Федеральное государственное бюджетное образовательное учреждение

Высшего профессионального образования

"Тольяттинский государственный университет"

Институт математики, физики и информационных технологий

Кафедра Прикладной математики и информатики

Профиль Математическое моделирование

Отчет по самостоятельной работе

по дисциплине "Получисленные алгоритмы"

на тему: Эмпирические критерии проверки случайных последовательностей

Выполнила: Парамонова К.С.

группа ПМИм-1301

Принял: Мельников Б.Ф.

Тольятти 2014 г.

***Содержание***

Эмпирические критерии проверки случайных последовательностей. Критерий частот, серий, интервалов, разбиений, перестановок, монотонности, конфликтов

Линейный конгруэнтный метод

Основные критерии проверки случайных наблюдений

Эмпирические критерии

Критерий серий

Покер-критерий (критерий разбиений)

Критерий перестановок

Критерий монотонности

Критерий конфликтов

***Эмпирические критерии проверки случайных последовательностей. Критерий частот, серий, интервалов, разбиений, перестановок, монотонности, конфликтов***

Числа, которые выбираются случайным образом, находят множество полезных применений: моделирование.

Это позволяет сделать модели похожими на реальные явления; выборочный метод. Часто невозможно исследовать все варианты, но случайная выборка обеспечивает понимание того, что можно назвать "типичным поведением"; численный анализ.

Для решения сложных задач численного анализа была разработана техника, использующая случайные числа; компьютерное программирование. Случайные числа являются хорошим источником данных для тестирования эффективности компьютерных алгоритмов; принятие решений; эстетика; развлечения.

Равномерным распределением на конечном множестве чисел называется такое распределение, при котором любое из возможных чисел имеет одинаковую вероятность появления. Если не задано определенное распределение на конечном множестве чисел, то принято считать его равномерным [3]. Джон фон Нейман первым предложил в 1946 г. идею возвести в квадрат случайное число и выделить из него средние цифры. Например, необходимо получить новое десятизначное число по числу *5772156649*. Возводим имеющееся число в квадрат и выделяем средние 10 цифр:

.

Новым сгенерированным числом будет число *7923805949.* Эта последовательность не случайна, она кажется такой. Последовательности, генерируемые детерминистическим путем, таким как этот, называются псевдослучайными или квазислучайными.

Метод середины квадратов фактически является сравнительно бедным источником случайных чисел. Опасность состоит в коротком цикле (периоде) повторяющихся элементов. Рассмотрим методы генерирования последовательности случайных дробей [3], т.е. случайных действительных чисел *Un*, равномерно распределенных между нулем и единицей. Будем генерировать целое число *Хn* между нулем и некоторым числом *U*, тогда дробь при m >= U

*Un = Хn / m*

будет принадлежать [0,1]. Обычно m выбирают равным размеру компьютерного слова. Обозначим *m = be*, где *b* - основание системы счисления, используемой ЭВМ, а *e* - число разрядов машины. Поэтому *Хn* можно по традиции рассматривать как целое число, занимающее все компьютерное слово с точкой в правом конце слова, а *Un* - как содержимое того же слова с точкой в левом конце слова.

# ***Линейный конгруэнтный метод***

Выберем четыре числа [3]:

*m* - модуль, *m > 0*;

*a* - множитель, *0 Ј a <* *m*;

*c -* приращение, *0 Ј с < m*;

*Х0 -* начальное значение, *0 Ј Х0 < m*.

Последовательность случайных чисел *{Хn}* можно получить, полагая

*Хn+1 = (a Хn + c)* mod *m, n і 0*, (4.1)

где mod - операция взятия остатка от деления. Последовательность (4.1) называется линейной конгруэнтной последовательностью. Например, для *m = 10* и *Х0 = a = c = 7* получим последовательность 7, 6, 9, 0, 7, 6, 9, 0, 7…

В примере отражен факт, что конгруэнтная последовательность всегда образует петли, т.е. обязательно существует цикл, повторяющийся бесконечное число раз. Это свойство является общим для всех последовательностей вида *Xn+1 = f (Xn),* где ***f*** - функция, преобразующая конечное множество само в себя. Повторяющиеся циклы называют периодами. Задача генерации случайных последовательностей состоит в использовании относительно длинного периода, что связано с выбором довольно большого ***m***, так как период не может иметь больше ***m*** элементов. Другой фактор - скорость генерирования. При выборе множителя ***а*** следует принимать во внимание выводы следующей теоремы.

*Пример.* Для ***m = 120*** назначить ***а*** и построить последовательность случайных чисел в интервале **[0,1].**

*Решение*. Выберем *с* = 7, *a* = 5, тогда при *x0*= 1 получим следующие числа: *X1**= 12, X2 = 67, X3**= 102, …* Далее генерируем *u0*= 1/120, *u1*= 12/120, *u2*= 67/120, *u3*= 102/120, …

В связи с довольно простым получением случайных чисел возникает такая общепринятая ошибочная идея - достаточно взять хороший генератор случайных чисел и немного его модифицировать для того, чтобы получить "еще более случайную" последовательность. Часто это не так. Например, известно, что формула (4.1) позволяет получить более или менее хорошие случайные числа. Может ли последовательность, полученная из

*Xn+1**= ( (a Xn)* mod *(m + 1) + c)* mod *m*

быть еще более случайной? Ответ: новая последовательность является менее случайной с большей долей вероятности, т.е. мы попадаем в область генераторов типа *Xn+1**= f (Xn)* с выбранной произвольно функцией *f*. Доказано, что такие последовательности, ведут себя хуже, чем последовательности, полученные при использовании более регулярных функций (4.1).

Отметим, что период линейной конгруэнтной последовательности довольно длинный; когда ***m*** приблизительно равно длине слова компьютера, обычно получаются периоды порядка *109* или больше, а в типичных вычислениях используется лишь малая часть всей последовательности.

Рассмотрим аддитивный генератор, предложенный Дж.Ж. Митчелом и Д.Ф. Муром:

*Xn**= (Xn-24 + Xn-55)* mod *m, n і 55*, (4.2)

где *m* - четное число, числа *24* и *55* - специальные числа, выбранные для того, чтобы определить такую последовательность, младшие значащие двоичные разряды (*Xn**mod 2*) которой имеют длину периода *255 - 1*.

Алгоритм В. (Аддитивный генератор чисел). В ячейки памяти *Y [1], Y [2], …, Y [55]* записано множество значений *X54*, *X53*, …, *X0* соответственно; в начале полагают ***j*** = 24, ***k*** = 55.

При реализации этого алгоритма на выходе последовательно получаем числа *X55*, *X56*, …

В1. [Суммирование] (Если на выходе мы оказываемся в точке *Xn*, то

*Y [j]* = *Xn-24*, а *Y [k]* = *Xn-55*.

*Y [k] ¬ (Y [k] + Y [j])* mod *2e*.

В2. [Продвижение] Уменьшим *j* и *k* на 1. Если *j = 0*, то присвоим *j ¬ 24*, иначе, если *k = 0*, присвоить *k ¬* *55*.

Числа 24 и 55 в выражении (4.2) называются запаздыванием, а числа *Xn*, определенные в (4.2) - последовательностью Фибоначчи с запаздыванием.

Возможно построить достаточно хороший генератор случайных чисел, используя всевозможные линейные комбинации *Xn-1*, …, *Xn-k* для малых *k*. В этом случае наилучший результат получается, когда модуль ***m*** является большим простым числом; например, **m -** наибольшее простое число, которое можно записать одним компьютерным словом.

Формула для генерации может быть выбрана такой:

*Xn**= (a1 Xn-1 + … + ak Xn-k)* mod *p* (4.3)

с периодом *pk - 1*. Здесь *X0, X1, …, Xk-1* могут быть выбраны произвольно, но не равные нулю одновременно.

Обратимся к процессу генерирования случайных чисел "0" и "1" по формуле (4.3). Зададим произвольно ненулевое двоичное слово (1 0 1 1), то есть ***X1*** *= 1, X****2*** *= 0,* ***X3*** *= 1,* ***X4*** *= 1;* ***k***= 4; ***p*** = 2. Зададим произвольно коэффициенты ***а1*, *а2*, *а3*, *а4*** двоичным словом (0 0 1 1), то есть ***а1* *= 0, а2 = 0, а3 = 1, а4 = 1*.** Перепишем (4.3) с учетом введенных величин, получим:

*Xn**= (a1 ЧXn-1 + a2 ЧXn-2 + a3 ЧXn-3 + a4 Ч Xn-4)* mod *2*

*Xn**= (0 ЧXn-1 + 0 ЧXn-2 + 1 ЧXn-3 + 1 Ч Xn-4)* mod *2*

***Xn = (Xn-3 + Xn-4)* mod *2.***

Откуда следует, что

*X5**= (X2 + X1)* mod *2 = 1.6**= (X3 + X2)* mod *2 = 1; X7**= (X4 + X3)* mod *2 = 0;8**= (X5 + X4)* mod *2 = 0; X9**= (X6 + X5)* mod *2 = 0;10**= (X7 + X6)* mod *2 = 1; X11**= (X8 + X7)* mod *2 = 0;12**= (X9 + X8)* mod *2 = 0*; … и т.д.

Сгенерированные числа (коды) имеют значения:

(начальное число); 1100; 0100; 1101; 0111; 1000; 1001 …

Период, т.е. число шагов, через которое начинается повтор, равняется *24 - 1 = 15*.

# ***Основные критерии проверки случайных наблюдений***

Статистические критерии [3] отвечают на вопрос: достаточно ли случайной будет последовательность. Если критерии *T1, T2, …, Tn* подтверждают, что последовательность ведет себя случайным образом, это еще не означает, вообще говоря, что проверка с помощью *Tn+1*-го критерия будет успешной. Однако каждая успешная проверка дает больше оснований, подтверждающих случайность последовательности. Обычно к последовательности применяется несколько статистических критериев и, если она удовлетворяет этим критериям, то последовательность считается случайной.

Различают два вида статистических критериев: *эмпирические и теоретические.* Эмпирические критерии основаны на использовании определенных статистик. Теоретические критерии основаны на использовании теоретико-числовых методов. Рассмотрим критерий "хи - квадрат" (c2-критерий) [3]. Он является основным эмпирическим критерием, используемым в сочетании с другими критериями. Прежде чем рассматривать идею в целом, проанализируем частный пример применения c2-критерия к бросанию игральной кости.

Используем 2 игральные кости, каждая из которых независимо допускает выпадение значений 1, 2, …, 6 с равной вероятностью. В следующей таблице дана вероятность получения определенной суммы ***S*** при одном бросании игральных костей:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Значение *S* | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| Вероятн. *PS* |  |  |  |  |  |  |  |  |  |  |  |

Имеем всего *36* возможных результатов бросания. Рассмотрим результаты бросания. Например, величина ***4*** может быть получена тремя способами: ***1 + 3, 2 + 2, 3 + 1***. Это составляет . Аналогично определяются оставшиеся вероятности *PS*. Если бросать игральную кость ***n*** раз, то в среднем мы должны получить величину ***S*** примерно ***n Ч PS*** раз, но это не совсем так. Например, при ***144***бросаниях были получены следующие результаты:

Таблица 4.1

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Величина *S* | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| Наблюдаемое число, *YS* | 2 | 4 | 10 | 12 | 22 | 29 | 21 | 15 | 14 | 9 | 6 |
| Ожидаемое число, *n pS* | 4 | 8 | 12 | 16 | 20 | 24 | 20 | 16 | 12 | 8 | 4 |

Заметим, что во всех случаях наблюдаемое число отличается от ожидаемого. Введем в рассмотрение число :

 (4.6)

Эта статистика называется статистикой "хи - квадрат" наблюдаемых значений *Y2 … Y12* при бросании игральных костей. Используя табл.4.1, получим:

.

Формулу (4.6) перепишем следующим образом:

,(4.7)

Так как

,

*Y1 + Y2 +…+ Yk = n*, *p1 + p2 +…+ pk =1*.

Чтобы воспользоваться c2 - статистикой проводят несколько экспериментов, затем вычисляют числа . Далее используют известные таблицы c2-распределения имеющие вид [4]:

Таблица 4.2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | p = 99% | p = 95% | p = 75% | p = 50% | p = 25% | p = 5% | p = 1% |
| … |  |  |  |  |  |  |  |
| 2,5583,9406,7379,34212,5518,3123,21 |  |  |  |  |  |  |  |
| … |  |  |  |  |  |  |  |

Здесь *p* - процентные точки c2 - распределения; m *= k - 1* - число степеней свободы, что на единицу меньше, чем число категорий. Внутри клеток таблицы стоят числа .

Предположим, что были сделаны три эксперимента с генерированием случайных последовательностей и получены числа:

, , .

Сравнивая эти величины со значениями таблицы 4.2 при 10 степенях свободы, мы видим, что *1*гораздо больше, чем 23.21, а это может произойти только в 1% случаев. В связи с этим эксперимент 1 демонстрирует значительное отклонение от случайного поведения. *2*показывает не лучшие свойства, так как результаты слишком близки к ожидаемым. Наконец, значение *3* находится между 75 и 50 - процентной точками. Таким образом, наблюдение является удовлетворительно случайным по отношению к этому критерию.

Отметим, таблица 4.2 - это только приближенные значения c2 распределения, которое является предельным распределением случайной величины  формулы (4.7).

Поэтому табличные значения близки к реальным только при больших ***n***.

Насколько большими должны быть ***n***? Эмпирическое правило гласит: нужно взять ***n*** настолько большим, чтобы все значения ***n pS*** были больше или равны пяти.

# ***Эмпирические критерии***

Эмпирические критерии традиционно применяются для проверки, будет ли последовательность случайной.

Обычно каждый такой критерий применяется к последовательности

*{Un} = U0, U1, U2, …* (4.8)

действительных чисел, которые предполагаются независимыми и равномерно распределенными в интервале (0,1).

Если критерии используются для целочисленных последовательностей, то используется вспомогательная последовательность

*{Yn} = Y0, Y1, Y2, …,* (4.9)

определенная правилом

*Yn = [d Ч Un] (*4.10)

Это последовательность целых чисел, распределенных в интервале (*0, d-1).* Число ***d*** выбирается таким образом, чтобы сделать все *Yi* - целыми. Обычно ***d*** выбирается достаточно большим, но не настолько большим, чтобы критерий стал практически неприменим.

*Критерий равномерности (критерий частот)*

Первое требование, предъявляемое к последовательности (4.8) состоит в том, чтобы ее члены были числа, равномерно распределенные между *0* и *1*. Существуют *2* способа проверить это:

. Использовать критерий Колмогорова-Смирнова [2].

. Использовать c2-критерий.

Для того чтобы применить c2-критерий, используется последовательность (4.10) вместо (4.8). Для каждого ***r***, *0 Ј r < d*, подсчитывается число случаев, когда *Yj = r*. Затем применим c2 - критерий, принимая число категорий *k = d,* а вероятности *pS = 1/d* для каждой категории. *d* - число, равное, например, 64 или 128.

# ***Критерий серий***

Более общее требование к последовательности состоит в том, чтобы пары последовательных чисел были равномерно распределены независимым образом. В критерии серий подсчитывается число случаев, когда пара (*Y2j, Y2j+1) = (q, r)* для *0 Ј j < n*.

Такая операция осуществляется для каждой пары целых чисел *q* и *r*, таких, что *0 Ј q< r < d*. Затем применяется c2 - критерий к этим *k = d2* категориям, где *1/d2* - вероятность отнесения пары чисел к каждой из категорий. При этом *d* выбирается таким образом, чтобы *n>>k*, например *n і 5d2*.

# ***Покер-критерий (критерий разбиений)***

"Классический" покер-критерий рассматривает *n* групп по пять последовательных целых чисел

*{Y5j, Y5j+1, Y5j+2, Y5j+3, Y5j+4}* для *0 Ј j < n*

и проверяет, какие из следующих семи комбинаций соответствуют таким пятеркам чисел (порядок не имеет значения):

Все числа разные: *a b c d e*

Одна пара: *a a b c d*

Две пары: a a b b c

Три числа одного вида: a a a b c

Полный набор: a a a b b

Четыре числа одного вида: a a a a b

Пять чисел одного вида: a a a a a

c2 - критерий основан на подсчете числа комбинаций пятерок в каждой из семи категорий имеющих место в n - группах, состоящих из 5 элементов.

В [3] можно ознакомиться с другими известными критериями, которые традиционно применяются для проверки, будет ли последовательность случайной.

Возникает вопрос: "Зачем применять такое количество критериев?" Необходимость проверки последовательности с помощью нескольких критериев позволяет сделать вывод о случайности генерируемых чисел, а значит и точности моделирования процессов с их использованием.

*Критерий интервалов*

случайная последовательность конгруэнтный интервал

Этот критерий используется для проверки длины "интервалов" между появлением *Uj* на определенном отрезке. Если α и β - два действительных числа, таких что 0≤α≤β≤1, то рассмотрим длины подпоследовательностей *Uj, Uj+1,… Uj+r, в* которых *Uj+r* лежит между α и β, а другие *Us* не лежат между этими числами.

# ***Критерий перестановок***

Разделим последовательность на выходе на n групп по t элементов в каждой, т.е. (*Ujt, Ujt+1, Ujt+t-1*) для 0 ≤ j < п. Элементы в каждой группе можно упорядочивать t! различными способами. Подсчитывается число групп с любым возможным порядком и применяется c2 - критерий с *к = t!* возможными категориями и вероятностью *1/t!* для каждой категории.

Например, если t = 3, существует шесть возможных категорий, а именно *U3j< U3j+1< U3j+2* или *U3j< U3j+2<U3j+1* или …, или *U3j+2< U3j+1< U3j.* В этом критерии предполагается, что U*s* не могут быть равны между собой, т.е. вероятность того, что *Us* **=** *Uj* при s ≠j равна нулю.

# ***Критерий монотонности***

В последовательности можно проверять распределение восходящих и нисходящих серий. Имеется в виду проверка распределений длин монотонных частей заданной последовательности (отрезки возрастания или убывания).

В качестве примера точного определения серии рассмотрим последовательность цифр "1298536704". Проводя вертикальные линии слева, справа, а также между Xj и Xj+i всякий раз, когда Xj > Xj+i, получим

|12 9 | 8 | 5 | 3 6 7 | 0 4 |. (9)

Таким образом выделяются восходящие серии: сначала - серия длиной 3, затем - две серии длиной 1, затем - снова серия длиной 3 и, наконец, серия длиной 2.

# ***Критерий конфликтов***

c2 - критерий можно применять только тогда, когда ненулевое число элементов ожидается в каждой категории. Но существует критерий другого вида, который можно использовать, когда число категорий намного больше числа наблюдений. Этот критерий имеет отношение к рандомизации - важному методу информационного поиска.

Предположим, что имеется т урн, и поместим в них наудачу п шаров, причем т намного больше *n*. Большинство шаров попадет в пустые урны, но если шар попадет в урну, в которой уже содержится хотя бы один шар, то будем говорить, что произошел "конфликт”. Критерий конфликтов подсчитывает число конфликтов, и генератор удовлетворяет этому критерию, если не возникает слишком много или слишком мало конфликтов.

Для определенности предположим, что т = 220 и п = 214. Тогда в среднем на 64 урны приходится только один шар. Вероятность того, что в конкретную урну попадет ровно к шаров, равна pk - т~1) п~к, поэтому среднее число конфликтов в урне равно.