УДК 519.6 Вестник СПбГУ. Математика. Механика. Астрономия. 2018. Т. 5 (63). Вып. 3
MSC 68U20, 65C60

Генерирование больших последовательностей нормальных рекордных величин и максимумов

А. И. Пахтеев, А. В. Степанов

Балтийский федеральный университет имени Иммануила Канта,
Российская Федерация, 236041, Калининград, ул. А. Невского, 14

Для цитирования: *Пахтеев А. И., Степанов А. В.* Генерирование больших последовательностей нормальных рекордных величин и максимумов // Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия. 2018. Т. 5 (63). Вып. 3. С. 431–440. https://doi.org/10.21638/11701/spbu01.2018.307

В нашей недавней статье (2018) были разработаны алгоритмы генерирования нормальных рекордных величин. В статье было показано, что разработанные алгоритмы превосходят по быстродействию и эффективности работы известные на тот момент алгоритмы генерирования нормальных рекордных величин. Приводимый в настоящей работе алгоритм 2.2 является наиболее эффективным алгоритмом упомянутой выше статьи. Он позволяет генерировать «очень большие» последовательности нормальных рекордных величин (до двух миллиардов величин). В данной работе предлагаются два алгоритма генерирования максимумов нормальных выборок. Один из предложенных алгоритмов основывается на алгоритме 2.2 и позволяет генерировать максимумы «больших» выборок. В статье также предложен алгоритм генерирования рекордных моментов в общем непрерывном случае.

Ключевые слова: рекордные моменты и величины, максимумы, нормальное распределение, метод выборки с отклонением, метод обратных преобразований, время работы программы.

1. Введение. Развитие теории экстремальных порядковых статистик и рекордов является актуальным в связи с различными приложениями, возникающими в метеорологии, гидрологии, в страховом и финансовом бизнесе. Перепады температур и атмосферного давления, паводки рек, спортивные достижения, страховые и финансовые риски, различные модели, связанные с временами обслуживания, коррозией металлов, сопротивлением материалов, все это и многое другое, прекрасно описывается математическим аппаратом этой теории. Более подробную информацию по этой тематике можно найти в книгах [1–3].

Интересным и востребованным направлением дальнейшего развития теории экстремальных порядковых статистик и рекордов является разработка алгоритмов генерирования этих случайных величин. Приведем пример того, где подобные алгоритмы находят применение. Предположим, что разработчиков некоторого медицинского препарата интересует влияние препарата на показатели артериального давления пациентов. Клинические исследования показывают, что показатели артериального давления после применения данного препарата имеют нормальное распределение. Генерирование максимумов и минимумов больших нормальных выборок позволяет моделировать поведение максимумов и минимумов артериального давления пациентов при массовом применении препарата.

В рамках данной работы мы ограничимся рассмотрением алгоритмов генерирования максимумов и рекордов. Первые алгоритмы генерирования максимумов появились давно. Приведем один из типичных алгоритмов, основанных на прямом методе.

Прямой метод генерирования максимумов. *Генерируется n случайных величин. Затем из них выбирается максимальная.*

Отметим недостатки этого метода. При «больших» n метод является ресурсозатратным. Он также не позволяет генерировать «большие» значения максимумов в случае, когда распределения имеют «тонкие» хвосты. Так, в случае стандартного нормального распределения, пользуясь прямым методом генерирования максимумов, нельзя сгенерировать (даже с помощью самых современных компьютеров и соответствующих программных продуктов) нормальный максимум, значение которого превысило бы число 30. Авторы данной работы, пользуясь программой MatLab и компьютером AMD FX(tm)-8350 Eight-Core Processor 4.00GHZ 16 Gb, смогли сгенерировать прямым методом за время 4678.210569 секунды нормальный максимум со значением 6.872549. В третьем разделе данной работы мы предложим альтернативные алгоритмы генерирования максимумов. Один из этих алгоритмов позволит генерировать максимумы, имеющие «большие» значения.

Пусть $X_1, X_2, \ldots$ — последовательность независимых случайных величин с общей непрерывной функцией распределения F. Определим последовательности рекордных моментов $L(n)$ и рекордных величин $X(n)$ следующим образом:

$$L(1) = 1,$$

$$L(n+1) = \min\left\{j : j > L(n),\, X_j > X_{L(n)}\right\}, \quad X(n) = X_{L(n)} \quad \text{для } n \geqslant 1.$$

Первые алгоритмы генерирования рекордных величин, предложенные в работах [4–10], появились относительно недавно. Существует несколько различных алгоритмов генерирования рекордов. Первый и наиболее простой алгоритм генерирования основан на прямом методе.

Прямой метод генерирования рекордных моментов и величин. *Предположим, что величина $X(1) = X_1$ уже сгенерирована. Положим $L(1) = 1$. Для дальнейшего генерирования последовательностей рекордных моментов и величин применим рекуррентный подход, который предполагает, что случайные величины $X(n), L(n)$ $(n \geqslant 1)$ уже сгенерированы. Продолжим генерировать величины X_i $(i > L(n))$ до тех пор, пока одна из них, скажем X_j, не станет больше чем $X(n)$. Тогда положим $X(n+1) = X_j$ и $L(n+1) = j$.*

Прямой метод генерирования рекордов позволяет генерировать рекорды для любых распределений. Однако, если требуется сгенерировать «большую» последовательность рекордов, то алгоритм, основанный на прямом методе, так же как и в случае с максимумами, становится медленным и ресурсозатратным.

Известно, что последовательность рекордных величин образует цепь Маркова, причем выполняется равенство

$$P(X(n+1) < x_{n+1} \mid X(n) = x_n) = \frac{F(x_{n+1}) - F(x_n)}{1 - F(x_n)} \qquad (x_{n+1} > x_n). \tag{1.1}$$

Если обратная функция F^{-1} может быть найдена явно, то для генерирования рекордных величин можно применять метод обратных преобразований. Пользуясь рекуррентным подходом и равенством (1.1), полагаем $X(n+1)$ равным

$$F^{-1}(U(1-F(X(n)))+F(X(n))),$$

где U — генерация случайного числа, а $X(n)$ — предыдущая рекордная величина. Если же обратная функция F^{-1} не может быть получена аналитически, то для генерирования рекордных величин можно применять либо таблицы обратной функции F^{-1}, либо метод выборки с отклонением.

В нашей недавней статье [4] были предложены алгоритмы генерирования нормальных рекордных величин. В статье было показано, что данные алгоритмы превосходят по быстродействию и эффективности работы известные на тот момент алгоритмы генерирования нормальных рекордных величин, изложенные в статье [5]. Важной положительной особенностью алгоритмов статьи [4] является то, что они позволяют эффективно генерировать «очень большие» последовательности рекордных величин.

Дальнейшее содержание нашей работы следующее. Во втором разделе работы предлагается алгоритм генерирования рекордных моментов в общем непрерывном случае. Алгоритм основан на методе обратных функций и позволяет генерировать «большие» последовательности рекордных моментов. Кроме того, во втором разделе данной работы приводится (взятый из [4]) наиболее эффективный алгоритм генерирования рекордных величин — алгоритм 2.2. Алгоритм основан на рекуррентном подходе и методе выборки с отклонением. В [4] было показано, что данный алгоритм быстр и эффективен и позволяет генерировать «очень большие» последовательности нормальных рекордов. В третьем разделе данной работы разрабатываются алгоритмы генерирования максимумов нормальных выборок. Один из алгоритмов генерирования максимумов (алгоритм 3.4) основывается на алгоритме 2.2 и также позволяет генерировать максимумы «больших» выборок.

2. Генерирование рекордов. ***2.1. Генерирование рекордных моментов.*** В этом разделе предложим алгоритм генерирования рекордных моментов в общем непрерывном случае. Пусть $X_1, X_2, \ldots$ — последовательность независимых случайных величин с общей непрерывной функцией распределения. Из теории рекордов известно равенство (см., например, [2, стр. 93])

$$P(L(n+1)=l_{n+1} \mid L(n)=l_n)=\frac{l_n}{(l_{n+1}-1)l_{n+1}} \qquad (l_{n+1}>l_n\geq n\geq 1).$$

Из последней формулы, в частности, следует соотношение

$$F_{L(n+1)|L(n)}(l_{n+1} \mid l_n)=1-\frac{l_n}{l_{n+1}-1} \qquad (l_{n+1}>l_n\geq n\geq 1). \tag{2.1}$$

Равенство (2.1) позволяет генерировать дискретные рекордные моменты $L(n)$ $(n\geq 2)$ методом обратных функций. Представим соответствующий алгоритм генерирования.

Алгоритм 2.1. *Последовательность $L(n)$ $(n \geqslant 1)$ может быть сгенерирована следующим образом. Положим $L(1)=1$. Для генерирования $L(n)$ $(n\geq 2)$ воспользуемся рекуррентным методом. Пусть величина $L(n)=l_n$ $(l_n\geq n\geq 1)$ уже сгенерирована. Генерируем случайное число $U=u$. Найдем то единственное $l_{n+1}>l_n$ такое, что $u\in(l_n/l_{n+1}, l_n/(l_{n+1}-1)]$. Положим $L(n+1)=l_{n+1}$.*

Алгоритм 2.1 позволяет быстро генерировать «большое» количество рекордных моментов. Ниже приведем некоторые результаты эксперимента статистического моделирования:

$$L(10) = 2069,$$

$$L(50) = 1.532414499502304610222080 * 10^{24},$$

$$L(100) = 1.22257703556981686411978507483344476203739250688 * 10^{47},$$

$$L(500) = 3.769683048131419483005179187473069837971396732 9208 * 10^{208},$$

$$L(700) = 6.78131345365948869699042844564203358866529723 59593 * 10^{299},$$

$$L(730) = 6.66278033738094356064769273721994859511659864 58002 * 10^{307}.$$

Время работы программы составило 10.724324 секунды. Дальнейшей работе алгоритма 2.1 помешало следующее обстоятельство. Реализация алгоритма 2.1 осуществлялась в программе MatLab. К сожалению, в программе MatLab нельзя работать с числами, большими $1.797693134862316 * 10^{308}$.

2.2. Генерирование нормальных рекордных величин. В этом разделе приведем алгоритм генерирования нормальных рекордных величин, взятый из работы [4]. Алгоритм основан на методе выборки с отклонением. Данный метод генерирования случайных величин можно найти, например, в [11, стр. 19] или в [12, стр. 71].

Пусть Φ — стандартное нормальное распределение и ϕ — соответствующая плотность, т. е.

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \quad (x \in \mathbb{R}).$$

Пусть в дальнейшем X_i $(i \geqslant 1)$ — независимые случайные величины со стандартным нормальным распределением Φ и $X(n)$ $(n \geqslant 1)$ — соответствующие нормальные рекордные величины. Из формулы (1.1) следует, что условная плотность величины $X(n+1)$ имеет вид

$$\varphi_{X(n+1)|X(n)}(x_{n+1} \mid x_n) = \frac{\phi(x_{n+1})}{1 - \Phi(x_n)} \quad (x_{n+1} > x_n). \tag{2.2}$$

Пользуясь (2.2) и методом выборки с отклонением, в [4] был получен следующий алгоритм генерирования нормальных рекордных величин. Пусть $\beta_n^* = (x_n + \sqrt{x_n^2 + 4})/2$.

Алгоритм 2.2. *Последовательность $X(n)$ $(n \geqslant 1)$ может быть сгенерирована следующим образом.*

ШАГ 1. Прямым методом генерируем рекордные величины $X(1) = X_1$, $X(2), \ldots, X(i)$ $(i \geqslant 1)$ до тех пор, пока величина $X(i)$ не станет положительной.

Для $n \geqslant i$ применим метод выборки с отклонением и следующий рекуррентный подход. Предположим, что величина $X(n) = x_n$ уже сгенерирована.

ШАГ 2. Генерируем случайные числа $U_1 = u_1, U_2 = u_2$.

ШАГ 3. Пусть $Y = y = x_n - \log u_1/\beta_n^*$. Если

$$-2 \log u_2 > (y - \beta_n^*)^2, \tag{2.3}$$

положим $X(n+1) = y$, иначе вернемся к шагу 2.

Замечание. Поясним, почему на шаге 1 алгоритма 2.2 мы вынуждены генерировать отрицательные нормальные рекордные величины прямым методом. Использование метода выборки с отклонением предполагает, что в алгоритме 2.2 плотность $f_{X(n+1)|X(n)}(x_{n+1} \mid x_n)$, задаваемая (2.2), приближается условной плотностью $g(x_{n+1} \mid x_n, \beta_n) = \beta_n e^{-\beta_n(x_{n+1}-x_n)}$ $(x_{n+1} > x_n)$, где $\beta_n > 0$ выбирается так, чтобы функция g аппроксимировала f «наилучшим» образом. В случае положительных x_n формы кривых $f_{X(n+1)|X(n)}(x_{n+1} \mid x_n)$ и $g(x_{n+1} \mid x_n, \beta_n)$ схожи, и удается подобрать соответствующее β_n $(= \beta_n^*)$. В случае отрицательных x_n формы g и f различны и подобрать соответствующее β_n не удается.

Отметим также следующее. Пусть τ (≥ 1) — число стандартных нормальных величин таких, что $X_1 < 0, \ldots, X_{\tau-1} < 0, X_\tau > 0$. Несложно показать, что $E\tau = 2$. Таким образом, генерирование отрицательных нормальных рекордов на шаге 1 прямым методом практически не замедляет работу алгоритма 2.2.

В [4] было показано, что основанный на методе выборки с отклонением алгоритм 2.2 со временем работает так же, как и типичный алгоритм, основанный на методе обратного преобразования. При длительной работе алгоритма 2.2 почти каждая генерация Y принимается и становится новой рекордной величиной. Алгоритм 2.2, соответственно, быстр и эффективен. Как уже отмечалось во введении, пользуясь прямыми методами генерирования максимумов или рекордов, нельзя сгенерировать (даже с помощью самых современных компьютеров и соответствующих программных продуктов) стандартную нормальную случайную величину, которая превышала бы значение 30. С помощью алгоритма 2.2 можно генерировать «очень большие» нормальные рекордные величины. Так, пользуясь программой MatLab (на компьютере AMD FX(tm)-8350 Eight-Core Processor 4.00GHZ 16 Gb), авторы данной работы сгенерировали следующую последовательность нормальных рекордов:

$$\begin{aligned} X(1000) &= 43.708507338291, \\ X(10000) &= 140.401995284972, \\ X(100000) &= 447.202565606248, \\ X(1000000) &= 1414.590966639376, \\ X(10000000) &= 4472.657006446189, \\ X(100000000) &= 14142.375314642721, \\ X(1000000000) &= 44721.300268966495, \\ X(2000000000) &= 63251.083009000470. \end{aligned}$$

Время работы программы составило 18012.4227 секунд.

3. Генерирование нормальных максимумов. Обозначим через $M_n = \max\{X_1, \ldots, X_n\}$ максимум выборки независимых случайных величин со стандартной нормальной функцией распределения Φ. В этом разделе рассмотрим два метода генерирования максимумов нормальной выборки.

Изложим первый метод. Известно, что функция распределения и плотность максимума выборки в нормальном случае задаются следующими формулами:

$$F_{M_n}(m_n) = \Phi^n(m_n), \quad f_{M_n}(m_n) = n\phi(m_n)\Phi^{n-1}(m_n).$$

Несложно показать, что условное распределение величины M_{n+1} при условии $M_n =$

m_n имеет вид

$$F_{M_{n+1}|M_n}(m_{n+1} \mid m_n) = \begin{cases} 0 & \text{при } m_{n+1} < m_n, \\ \Phi(m_{n+1}) & \text{при } m_{n+1} \geq m_n. \end{cases}$$

Положим следующий алгоритм в основу первого метода генерирования максимумов.

Алгоритм 3.3. *Последовательность M_n $(n \geqslant 1)$ может быть сгенерирована следующим образом.*

ШАГ 1. Генерируем величину $M_1 = X_1$.

Для $n \geqslant 1$ применим следующий рекуррентный подход. Предположим, что величина $M_n = m_n$ уже сгенерирована.

ШАГ 2. Генерируем случайное число $U = u$.

ШАГ 3. Если $u > \Phi(m_n)$, положим $M_{n+1} = m_{n+1} = \Phi^{-1}(u)$, в противном случае $M_{n+1} = m_n$. Вернемся к шагу 2.

Отметим, что $\Phi^{-1}(u)$ на шаге 3 ищется при помощи встроенных функций, основанных на таблицах (см, например, [13]). Алгоритм 3.3 обладает следующей особенностью. При «больших» n вероятность события $\{u > \Phi(m_n)\}$ мала и чем дольше будет работать алгоритм 3.3, тем реже будут обновляться значения максимума. Поскольку алгоритм обрабатывает все максимумы, то работает он достаточно медленно. С помощью алгоритма 3.3 на нашем компьютере был получен максимум всего лишь со значением 6.5322. Время работы программы составило 928.4323 секунды. Очевидно, данный алгоритм генерирования максимумов сопоставим по эффективности работы и скорости с алгоритмом генерирования, основанным на прямом методе.

Предложим второй алгоритм генерирования нормальных максимумов. Он основан на алгоритме генерирования рекордных величин — алгоритме 2.2.

Алгоритм 3.4. *Последовательность M_n $(n \geqslant 1)$ может быть сгенерирована следующим образом.*

ШАГ 1. Прямым методом генерирования рекордных моментов и величин генерируем рекордные величины $X(1) = X_1 = x_1, X(2) = x_2, \ldots, X(i) = x_i$ $(i \geq 1)$ и рекордные моменты $L(1) = 1, L(2) = l_2, \ldots, L(i) = l_i$ до тех пор, пока величина $X(i) = x_i$ не станет положительной. При этом полагаем $M_i = X_1$, если $i = 1$, и

$$M_{l_k+1} = \ldots = M_{l_{k+1}-1} = x_k, \quad M_{l_{k+1}} = x_{k+1},$$

если $1 \leq k < i$.

Для $n \geqslant i$ применяем метод выборки с отклонением и следующий рекуррентный подход. Предполагаем, что величины $L(n) = l_n$ и $M_{l_n} = X(n) = x_n$ уже сгенерированы.

ШАГ 2. Генерируем случайные числа $U_1 = u_1, U_2 = u_2$.

ШАГ 3. Пусть $Y = y = x_n - \log u_1/\beta_n^*$. Если

$$-2\log u_2 > (y - \beta_n^*)^2,$$

полагаем $X(n+1) = x_{n+1} = y$. Иначе, возвращаемся к шагу 2.

ШАГ 4. Генерируем случайное число $V = v$ и полагаем

$$L(n+1) = l_{n+1} = l_n + \left\lceil \frac{-\log(v)x_n}{\phi(x_n)} \right\rceil,$$

где $\lceil x \rceil$ — наименьшее целое число, большее или равное x. При этом считаем, что

$$M_{l_n+1} = \ldots = M_{l_{n+1}-1} = x_n, \quad M_{l_{n+1}} = x_{n+1}.$$

Обоснование алгоритма 3.2. Пусть $X_1, X_2, \ldots$ — последовательность независимых случайных величин с общей непрерывной функцией распределения Φ. Известно (см., например, [14]), что последовательность векторов $(L(n), X(n))$ образует цепь Маркова, причем

$$P(L(n+1) = l_{n+1}, X(n+1) < x_{n+1} \mid L(n) = l_n, X(n) = x_n) = \\ = (\Phi(x_{n+1}) - \Phi(x_n))\Phi^{l_{n+1}-l_n-1}(x_n) \quad (x_{n+1} > x_n, l_{n+1} > l_n).$$

Отсюда следуют равенства

$$P(L(n+1) \le k \mid L(n) = l_n, X(n) = x_n) = 1 - \Phi^{k-l_n}(x_n) \quad (k > l_n) \tag{3.1}$$

и

$$P(X(n+1) < x_{n+1} \mid L(n) = l_n, X(n) = x_n) = \\ = P(X(n+1) < x_{n+1} \mid X(n) = x_n) = \frac{\Phi(x_{n+1}) - \Phi(x_n)}{1 - \Phi(x_n)} \quad (x_{n+1} > x_n). \tag{3.2}$$

Из последней формулы, в частности, можно получить условную плотность $\varphi_{X(n+1)|X(n)}(x_{n+1} \mid x_n)$, фигурирующую в (2.2). Таким образом, если известен случайный вектор $(L(n), X(n))$, то генерирование вектора $(L(n+1), X(n+1))$ осуществляется следующим образом. На шаге 3 генерируется $X(n+1)$ с помощью алгоритма 2.2, использующего, в свою очередь, плотность $\varphi_{X(n+1)|X(n)}(x_{n+1} \mid x_n)$. Для генерирования $L(n+1)$ используется (3.1). Из (3.1) следует, что $L(n+1)$ можно генерировать методом обратных функций, при этом необходимо сравнивать $\Phi^{l_{n+1}-l_n}(x_n)$ с генерацией случайного числа v. Такое сравнение происходит на шаге 4. На шаге 4 следовало бы записать следующее:

$$L(n+1) = l_{n+1} = l_n + \left\lceil \frac{-\log v}{-\log(\Phi(x_n))} \right\rceil.$$

Однако из-за ограничений программы MatLab происходит деление на ноль при $x_n > 8.3$. Для устранения этой технической проблемы мы заменяем $-\log(\Phi(x_n))$ эквивалентной величиной $\phi(x_n)/x_n$. После такой замены алгоритму удается генерировать до значения 37.5225880526932. Дальнейшему генерированию последовательности максимумов, как и в случае с рекордными моментами, препятствуют возможности программы MatLab.

Алгоритм 3.4 использовался для генерирования максимумов. Мы сгенерировали следующую последовательность нормальных максимумов:

$$\begin{aligned} M_{L(1)} = M_1 &= -0.0255282267262544, \\ M_{L(2)} = M_2 &= 1.40598025651916, \\ &\ldots \\ M_{L(716)} = M_{3.11265542659599242032398324031405288617914616718 6*10^{303}} &= 37.2639369919189, \end{aligned}$$

$$
\begin{aligned}
& & & \dots & & \\
M_{L(717)} &= M_{8.82792199192248824634648934325819869390031834363 09*10^{305}} &= \quad & 37.3765732886911, \\
& & & \dots & & \\
M_{L(718)} &= M_{2.26538171575234271126209769880905465005284339898 19*10^{306}} &= \quad & 37.4411205861029, \\
& & & \dots & & \\
M_{L(719)} &= M_{4.14619479156819483199836519379676284175893508865 05*10^{306}} &= \quad & 37.4563899796549, \\
& & & \dots & & \\
M_{L(720)} &= M_{6.02004642029731397413359385598285737225293839554 99*10^{306}} &= \quad & 37.4688264102772, \\
& & & \dots & & \\
M_{L(721)} &= M_{1.49730992373607024235376451330617106397864415682 74*10^{308}} &= \quad & 37.5225880526932.
\end{aligned}
$$

Время работы программы составило 11.348971 секунды. Отметим, что алгоритм 3.4 значительно превосходит по быстродействию и эффективности работы как алгоритм, основанный на прямом методе, так и алгоритм 3.3.

Генерируемые с помощью алгоритма 3.4 последовательности нормальных максимумов M_n $(n \geq 1)$ при больших n удовлетворяют следующему асимптотическому соотношению:

$$
\frac{M_n}{\sqrt{2\log n}} \overset{\text{п.н.}}{\to} 1 \quad (n \to \infty),
$$

известному в теории экстремальных величин (см., например, [15, стр. 203]).

Авторы работы благодарны двум рецензентам за замечания, позволившие улучшить содержание работы.

Литература

1. *Arnold B. C., Balakrishnan N., Nagaraja H. N.* Records. New-York: John Wiley & Sons. 1998.
2. *Невзоров В. Б.* Рекорды. Математическая теория. М.: ФАЗИС, 2000.
3. *Ahsanullah M., Nevzorov V. B.* Record via Probability Theory. Atlantis Press, 2015.
4. *Pakhteev A., Stepanov A.* On Simulation of Normal Records // Communication in Statistics — Simulation and Computation. 2018 (to appear). https://doi.org/10.1080/03610918.2018.1457692.2018
5. *Balakrishnan N., So H. Y., Zhu X. J.* On Box-Muller Transformation and Simulation of Normal Record Data // Communication in Statistics — Simulation and Computations. 2016. Vol. 45. P. 3670–3682.
6. *Bairamov I., Stepanov A.* Numbers of near bivariate record-concomitant observations // J. of Multivariate Analysis. 2011. Vol. 102. P. 908–917.
7. *Luckett D. J.* Statistical Inference Based on Upper Record Values. PhD thesis. The College of William and Mary. 2013.
8. *Nevzorov V. B., Stepanov A.* Records with confirmation // Statist. Probab. Lett. 2014. Vol. 95. P. 39–47.
9. *Pakhteev A., Stepanov A.* Simulation of Gamma Records // Statist. Probab. Lett. 2016. Vol. 119. P. 204–212.
10. *Stepanov A., Berred A., Nevzorov V. B.* Concomitants of records: Limit results, generation techniques, correlation // Statistics & Probability Letters. 2016. Vol. 109. P. 184–188.
11. *Ермаков С. М., Сипин А. С.* Метод Монте-Карло и параметрическая разделимость алгоритмов. СПб.: Изд-во С.-Петерб. ун-та, 2014.

12. *Ross S. M.* Simulation. 4th ed. Elsevier, 2006.
13. *Wichura M. J.* Algorithms AS 241: The percentage points of the normal distribution // J. Royal Statistical Society. Ser. C. 1988. Vol. 37. P. 477–484.
14. *Stepanov A.* Conditional moments of record times // Statist. Pap. 2003. Vol. 44, N 1. P. 131–140.
15. *Галамбош Я.* Асимптотическая теория экстремальных порядковых статистик. М.: Наука, 1984.

Статья поступила в редакцию 18 февраля 2018 г.; рекомендована в печать 22 марта 2018 г.

Контактная информация:

Пахтеев Артем Игоревич — аспирант; mir123i3@gmail.com
Степанов Алексей Васильевич — д-р физ.-мат. наук, проф.; alexeistep45@mail.ru

Generation of large sequences of normal record values and maxima

A. I. Pakhteev, A. V. Stepanov

Immanuel Kant Baltic Federal University, ul. A. Nevskogo, 14, Kaliningrad, 236041, Russian Federation

For citation: Pakhteev A. I., Stepanov A. V. Generation of large sequences of normal record values and maxima. *Vestnik of Saint Petersburg University. Mathematics. Mechanics. Astronomy*, 2018, vol. 5 (63), issue 3, pp. 431–440. https://doi.org/10.21638/11701/spbu01.2018.307

In our recent paper (2018), algorithms of generation of normal record values were developed. It was shown there that the developed algorithms are more efficient and speedy than all existing at that time algorithms of generation of normal record values. Presented in this paper algorithm 2.2 is the most efficient algorithm amongst the algorithms of the paper of Pakhteev and Stepanov. It allows to generate “large” sequences of record values (up to two billion record values). In the present paper, two algorithms of generation of normal maxima are proposed. One of the proposed algorithms is based on algorithm 2.2. It also allows to generate maxima of samples of “large” sizes. An algorithm of generation of record times in general continuous case is also proposed in the present paper.

Keywords: records, maxima, normal distribution, rejection method, inverse-transform method, generation techniques, elapsed time.

References

1. Arnold B. C., Balakrishnan N., Nagaraja H. N., *Records* (John Wiley & Sons, New York, 1998).
2. Nevzorov V. B., *Records. Mathematical theory* (Fasiz Publ., Moscow, 2000) [in Russian].
3. Ahsanullah M., Nevzorov V. B., *Record via Probability Theory* (Atlantis Press, 2015).
4. Pakhteev A., Stepanov A., “On Simulation of Normal Records”, *Communication in Statistics — Simulation and Computation* (to appear, 2018). https://doi/org/10.1080/03610918.2018.1457692
5. Balakrishnan N., So H. Y., Zhu X. J., “On Box-Muller Transformation and Simulation of Normal Record Data”, *Communication in Statistics — Simulation and Computations* **45**, 3670–3682 (2016).
6. Bairamov I., Stepanov A., “Numbers of near bivariate record-concomitant observations”, *J. of Multivariate Analysis* **102**, 908–917 (2011).
7. Luckett D. J., *Statistical Inference Based on Upper Record Values* (PhD thesis. The College of William and Mary, 2013).
8. Nevzorov V. B., Stepanov A., “Records with confirmation”, *Statist. Probab. Lett.* **95**, 39–47 (2014).
9. Pakhteev A., Stepanov A., “Simulation of Gamma Records”, *Statist. Probab. Lett.* **119**, 204–212 (2016).
10. Stepanov A., Berred A., Nevzorov V. B., “Concomitants of records: Limit results, generation techniques, correlation”, *Statistics & Probability Letters* **109**, 184–188 (2016).

11. Ermakov S. M., Sipin A. S. Monte Carlo method and parametric algorithm discriminatingly (St. Petersburg Univ. Press, St. Petersburg, 2014) [in Russian].

12. Ross S. M., *Simulation* (4th ed., Elsevier, 2006).

13. Wichura M. J., "Algorithms AS 241: The percentage points of the normal distribution", *J. Royal Statistical Society. Ser. C* **37**, 477–484 (1988).

14. Stepanov A., "Conditional moments of record times", *Statist. Pap.* **44**(1), 131–140 (2003).

15. Galambosh Ya., *Asimptotic theory of extremal order statistic* (Nauka Publ., Moscow, 1984) [in Russian].

Author's information:

Artem I. Pakhteev — mir123i3@gmail.com
Alexei V. Stepanov — alexeistep45@mail.ru