U.PORTO

FEP

ECONOMIA E GESTÃO

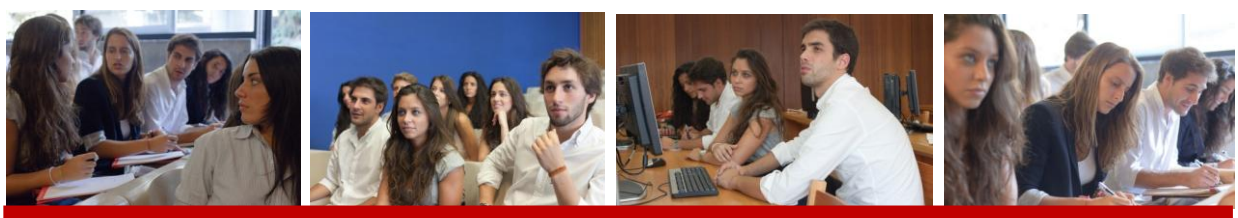
Estatística Aplicada

Assessment Work of Simulation Module

Daniel Fernando Silva

(199804052)

Dezembro 2013



Índice

1. Introdução

A simulação é o conceito base deste trabalho e pode ser vista como uma ferramenta que permite empregar técnicas matemáticas em ambiente computacional com o propósito de imitar um processo ou operação do mundo real. A simulação possibilita a realização de experiências em problemas de elevada complexidade sem se ter de recorrer ao problema real.

Para levar a cabo os exercícios propostos foi utilizado o *software* R. O recurso a um *software* para concretizar a simulação é importante na medida em que possibilita a execução do modelo com o intuito de o analisar e de obter as medidas de performance necessárias.

Este trabalho decorre da resolução de um problema apresentado na disciplina de Estatística Aplicada no âmbito **da geração de variáveis aleatórias discretas**.

Este trabalho encontra-se estruturado da seguinte forma: inicialmente é apresentado o exercício proposto (ponto 2) e seguidamente as distribuições alvo de simulação (ponto 3).

Na fase seguinte é efetuada uma breve fundamentação teórica das metodologias utilizadas dirigidas ao problema concreto (gerar de variáveis aleatórias discretas). Seguidamente as metodologias são aplicadas ao problema proposto e são interpretados dos resultados obtidos (ponto 4).

No final é efetuado um teste à qualidade de ajustamento para cada uma das amostras geradas pelos diferentes métodos e analisados os QQ plots para o método da inversa (ponto 5). No ponto 6 são apresentadas as conclusões.

No final de cada ponto, se aplicável, existe a referência ao *script* R utilizado que é parte integrante do presente relatório.

2. Problema a resolver

Considere 2 variáveis aleatórias X e Y independentes tais que:

$X \sim Po(1.5)$

Y ~Uniforme discreta

yi	0	1	2
f(yi)	1/3	1/3	1/3

a) Determine um algoritmo para gerar a lei associada à variável aleatória X e simule amostras de várias dimensões dessa distribuição, sem utilizar o comando "rpois". Determine outro algoritmo para gerar a lei associada à variável aleatória Y e simule amostras de várias dimensões dessa distribuição.

b) Determine um algoritmo para gerar a lei associada à variável aleatória Z = X + Y e simule amostras de várias dimensões dessa distribuição, sem utilizar o comando "rpois".

3. Distribuições do exercício

3.1. Distribuição de Poisson

Na teoria da probabilidade e na estatística, a distribuição de Poisson é uma distribuição de probabilidade de variável aleatória discreta que expressa a probabilidade de uma série de eventos ocorrer num certo período de tempo se estes eventos ocorrem independentemente de quando ocorreu o último evento.

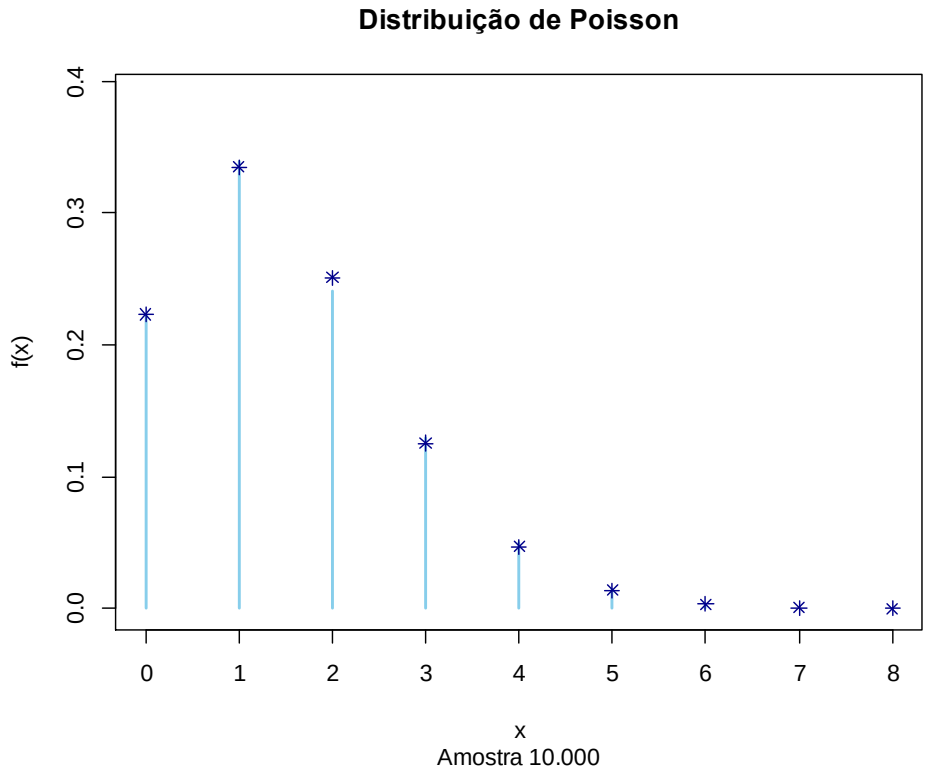
Uma distribuição de Poisson de parâmetro λ é muitas vezes utilizada para caracterizar o comportamento de variáveis do tipo:

X- "Número de ocorrências de um acontecimento A por um determinado intervalo de tempo ou espaço"

E escreve-se X $\sim \mathrm{Po}(\lambda)$. A função de probabilidade é dada por:

$$f(x) = \frac{e^{-\lambda}\lambda^{x}}{x!}, \text{para } x = 0,1,2,\dots,\lambda > 0$$

No exercício $\lambda = 1.5$. Utilizando a função *rpois* do R na geração de 10.000 valores aleatórios obtemos a seguinte função de probabilidade.



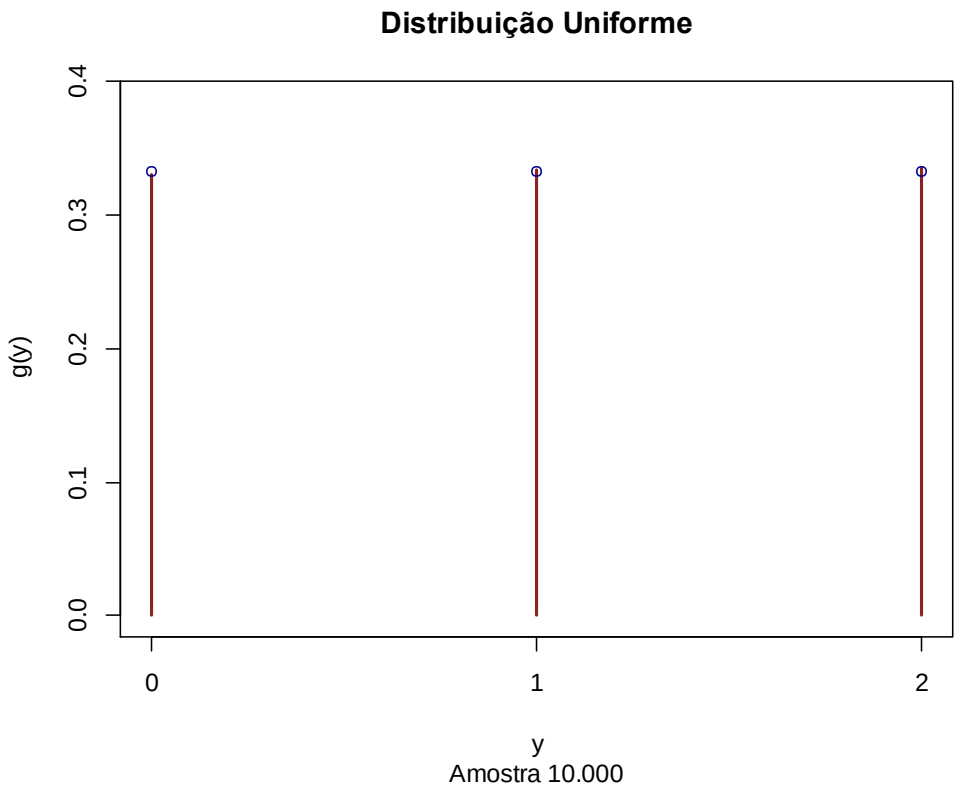
3.2 Distribuição Uniforme Discreta

Na teoria da probabilidade e na estatística, a distribuição Uniforme Discreta é uma distribuição de probabilidade que assume um número finito de valores com a mesma probabilidade.

A variável aleatória Y segue uma distribuição uniforme discreta em n pontos, se todos os n valores da variável ocorrem com igual probabilidade, ou seja, tem função probabilidade:

$$f(yi) = P(Y = yi) = \frac{1}{n}, \text{para todos os valores possíveis } y_i \; i = 1, \dots, n$$

No exercício a variável Y pode assumir 3 valores (0,1,2) com igual probabilidade. Utilizando a função *sample* para estimar 10.000 pontos obtemos a seguinte função probabilidade.



3.3 Distribuição Z = X + Y

A função Z resulta do somatório das funções X e Y. Para calcular a probabilidade para um determinado valor de Z deve-se partir de todas as combinações de valores de X e de Y que originam o valor de Z.

Seja h(z) a função de probabilidade de Z, h(z) = P(Z = z)

Seja f(x) a função de probabilidade de X, f(x) = P(X = x)

Seja g(y) a função de probabilidade de Y, g(y) = P(Y = y)

Ou seja,

h(0) = P(Z = 0) = P(X=0 e Y=0) = f(0) x g(0)

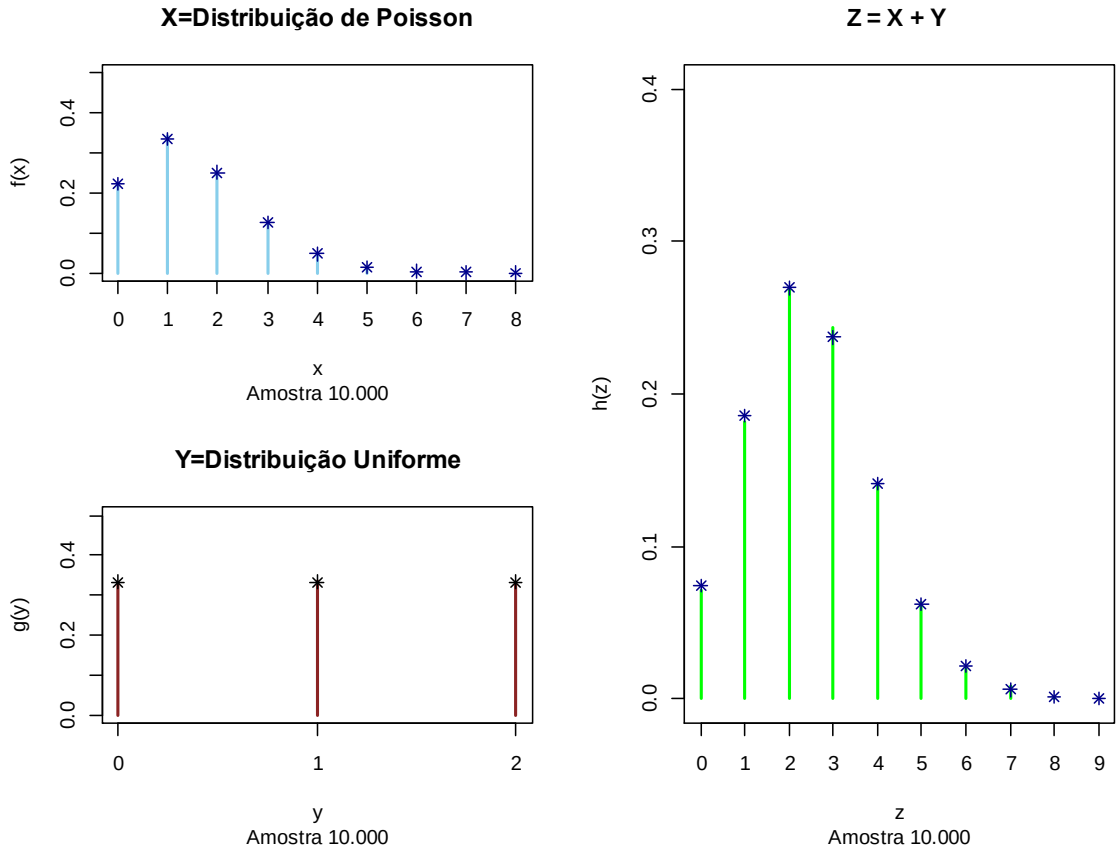
h(1) = P(Z = 1) = P(X=1 e Y=0) + P(X=0 e Y=1) = f(1) x g(0) + f(0) x g(1)

h(2) = P(Z = 2) = P(X=2 e Y=0) + P(X=1 e Y=1) + P(X=0 e Y=2) = f(2) x g(0) + f(1) x g(1) + f(0) x g(2)

...

h(z) = P(Z = z) = P(X=z e Y=0) + P(X=z-1 e Y=1) + P(X=z-2 e Y=2) = f(z) x g(0) + f(z-1) x g(1) + f(z-2) x g(2), para z >=2

A representação gráfica da função probabilidade é a seguinte (utilização da função *sample* e *rpois)*.



No quadro seguinte é apresentada a função probabilidade associada aos 10 primeiros valores de Z.

zi	0	1	2	3	4	5	6	7	8	9	10
h(zi)	0,0744	0,1859	0,2696	0,2371	0,1412	0,0622	0,0216	0,0061	0,0015	0,0003	0,0001

Programa R do ponto 3.1: Distribuições_exercicio.R

4. Fundamentação teórica e resolução do exercício

4.1 Gerar variáveis aleatórias discretas

O objetivo dos métodos estatísticos de geração de variáveis aleatórias (v.a.) é gerar variáveis aleatórias independentes X1,…,Xn com função distribuição $F_{x(x)}= P(Xi \leq x)$ e função probabilidade ou função densidade de probabilidade fx.

Usualmente parte-se de uma sequência de números com distribuição Uniforme entre 0 e 1:

$$F_U(u) = P(U_i \leq u) = u, \text{ para todo o } u \in [0,1] \text{ e para todo o } i$$

A partir desta sequência é possível gerar outras sequências com distribuições arbitrárias. É importante destacar que uma sequência de números, gerada deterministicamente por computador é necessariamente determinística (pseudoaleatória)[1].

Existem vários métodos para gerar variáveis aleatórias. Alguns métodos são de aplicação generalizada, dado que podem ser aplicados a várias distribuições estatísticas, como é o caso do método da inversão ou da rejeição utilizados no presente estudo. Porém, também existem métodos mais especializados que são desenvolvidos e aplicados a determinada distribuição[2].

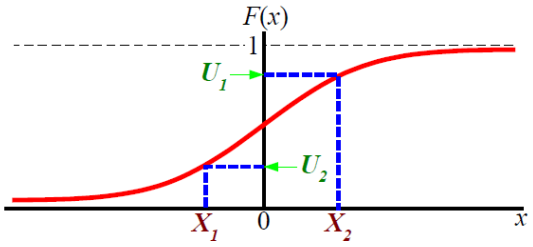
4.2 Método da inversão

Suponhamos que queremos gerar uma variável aleatória X com uma função de distribuição $F_{(X)}$ contínua e estritamente crescente.

Então, nessas condições poderemos gerar uma variável aleatória com distribuição F utilizando o seguinte algoritmo:

1. Gerar um número aleatório U.
2. Devolver X = $F^{-1}(U)$.

$$\begin{aligned} P[X \leq x] &= P[F^{-1}(U) \leq x] \\ &= P[U \leq F(x)] \\ &= F(x) \end{aligned}$$



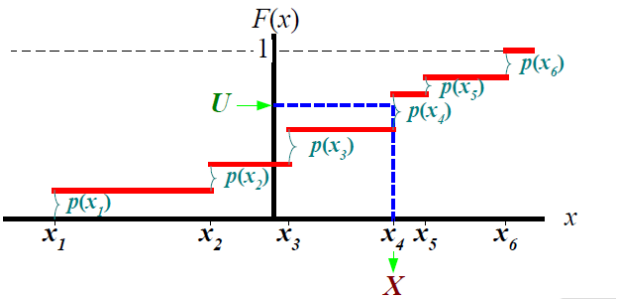
[1] Esta é a justificação para o facto de ao utilizar o comando s*et.seed()* do R se conseguir 'fixar' os valores aleatoriamente gerados.
[2] Estes métodos normalmente estão associados a variáveis aleatórias contínuas, onde por exemplo é difícil obter a função inversa da distribuição.

Também se pode aplicar o método da transformada inversa a variáveis discretas. Como só existe a função inversa para alguns pontos é necessário fazer algumas transformações.

Para gerar variáveis aleatórias discretas, com uma distribuição $F_{(x)}$:

1. Gerar um número aleatório U.
2. Determinar o menor inteiro positivo tal que:

   $U \leq F(x_1)$, e devolver x_1



Considere-se uma v.a. X com função probabilidade:

$$P(X=x_j) = p_j, j = 0,1,\ldots,\sum_j pj = 1$$

A função de distribuição de X é definida como:

$$F(x) = \begin{cases} 0 & se \quad x < x_1 \\ p_1 & se \quad x_1 \leq x < x_2 \\ p_1 + p_2 & se \quad x_2 \leq x < x_3 \\ \vdots & \\ 1 & se \quad x \geq x_n \end{cases}$$

Gera-se um número uniformemente distribuído (0,1) e estabelece-se:

$$X = \begin{cases} x_1 & se \quad U < p_0 \\ x_2 & se \quad p_0 \leq U < p_0 + p_1 \\ \vdots & \\ x_j & se \quad \sum_{i=1}^{j-1} p_i \leq U < \sum_{i=1}^{j} p_i \\ \vdots & \end{cases}$$

$$P(X = x_j) = P\left(\sum_{i=0}^{j-1} pi \leq U < \sum_{i=0}^{j} pi\right) = P\left(U < \sum_{i=0}^{j} pi\right) - P\left(U < \sum_{i=0}^{j-1} pi\right)$$
$$= \sum_{i=0}^{j} pi - \sum_{i=0}^{j-1} pi = pj$$

4.2.1 Distribuição de Poisson

Função distribuição:

$$X \sim Po(\lambda) <=> P(X = i) = \frac{e^{-\lambda}\lambda^{i}}{i!} = p_i$$

Utilizando o método da transformação inversa:

$$P(X = i)\ se \sum_{j=0}^{i-1} pj \leq U \leq \sum_{j=0}^{i} pj$$

$$p_{i+1} = \frac{e^{-\lambda}\lambda^{i+1}}{(i+1)!} = \frac{\lambda}{i+1} p_i$$

Facilmente se obtém: $p_0 = e^{-\lambda}$

O algoritmo que se segue é o indicado para a criação de uma distribuição de Poisson com parâmetro λ.

Algoritmo:

Passo 1: Gerar U ~ U [0,1]

Passo 2: Inicializar i = 0; p = $e^{-\lambda}$ e F = p

Passo 3: Enquanto F ≤ U

Passo 3.1.: Atualizar i = i+1

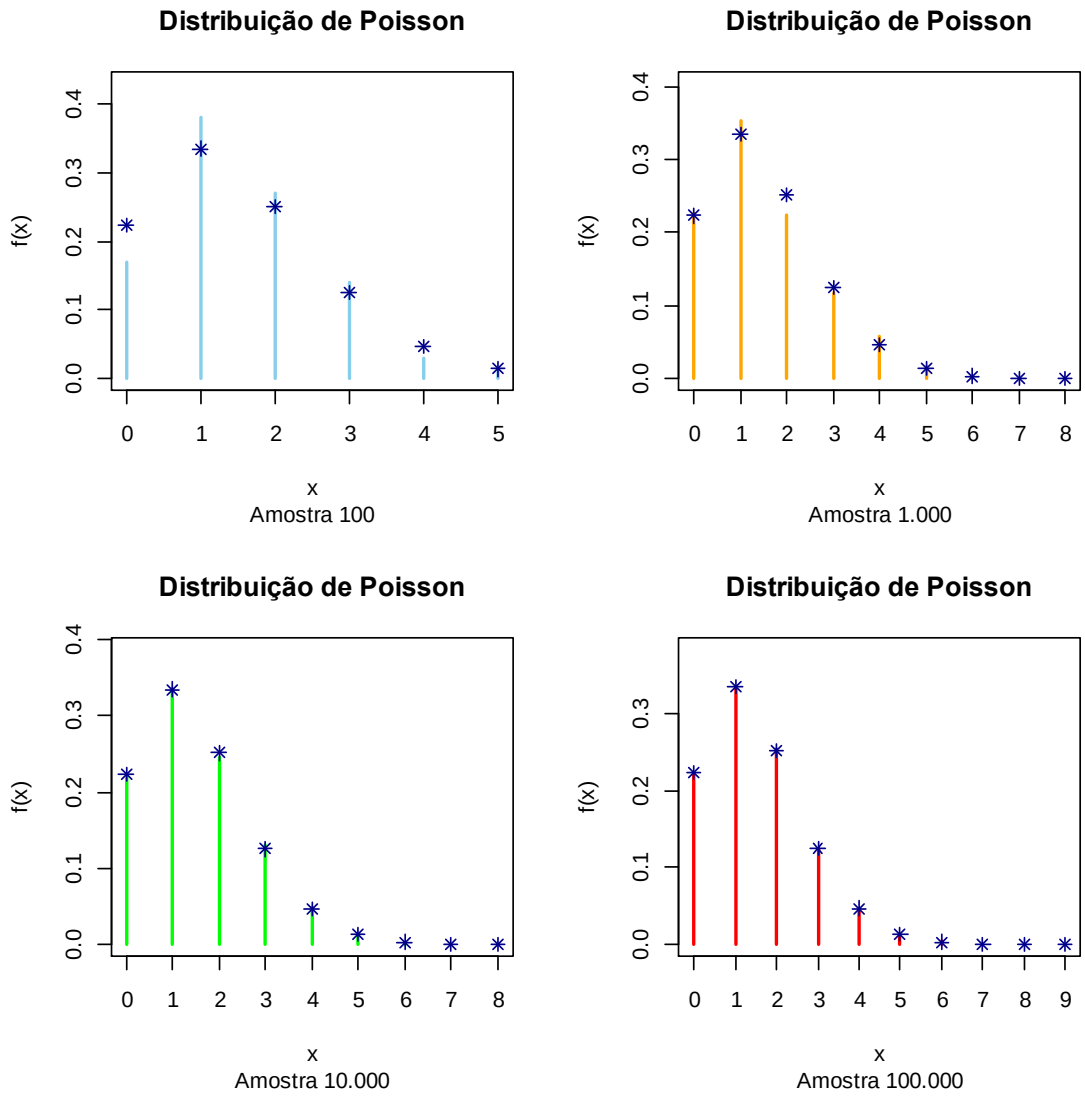
Passo 3.2.: Considerar p = $p\frac{\lambda}{i}$

Passo 3.3.: Considerar F = F+p

Passo 4: Fim do ciclo

Passo 5: Considerar X = i.

Seguidamente são apresentados graficamente os resultados obtidos para um $\lambda = 1.5$ e os impactos da alteração da dimensão na amostra gerada.



Programa R do ponto 4.2.1: Inversa_poisson.R

4.2.2 Distribuição Uniforme Discreta

Resolução 1

Seja Y uma v.a. com função probabilidade e F_y dadas por

yi	0	1	2
f(yi)	1/3	1/3	1/3

e $F_y(y) = \begin{cases} 0\ se\ y < 0 \\ \frac{1}{3}\ se\ 0 \leq y < 1 \\ \frac{2}{3}\ se\ \ 1 \leq y < 2 \\ 1\ se\ y \geq 2 \end{cases}$

Pelo método da inversão obtém-se:

$$Y = F_y^{-1}(U) = \begin{cases} 0\ se\ U \leq \frac{1}{3} \\ 1\ se\ \frac{1}{3} < U \leq \frac{2}{3} \\ 2\ \ se\ U > \frac{2}{3} \end{cases}$$

Algoritmo:

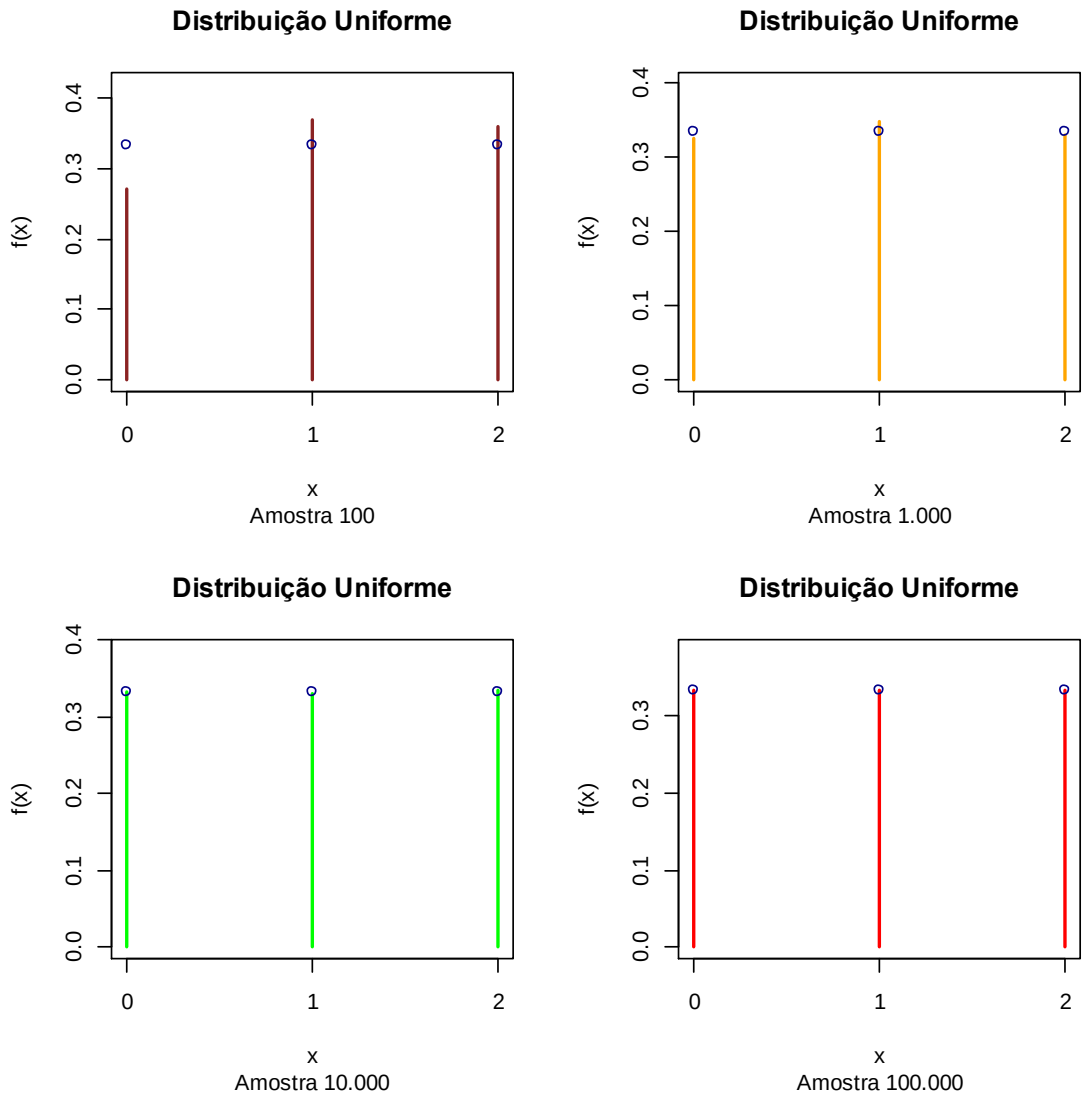
Passo 1: Gerar U ~ U [0,1]

Passo 2: Se U $\leq \frac{1}{3}$ então X=0

Passo 3: Se U $\leq \frac{2}{3}$ então X=1

Passo 4: Se U> $\frac{2}{3}$ então X=2

Seguidamente são apresentados graficamente os resultados obtidos para uma distribuição uniforme discreta com 3 valores possíveis e por definição probabilidade $\frac{1}{3}$.



Resolução 2

Na distribuição uniforme as probabilidades são iguais:

$$P(X=j) = \frac{1}{n}, para\ j = 1,2, \dots, n$$

Para simular valores para esta distribuição gera-se um valor aleatório U e considera-se:

$$X = j\ se\ \frac{j-1}{n} \leq U < \frac{j}{n}$$

Logo,

$$X = Int(nU) + 1$$

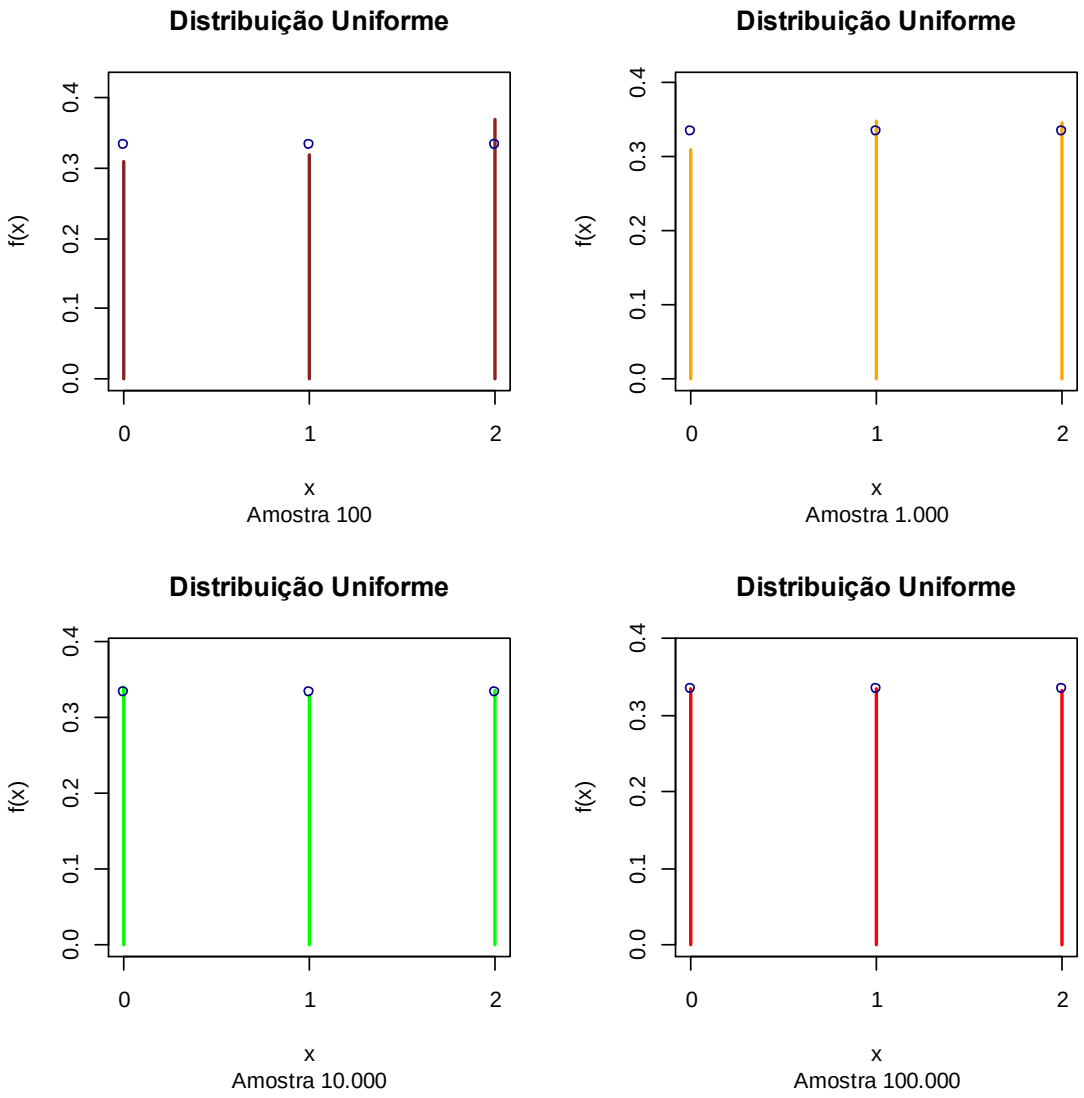
onde $Int(x)$ representa a parte inteira de x

Algoritmo:

Passo 1: Gerar U ~ U [0,1]

Passo 2: X=$Int(nU) + 1$

Para n=3 obtém-se os seguintes resultados:



Programa R do ponto 4.2.2: Inversa_uniforme.R

4.2.3 Distribuição Z = X + Y

Resolução 1

A forma mais simples de obter pelo método da inversa a distribuição Z é gerar as distribuições X e Y de forma independente e efetuar a combinação linear das distribuições dentro do algoritmo (passo 8).

Algoritmo:

Passo 1: Gerar U ~ U [0,1]

Passo 2: Inicializar i = 0; p = $e^{-\lambda}$ e F = p

Passo 3: Enquanto F ≤ U

Passo 3.1.: Atualizar i = i+1

Passo 3.2.: Considerar p = $p\frac{\lambda}{i}$

Passo 3.2.: Considerar F = F+p

Passo 4: Fim do ciclo

Passo 5: Considerar X = i.

Passo 6: Gerar U ~ U [0,1]

Passo 6.1: Se U ≤ $\frac{1}{3}$ então y=0

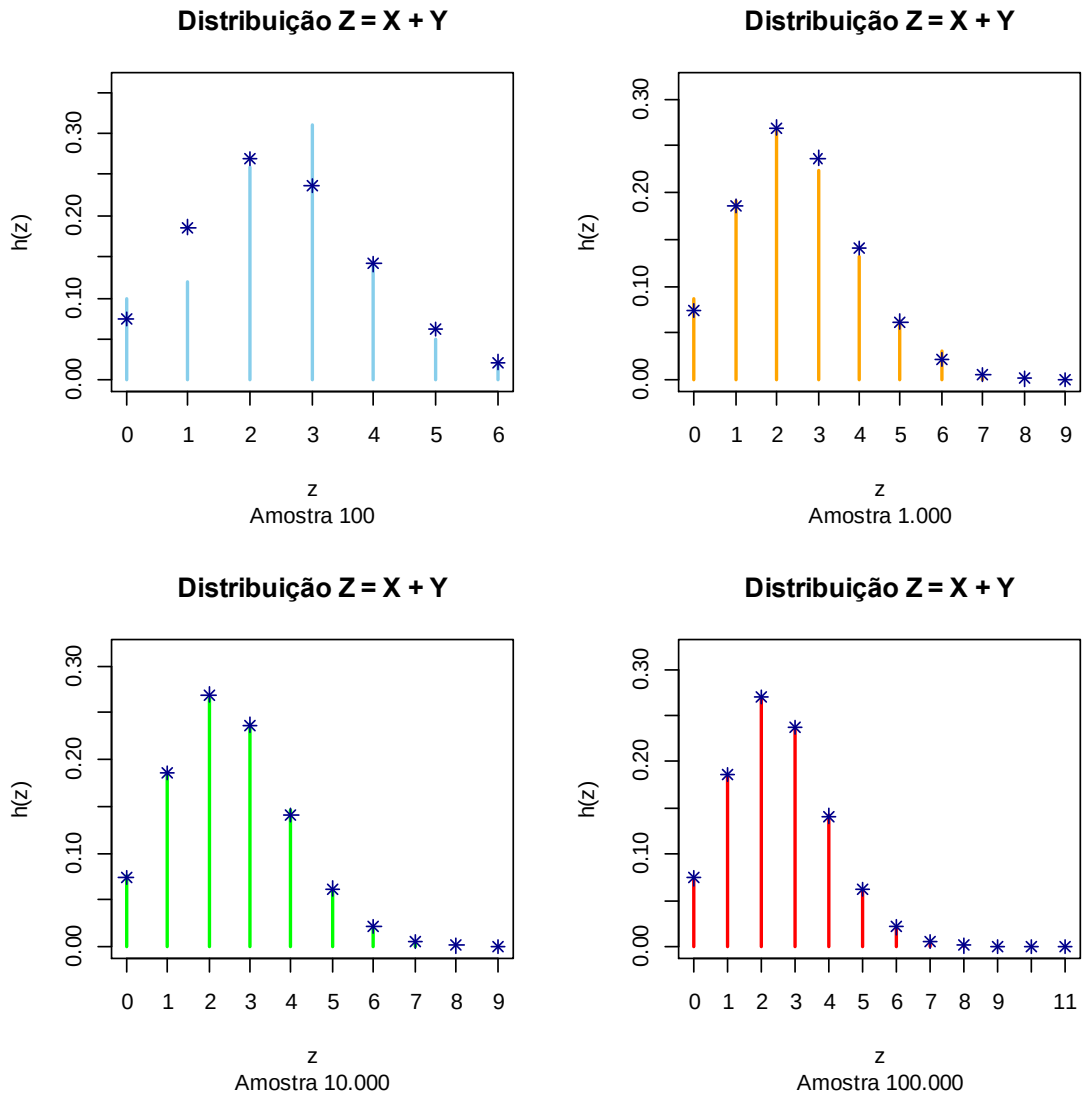
Passo 6.2: Se U ≤ $\frac{2}{3}$ então y=1

Passo 6.3: Se U > $\frac{2}{3}$ y=2

Passo 7: Considerar Y = y.

Passo 8: Z=X+Y

Seguidamente são apresentados graficamente os resultados obtidos para a distribuição Z, obtida pela combinação linear das distribuições X e Y.



Resolução 2

Utilizando a função de probabilidade de Z e adaptando o algoritmo definido no ponto 4.2.1.

Pelo método da transformação inversa obtém-se:

$$P(Z = i)\ se \sum_{j=0}^{i-1} pj \leq U \leq \sum_{j=0}^{i} pj$$

Função de probabilidade Z.

h(0) = P(Z = 0) = P(X=0 e Y=0) = f(0) x g(0)

h(1) = P(Z = 1) = P(X=1 e Y=0) + P(X=0 e Y=1) = f(1) x g(0) + f(0) x g(1)

h(2) = P(Z = 2) = P(X=2 e Y=0) + P(X=1 e Y=1) + P(X=0 e Y=2) = f(2) x g(0) + f(1) x g(1) + f(0) x g(2)

...

h(z) = P(Z = z) = P(X=z e Y=0) + P(X=z-1 e Y=1) + P(X=z-2 e Y=2) = f(z) x g(0) + f(z-1) x g(1) + f(z-2) x g(2), para z >=2

O algoritmo que se segue é o indicado para a criação da distribuição de Z.

Algoritmo:

Passo 1: Gerar U ~ U [0,1]

Passo 2: Inicializar i = 0; p = h(0)=P(Z = 0) = P(X=0 e Y=0) = f(0) x g(0) e H = p

Passo 3: Enquanto H ≤ U

Passo 3.1.: Atualizar i = i+1

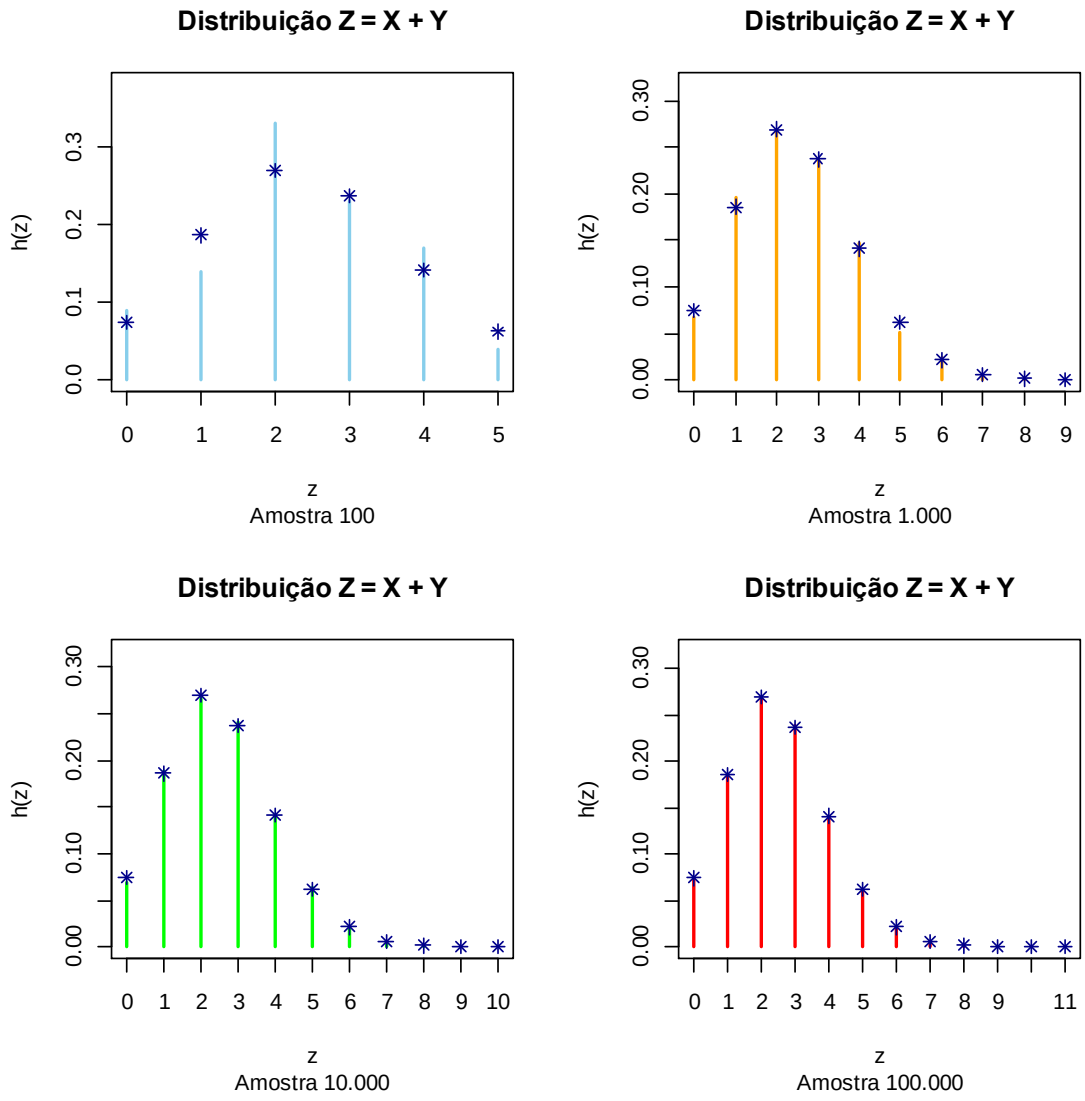
Passo 3.2.: Considerar p = h(i)

Passo 3.2.: Considerar H = H+p

Passo 4: Fim do ciclo

Passo 5: Considerar Z = i.

Seguidamente são apresentados graficamente os resultados obtidos para a distribuição Z pelo método da inversa.



Resolução 3

Utilizando a função de probabilidade acumulada de Z e adaptando o algoritmo definido no ponto 4.2.2.

Seja Z uma v.a. com função probabilidade h(zi) e probabilidade acumulada H(zi) para os primeiros 10 valores da distribuição Z:

zi	0	1	2	3	4	5	6	7	8	9	10
h(zi)	0,0744	0,1859	0,2696	0,2371	0,1412	0,0622	0,0216	0,0061	0,0015	0,0003	0,0001

zi	0	1	2	3	4	5	6	7	8	9	10
H(zi)	0,0744	0,2603	0,5299	0,7670	0,9082	0,9704	0,9920	0,9981	0,9996	0,9999	1,0000

Pelo método da inversão obtém-se:

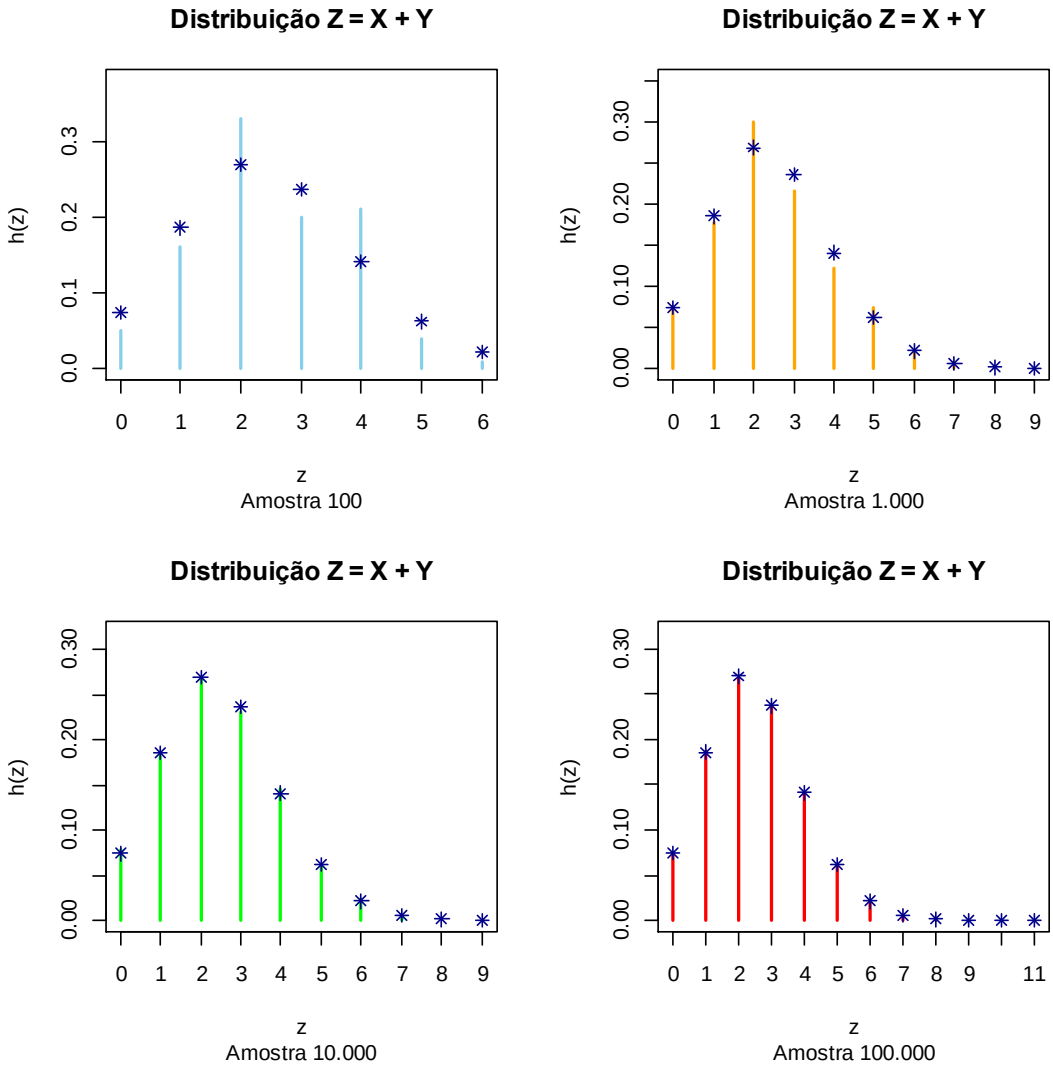
$$Z = F_z^{-1}(U)$$

Algoritmo:

Passo 1: Gerar U ~ U [0,1]

Passo 2: $Se\ U \leq H(0)$ então Z=0, $Se\ U \leq H(1)$, então Z=1… $Se\ U \leq H(10)$, então Z=10

equivale a Z= n.º de elementos de H(zi) que satisfaz a condição U >= H(zi) [3]



Programa R do ponto 4.2.3: Inversa_distribuição_Z.R

[3] Esta segunda formulação surgiu como a forma mais eficaz de operacionalizar o algoritmo no *software*.

4.3 Método da rejeição

O método da aceitação-rejeição é um método muito útil e de aplicação geral para gerar variáveis aleatórias.

Suponhamos que queremos simular uma distribuição discreta com função massa:

$$p_j, j \geq 0$$

E que existe um método eficiente para simular $q_j, j \geq 0$ onde:

$$\frac{p_j}{q_j} \leq c, \qquad \text{para todo j enquanto } p_j > 0$$

em que c é uma constante positiva

$$p_j \leq cq_j$$

O método da aceitação-rejeição é o seguinte:

1. Gerar Y com da função massa q_j
2. Gerar U ~ U [0,1]
3. Se U < $\frac{p_j}{cq_j}$, aceitar Y, para valores de X.

   Caso contrário regressar a 1.

$Provar\ que: P(X = j) = p_j$

Considerar:

$$P(Y = j, aceitar) = P(Y = j)P(aceitar) = q_j \frac{p_j}{cq_j} = \frac{pj}{c}$$

$$\sum_j P(Y = j, aceitar) = \sum_j \frac{p_j}{c} = \frac{1}{c} \qquad \begin{cases} P(aceitar) = \frac{1}{c} \\ P(rejeitar) = 1 - \frac{1}{c} \end{cases}$$

$$P(X = j) = \sum_n P(j \text{ é } aceite\ na\ nésima\ iteração) = \sum_n \left(1 - \frac{1}{c}\right)^{n-1} \frac{1}{c} = p_j$$

4.3.1 Distribuição de Poisson

A função de distribuição de Poisson assume um conjunto de valores inteiros infinitos (X). Para valores de X> λ a probabilidade associada a X diminui de forma considerável (primeiramente a ritmos crescentes e depois decrescentes), tornando-se muito pequena quando X se afasta consideravelmente de c. Para λ=1.5 e para a resolução do exercício optou-se por considerar como limite X=7 que já apresenta uma probabilidade associada muito reduzida (função do R: *dpois(7,1.5)* = 0.000756).

Algoritmo:

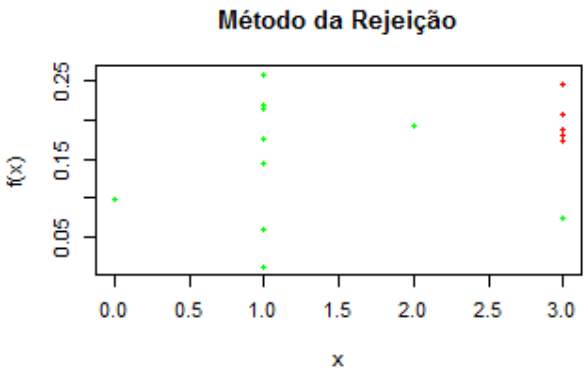
Passo 1: Gerar Y = $Int(nU)$

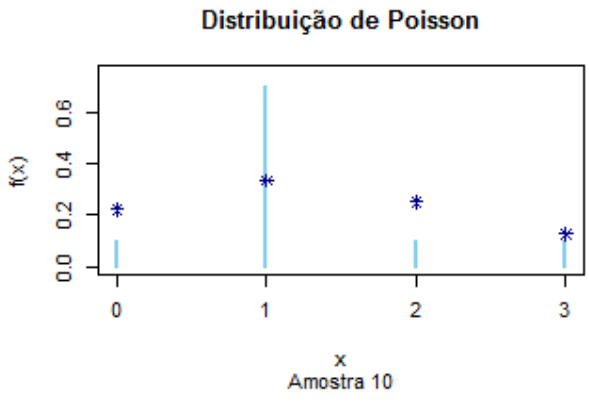
Passo 2: Gerar U ~ U [0,1]

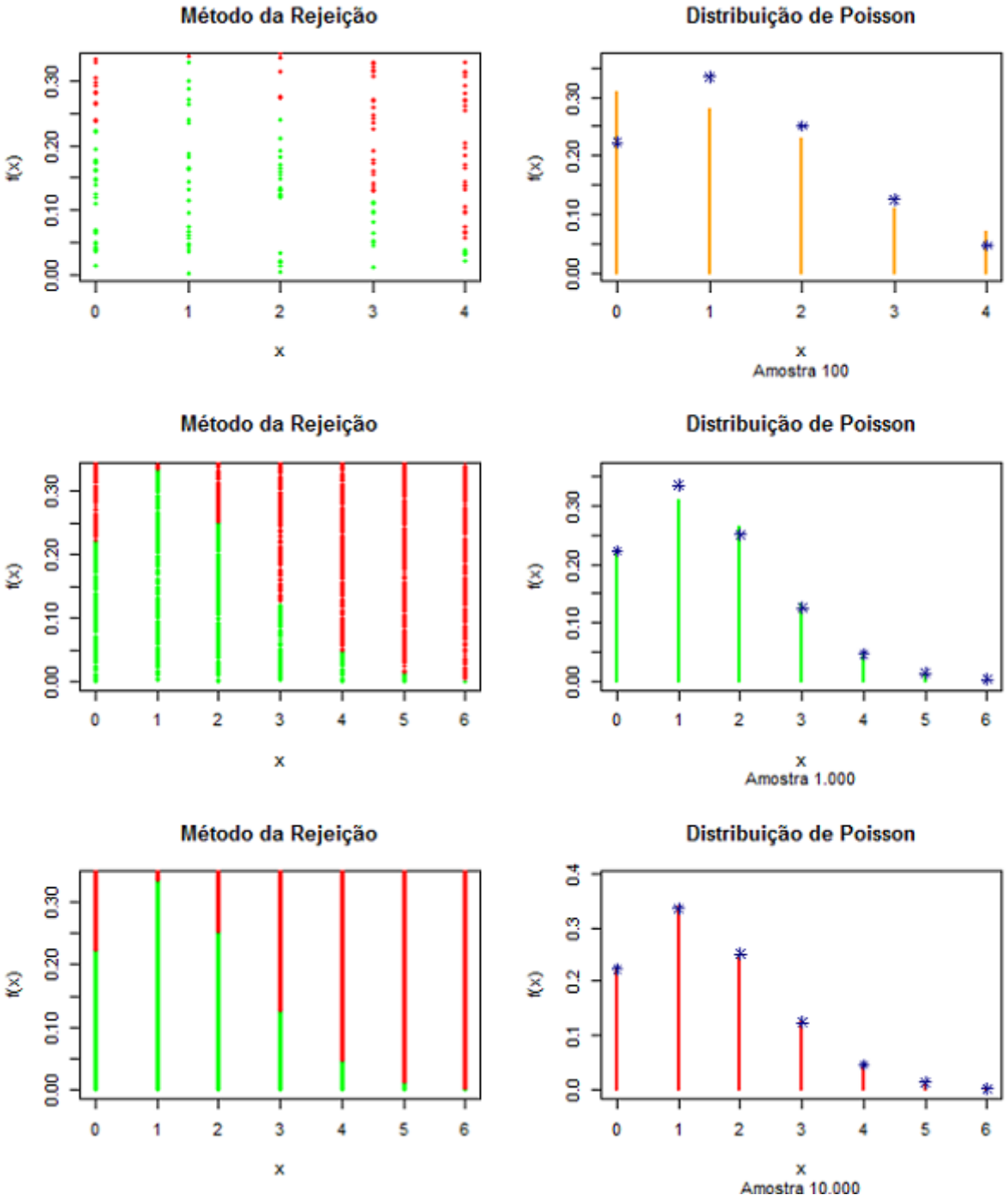
Passo 3: Aceitar Y para valores de X se U ≤ *dpois*(Y, lambda) ou U ≤ (*e*^(-λ)* λ ^Y) / Y!

Passo 4: Repetir

Para n=7 e λ =1.5 obteve-se os resultados apresentados em seguida. Os gráficos da esquerda representam os pontos aceites (verdes) e rejeitados (vermelhos), os da direita a comparação dos valores obtidos (aceites) com os da distribuição em causa.







Programa R do ponto 4.3.1: Rejeicao_poisson.R

4.3.2 Distribuição Uniforme

A função de distribuição uniforme discreta assume um conjunto de valores com igual probabilidade. No exercício a função pode assumir 3 valores (0,1,2). A probabilidade associada a cada elemento é 1/3.

Algoritmo:

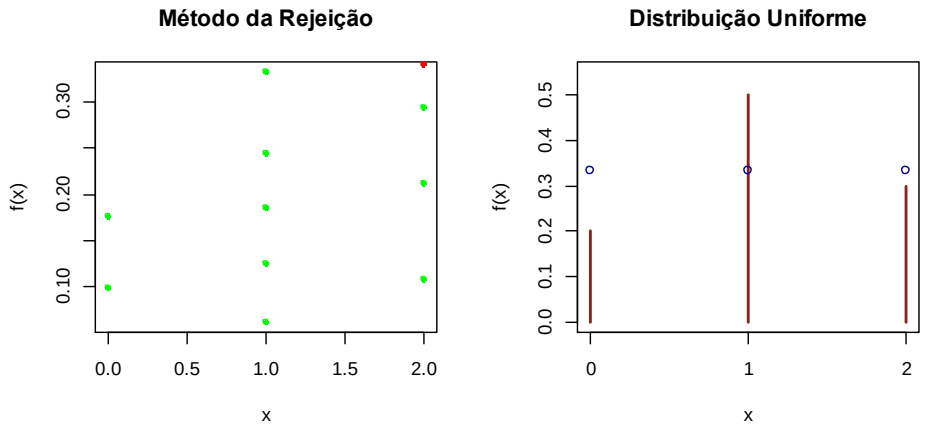
Passo 1: Gerar Y = $Int(nU)$

Passo 2: Gerar U ~ U [0,1]

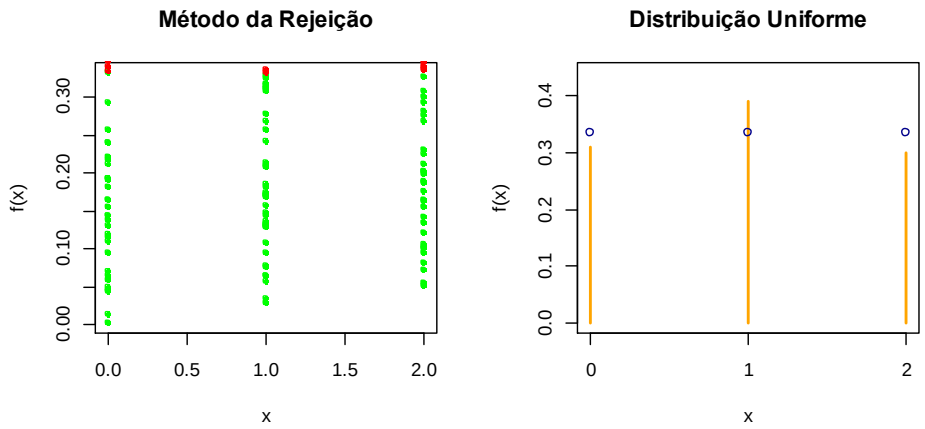
Passo 3: Aceitar Y para valores de X se U $\leq$ 1/n

Passo 4: Repetir

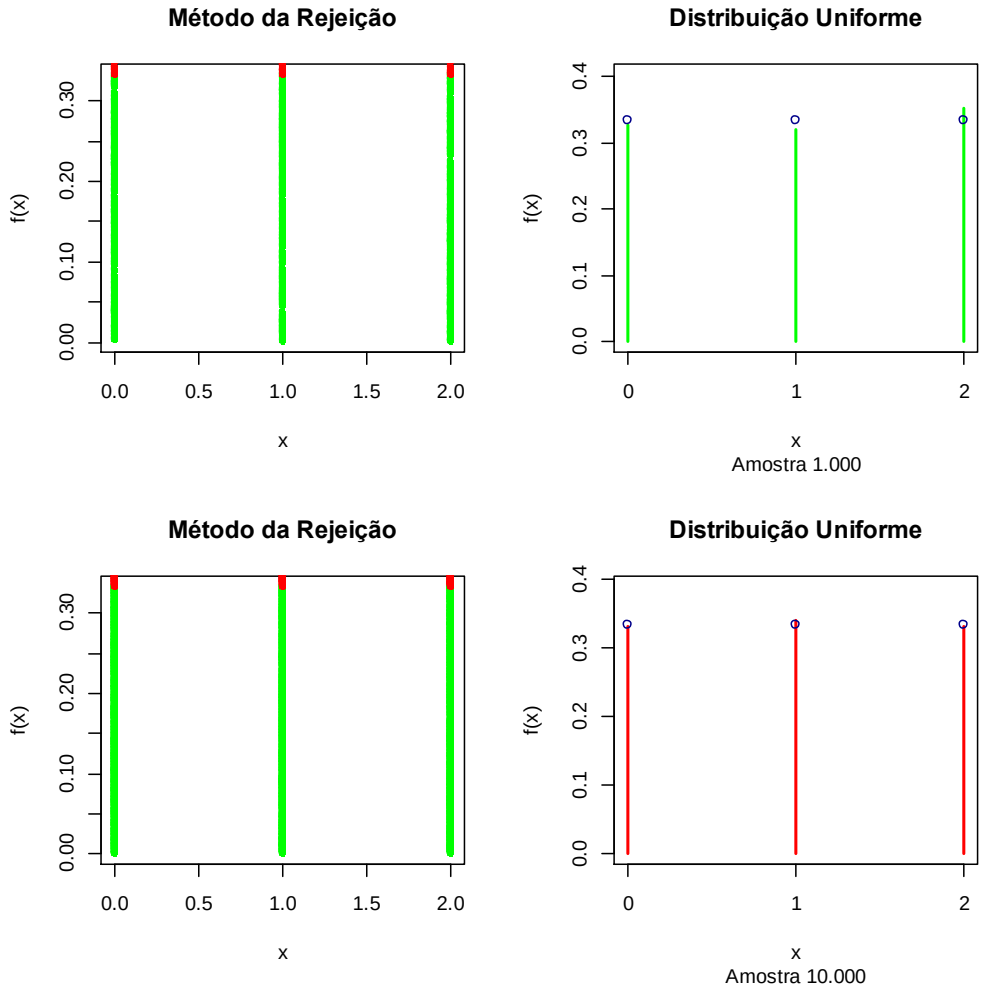
Para n=3 obteve-se os resultados apresentados em seguida. Tal como no ponto anterior, os gráficos da esquerda representam os pontos aceites (verdes) e rejeitados (vermelhos), os da direita a comparação dos valores obtidos (valores aceites) com os da distribuição em causa.



Amostra 10



Amostra 100



Programa R do ponto 4.3.2: Rejeicao_unifirme.R

4.3.3 Distribuição Z = X + Y

A função de distribuição Z assume um conjunto de valores inteiros infinitos (Z). Para Z= 10 a probabilidade associada é 0.001. Para a resolução do exercício optou-se por considerar como limite Z=10 que já apresenta uma probabilidade associada muito reduzida.

Algoritmo:

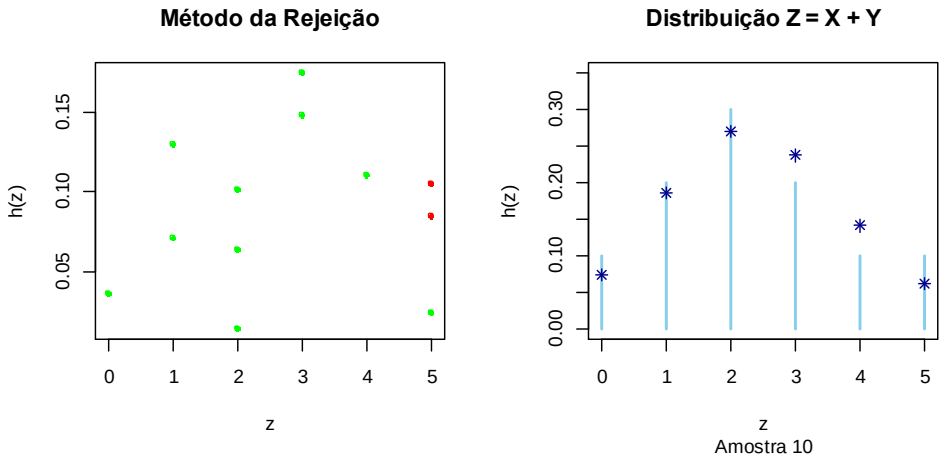
Passo 1: Gerar Y = $Int(nU)$

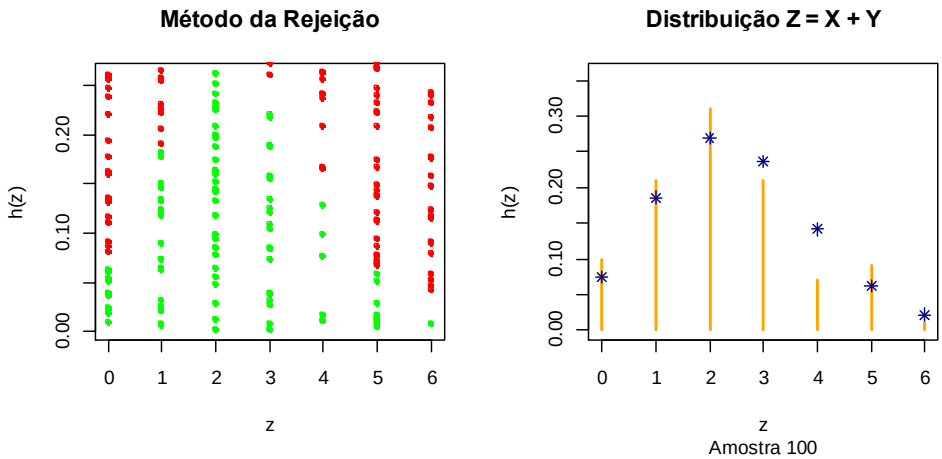
Passo 2: Gerar U ~ U [0,1]

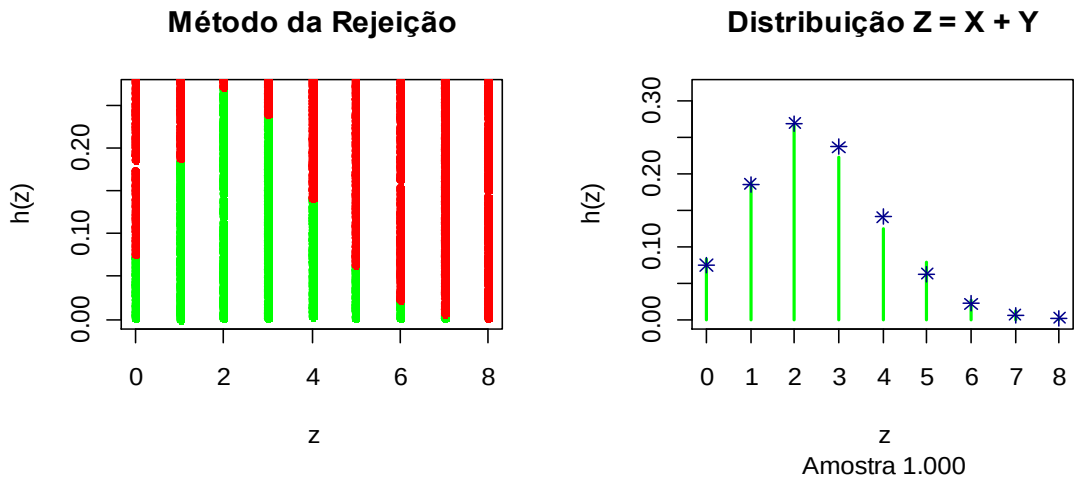
Passo 3: Aceitar Y para valores de Z se U $\leq F_Z(Y)$

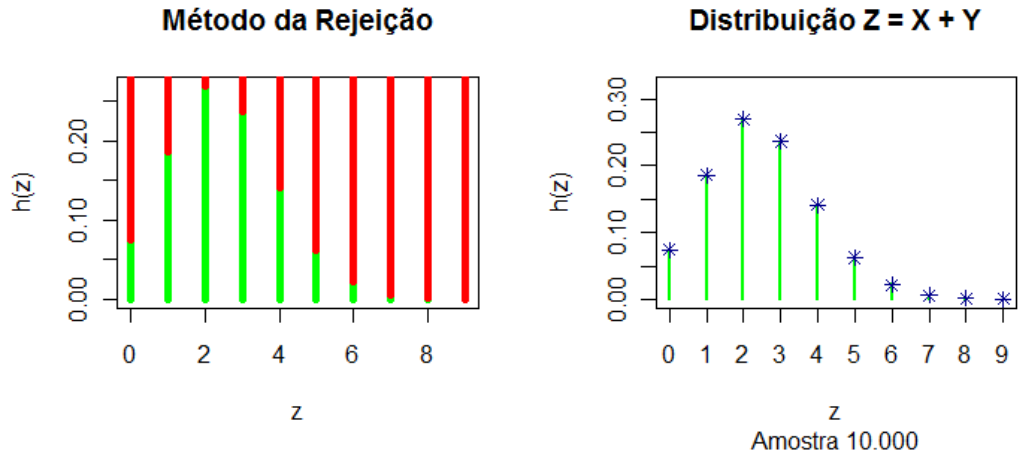
Passo 4: Repetir

Para n=10 obteve-se os resultados apresentados em seguida. Os gráficos da esquerda representam os pontos aceites (verdes) e rejeitados (vermelhos), os da direita a comparação dos valores obtidos (aceites) com os da distribuição em causa.









Programa R do ponto 4.3.3: Rejeicao_Z.R

5. Avaliação da qualidade do ajustamento

Para além da comparação entre o gráfico de barras e a função probabilidade para verificar o ajustamento, existem outras metodologias. Neste ponto vamos explorar o **teste de ajustamento do qui-quadrado** e analisar os **QQ Plot** associados a cada uma das distribuições.

A vantagem de efetuar um teste de ajustamento, face às restantes alternativas apresentadas, é que se obtém um número (*p-value)* que pode ser utilizado para medir e comparar a qualidade do ajustamento.

5.1 Teste de ajustamento a uma distribuição teórica

O teste qui-quadrado permite avaliar a aderência entre uma distribuição de frequências, associada a uma amostra, constituída por observações expressas numa escala e uma distribuição teórica.

O teste qui-quadrado para a avaliação da qualidade de ajustamento baseia-se na comparação da distribuição dos dados amostrais com a distribuição teórica. O princípio básico do método é comparar proporções, ou seja, as possíveis divergências entre as frequências observadas e esperadas para um certo evento. Pode-se afirmar que dois grupos se comportam de forma semelhante se as diferenças entre as frequências observadas e as esperadas em cada categoria forem próximas a zero.

A hipótese nula (H0) e a hipótese alternativa (H1) as seguintes:

H0: A amostra segue uma determinada distribuição teórica

H1: A amostra não segue tal distribuição

O teste do qui-quadrado apenas tem resultados robustos se a dimensão da amostra for superior a 30 e se a frequência esperada para cada classe for não inferior a 5. Para satisfazer a segunda condição procedeu-se à agregação de classes adjacentes.

A tabela seguinte apresenta os p-values referentes ao teste qui-quadrado.

	Dimensão			
	100	**1.000**	**10.000**	**100.000**
X = Poisson				
Inversão	0,53	0,22	0,24	0,38
Rejeição	0,32	0,65	0,69	-
Y = Uniforme				
Inversão - R1	0,40	0,64	0,83	0,98
Inversão - R2	0,73	0,24	0,40	0,46
Rejeição	0,48	0,41	0,55	-
Z = X + Y				
Inversão - R1	0,30	0,26	0,43	0,89
Inversão - R2	0,27	0,83	0,66	0,96
Inversão - R3	0,14	0,14	0,18	0,63
Rejeição	0,63	0,14	0,83	-

p-values do teste qui-quadrado

Face aos resultados obtidos podem-se retirar as seguintes conclusões:

- Para um nível de significância de 5% não se rejeita H0 (*p-value* >5%), pelo que se concluí que cada uma das amostras segue as respetivas distribuições teóricas. Conclui-se assim que os algoritmos construídos são eficazes independentemente da dimensão da amostra.
- A comparação dos *p-values* para tentar inferir qual o melhor algoritmo é complicado na medida em que para cada geração de valores vai-se obter amostras diferentes e consequentemente *p-values* também diferentes. Apesar desta limitação pode-se concluir que o aumento da dimensão, na maior parte das vezes, implica um aumento do ajustamento da amostra à distribuição teórica (aumento do *p-value* associado). No entanto, nem sempre se verificou uma melhoria do ajustamento com o aumento da amostra.

5.2 QQ plot - Quantil-Quantil plot

Outra maneira de comprovar a adequação de um determinado modelo estatístico aos dados empíricos é através do método gráfico de probabilidade ou QQ plot (Quantil-Quantil plot). Por definição, o QQ plot é um método gráfico que permite diagnosticar as diferenças entre a distribuição de probabilidade de uma população a partir da qual é retirada uma amostra e uma distribuição de comparação (distribuição teórica neste caso).

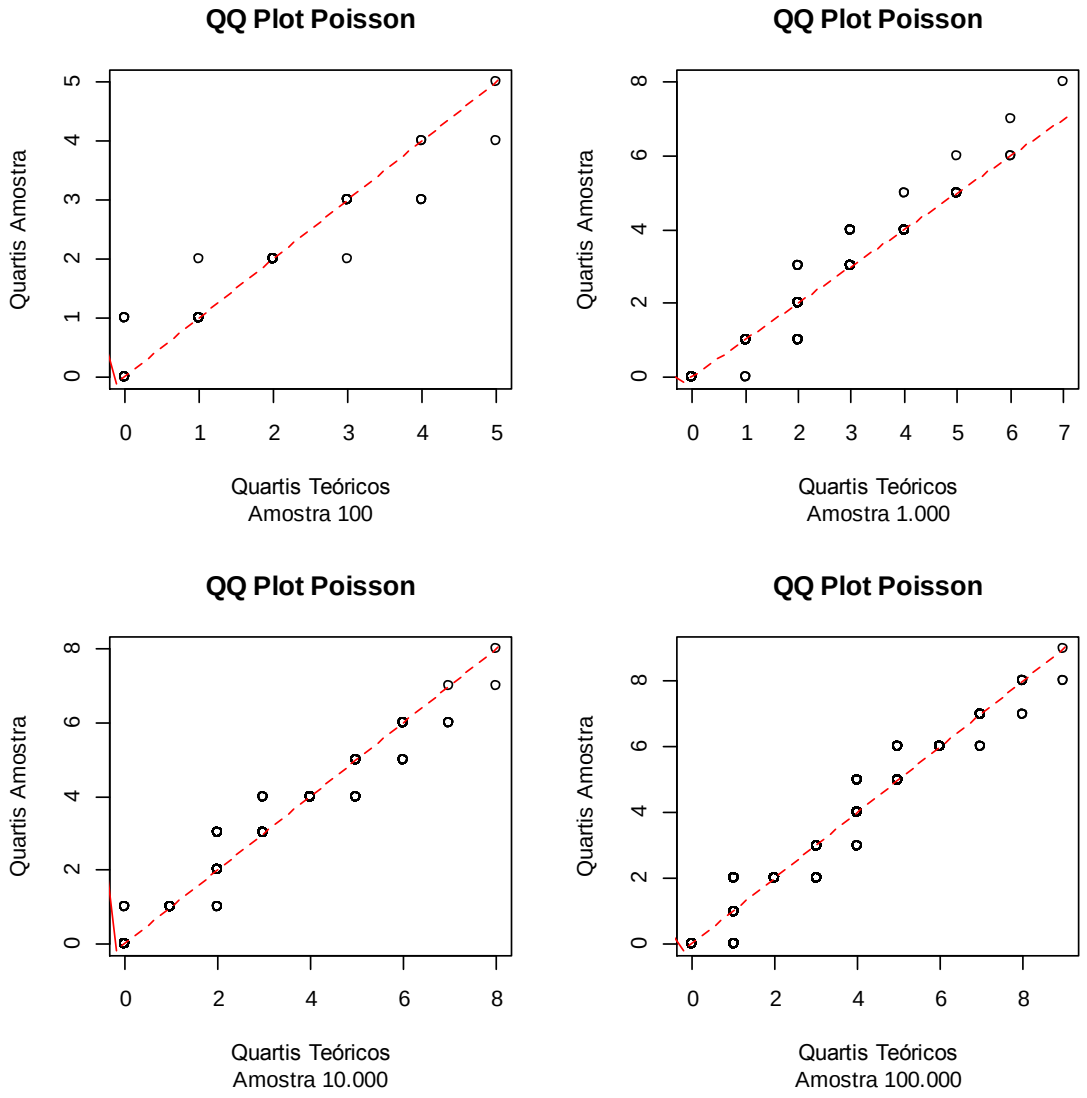
Apesar da subjetividade inerente ao método, este permite uma confirmação visual rápida da qualidade do ajustamento de determinado modelo às amostras.

Se a amostra for adequada à distribuição teórica, os pontos no gráfico distribuem-se aleatoriamente próximos de uma linha reta (assinalada a vermelho nos gráficos).

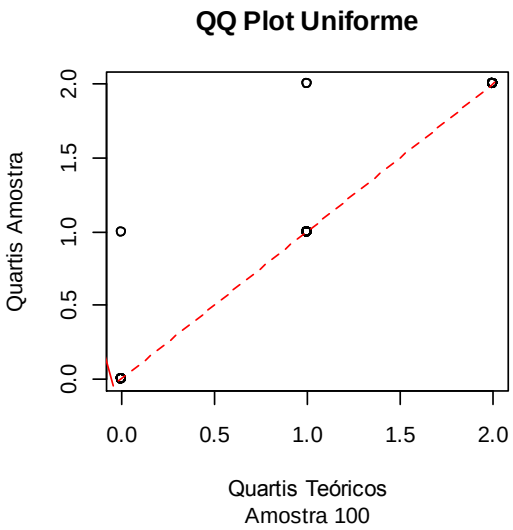
No caso de variáveis discretas há muitos pontos iguais, pelo que é difícil através da visualização inferir sobre melhorias no ajustamento face à variação da dimensão da amostra. Como já foi constatado anteriormente (principalmente utilizando o teste do qui-quadrado), verificamos que os pontos encontram-se mais ou menos em torno e próximos da reta esperada, pelo que se conclui que os algoritmos produziram os resultados esperados.

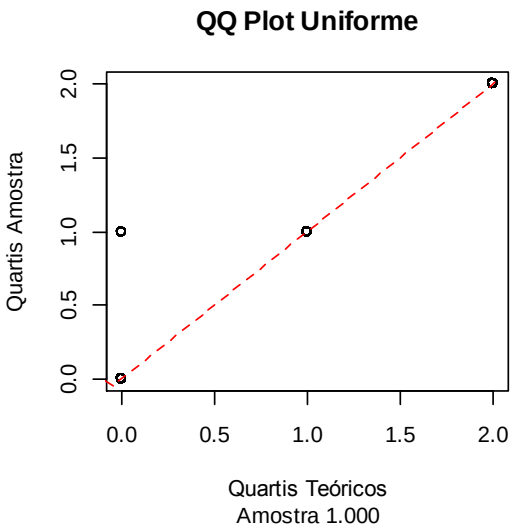
Por se entender que os QQ plots não acrescentam informação adicional às conclusões do presente relatório optou-se por apenas apresentar os gráficos referentes ao método da inversa para cada uma das distribuições.

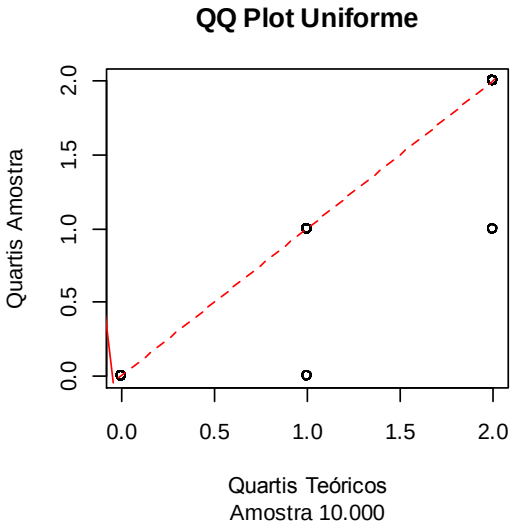
Método: Inversa - QQ Plot Po (1.5)

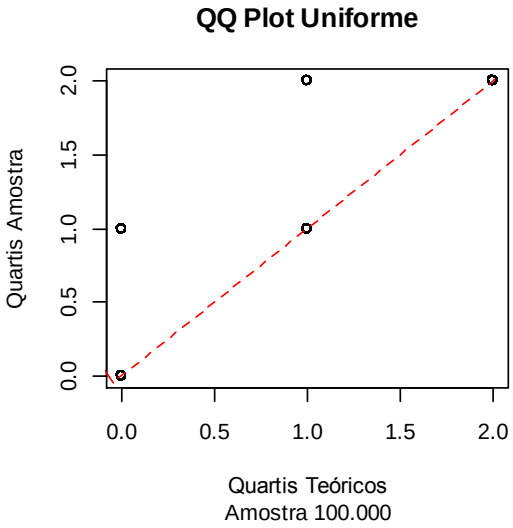


Método: Inversa – Uniforme: algoritmo da resolução 1

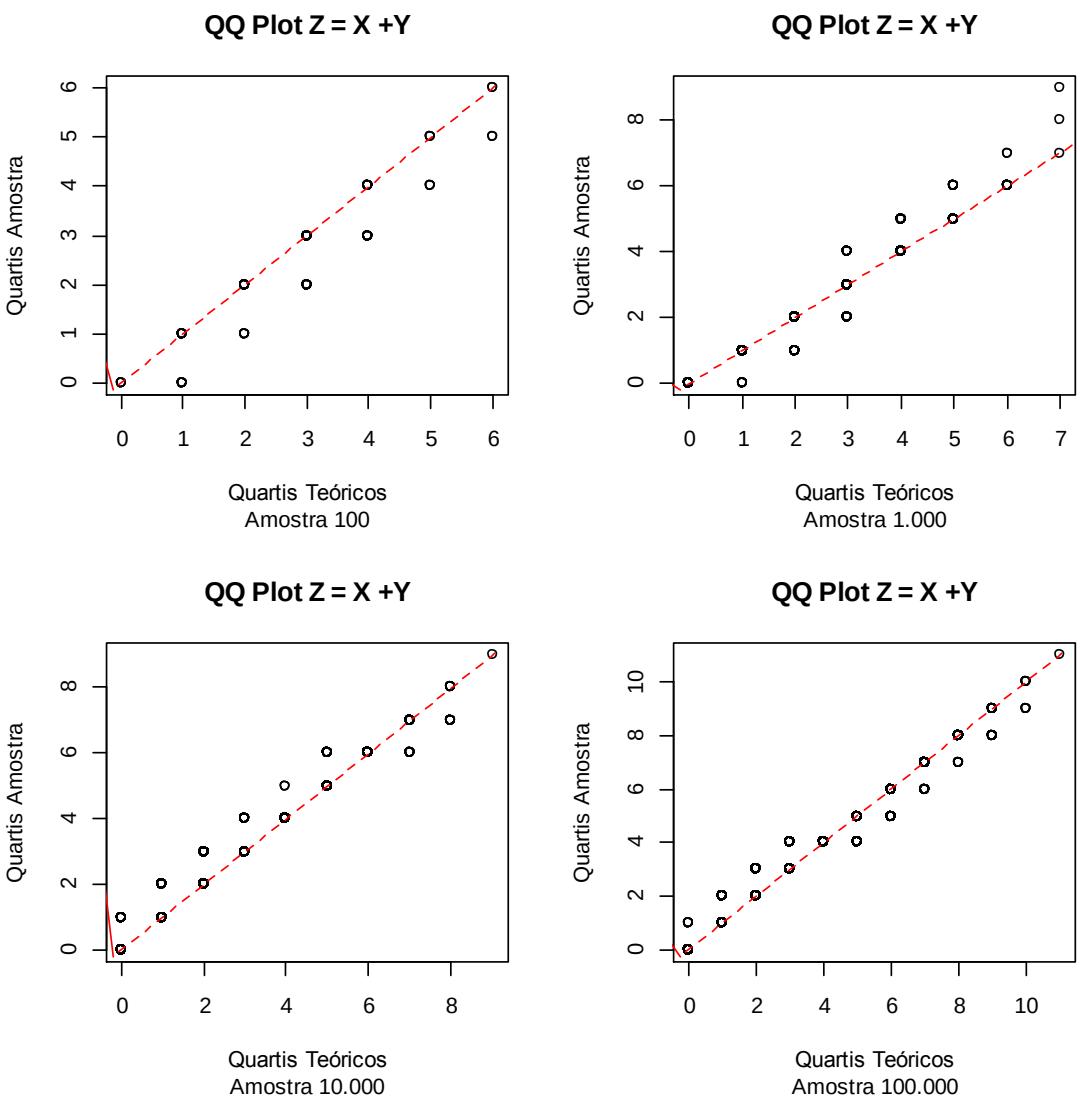








Método: Inversa – Z=X + Y: algoritmo da resolução 1



6. Conclusões

A realização deste trabalho permitiu compreender e aprofundar conhecimentos relativos à geração de variáveis aleatórias discretas. Neste estudo foram aplicados 2 métodos de aplicação genérica: o método da inversão e da rejeição e calculados 9 algoritmos.

Ambos os métodos tiveram bons resultados, isto é, obteve-se amostras que correspondem às funções distribuições teóricas esperadas. Analisando o *p-value* do teste do qui-quadrado ou efetuando uma avaliação gráfica (comparação gráfico de barras/função de probabilidade e QQ plots), de uma forma geral, pode concluir-se que o aumento da amostra permite um incremento do ajustamento à distribuição teórica. No entanto, nem sempre acontece.

A utilização do teste do qui-quadrado permitiu verificar com exatidão a qualidade de ajustamento, uma vez que os métodos gráficos estão sujeitos a alguma subjetividade.

Em termos de métodos aplicados, não foi possível encontrar o método mais eficaz (medido em termos de *p-value*) na geração de amostras de igual dimensão. No entanto, o método da rejeição, tendo em conta os algoritmos desenvolvidos, revelou-se mais exigente quer ao nível da operacionalização, quer ao nível de tempo de processamento[4]. Apesar dos métodos terem igual exatidão, o método da inversão revela-se melhor em termos de eficiência (mais rápido) e de complexidade (mais simples de operacionalizar), pelo que se pode concluir que é o melhor na geração destas variáveis aleatórias discretas considerando os algoritmos desenvolvidos e o *software* utilizado.

[4] Esta foi a razão pela qual não se apresentou resultados para uma amostra de dimensão 100.000.

7. Bibliografia

Acetatos do Módulo de Simulação de Estatística Aplicada, professor Jorge Pereira.

Livros

- Ross, S.M., Simulation, 5ª Edição, Academic Press , 2013
- Fishman , George S. , Monte Carlo - Concepts, Algorithms and Applications, Springer-Verlag New York, Inc., 1996
- Kroese ,Dirk P., Monte Carlo Methods, Department of Mathematics School of Mathematics and Physics The University of Queensland,2011[5]
- Figueiredo, Fernanda et al, *Estatística Descritiva e Probabilidades – Problemas resolvidos e propostos com aplicações em R*, Escolar Editora, 2007

Outros materiais

- Generating discrete random variables, Math 276 Actuarial Models, Spring 2008 semester, EA Valdez, University of Connecticut – Storrs, Lecture Week 3
- Provete et al, Estatística aplicada à ecologia usando o R, São José do Rio Preto, SP, Abril 2011

[5] Lecture notes for a graduate course on Monte Carlo methods given at the 2011 Summer School of the Australian Mathematical Science Institute (AMSI)