

# Universidade de Aveiro

**MPEI 2020/21**

**1º guião para avaliação**

**Joaquim Andrade nº93432**

**Francisco Silva nº93400**



## Guião 2 Parte 3 -- Avaliação

1. A) Neste exercício criámos, com uma função for que percorre o número de brinquedos em cada caixa, criando para cada brinquedo três números entre 0 e 1, e cada um desses vai ser comparado com uma das probabilidades de existir um erro do processo do brinquedo. Caso um ou mais dos números seja inferior à uma das probabilidades de se estragar, adicionamos um objeto à variável estragados.

No fim de verificar a caixa, caso o número de objetos estragados seja maior que um a condição A está verificada.

Repetimos o processo acima N (1e6 vezes), e dividimos a quantidade de vezes em que a foi verificada pelo mesmo.

Após observarmos uma oscilação baixa de resultados obtivemos, por simulação, que a probabilidade de A é aproximadamente 0.1274 .

```

N= 1e6;                                %numero de experiencias
p1=0.002;                               %probabilidade de estragar peça
p2=0.005;                               %probabilidade de estragar peca2
pmontagem= 0.01;                        %probabilidade de montagem
n=8;                                    %numero de brinquedos na caixa
j=0;                                    %variavel
estraga=0;                              %variavel2
for f=1: N
    for i=1 :n
        if (rand <p1)|| (rand <p2)|| (rand <pmontagem) %se algum dos
objetos estiver estragado ou o processo de montagem correr mal entra
no ciclo
            estraga= estraga+1;          %adiciona um aos estragados
        end
    end
    if estraga>=1
        j=j+1;                          %número de vezes em que a condição A se verifica
    end
    estraga=0;                           %inicializa o numero de
objeto de estragados a 0 para a próxima experiência
end
probsimulacao= j/N;                     %prob simulação

```

b) Com um código baseado na primeira alínea, caso usámos um número rand e cada vez que este era menor que a percentagem de erro em processo de montagem incrementámos o número de estragados.

Caso exista pelo menos um estragado na caixa, incrementamos uma nova variável que conta o número de brinquedos estragados cada vez que a caixa satisfaz a condição A (variável l).

Repetimos o processo até a oscilação de resultados ser menor. Por fim dividimos l por o número de vezes que a caixa teve pelo menos 1 objeto estragado e obtemos a média de 1,0353 brinquedos estragados por caixa.

```

N= 1e6;                                %numero de experiencias
pmontagem= 0.01;                       %probabilidade de montagem
n=8;                                    %numero de brinquedos na caixa
j=0;                                    %variavel
l=0;                                    %variavel2
estraga=0;
for f=1: N
    for i=1 :n
        if (rand<pmontagem) %se algum dos objetos estiver estragado
            ou o processo de montagem correr mal entra no ciclo
                estraga= estraga+1;          %adiciona um aos
            estragados
        end
        end
        if estraga>=1
            j=j+1;                          %número de vezes em que a
            condição A se verifica
            l=l+estraga;                    %número total de
            brinquedos estragados de todas as vezes que a condição A se verifica
        end
        estraga=0;                          %inicializa o numero de
        objeto de estragados a 0 para a próxima experiência
    end
nmedtoys= l/j;                           %prob simulação

```

2.A) Esta simulação foi praticamente igual á simulação executada na primeira alínea do exercício 1.A) alterando-o para incrementar apenas quando não existisse nenhum brinquedo estragado na caixa.

Repetidas vezes suficientes o experimento obtivemos 0.8724 como a probabilidade de não existirem brinquedos estragados numa caixa de 8 brinquedos.

```

N= 1e6;                                %numero de experiencias
p1=0.002;                              %probabilidade de estragar peça
p2=0.005;                              %probabilidade de estragar peca2
pmontagem= 0.01;                       %probabilidade de montagem
n=8;                                   %numero de brinquedos na caixa
j=0;                                   %variavel
estragea=0;                            %varivael2
for f=1: N
    for i=1 :n
        if (rand<p1)|| (rand<p2)|| (rand<pmontagem) %se algum dos
objetos estiver estragado ou o processo de montagem correr mal entra
no ciclo
            estragea= estragea+1;          %adiciona um aos
estrageados
        end
    end
    if estragea==0
        j=j+1;                          %número de vezes em que a
condição A se verifica
    end
    estragea=0;                          %inicializa o numero de
objeto de estragados a 0 para a próxima experiência
end
probsimulacao= j/N;                    %prob simulação

```

b)O valor teórico do evento B obtido é 0.8274, que é exatamente igual ao valor obtido no exercício em cima. Concluo então que a probabilidade de simulação é igual á teórica.

```

p= (1-0.01)*(1-0.002)*(1-0.005);
n=8;
k=8;
probteo= factorial(n)/(factorial(n-k)*factorial(k))*p^k*(1-p)^(n-k);

```

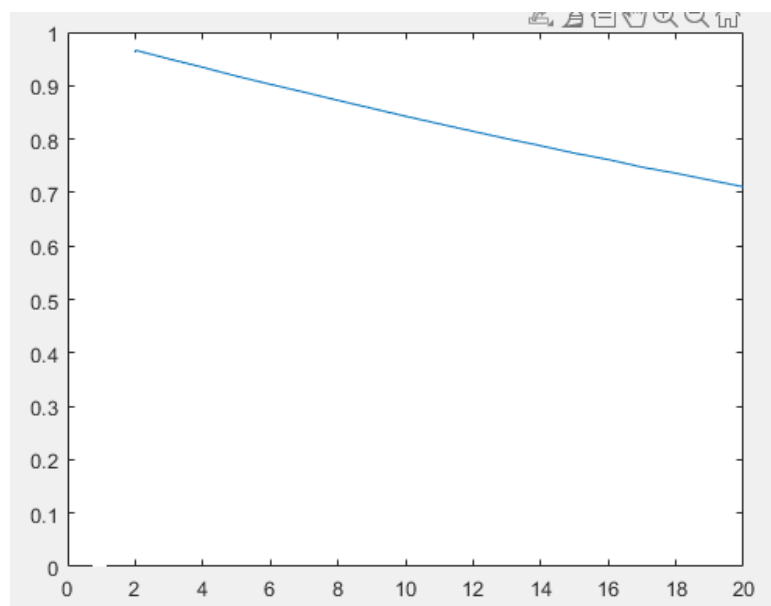
C) Para executar esta função usámos o código escrito na pergunta 2.A) e adicionámos um ciclo alterando o número de brinquedos cada vez que a simulação de um era executada, guardando os resultados num vetor.

Por fim colocámos o vetor na função plot().

```

N= 1e6;                                %numero de experiencias
p1=0.002;                              %probabilidade de estragar peça
p2=0.005;                              %probabilidade de estragar peca2
pmontagem= 0.01;                       %probabilidade de montagem
j=0;                                   %variavel
estragea=0;                            %varivael2
probsimulacao= zeros(20, 1);
for n=2: 20
    for f=1: N
        for i=1 :n
            if (rand <p1)|| (rand<p2)|| (rand <pmontagem) %se algum
dos objetos estiver estragado ou o processo de montagem correr mal
entra no ciclo
                estragea= estragea+1;          %adiciona um aos
estrageados
            end
        end
        if estragea==0
            j=j+1;                            %número de vezes em
que a condição A se verifica
        end
        estragea=0;                          %inicializa o numero
de objeto de estrageados a 0 para a próxima experiência
    end
    probsimulacao(n)= j/N;                  %prob simulação
    j=0;
end
plot(probsimulacao);

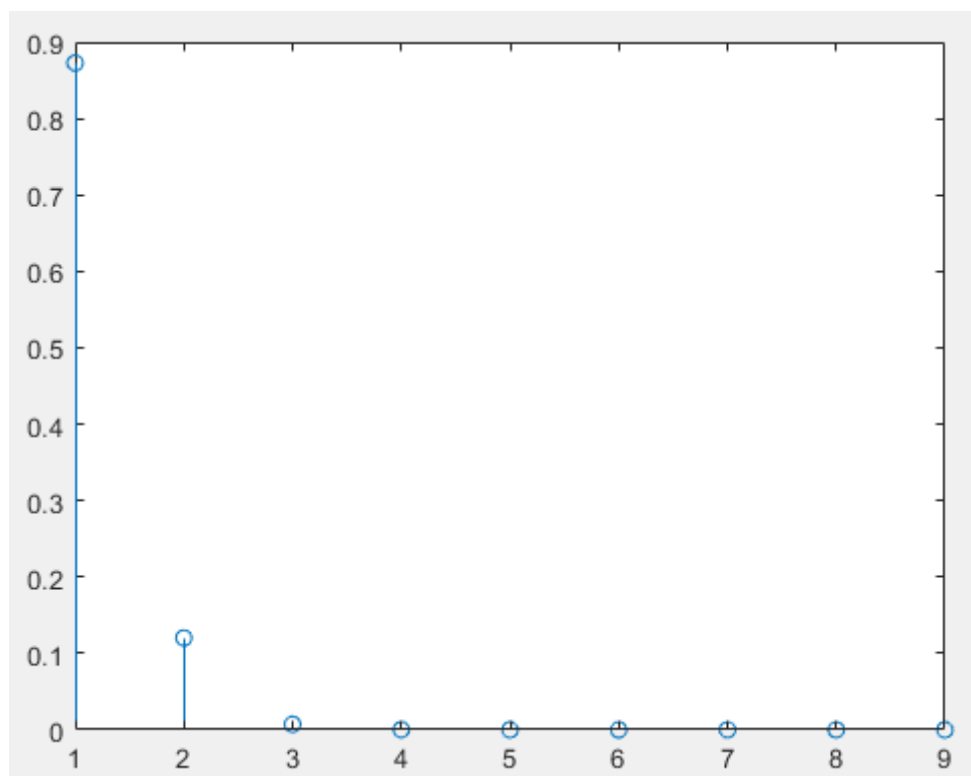
```



D) Verificámos que o número máximo de brinquedos para manter a caixa com 0 brinquedos estragados seriam 6, com uma probabilidade de 0.9028 de não existir nenhum estragado, seguido de sete com uma probabilidade de 0.8875, já inferior ao pretendido.

3.A) Usando novamente como base o código já escrito no problema 2.A) desta vez com uma função for a incrementar o número da condição que verifica o número de objetos estragados por cada caixa. No fim criamos um vetor ao qual atribuímos a probabilidade de cada uma das possibilidades de um objeto estragado.

Por fim aplicamos a função stem() e obtemos o gráfico desejado.



B) A probabilidade de  $X \geq 2$  é a soma de todos os superiores a 2 juntamente com 2, que é 0.0074. Concluindo que a probabilidade de existir mais que 1 brinquedo estragado numa caixa de 8 brinquedo é inferior a 1%.

C) valor esperado= 0.1352

variância= 0.1332

desvio padrão= 0.0666

```

N= 1e6;                                %numero de experiencias
p1=0.002;                               %probabilidade de
estragar peça                           %probabilidade de
p2=0.005;                               %probabilidade de montagem
estragar peca2                           %probabilidade de montagem
pmontagem= 0.01;                         %probabilidade de montagem
j=0;
n=8;
estraga=0;                               %varivael2
probsimulacao= zeros(n, 1);
for k=0 : n
    for f=1: N
        for i=1 :n
            if (rand <p1)|| (rand<p2)|| (rand <pmontagem) %se algum
dos objetos estiver estragado ou o processo de montagem correr mal
entra no ciclo
                estraga= estraga+1;        %adiciona um aos
            estragados
        end
    end
    if estraga==k
        j=j+1;                            %número de vezes em que
a condição A se verifica
    end
    estraga=0;                            %inicializa o numero
de objeto de estragados a 0 para a próxima experiência
    end
    probsimulacao(k+1)= j/N;               %prob simulação
    j=0;
end
stem(probsimulacao);
prob=0; % 2.b) X>=2
for i=3: n
    prob = prob+probsimulacao(i);
end
valoresperado=0; %valor esperado
for i=0: n
    valoresperado= probsimulacao(i+1)*i+valoresperado;
end
variancia=0; %variancia
for i=0: n
    variancia= variancia+ ((i-valoresperado)^2 * probsimulacao(i+1)) ;
end
desviopadrao= variancia^1/2;%desviopadrao

```

D) B)  $\text{prob}(X \geq 2) = 0.0293$

D) C) valor esperado= 0.2703      variância= 0.2649

desvio padrão= 0.1324

Os resultados são aproximadamente o dobro dos resultados para 8, porque os resultados são diretamente proporcionais ao número de acontecimentos possíveis.

```

N= 1e6;                                %numero de experiencias
p1=0.002;                               %probabilidade de
estragar peça
p2=0.005;                               %probabilidade de
estragar peça2
pmontagem= 0.01;                        %probabilidade de montagem
j=0;
n=16;
estraga=0;                              %variavel2
probsimulacao= zeros(n, 1);
for k=0 : n
    for f=1: N
        for i=1 :n
            if (rand <p1)|| (rand<p2)|| (rand <pmontagem) %se algum
dos objetos estiver estragado ou o processo de montagem correr mal
entra no ciclo
                estraga= estraga+1;        %adiciona um aos
estragados
            end
        end
        if estraga==k
            j=j+1;                        %número de vezes em que
a condição A se verifica
        end
        estraga=0;                        %inicializa o numero
de objeto de estragados a 0 para a próxima experiência
    end
    probsimulacao(k+1)= j/N;              %prob simulação
    j=0;
end
stem(probsimulacao);
prob=0; % 2.b)  $X \geq 2$ 
for i=3: n
    prob = prob+probsimulacao(i);
end
valoresperado=0; %valor esperado
for i=0: n
    valoresperado= probsimulacao(i+1)*i+valoresperado;
end
variancia=0; %variancia
for i=0: n
    variancia= variancia+ ((i-valoresperado)^2 * probsimulacao(i+1)) ;
end
desviopadrao= variancia^1/2;%desviopadrao

```



**4.A)** A probabilidade de uma caixa passar a vistoria é de quase 1(0.9920), no entanto, os padrões de qualidade podem não estar a ser mantidos. A verificação de apenas um é insuficiente para manter os padrões da empresa.

```

N= 1e6;                                %numero de experiencias
p1=0.002;                               %probabilidade de estragar peça
p2=0.005;                               %probabilidade de estragar peca2
pmontagem= 0.001;                       %probabilidade de montagem
j=0;
n=20;
random=randperm(20, 20);
estrage=0;
caixa= zeros(20, 1);

for f=1: N
    for i=1 :n
        if (rand <p1)|| (rand<p2)|| (rand <pmontagem) %se algum dos
objetos estiver estragado ou o processo de montagem correr mal entra
no ciclo
            caixa(random(i))=1;
        end
    end
    if caixa(1)== 0
        j=j+1;                           %número de vezes em que a
condição A se verifica
    end
    estrage=0;                           %inicializa o numero de
objeto de estragados a 0 para a próxima experiência
    caixa= zeros(20, 1);
end
probsimulacao= j/N;                     %prob simulação

```

#### 4.B) O número mínimo para alcançar o 90% são a verificação de 7 brinquedos.

```

N= 1e5;                                %numero de experiencias
p1=0.002;                               %probabilidade de
estragar peça
p2=0.005;                               %probabilidade de
estragar peca2
pmontagem= 0.001;                       %probabilidade de
montagem
j=0;
g=0;
n=20;
random=randperm(20, 20);
estraga=0;
caixa= zeros(20, 1);
probsimulacao= zeros(20, 1);
for m=1: 20
    for f=1: N
        for i=1 :n
            if (rand <p1)|| (rand<p2)|| (rand <pmontagem) %se algum
dos objetos estiver estragado ou o processo de montagem correr mal
entra no ciclo
                estraga= estraga +1;
                caixa(random(i))=1;
            end
        end
        for i=1: m
            if caixa(i)== 1
                j=j+1;
                break;
            end
        end
        if estraga==0
            g=g+1;
        end
        estraga=0;                                %inicializa o numero
de objeto de estragados a 0 para a próxima experiência
        caixa= zeros(20, 1);
    end
    probsimulacao(m)= g/ (N-j);
    g=0;
    j=0;
end

```

**FIM**