Testing the Reliability of Statistical Tests for Pseudorandom Number Generators

Hiroshi Haramoto (Ehime Univ.)
joint work with Makoto Matsumoto (Hiroshima Univ.)

July, 3rd, 2018

This work is supported by JSPS Grant # 16K13750 and # 17K141234
and JST CREST
The computation in this work has been done using the facilities of the Institute of Statistical Mathematics

Introduction

- Our purpose is to check the accuracy of the approximation formula for p-values used in statistical tests (which may cause errorneous results).
- A three-level procedure gives an experimental evaluation for the accuracy.
- In MCM 2017, we showed the results of the three-level test on the current version of TestU01
  $\Longrightarrow$ some tests seem to have defects
- We will show the results of the three-level test on
  - the current version of NIST SP800-22
  - the current version of TestU01

  and their modifications.

- We will also show an investigation of the LIL test (Wang and Nicol, '15).

Statistical tests for PRNGs and their defects

- Does a statistical test yield correct p-values?
  - Some tests are based on incorrect mathematical analyses.
  - Computational errors may not be negligible.
  - Choice of parameters in tests is done by experience.
  - Poor implementations may damage the credibility of tests.
- To find such errors by theory is often difficult.
- Experimental methods may be useful.
- We introduce a three-level test to check the reliability of statistical tests.

Statistical tests for PRNGs and their defects

The NIST test suite (SP800-22) is one of the most used test suites:

- the Lempel-Ziv test: removed from the suite (based on an incorrect mathematical model).
- the Discrete Fourier Transform test: a change of several parameters is proposed.
- the Maurer's Universal statistical test: a better approximation has been proposed.
- the Overlapping template matching test: there are several studies on affects of approximation errors.
- etc.

The latest version of the suite addresses some of these problems, but not all.

Statistical tests for PRNGs

- Test the null hypothesis

$$a_1, \ldots, a_n\text{: u.i.i.d. in } I = \{0,1\} \text{ (resp. } I = [0,1)).$$

  Put $\boldsymbol{a} := (a_1, \ldots, a_n)$. ($a_k$ represents the k-th output of the tested PRNGs.)
- A test statistic is a function of n-variables

$$f : I^n \to \mathbb{R}.$$

- Let $X_1, \ldots, X_n$ be u.i.i.d. random variables in I^n. If the p-value

$$\mathbb{P}\left(f(\boldsymbol{a}) < f(\boldsymbol{X})\right) \qquad (\boldsymbol{X} := (X_1, \ldots, X_n))$$

  is too close to 0 or too close to 1, the null hypothesis is rejected.
- It is necessary that we can compute the exact p-values or have approximation formulas accurate enough.

Three-level test to check the accuracy of aproximated p-values

- Okutomi and Nakamura proposed a three-level test ('12).
- Their intention was to develop a multi-level test on PRNGs.
- We found out effectiveness of the three-level procedure to check the accuracy of aproximated p-values (not effective as statistical tests on PRNGs).

Lemma

If a statistic f has a continuous distribution and $X_1, \ldots, X_n$ are i.i.d. on $[0, 1)$, then the p-values are i.i.d. on $[0, 1]$.

Three-level test for approximated p-values

Check the accuracy of the approximated p-values of f with sample size n

- Use the approximation formula to obtain approximated p-values

$$\mathbb{P}(f(\boldsymbol{a}_1) < f(\boldsymbol{X})), \ldots, \mathbb{P}(f(\boldsymbol{a}_{NN'}) < f(\boldsymbol{X}))$$

  from a uniform randomnumber sequence $\boldsymbol{a}_i \in I^n$
- For $i = 1, \ldots, N'$, count the number T_i of p-values $\geq \alpha$ in

$$\mathbb{P}(f(\boldsymbol{a}_{1+(i-1)N}) < f(\boldsymbol{X})), \ldots, \mathbb{P}(f(\boldsymbol{a}_{iN}) < f(\boldsymbol{X})).$$

- Test the following null hypothesis via a χ^2 (GOF) test.

$$T_1, \ldots, T_{N'} \sim_{i.i.d.} B(N', 1-\alpha).$$

- If the resulting p-value is extremely small (e.g. $< 10^{-10}$), the accuracy of the approximated p-values with respect to f is judged to be too low to use in the test.

Results on the NIST test suite and Crush in TestU01

- Apply the three-level test to the NIST test suite and TestU01.
- Use Mersenne Twister (MT) and SHA1 algorithm.
- Parameters: $N = 1,000$, $N' = 1,000$, $\alpha = 0.01$.

	p-value(Current)		p-value(Modified)	
Test Name	MT	SHA1	MT	SHA1
Longest Run	3.9E−5	1.3E−8	0.44	0.0011
DFT	4.1E−119	7.2E−116	0.19	0.026
O. Temp.	7.5E−80	5.6E−73	0.70	0.88
Universal	8.7E−76	4.1E−66	0.99	0.77
Rand. Exc.	4/8	4/8	8/8	8/8
SampleCorr	1.8E−222	5.5E−237	0.50	0.83
Savir2	2.7E−49	9.9E−32	1.1E-06	1.6E-05
Run	ε, ε	ε, ε	0.66, 0.72	0.30, 0.048
LempelZiv	ε	ε	-	-
Fourier3	ε, ε, ε	ε, ε, ε	-	-

ε : p-value $< 10^{-300}$

Modifications for the NIST test suite

Actual NIST test suite works correctly even if the three-level test reports small p-values.

Table: p-values of one-level tests and a two-level test for MT

Test Name	first level ($n = 10^6$)					second level ($N = 10^3$)
	1st	2nd	3rd	4th	5th	
Longest Run	0.15	0.39	0.64	0.029	0.47	0.88
DFT	0.48	0.44	0.31	0.89	0.66	0.41
O. Temp.	0.58	0.69	0.18	0.47	0.99	0.15
Universal	0.78	0.96	0.083	0.40	0.38	0.99

Table: p-values of one-level tests and a two-level test for SHA1

Test Name	first level ($n = 10^6$)					second level ($N = 10^3$)
	1st	2nd	3rd	4th	5th	
Longest Run	0.65	0.50	0.69	0.44	0.052	0.64
DFT	0.73	0.038	0.13	0.77	0.34	0.034
O. Temp.	0.21	0.75	0.91	0.087	0.76	0.14
Universal	0.32	0.33	0.63	0.89	0.090	0.083

Modifications for the NIST test suite

- Test for the Longest Run of Ones in a Block
  - Divides an n-bit sequence into consecutive m-bit blocks.
  - Counts the number of blocks ν_i whose longest runs of ones is i
  - Computes

$$\chi^2 = \sum_i (\nu_i - \lfloor n/m \rfloor \pi_i)^2 / (\lfloor n/m \rfloor \pi_i)$$

    (π_i : the theoretical probability that an m-bit block has the longest runs of ones as i)
  - Test the null hypothesis by a χ^2 GOF test.
  - the NIST suite uses 4-digit values of π_is from an approximation formula.
  - We replaced these values by the exact probabilities with 15-digit accuracy.

Modifications of the NIST test suite

- The Discrete Fourier Transform test
  Replace the statistic

$$(O_h - 0.95n/2)/\sqrt{0.05 \cdot 0.95n/4}$$

  with Pareschi-Rovatti-Setti's recommendation ('10)

$$(O_h - 0.95n/2)/\sqrt{0.05 \cdot 0.95n/3.8}$$

  (O_h : # of the discrete Fourier coefficients $< h$ (a constant))
- The Maurer's Universal Test
  Adopt a better statistics (Coron, '98)
- The Overlapping Template Matching test
  Remove a bug in the source code of the NIST suite (H–M, submitted)

Modifications of the NIST test suite

- The Random Excursion test
  Change the constraint $J \geq 500$ into $J \geq 2000$ (H–M, submitted).

Table: p-values on the Random Excursions test

	$J \geq 500$		$J \geq 2000$	
x	MT	SHA1	MT	SHA1
-4	1.0E−10	9.1E−20	1.1E−03	1.2E−04
-3	3.2E−06	8.1E−07	8.4E−02	4.1E−01
-2	4.5E−01	3.1E−01	3.2E−01	1.1E−01
-1	6.4E−02	8.9E−01	2.8E−02	2.8E−01
1	9.0E−02	6.3E−01	2.5E−01	4.7E−01
2	3.8E−02	5.8E−02	8.2E−02	1.7E−01
3	3.5E−06	2.5E−08	5.5E−03	9.1E−05
4	6.7E−16	4.7E−18	5.3E−02	4.0E−06

Modifications of TestU01

We found the following flaws in TestU01 by the three-level test .
We proposed to change several statistics (submitted).

- The `svaria_SampleCorr` test

  wrong : $\sum_{i=1}^{n-1} \left(X_i X_{i+k} - \frac{1}{4} \right)$

  correct : $\sum_{i=1}^{n-1} \left(X_i - \frac{1}{2} \right) \left(X_{i+k} - \frac{1}{2} \right)$

- The `sstring_Run` test
  - wrong : $\frac{X - 4n}{\sqrt{8n}}$ correct : $\frac{X - 4n}{\sqrt{4n}}$
  - wrong : $\sum_{i=1}^{2n} \frac{(X_i - np_i)^2}{np_i(1-p_i)}$ correct : $\sum_{i=1}^{2n} \frac{(X_i - np_i)^2}{np_i}$

Modifications of TestU01

The `smarsa_Savir2` test

- Generates a random integer I_1 in $\{1, \ldots, m\}$.
- Generates a random integer I_2 in $\{1, \ldots, I_1\}$.
- Generates a random integer I_3 in $\{1, \ldots, I_2\}$.
- $\ldots$
- Generates a random integer I_t in $\{1, \ldots, I_{t-1}\}$.
- Repeats this procedure n times to obtain n values of I_t.
- Compares their empirical distribution with the theoretical one via a χ^2 test.

Parameters : m, t, n.

Experiments : we vary m, t and n and apply the three-level test.

Modifications of TestU01

We detected some error in p-values in this test.

Table: p-values of the three level test with $n = 10000$, $m = 1024$

t	5	6	7	8	9	10
MT	1.1E-12	1.6E-15	1.1E-16	0.0E+00	0.0E+00	0.0E+00

Table: p-values of the three level test with $n = 10000$, $m = 16384$

t	5	6	7	8	9	10
MT	3.8E-03	2.0E-013	1.8E-14	7.8E-16	0.0E+00	0.0E+00

Table: p-values of the three level test with $n = 10000$, $m = 131072$

t	5	6	7	8	9	10
MT	1.8E-01	1.4E-01	3.7E-06	1.9E-05	1.8E-08	0.0E+00

Table: p-values of the three level test with $n = 10000$, $m = 1048576$

t	5	6	7	8	9	10
MT	5.8E-05	1.9E-14	2.2E-16	0.0E+00	0.0E+00	0.0E+00

Modifications of TestU01

Table: p-values of the three level test with $n = 10000000$, $m = 1024$

t	5	6	7	8	9	10
MT	2.8E-01	2.3E-02	1.0E-01	2.0E-03	2.7E-05	6.6E-05

Table: p-values of the three level test with $n = 10000000$, $m = 16384$

t	5	6	7	8	9	10
MT	3.3E-01	5.2E-05	4.5E-06	1.4E-05	5.6E-12	3.9E-04

Table: p-values of the three level test with $n = 10000000$, $m = 131072$

t	5	6	7	8	9	10
MT	2.7E-05	2.0E-04	1.7E-07	2.7E-07	1.2E-04	4.7E-07

Table: p-values of the three level test with $n = 10000000$, $m = 1048576$

t	5	6	7	8	9	10
MT	4.8E-07	1.1E-07	5.1E-08	0.0E+00	2.2E-10	1.3E-09

The Birthday Spacings test

The `smarsa_Birthdayspacings` test

- Fix an integer k, and divide $[0,1)^t$ into k subcubes.
- Generate n points in $[0,1)^t$ using nt output values of a PRNG.
- $I_1 \leq I_2 \leq \ldots \leq I_n$ be the (sorted) subcube numbers where these n points fall.
- Compute the spacings $I_{j+1} - I_j$ for $1 \leq j < n$, and count the number Y of collisions between these differences.
- If the output values of the PRNG are i.i.d. over $[0,1)$, the distribution of Y is approximated by $\mathrm{Pois}(n^3/4k)$.

Example (Knuth, '98)

For $k = 2^{25}$, $n = 512$ and $t = 1$,

Y	0	1	2	≥ 3
Exact Probability	0.3688	0.3690	0.1835	0.07869
Poisson Approx.	0.3679	0.3679	0.1839	0.08030

Results of the Birthday Spacings test

For $k = 2^{25}$, $n = 512$ and $t = 1$,

Y	0	1	2	≥ 3
Exact Probability	0.3688	0.3690	0.1835	0.07869
Poisson Approx.	0.3679	0.3679	0.1839	0.08030

To check the accuracy of the approximation of $\mathbb{P}(Y \leq 0)$

- Compute the NN' of the approximated p-values $\mathbb{P}(Y \leq y)$ of the Birthday Spacings test by Poisson approximation.
- Count the number T_i of p-values ≤ 0.3688.
- Test the null hypothesis $T_1, \ldots, T_{N'} \sim_{i.i.d.} B(N, 1 - 0.3688)$.

For $\mathbb{P}(Y \leq 1)$, count the number of p-values $\leq 0.3688 + 0.3690$, and test the null hypothesis
$T_1, \ldots, T_{N'} \sim_{i.i.d.} B(N, 1 - 0.3688 - 0.3690)$.

Results of three level test for Birthday Spacings test

- Left p-values $\mathbb{P}(Y \leq y)$

Table: p-values of the three-level test on the Birthday Spacings test

y	0	1	2
MT	0.081	0.97	0.99
SHA1	0.078	0.62	0.47

- Right p-values $\mathbb{P}(Y \geq y)$

Table: p-values of the three-level test on the Birthday Spacings test

y	1	2	3
MT	0.22	0.97	0.080
SHA1	0.078	0.62	0.47

The LIL test (Wang and Nicol, '15)

- $S^{lil} := \dfrac{\sum_{i=1}^{n}(2X_i - 1)}{\sqrt{2n \ln \ln n}}$.
- $\mathbb{P}(S^{lil} \in (a, b)) \approx \Phi(b\sqrt{2 \ln \ln n}) - \Phi(a\sqrt{2 \ln \ln n})$
  $=: \mathbb{P}_{ap}(S^{lil} \in (a, b))$
  (Φ : the CDF of the standard normal distribution)
- Divides $\mathbb{R}$ into a finite number of subintervals

  $P_0 = (-\infty, -1), P_i = [-1+0.05(i-1), -1+0.05i), P_{41} = [1, \infty)$

- Empirical distribution :
  $\nu(P_i) = |\{j \mid S_j^{lil} \in P_i\}|/m$ for S_1^{lil}, ..., S_m^{lil}
- Theoretical approximated distribution :
  $\mu(P_i) = \mathbb{P}_{ap}(S^{lil} \in P_i)$
- Compares ν with μ by the distance functions
  - $d_1(\nu, \mu) := \dfrac{1}{\sqrt{2}}\sqrt{\sum_i (\sqrt{\nu(P_i)} - \sqrt{\mu(P_i)})^2}$
  - $d_2(\nu, \mu) := \dfrac{1}{42}\sqrt{\sum_i (\nu(P_i) - \mu(P_i))^2}$

The LIL test (Wang and Nicol, '15)

- The tested PRNG is rejected if the distance is greater than a threshold.
- Wang and Nicol determined the threshold of d_1 and d_2 from experiments by MT with $m = 5000, 10000, \ldots, 55000$.
- They used a fitting curve and gave the thresholds: $3.3809m^{-0.541}$ for d_1 and $0.3404m^{-0.583}$ for d_2

Even if we test MT five times, three of 5 are rejected by their criterion of d_2.

Table: Results of the LIL test with $n = 2^{31}$ and $m = 55000$

	threshold	1st	2nd	3rd	4th	5th
d_1	9.2E-03	8.4E-03	1.1E-03	1.3E-03	8.6E-03	8.6E-03
d_2	5.9E-04	5.7E-04	8.0E-04	8.8E-04	5.9E-04	6.1E-04

Investigation of the LIL test

$$S_j^{lil} := \frac{\sum_{i=1}^{n}(2X_i - 1)}{\sqrt{2n \ln \ln n}} \quad (j = 1, \ldots, m)$$

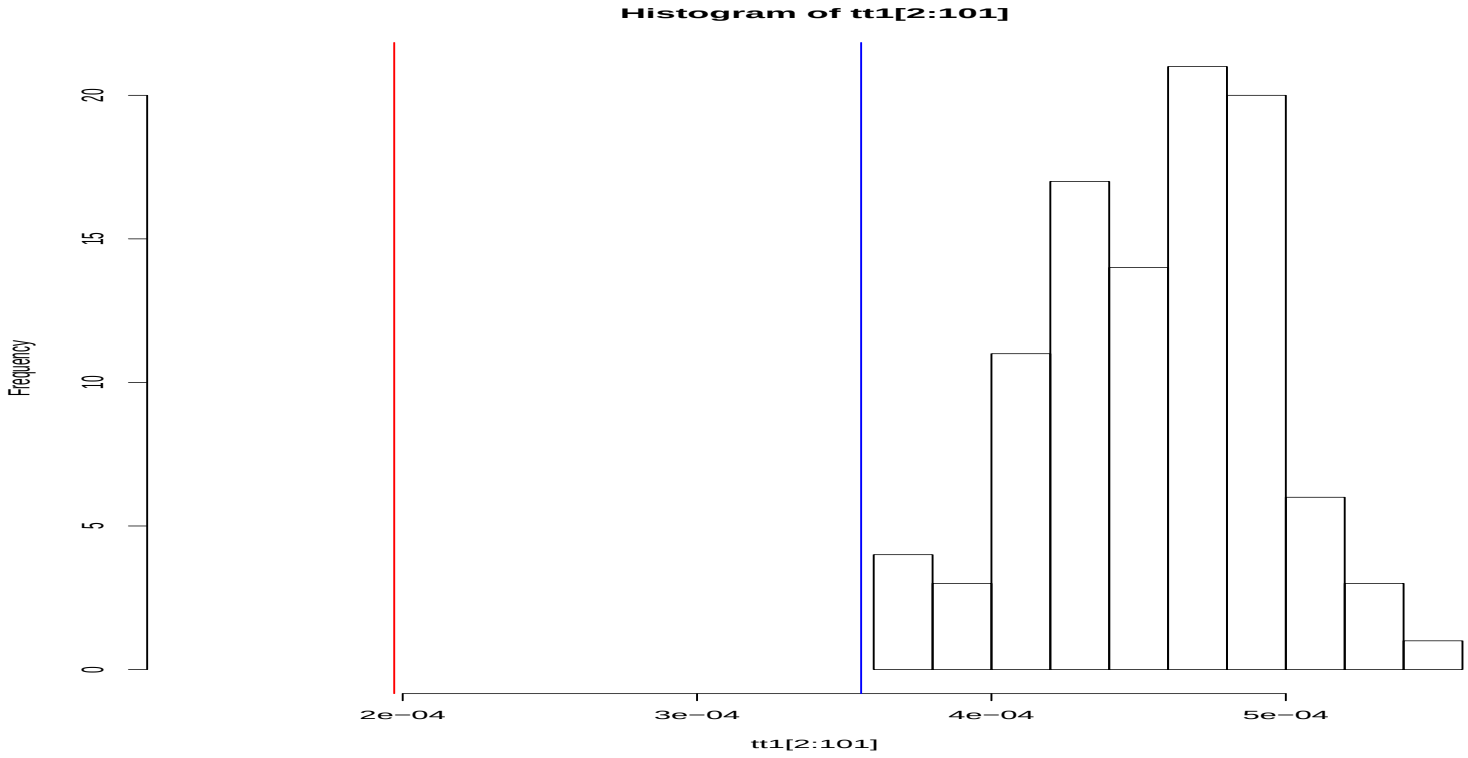
Assume that $m = n$ and $X_1, \ldots, X_n$ are u.i.i.d. on $\{0, 1\}$. We compute the distance d_1' and d_2' between the theoretical distribution of $\mathbb{P}(S^{lil} \in P_i)$ and its approximated distribution of $\mathbb{P}_{ap}(S^{lil} \in P_i)$.

n	2^{26}	2^{27}	2^{28}	2^{29}	2^{30}
$3.3809m^{-0.541}$	2.0E-04	1.4E-04	9.3E-05	7.4E-06	4.4E-05
d_1'	3.6E-04	1.5E-04	1.7E-04	8.4E-05	7.5E-05
$0.3404m^{-0.583}$	9.3E-06	6.2E-06	4.1E-06	2.8E-06	1.8E-06
d_2'	2.9E-05	1.5E-05	1.4E-05	7.4E-06	6.3E-06

$\Longrightarrow$ $3.3809m^{-0.541}$ and $0.3404m^{-0.583}$ give too small thresholds.

Investigation of the LIL test

Generate 100 independent copies $S_1^{lil}, \ldots, S_{100}^{lil}$, and compute d_1



Red : Wang and Nicol's threshold, Blue : d_1'

Future works

- Give a clearer criterion on appropriate parameters for the `smarsa_Savir2` test.
- Check the accuracy of
  - the `snpair_ClosePairs` test
  - the `snpair_ClosePairsBitMatch` test
  - the `sknuth_CollisionPermut` test
- Check the two-level test of the `smarsa_Birthdayspacings` test in Crush and BigCrush.
- Examine L'Ecuyer and Simard's recommendation on the `smarsa_Birthdayspacings` test

$$4n \leq k^{5/12}/N^{1/3}$$

by the three level procedure.

Thank you for your attention.