Исследование некоторой псевдослучайной последовательности

Ульянова Анастасия Евгеньевна, гр. 522

Санкт-Петербургский государственный университет
Математико-механический факультет
Кафедра статистического моделирования

Научный руководитель: к.ф.-м.н. Товстик Т.М.
Рецензент: к.ф.-м.н. Москалева Н.М.

Санкт-Петербург
2007г.

Постановка задачи

- Дана некоторая псевдослучайная последовательность, описанная в статье Товстик Т.М. от 2006г.
- Необходимо провести тестирование данной последовательности, описать ее преимущества.
- По возможности модифицировать данный генератор псевдослучайных чисел (далее ГПСЧ), генерирующий эту последовательность.

Алгоритм Холтона

Определение

Одномерной последовательностью Холтона называется последовательность $h_m(i)$, где

$$i = 0, 1, 2, 3, \ldots, \quad m \in \mathbb{N}, \quad m \geq 2, \quad 0 \leq h_m(i) \leq 1.$$

Для получения $h_m(i)$ используется следующий алгоритм:

1. $h_m(0) = 0$
2. В m-ичной системе счисления число i записывается в виде:

$$i = a_j a_{j-1} \ldots a_1, \quad 0 \leq a_k \leq m - 1, \quad k \in [1, j].$$

3. Соответствующее псевдо-случайное число по алгоритму Холтона равно:

$$h_m(i) = 0, a_1 a_2 \ldots a_j.$$

Связь чисел i и $h_m(i)$ в десятичной и m-ичной системах такова:

$$i = \sum_{k=1}^{j} a_k m^{k-1}, \quad h_m(i) = \sum_{k=1}^{j} a_k m^{-k}.$$

Основной алгоритм генерации последовательности

В одномерном случае

Замечание

В алгоритме Холтона число i раскладывалось в m-ичную систему счисления.

Данный алгоритм является модификацией алгоритма Холтона:

1. Вместо i будем брать следующие значения

$$t_i = [i\sqrt{L \cdot i}],$$

   где $L \in \mathbb{N}$
2. Применим к t_i алгоритм Холтона.
3. Получим последовательность $x_i = h_m(t_i)$.

Замечание

В многомерном случае у различных одномерных последовательностей в данном алгоритме должны быть разные параметры m и L.

Параметризованный алгоритм

- Данный алгоритм является моей модификацией основного алгоритма:
- Вместо $t_i = [i\sqrt{L \cdot i}]$, будем брать следующие значения:

$$t_i = [i(L \cdot i)^k], \text{где } k \in [0, 1).$$

Замечание

При $k = \frac{1}{2}$ параметризованный алгоритм принимает вид алгоритма генерации последовательности.

Варианты выборок

Для тестирования были сгенерированы следующие последовательности

- Длиной в 100 000 по всем алгоритмам
- Выборки в 50 элементов, взятые из начала, середины и конца выборки в 100 000 элементов
- Выборки длиной в 100, 500, 1000 элементов.

А так же, для сравнения, с помощью встроенного ГПСЧ ОС Linux, были сгенерированы последовательности аналогичной длины.

Вычисление коэффициентов корреляции последовательности

Один из самых важных показателей, на который делался акцент при построении основного алгоритма.

Определение

Коэффициенты корреляции последовательности вычисляются по следующей формуле:

$$cor_k = \frac{\sum_{i=1}^{N}(x_i - \overline{x})(x_{i+k} - \overline{x})}{\sum_{i=1}^{N}(x_i - \overline{x})^2},$$

где $0 \leq k \leq M,\ M < N$.

Были сгенерированы последовательности с $2 \leq m \leq 10000$ и $1 \leq L \leq 1000$ длиной в 100 000 элементов и отобраны 4 выборки, обладающие самыми малыми и самыми большими попарными корреляциями.

Результаты теста на попарные корреляции

Таблица: Модули коэффициентов корреляций, выборка в $N = 100$ элементов, $M = 50$

m	L	Max
343	7	0.20
547	27	0.18
431	43	0.18
123	4	0.12
Встроенный ГПСЧ Linux		0.34

Таблица: Модули коэффициентов корреляций, выборка в $N = 100\ 000$ элементов, $M = 99\ 900$

m	L	Max
343	7	3.7410^{-3}
547	27	11.9010^{-3}
431	43	12.2610^{-3}
123	4	14.2810^{-3}
Встроенный ГПСЧ Linux		$43.13 \cdot 10^{-3}$

График коэффициентов корреляции

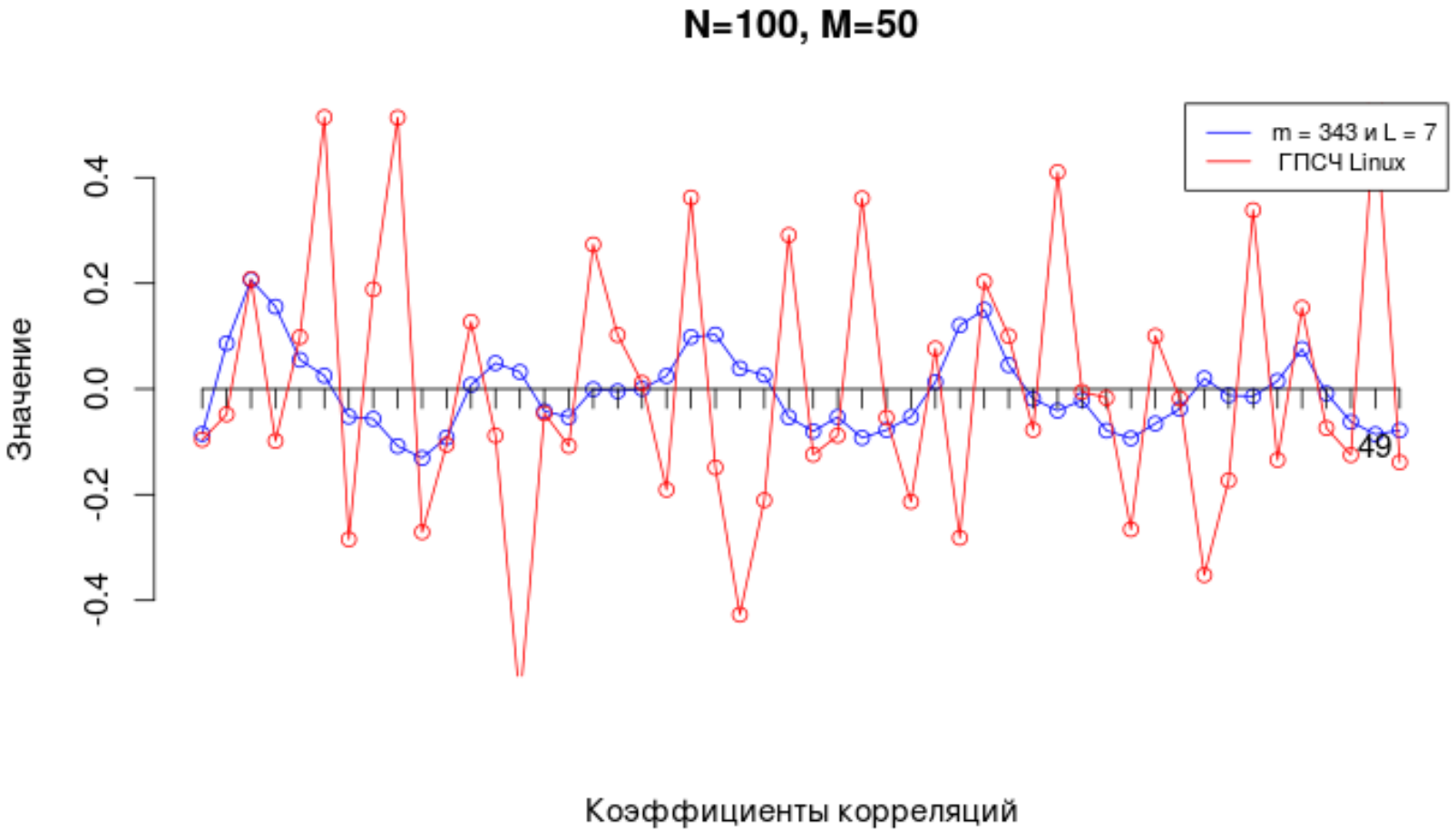

Рис.: График коэффициентов корреляции последовательности с параметрами $m = 343$ и $L = 7$ и генератора Linux, $N = 100, M = 50$

Другие статистические тесты

Дополнительные тесты

1. Оценка математического ожидания
2. Проверка на равномерность меры распределения

Результат

- Для выборок в 50, 100, 500 и 1000 значений по основному алгоритму отклонение от теоретической оценки прохождения каждого из тестов составило не более 4% из доступных для прохождения 10%.
- Для выборок в 100 000 отклонение составило не более 3%
- Встроенный ГПСЧ Linux не прошел тест на проверку равномерности меры распределения при выборке в 50 значений.
- ГПСЧ по параметризованному алгоритму показал средний результат. Поэтому дальнейшая его проверка в многомерном случае не имеет смысла.

Тесты в многомерном случае

Определение

Взаимные коэффициенты корреляции в многомерном случае вычисляются по следующей формуле:

$$cor_k(x,y) = \frac{\sum_{i=1}^{N}(x_i - \overline{x})(y_{i+k} - \overline{y})}{\sqrt{\sum_{i=1}^{N}(y_i - \overline{y})\sum_{i=1}^{N}(x_i - \overline{x})}},$$

где $0 \leq k \leq M,\ M < N$.

Результаты

1. Оценка математического ожидания в многомерном случае бессмысленна ввиду покоординатности метода.
2. Процент отклонения от теоретического значения проверки на равномерность меры распределения составил не более 3% у всех ГПСЧ.

Результат теста на корреляции (многомерный случай)

Таблица: Оценка коэффициентов корреляций, выборка в 100 000 трехмерных элементов

Основной параметр m	Дополнительный параметр L	Максимальное значение по модулю
152 898 398	9 17 13	$1.89\dot{1}0^{-4}$
675 32 543	90 10 13	$1.34\dot{1}0^{-4}$
7643 4685 6542	782 432 23	$13.27\dot{1}0^{-4}$
56 532 12	52 54 85	$19.62\dot{1}0^{-4}$
Встроенный ГПСЧ Linux		$14.82\dot{1}0^{-4}$

График коэффициентов взаимных корреляций

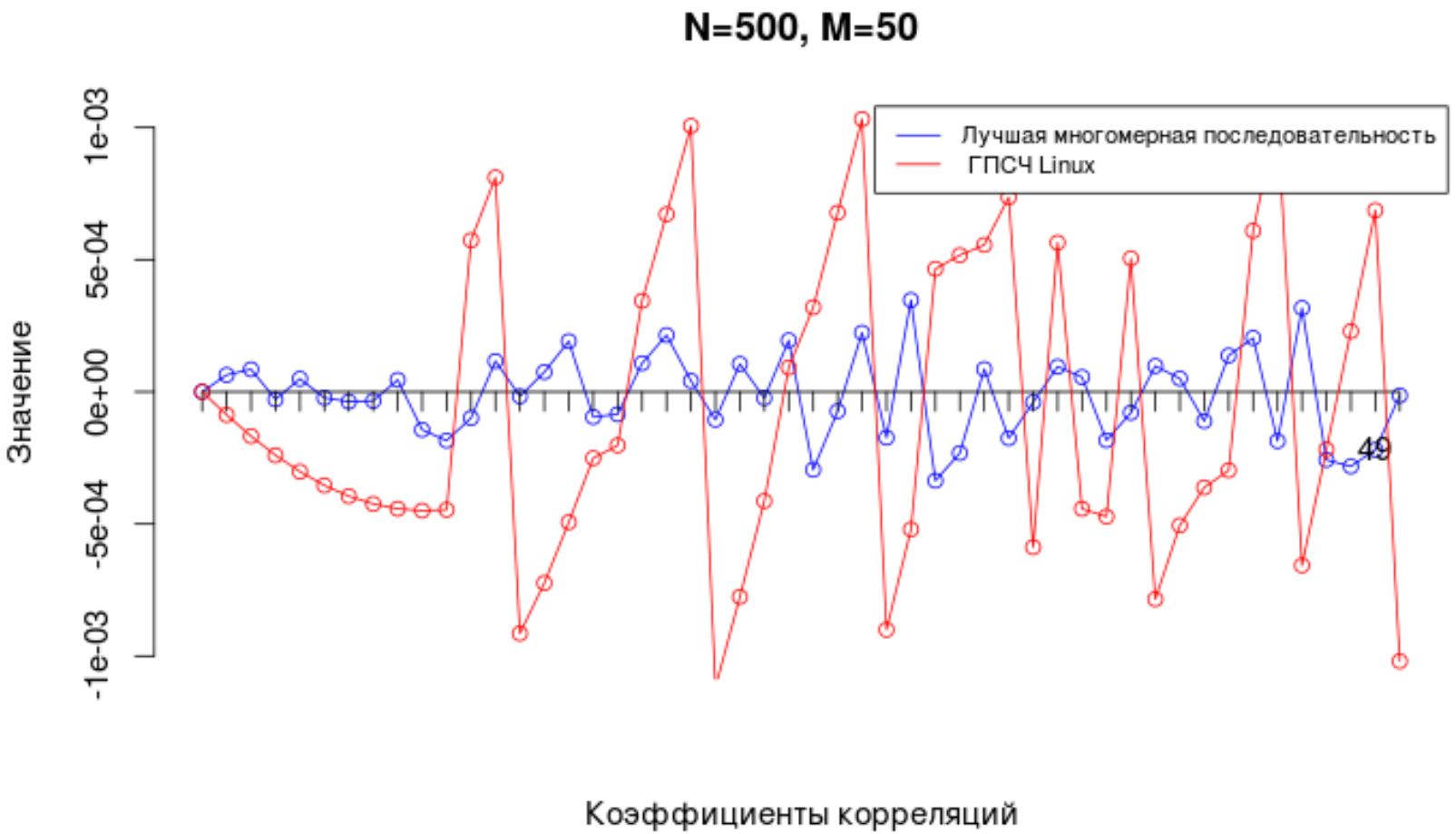

Рис.: График коэффициентов взаимных корреляций, посчитанных для трехмерной последовательности с параметрами $m = 152; 898; 398$ и $L = 9; 17; 13$ соответственно.

Описание

- NIST = National Institute of Standards and Technology, основанный в США в 1901 году, в данный момент является частью департамента торговли США.
- STS = Statistical Test Suite, специальный пакет тестов, разработанный для тестирования ГПСЧ.
- Содержит 16 тестов (всего 189 результатов), выявляющих различные дефекты двоичных последовательностей случайных чисел, например:
  - Большое количество нулей либо единиц в последовательности.
  - Большое количество серий из единиц.
  - Сжимаемость последовательности.
  - и т.д.

Результат тестов

#Test 99% - количество тестов, которые выполнились для более чем 99% сгенерированных последовательностей генератора. Это значит, что из $q = 100$ двоичных последовательностей, сгенерированных данным ГПСЧ, 99 (или 100), успешно прошли определенное количество тестов.

Таблица: Результат NIST STS, основной алгоритм

Основной параметр m	Дополнительный параметр L	#Test 99%	#Test 96%
343	7	172	185
123	4	177	184
547	27	171	183
431	43	174	181
Встроенный ГПСЧ Linux		159	183

ГПСЧ с параметризованным алгоритмом опять же показал результаты лучше, чем у ГПСЧ Linux, но хуже чем у основного алгоритма.

Моделирование нормального распределения

1. Преобразование величин производилось по следующей формуле

$$N_m(i) = (2x_i - 1)\sqrt{\frac{-2\ln s}{s}},$$

$$N_m(i+1) = (2x_{i+1} - 1)\sqrt{\frac{-2\ln s}{s}},$$

где $s = (2x_i - 1)^2 + (2x_{i+1} - 1)^2$, а $i = 1, 3, 5, \ldots$

2. Для проверки нормальности распределения пользовались критерием Пирсона, с уровнем доверия в 5% .
3. Для всех ГПСЧ достаточно 35 значений, для того чтобы пройти данный тест.

Моделирование экспоненциального распределения

1. Преобразование величин производилось по следующей формуле

$$E_m(i, \lambda) = -\frac{1}{\lambda} \ln x_i.$$

2. Для проверки экспоненциальности распределения пользовались критерием Шапиро-Уилка, по статистике:

$$W_E = \frac{\sum (x_i - \overline{x})^2}{(\sum x_i)^2}.$$

3. Гипотеза принималась, если выполнялось условие:

$$W_E \in [0.017, 0.041],$$

где границы интервала – табличные значения.

4. Так же, как и в случае нормального распределения, для всех ГПСЧ достаточно 35 значений, для того чтобы пройти данный тест.

Вывод

- Коэффициенты корреляций очень малы, что гарантирует достаточную независимость элементов случайной последовательности друг от друга.
- Прочие тесты показывают возможность данного генератора выполнять различные статистические задачи.
- Тесты NIST STS показывают, что данный генератор можно использовать как для стандартных задач, так и для различных статистических исследований.
- ГПСЧ по основному алгоритму удовлетворяет поставленным при разработке любого ГПСЧ требованиям, и случайность генерируемых им последовательностей имеет весьма высокий уровень.

Список литературы


Т.М. Товстик Сравнение некоторых статистических свойств квазислучайных и псевдо-случайных последовательностей // Вестник СПБГУ, Сер. 1, вып. 2. — 2006.


Nist. cryptographic toolkit. random number generation. — http://csrc.nist.gov/rng/.


Г.П. Акимова, Е.В. Пашкина, А.В.Соловьев Методологический подход к оценке качества случайных чисел и последовательностей // Труды ИСА РАН. — 2008.


Д.Э. Кнут Искусство программирования для ЭВМ — М., Издательский дом Вильямс, 2002.


С. М. Ермаков Метод Монте-Карло и смежные вопросы — Москва, 1971.


И.М. Соболь Численные методы Монте-Карло — Издательство Наука, 1973.


S.S.Shapiro M.B. Wilk An analysis of variance test for the exponential distribution data // Biometrika, Vol. 52, No. 3/4. pp. 591-611. — 1965.