Trabalho 1 - Métodos Probabilísticos

Modelação de Sistemas Complexos

Universidade de Aveiro

João Maria Machado, 89132

31 de março de 2022

1 Introdução

Métodos probabilísticos são a principal ferramenta de estudo de sistemas complexos. De modo geral na natureza, quando se estuda um sistema com profundidade suficiente este não se comporta de maneira determinística, logo a compreensão de métodos estatísticos é essencial no âmbito na compreensão e estudo de sistemas complexos.

Neste relatório serão resolvidos e discutidos os problemas propostos relativos ao tema. Para cada problema numa primeira etapa serão introduzidos de alguns conceitos teóricos fundamentais, numa segunda etapa serão estudados os algoritmos desenvolvidos no *matlab* para responder às questões propostas. Finalmente serão expostos os resultados obtidos bem como o seu enquadramento nos valores, *a priori*, esperados.

2 Probabilidade

No decorrer deste relatório serão exploradas variáveis aleatórias. Por definição, uma variável aleatória que depende de fatores aleatórios. Um dos primeiros conceitos estudados foi a determinação do valor médio e da variância de uma variável aleatória.

Uma forma de efetuar este estudo é considerando uma variável aleatória x que toma valores:

$$X = 1, 2, 3, ..M \tag{1}$$

de forma aleatória e uniforme, isto é, todos os valores têm probabilidade $P(x_i) = \frac{1}{M}$ de se verificarem. A média de uma variável pode ser determinada, de um modo geral, da seguinte forma:

$$\langle n \rangle = \sum_{n=0}^{\infty} nP(n) \tag{2}$$

Aplicando esta equação a X, obtém-se:

$$\langle x \rangle = \sum_{i=1}^{M} x_i P(x_i) \tag{3}$$

Como $P(x_i)$ é constante pode aplicar-se a seguinte simplificação:

$$\langle x \rangle = \frac{1}{M} \sum_{i=1}^{M} x_i. \tag{4}$$

Resolvendo a série geométrica no somatório obtém-se que:

$$\langle x \rangle = \frac{M+1}{2} \tag{5}$$

Analogamente é possível determinar a variância desta variável aleatória. A variância é dada por:

$$\sigma^2 = \langle x^2 \rangle - \langle x \rangle^2 \tag{6}$$

Substituindo, obtém-se que:

$$\sigma^2 = \frac{1}{M}\sum_{i=1}^{M} x_i^2 - \left(\frac{1}{M}\sum_{i=1}^{M} x_i\right) \tag{7}$$

Resolvendo as séries geométricas, obtém-se:

$$\sigma^2 = \frac{(x+1)(2x+1)}{6} - \frac{(x+1)^2}{4} \tag{8}$$

Simplificando, obtém-se:

$$\sigma^2 = \frac{x^2-1}{12} \tag{9}$$

Uma forma de validar estas deduções, é como por exemplo, verificar que $P(X < 60) = 0.6$. Como X é uma variável discreta pode-se denotar $P(X < 60)$ da seguinte forma:

$$P(X < 60) = P(X = 1) + P(X = 2) + P(X = 3) + \ldots P(X = 60) \tag{10}$$

Como se trata de uma distribuição de probabilidade uniforme (todos os acontecimentos são equiprováveis) pode escrever-se, como por exemplo:

$$P(X < 60) = 60 \times P(X = 1) \tag{11}$$

Como $P(x_i) = \frac{1}{M}$ e $M = 100$ temos que $P(x_i) = \frac{1}{100}$, logo:

$$P(X < 60) = 60 \times \frac{1}{100} = 0.6 \tag{12}$$

Estes conceitos foram validados numericamente com um auxílio de programa em *matlab*. Neste ficheiro numa primeira etapa foram gerados N números aleatórios entre 1 e 100. A figura 1 contém o trecho de código responsável pela geração de números aleatórios.

```
for i=1:N_val
    x(i)=randi(M);
end
```

Figura 1: Geração de Números aleatórios

Em seguida computaram-se os valores médios desta distribuição (quer analíticos, quer numéricos). A figura 2 contém o trecho de código por este processo:

```
%media
med=sum(x)/N_val;

med_anal=(M+1)/(2);

med_conv(j)=abs(med-med_anal);
```

Figura 2: Cálculo dos valores médios

Analogamente, repete-se o processo para o cálculo das variâncias e probabilidades. As figuras 5 e 3 contém os trechos de código utilizados neste processo.

```
%variancia

for i=1:N_val
    dif(i)=(x(i)-med)^2;
end

sigma=(1/N_val)*sum(dif);
sigma_anal=(M^2-1)/(12);
sigma_conv(j)=abs(sigma-sigma_anal);
```

Figura 3: Cálculo dos valores da variância

```
%Determinar p

y=x(x<60);
p=length(y)/N_val;
p_anal=0.6;
p_con(j)=abs(p-p_anal);
```

Figura 4: Cálculo dos valores da probabilidade

Finalmente, após todos os cálculos numéricos **verificou-se a lei dos grandes números**. Isto é, quando sé aumenta N os valores computados aproximam-se dos calores analíticos.

Resta agora concluir o último problema proposto nesta secção. Isto é, provar que dadas duas variáveis aleatórias discretas x e y (ambas definidas com recurso à função do *matlab rand(1)*) pretende-se mostrar que: uma terceira variável aleatória z definida por $z = x \times y$ tem a sua média definida por $\langle z \rangle = \langle x \rangle \times \langle y \rangle$.

Para esse efeito produziu-se o seguinte algoritmo de matlab, conforme descrito na imagem **??**

```
for i=1:length(N_val)

    N=N_val(i);

    x=rand(1,N);
    y=rand(1,N);
    z=x.*y;

    med_x=(sum(x)/N);
    med_y=(sum(y)/N);
    med_z=(sum(z)/N);


    med_z_anal=med_x.*med_y;

    conv(i)=abs(med_z_anal-med_z);

end
```

Figura 5: Algoritmo elaborado

Como se pode verificar, à medida que se aumenta a amostragem de x e y, os valores médios calculados indiretamente convergem para o resultado teórico esperado.

3 Densidade de Probabilidade

Para estudar a função densidade de probabilidade de uma determinada variável aleatória no *matlab*, em que a variável é gerada com a função *rand* no intervalo 0 e 1, gerou-se o histograma da variável aleatória em causa. A figura 6 contém o trecho de código utilizado para a geração do histograma em causa. Importa mencionar que quando se considera a forma de histograma (isto é, o seu contorno) este acaba por corresponder à função de distribuição de probabilidade, esta estratégia é empregue meramente para facilitar a questão da discretização da variável aleatória.

```
for k=1:length(y)-1

    lower=y(k);
    upper=y(k+1);
    p(k,N_vals)=length(x(x<=upper & x>=lower));
    bin_center(k)=(upper+lower)/2;


end


p(:,N_vals)=p(:,N_vals)/(sum(p(:,N_vals))*(dy));

m(:,N_vals)=sum(bin_center*p(:,N_vals)*dy);

sig(:,N_vals)=sum((bin_center(:)-m(:,N_vals)).^(2).*p(:,N_vals))*dy;
```

Figura 6: Cálculo da Função de distribuição de probabilidade

A figura 8 contém as funções obtidas para

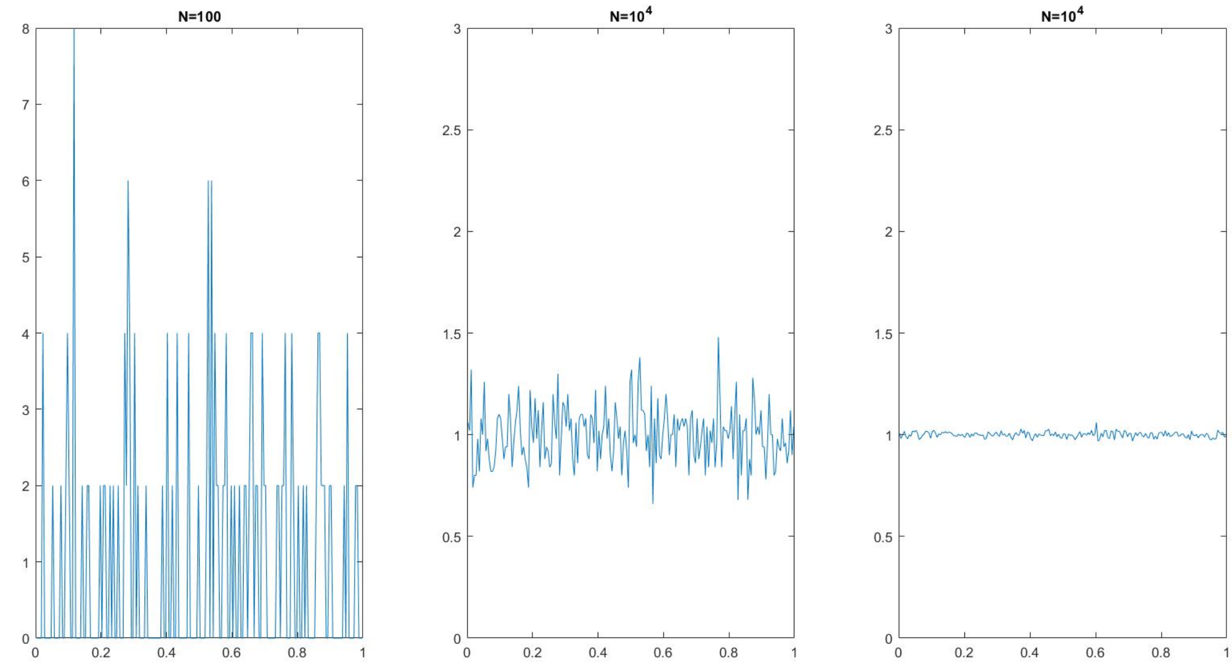


Figura 7: Funções obtidas

Um outro aspeto que importa mencionar é que também foram computados os valores da média e da variância e que também se observou uma convergência para o valor teórico esperado com o aumento de N.

Finalmente este processo foi repetido considerando uma nova variável aleatória $y = \sqrt{rand(1)}$, de onde se obtiveram as seguintes funções:

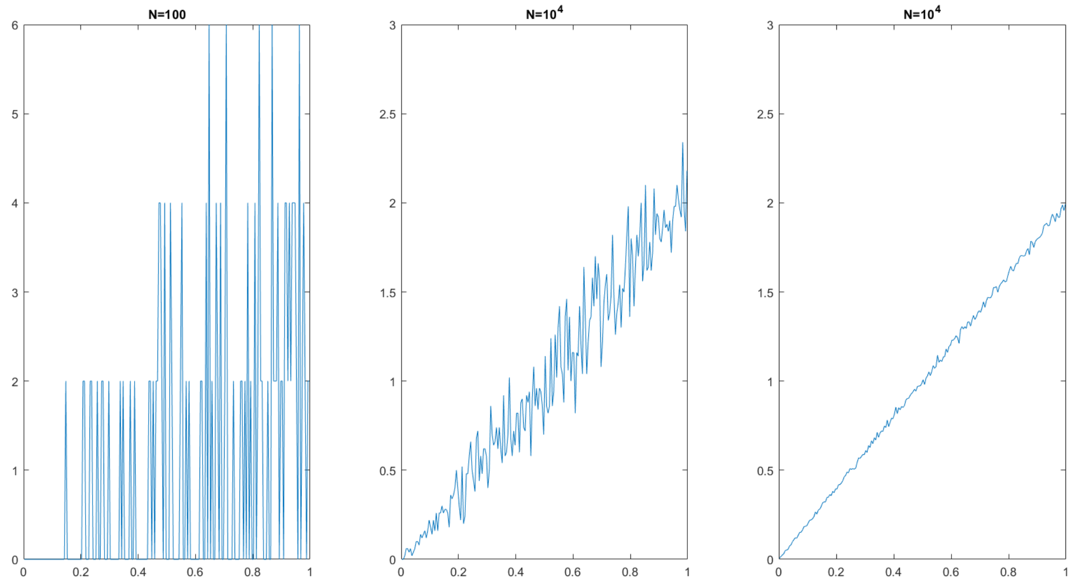


Figura 8: Funções obtidas

Novamente, verifica-se que à medida que há um aumento de N, há uma convergência para os valores teóricos. Como se pode verificar de um gráfico para o outro ocorre uma transformação de variáveis que se manifesta num crescimento (não local) da função de distribuição de probabilidade.

4 Teorema do Limite Central

Para estudar o teorema do limite central primeiro começou-se por gerar n números aleatórios x_i com média $\langle x \rangle$ e variância σ^2, usando, como por exemplo, a função *rand* do *matlab*. A partir desta distribuição define-se o número aleatorio Y em que:

$$Y = \frac{1}{n} \sum_{i=1}^{2} x_i \tag{13}$$

O objetivo consiste em dado Y, encontrar a função de distribuição $P(Y)$ mostrando que o valore de Y é igual a $\langle x \rangle$, calcular a variância de Y, dada por:

$$\Lambda^2 = \langle (Y - \langle Y \rangle)^2 \rangle \tag{14}$$

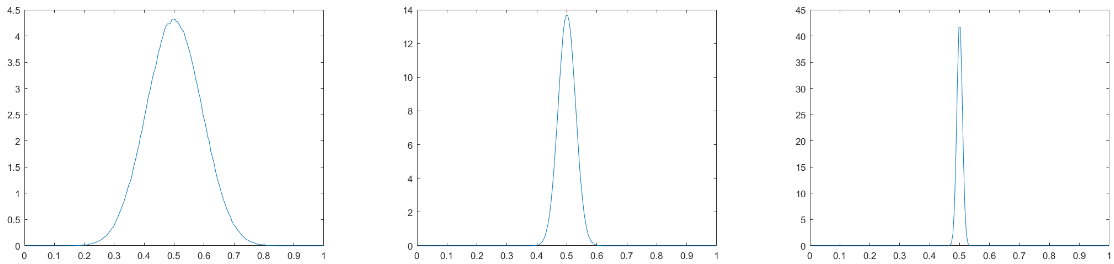
e mostrar que Λ tende para $\frac{\sigma^2}{n}$ quando o número de amostras de Y vai para infinito.

Para tal efeito foi implementado o algoritmo fornecido em *matlab*. A figura 9 contém o código elaborado para a implementação deste algoritmo

```
for j=1:N
    x=rand(1,n(i));
    Y_med(j)=(1/n(i))*sum(x);
end

[M,vert]=histcounts(Y_med,k);
P=M/(N*dy);

norm=sum(P*dy);

figure(i)
plot(y_centre_bins(1:end-1),P)


y_mean(i)=sum(vert(1:end-1).*P*dy);
y_sig(i)=sum(dy*P.*(vert(1:end-1)-y_mean(i)).^2);


x_med(i)=(1/N)*sum((1/n(i))*sum(x));
x_sig(i)=(1/N)*sum((1/n(i))*sum((x-x_med(i)).^2));
```

Figura 9: Implementação do Algoritmo

A partir deste algoritmo foi possível obter as seguintes funções de distribuição de probabilidade:



((a)) ((b)) ((c))

Figura 10: Funções de distribuição de probabilidade com o aumento de N (isto é, do tamanho da amostra)

Como se pode verificar à medida que se aumenta o tamanho da amostra a distribuição aproxima-se cada vez mais de uma distribuição normal ideal. Um outro aspeto que também se verifica com esta variação da amostra que quando esta aumenta Λ tende, de facto, para $\frac{\sigma^2}{n}$, verificando o teorema do limite central.

5 Lançamento de Bolas

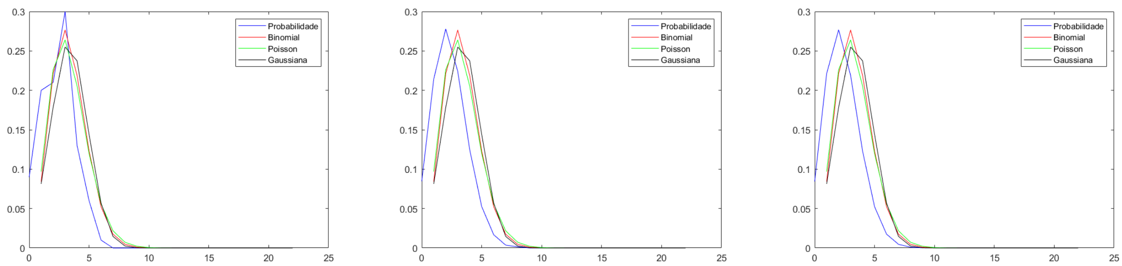
Nesta secção do relatório, pretendem-se estudar diferentes categorias de distribuição de probabilidade. Para tal efeito considera-se o seguinte problema: existem M caixas e N bolas. Atiram-se as bolas para as caixas. As N bolas são distribuídas pelas caixas de forma aleatoriamente uniforme. Desse modo, quer-se determinar a probabilidade de encontrar n bolas numa determinada caixa.

Para tal efeito seguiu-se e implementou-se o algoritmo fornecido em *matlab*. A figura 11 contém o trecho de código que implementa o algoritmo fornecido.

```
for i=1:length(k)
    for j=1:k(i)
        x=randi(M,[N,1]);
        n(j)=numel(x(x==3));

    end
    p=1/M;
    for h=0:N
        Ntr=sum((n==h)==1); %dupla indexação lógica
        P(h+1)=Ntr/k(i);
        Bin(h+1)=nchoosek(N,h)*(p^h)*(1-p)^(N-h); %Binomial
        Poi(h+1)=exp(-N*p)*((N*p).^h)/factorial(h); %Poisson
        Gau(h+1)=(1/sqrt(2*pi*N*p))*exp(-((h-N*p)^2)/(2*N*p)); %Gaussiana


    end
```

Figura 11: Implementação do Algoritmo

Através deste algoritmo gerou-se a seguintes funções de distribuição de probabilidades:



((a)) ((b)) ((c))

Figura 12: Funções de distribuição de probabilidade com o aumento de N. Neste caso mediu-se a probabilidade da bola cair na caixa 3

Como se pode verificar, à medida que se aumenta a amostragem as distribuições estatísticas aproximam cada vez melhor o valor da probabilidade real computado.

6 Conclusão

Neste trabalho foram estudadas variáveis aleatórias e distribuições estatísticas através da resolução de problemas práticos. Em todas as etapas os resultados numéricos obtidos convergiram para os resultados teóricos esperados, logo o autor considera que houve sucesso na realização deste relatório (e dos algoritmos implementados).

Uma possível maneira de melhor a qualidade dos resultados obtidos, isto é, obter uma maior convergência com o mesmo número de números aleatórios gerados, será, por exemplo, estudar mais atentamente o gerador de números pseudo-aleatórios do *matlab* e tentar aproximar o comportamento deste a um verdadeiro gerado de números aleatórios.