frac{1}{D} v^2, \quad (8.32)$$

onde  $a = x, y, z$  e  $D$  é a dimensão espacial.

(a) Verifique as expressões acima para um sistema bidimensional ( $D = 2$ ), supondo  $v = 1$ . Use o seguinte esquema:

- Gere  $N$  ângulos  $\varphi$  uniformemente distribuídos entre 0 e  $2\pi$ .
- Use-os para calcular  $v_x$  e  $v_y$  para cada partícula.
- Calcule as médias pedidas.

(b) Estenda para o caso tridimensional ( $D = 3$ ). O ângulo azimutal  $\varphi$  pode ser sorteado como no caso bidimensional, mas você deve ter cuidado com o sorteio do ângulo polar  $\theta$ . A integral sobre a superfície de uma esfera é

$$\int_0^{2\pi} d\varphi \int_{-1}^1 d(\cos \theta) = 4\pi \quad (8.33)$$

Assim, para ângulos uniformemente distribuídos, a probabilidade de ter uma direção com ângulo polar entre  $\theta$  e  $\theta + d\theta$ , e azimutal entre  $\varphi$  e  $\varphi + d\varphi$  é

$$\frac{1}{2} d(\cos \theta) \frac{1}{2\pi} d\varphi. \quad (8.34)$$

Em resumo, o esquema é

- Gere  $N$  ângulos  $\varphi$  uniformemente distribuídos entre 0 e  $2\pi$ .
- Gere  $N$  valores de  $\cos \theta$  uniformemente distribuídos entre  $-1$  e  $1$ . Para cada valor use a função `acos` da biblioteca matemática para calcular  $\theta$ .
- Use-os para calcular  $v_x$ ,  $v_y$ , e  $v_z$  para cada partícula.
- Calcule as médias pedidas.

6. Use a integração Monte Carlo para calcular a integral elíptica

$$K(m) = \int_0^{\pi/2} \frac{1}{\sqrt{1 - m \sin^2 \theta}} d\theta$$

para  $m = 0.2$ ,

- (a) usando uma distribuição uniforme  $P_X = 2x/\pi$
- (b) usando a distribuição  $P_X = 8x/\pi^2$ . O valor tabelado em *Handbook of Mathematical Functions*, editado por Abramowitz e Stegun é 1.659623598610528. Escolha o número de sorteios de forma a ter concordância de 3 casas decimais.

# Bibliografia

- [1] Intel Corporation. *IA-32 Intel Architecture Software Developers Manual*. <http://support.intel.com/design/Pentium4/documentation.htm>, 2004.
- [2] Winn L. Rosch. *The Winn L. Rosch Hardware Bible (6th Edition)*. Que, 2003.
- [3] <http://computer.howstuffworks.com/computer-memory.htm>.
- [4] [http://arstechnica.com/paedia/r/ram\\_guide/ram\\_guide.part1-1.html](http://arstechnica.com/paedia/r/ram_guide/ram_guide.part1-1.html).
- [5] H. Moysés Nussenzveig. *Curso de Física Básica 2*. Edgard Blucher, 1983.
- [6] Kittel e Kroemer. *Thermal Physics*. W.H. Freeman and Company, 1980.
- [7] S. R. A. Salinas. *Introdução à Física Estatística*. Edusp, 1997.
- [8] W. H. Press, B.P. Flannery, S. A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C. The Art of Scientific Computing*. Cambridge, 1988.
- [9] K. R. Symon. *Mechanics*. Addison-Wesley, 1972.
- [10] P. Collet and J-P. Eckmann. *Iterated Maps on the Interval as Dynamical Systems*. Birkhauser, 1980.
- [11] E. A. Jackson. *Perspectives of nonlinear dynamics*, volume 1. Cambridge, 1989.
- [12] H. Moysés Nussenzveig, editor. *Complexidade e Caos*. Editora UFRJ/COPEA, 1999.
- [13] P. M. Castro de Oliveira. Sistemas complexos. *Ciência Hoje*, 16:35, 1994.