Gerador de números aleatórios

Lucas Monteiro (fc52849@alunos.fc.ul.pt)

Implementação e teste de diversas sequências de números pseudo-aleatórios geradas por geradores lineares congruentes, utilizando o teste do quadrado, cubo e χ^2; e estudo da distribuição de probabilidade não uniforme dum círculo.

Introdução

Um gerador de números aleatórios (***gna***) é a geração de uma sequência de valores preferencialmente descorrelacionados entre eles e uniformemente distribuídos num certo intervalo. Os números *aleatórios* gerados durante este trabalho vão sê-lo através de algoritmos, logo a geração não é verdadeiramente aleatória, daí chamar-se a este tipo de algoritmos geradores de números pseudo-aleatórios. Acrescenta-se assim ás características de um bom *gna* o facto de ter um longo período ou a capacidade de gerar um grande número de valores até estes se repetirem criando, assim, um ciclo.

1. Implementação de um gerador linear congruente

Começou-se por escrever o código para gerar números aleatórios usando o método congruente (***MC***). Este tipo de gerador é definido pela relação de recorrência (1):

$$X_{n+1} = (c * X_n + a) \bmod p \qquad (1)$$

,onde **c** é o multiplicador, **a** o desvio e **p** o módulo escolhido. Também se define o valor inicial ($\mathbf{X_0}$) ou *seed* no intervalo $[0; p[$ e **n** o número de valores aleatórios que se pretende gerar.

Para testar o código implementado escolheu-se valores relativamente baixos ($<$50) para os parâmetros c e p, nomeadamente $c = 3$ e $p = 31$. Os valores por defeito dos restantes parâmetros foram definidos como $a = 0$, $X_0 = 7$ e $n = 100$. Chamou-se esta sequência de (i).

Para verificar a correlação entre os números gerados seguiu-se para o teste do quadrado, que consiste em desenhar um gráfico bi-dimensional onde as coordenadas de cada ponto correspondem a dois números gerados seguidos (X_n e X_{n+1}). Com este método pretende-se melhor visualizar os dados e perceber a sua correlação, ou descorrelação, pela emergência de padrões, por exemplo retas, no gráfico.

Visualizou-se então os números gerados pelo *MC* utilizando os parâmetros já definidos. Estes claramente estão fortemente correlacionados, visto que várias retas se notam no gráfico. Notou-se que bastavam cerca de uma dezena de pontos para o padrão começar a formar-se e que o período da sequência era 30.

Também foi utilizado o método do cubo que é em tudo semelhante ao método do quadrado exceto que em três dimensões em vez de duas. Como era de esperar também neste teste se nota a correlação entre os números gerados. Ambos os métodos estão representados na Figura 1.

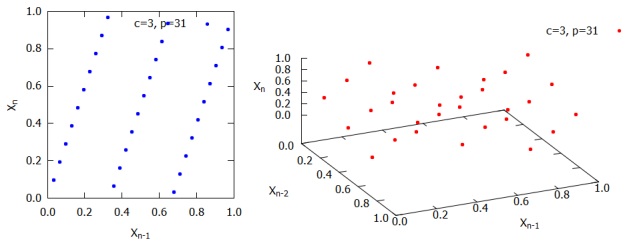

Figura 1. Visualização da correlação dos pontos gerados pelo método congruente com c = 3 e p = 31, utilizando o método do quadrado (á esquerda) e do cubo (á direita)

Visto que estes parâmetros produziam valores correlacionados seguiu-se para o teste de outros, nomeadamente (ii) $c = 8121$, $p = 134456$ e (iii) $c = 16807$, $p = 2^{16}$.

Na sequência (ii) nota-se ainda correlação entre os pontos, embora menos que na (i), no entanto na (iii) obteve-se resultados 'bons', com números aparentemente descorrelacionados. O teste do quadrado de (ii) e (iii) podem se visualizar na Figura 2.

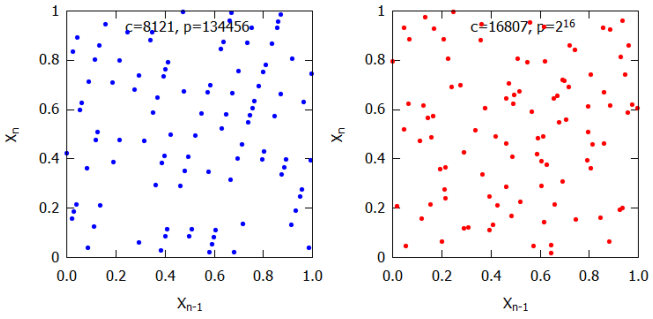

Figura 2. Teste do quadrado em valores gerados pelo método congruente com $c = 8121$, $p = 134456$ e $c = 16807$, $p = 2^{16}$

Também se testou outros *gna* implementados em C++, nomeadamente o (iv) ***rand()*** e o (v) ***drand48()***, obtendo também resultados descorrelacionados.

2. Gerar pontos uniformemente num círculo

Seguidamente, pretendeu-se gerar pontos num círculo sem rejeição. Para isso implementou-se o código (2) para transformar coordenadas polares em cartesianas, onde $\boldsymbol{r}$ e $\boldsymbol{\theta}$ são valores do intervalo $[0; 1[$ gerados aleatoriamente.

$$x = r\cos(2\pi\theta) \text{ e } y = r\sin(2\pi\theta)\ ; \quad r, \theta \in [0; 1[\qquad (2)$$

Foram gerados 500 números aleatórios em círculo utilizando o método congruente com $c = 16807$, $p = 2^{16}$ e a transformação (2). Para cada parâmetro do circulo (r e θ) foi utilizada uma *seed* diferente, $X_{r0} = 0$ e $X_{\theta 0} = 59169$, pois, caso contrário, r e θ aleatórios estariam correlacionados.

Notou-se então, como se observa na Figura 4, que a distribuição dos pontos não é uniforme. Isto deve-se ao fato dos pontos serem uniformemente gerados em r e θ. Assim um ponto aparecer numa circunferência de raio r tem o dobro da probabilidade de aparecer numa de $2r$ pois o comprimento da circunferência cresce linearmente com r.

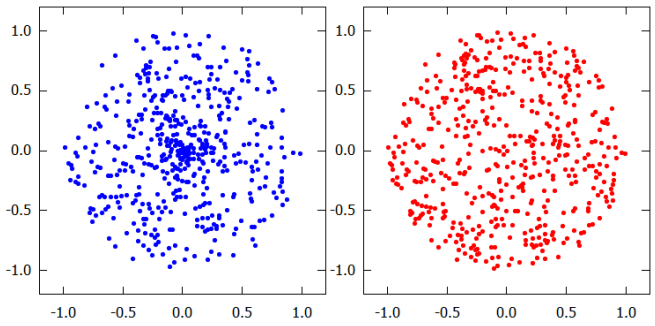

Figura 3. Representação gráfica de pontos gerados não uniforme e uniformemente num círculo sem rejeição, pelo *MC*

Queremos então que a função densidade de probabilidade (**f(r')**) cresça linearmente com r'. Sabendo que esta tem área igual a 1 e $r' \in [0;1[$ temos que $f(r') = 2r'$. Sabe-se que se f(r') é contínua, e a sua função distribuição acumulada é invertível, então esta pode ser simulada a partir de uma distribuição uniforme. Tomamos então o inverso da função distribuição acumulada ($\boldsymbol{F^{-1}(r)}$) para obter a transformação que transforma r em r'.

$$F(r') = \int_0^{r'} f(r')dr' = r'^2 \Rightarrow r' = F^{-1}(r) = \sqrt{r}. \quad (3)$$

Substituindo em (2) o r por r' obtemos a distribuição uniforme no círculo (4).

$$x = \sqrt{r}\cos(2\pi\theta) \text{ e } y = \sqrt{r}\sin(2\pi\theta)\ ;\ \ r, \theta \in [0;1[\quad (4)$$

3. Teste do χ^2

O teste do quadrado e do cubo são bons para visualizar a correlação entre os valores, no entanto, é difícil de entender o quão correlacionados estes estão.

Para isso fazemos o Teste χ^2 de Pearson, que consiste em dividir o intervalos de valores gerados em $\boldsymbol{k}$ intervalos de tamanho igual e, uma vez gerados os n valores aleatórios, ver quantos destes ($\boldsymbol{N_i}$) estão compreendidos nos k intervalos i. Sabendo isto, calcula-se o valor de χ^2 pela fórmula (5), onde $\boldsymbol{p_i}$ é a probabilidade de um número estar num intervalo i. Como neste caso a distribuição é uniforme temos que $p_i = \frac{1}{k}$.

$$\chi^2 = \sum_{i=1}^{k} \frac{(N_i - np_i)^2}{np_i} \overset{p_i=\frac{1}{k}}{=} \sum_{i=1}^{k} \frac{k(N_i - \frac{n}{k})^2}{n} \quad (5)$$

Utilizando os valores predefinidos dos parâmetros a, X_0 e n, e com $np_i \geqslant 5$, calculou-se o χ^2 de cada um dos *gna* anteriormente descritos, com $k = 10$ e para diferentes n. Os valores obtidos foram representados na Tabela 1, onde os valores de χ^2 estão em baixo, com $n = 100$ á direita, $n = 500$ no centro e $n = 1000$ á esquerda, de cada sequência. Os valores foram todos arredondados por defeito á primeira casa decimal.

Tabela I. Valores de χ^2 para diferente sequências de números aleatórios, onde $n = 100$, $n = 500$ e $n = 1000$, respetivamente

(i)			(ii)			(iii)			(iv)			(v)		
1.	0.2	0.1	2.8	1.	0.2	7.4	7.6	4.9	4.2	6.4	4.8	12.	8.2	14.6

Utilizando os valores da tabela em anexo para $k = 10$, ou 9 graus de liberdade, vemos que procuramos 'bons' valores de χ^2 entre 6 e 8, rejeitando logo á partida sequências com valores abaixo de 3 e acima de 17. Salienta-se também que é importante testar para vários valores de n pois um gerador pode aparentar ser 'bom' para certos valores de n mas não o ser para outros, como acontece com a sequência (iv).

Assim, rejeita-se logo á partida os geradores (i) e (ii) por apresentarem valores de χ^2 muito baixos. Os geradores (iv) e (v) apresentam resultados razoáveis, embora mais baixos e altos, respectivamente, dos desejáveis; e a sequência (iii) é a que apresenta melhores resultados.

Ainda se testou o gerador (iii) com $n = 500$, $k = 10$ e diferentes valores de X_0, como se vê na Tabela 2. Os valores oscilaram bastante ao ponto de haverem valores muito próximos de 0 e outros muito altos ($\chi^2 = 500$).

Tabela II. Valores de χ^2 de (iii) para diferentes valores de X_0

X_0	3	7	12	365	1872	25771	59169	117649	248832
χ^2	10.	7.6	10.	5.2	0.1	9.9	5.7	6.7	500

Conclusão

Com este exercício compreende-se a necessidade de testar diferentes geradores de modo a escolhermos o mais indicado. Notou-se que os geradores são muito sensíveis ás escolhas dos diferentes parâmetros o que leva a um cuidado adicional na sua escolha. Dos testados, viu-se que o gerador mais indicado foi o do *MC* com $c = 16807$, $a = 0$, $p = 2^{16}$ e $X_0 = 7$.

Também se mostrou que é possível impor uma distribuição uniforme de pontos aleatórios sobre uma distribuição não uniforme, através de uma transformação, de modo a, por exemplo, gerar pontos uniformemente num círculo.