Stochastic Simulation
Testing random number generaters

Bo Friis Nielsen

Applied Mathematics and Computer Science

Technical University of Denmark

2800 Kgs. Lyngby – Denmark

Email: bfn@imm.dtu.dk

Testing random number generaters

- Theoretical tests/properties
- Tests for uniformity
- Tests for independence

Characteristics of random number generators

Definition: A sequence of *pseudo-random* numbers U_i is a deterministic sequence of numbers in $]0,\ 1[$ having the same relevant statistical properties as a sequence of random numbers.

The question is what are relevant statistical properties.

- Distribution type
- Randomness (independence, whiteness)

Theoretical tests/properties

- Test of global behaviour (entire cycles)
- Mathematical theorems
- Typically investigates multidimensional uniformity

Testing random number generators

- Test for distribution type
  - Visual tests/plots
  - χ^2 test
  - Kolmogorov Smirnov test
- Test for independence
  - Visual tests/plots
  - Run test up/down
  - Run test length of runs
  - Test of correlation coefficients

Significance test

- We assume (known) model - *The hypothesis*
- We identify a certain characterising random variable - *The test statistic*
- We reject the hypothesis if the test statistic is an abnormal observation under the hypothesis

Key terms

- Hypothesis/Alternative
- Test statistic
- Significance level
- Accept/Critical area
- Power
- p-value

Multinomial distribution

- n items
- k classes
- each item falls in class j with probabibility p_j
- X_j is the (random) number of items in class j
- We write $\boldsymbol{X} = (X_1, \ldots, X_2) \;\sim \mathsf{Mul}(n, p_1, \ldots, p_k)$

Thus $X_j \sim \mathsf{Bin}(n, p_j)$ $\mathsf{E}(X_j) = np_j$, $\mathsf{Var}(X_j) = np_j(1 - p_j)$

And $\mathsf{E}\left(\frac{X_j - np_j}{\sqrt{np_j(1-p_j)}}\right) = 0 \quad \mathsf{Var}\left(\frac{X_j - np_j}{\sqrt{np_j(1-p_j)}}\right) = 1$

Thus $\frac{X_j - np_j}{\sqrt{np_j(1-p_j)}} \overset{n\to\infty}{\sim} \mathsf{N}(0, 1)$

Test statistic for $k - 2$

Recall $\frac{X_j - np_j}{\sqrt{np_j(1-p_j)}} \overset{n\to\infty}{\sim} \mathsf{N}(0,1)$

thus $\left(\frac{X_j - np_j}{\sqrt{np_j(1-p_j)}}\right)^2 = \frac{(X_j - np_j)^2}{np_j(1-p_j)} \overset{asymp}{\sim} \chi^2(1)$

Consider now the case $k = 2$

$$\frac{(X_1-np_1)^2}{np_1(1-p_1)} = \frac{(X_1-np_1)^2(p_1+1-p_1)}{np_1(1-p_1)} = \frac{(X_1-np_1)^2}{np_1} + \frac{(X_1-np_1)^2}{n(1-p_1)}$$
$$= \frac{(X_1-np_1)^2}{np_1} + \frac{(X_1-n-n(p_1-1))^2}{n(1-p_1)} = \frac{(X_1-np_1)^2}{np_1} + \frac{(-X_2+np_2)^2}{np_2}$$
$$= \frac{(X_1-np_1)^2}{np_1} + \frac{(X_2-np_2)^2}{np_2}$$

- the χ^2 statistic
- the proof can be completed by induction

Test for distribution type χ^2 test

The general form of the test statistic is

$$T = \sum_{i=1}^{n_{\text{classes}}} \frac{(n_{\text{observed},i} - n_{\text{expected},i})^2}{n_{\text{expected},i}}$$

- The test statistic is to be evaluated with a χ^2 distribution with df degrees of freedom. df is generally $n_{\text{classes}} - 1 - m$ where m is the number of estimated parameters.
- It is recommend to choose all groups such that $n_{\text{expected},i} \geq 5$

Test for distribution type Kolmogorov Smirnov test

- Compare empirical distribution function $F_n(x)$ with hypothesized distribution $F(x)$.
- For known parameters the test statistic does not depend on $F(x)$
- Better power than the χ^2 test
- No grouping considerations needed
- Works only for completely specified distributions in the original version

Empirical distribution

20 $N(0,1)$ variates (sorted):
-2.20, -1.68, -1.43, -0.77, -0.76, -0.12, 0.30, 0.39, 0.41, 0.44, 0.44, 0.71, 0.85, 0.87, 1.15, 1.37, 1.41, 1.81, 2.65, 3.69

X_i iid random variables with $F(x) = \mathsf{P}(X \leq x)$

Each leads to a (simple) random function $F_{e,i}(x) = \mathbf{1}_{\{\mathbf{X_i \leq x}\}}$

leading to $F_e(x) = \frac{1}{n}\sum_{i=1}^{n} F_{e,i}(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{\{\mathbf{X_i \leq x}\}}$

$\mathsf{E}\left(F_e(x)\right) = \mathsf{E}\left(\frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{\{\mathbf{X_i \leq x}\}}\right) = \frac{1}{n}\sum_{i=1}^{n} \mathsf{E}\left(\mathbf{1}_{\{\mathbf{X_i \leq x}\}}\right) = F(x)$

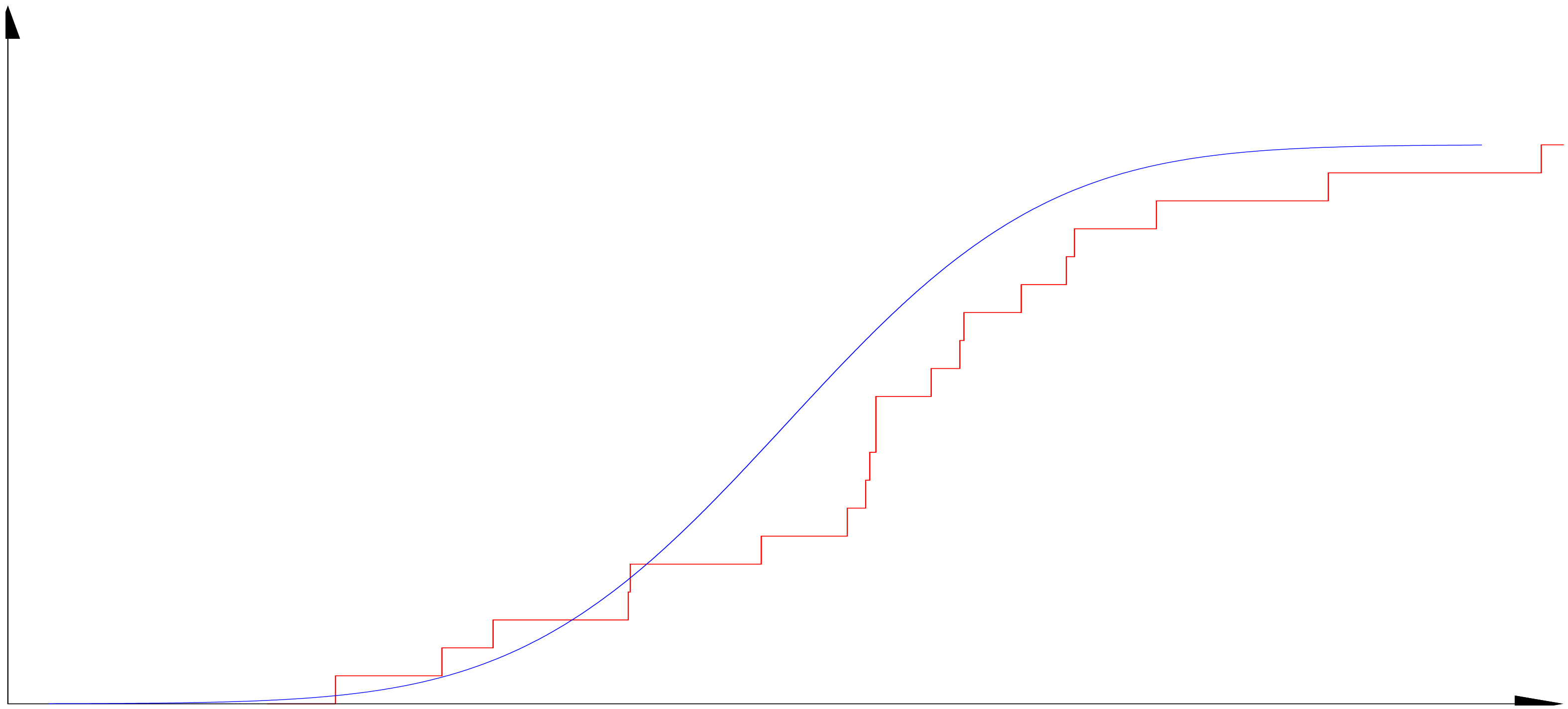
$\mathsf{Var}\left(F_e(x)\right) = \frac{1}{n^2} nF(x)(1-F(x)) = \frac{F(x)G(x)}{n}$

$F_e(x) \overset{n\to\infty}{\sim} \mathsf{N}\left(F(x), \frac{F(x)G(x)}{n}\right)$

In the limit $(n \to \infty)$ we have a random continuous function of x - a stochastic process, more particularly a Brownian bridge

Empirical distribution

20 $N(0,1)$ variates (sorted):
-2.20, -1.68, -1.43, -0.77, -0.76, -0.12, 0.30, 0.39, 0.41, 0.44, 0.44, 0.71, 0.85, 0.87, 1.15, 1.37, 1.41, 1.81, 2.65, 3.69



$$D_n = \sup_x \{|F_n(x) - F(x)|\}$$

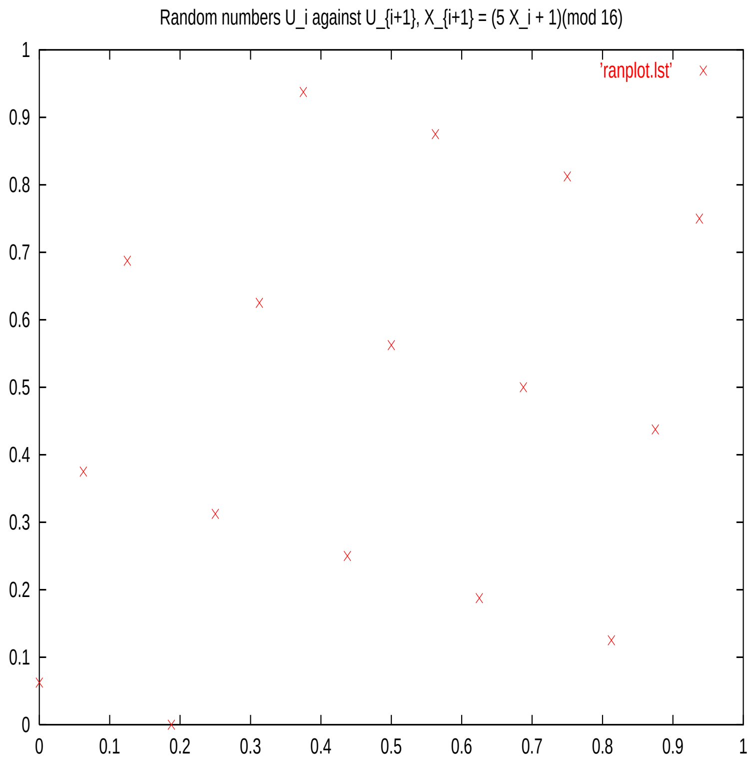
the test statistic follows Kolmogorovs distribution

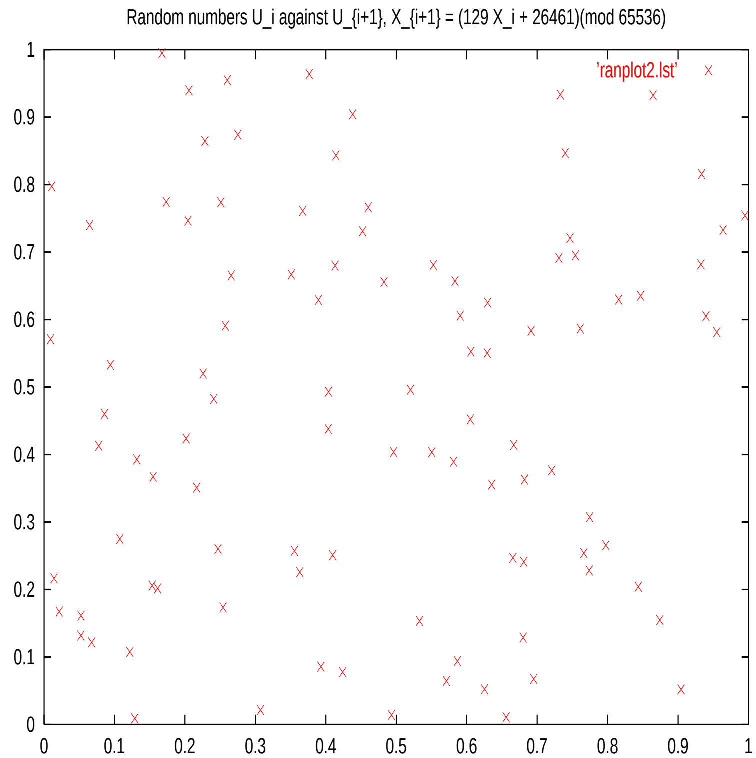
Test statistic and significance levels

		Level of significance $(1-\alpha)$				
Case	Adjusted test statistic	0.850	0.900	0.950	0.975	0.990
All parameters known	$\left(\sqrt{n} + 0.12 + \frac{0.11}{\sqrt{n}}\right) D_n$	1.138	1.224	1.358	1.480	1.628
$N(\bar{X}(n), S^2(n))$	$\left(\sqrt{n} - 0.01 + \frac{0.85}{\sqrt{n}}\right) D_n$	0.775	0.819	0.895	0.955	1.035
$\exp(\bar{X}(n))$	$\left(\sqrt{n} + 0.26 + \frac{0.5}{\sqrt{n}}\right)\left(D_n - \frac{0.2}{n}\right)$	0.926	0.990	1.094	1.190	1.308

Test for correlation - Visual tests

- Plot of U_{i+1} versus U_i





Indepedence test: Test for multidimensional uniformity

- In the two dimensional version test for uniformity of (U_{2i-1}, U_{2i})
- Typically χ^2 test
- The number of groups increases drastically with dimension

Run test I

Above/below

- The Wald-Wolfowitz run test, can be used by e.g. comparing with the median.
- The number of runs (above/below the median) is (asymptotically) distributed as

$$\mathsf{N}\left(2\frac{n_1 n_2}{n_1+n_2}+1, 2\frac{n_1 n_2(2n_1 n_2 - n_1 - n_2)}{(n_1+n_2)^2(n_1+n_2-1)}\right)$$

  where n_1 is the number of samples above and n_2 is the number below.
- The test statistic is the total number of runs $T = R_a + R_b$ with R_a (runs above) and R_b (runs below)

Run tests II

Up/Down from Knuth

A test specifically designed for testing random number generators is the following UP/DOWN run test, see e.g. Donald E. Knuth, The Art of Computer Programming Volume 2, 1998, pp. 66-.

The sequence:

$$0.54, 0.67, |0.13, 0.89, |0.33, 0.45, 0.90, |0.01, 0.45, 0.76, 0.82, |0.24, |0.17$$

has runs of length 2,2,3,4,1, ... i.e. runs of consecutively increasing numbers.

Run test II

Generate n random numbers. The observed number of runs of length $1, \ldots, 5$ and $\geq$6 are recorded in the vector $\boldsymbol{R}$. The test statistic is calculated by:

$$Z = \frac{1}{n-6}(\boldsymbol{R} - n\boldsymbol{B})^T A(\boldsymbol{R} - n\boldsymbol{B})$$

$$A = \begin{bmatrix} 4529.4 & 9044.9 & 13568 & 18091 & 22615 & 27892 \\ 9044.9 & 18097 & 27139 & 36187 & 45234 & 55789 \\ 13568 & 27139 & 40721 & 54281 & 67852 & 83685 \\ 18091 & 36187 & 54281 & 72414 & 90470 & 111580 \\ 22615 & 45234 & 67852 & 90470 & 113262 & 139476 \\ 27892 & 55789 & 83685 & 111580 & 139476 & 172860 \end{bmatrix} \qquad \boldsymbol{B} = \begin{bmatrix} \frac{1}{6} \\ \frac{5}{24} \\ \frac{11}{120} \\ \frac{19}{720} \\ \frac{29}{5040} \\ \frac{1}{840} \end{bmatrix}$$

The test statistic is compared with a $\chi^2(6)$ distribution. One should have $n > 4000$

Run test III

The-Up-and-Down Test This test is described in Rubinstein 81 “Simulation and the Monte Carlo Method” and Iversen 07 (in Danish).

The sequence:

$0.54, 0.67, 0.13, 0.89, 0.33, 0.45, 0.90, 0.01, 0.45, 0.76, 0.82, 0.24, 0.17$

is converted to

$<, >, <, >, <, <, >, <, <, <, >, >$

giving in total 8 runs of length $1, 1, 1, 1, 2, 1, 3, 2$

Run test III

The expected number of runs of length k is $\frac{n+1}{12}$, $\frac{11n-4}{12}$ for runs of length 1 and 2 respectively, and

$$\frac{2[(k^2+3k+1)n-(k^3+3k^2-k-4)]}{(k+3)!}$$

for runs of length $k < N-1$.

Define X to be the total number of runs, then

$$Z = \frac{X - \frac{2n-1}{3}}{\sqrt{\frac{16n-29}{90}}}$$

is asymptotically N(0,1).

Correlation coefficients

- the estimated correlation

$$c_h = \frac{1}{n-h} \sum_{i=1}^{n-h} U_i U_{i+h} \sim \mathsf{N}\left(0.25, \frac{7}{144n}\right)$$

Exercise 1

In this exercise you should implement everything including the tests (e.g. the chi-square and KS tests) yourself. Later, when your code is working you are free to use builtin functions.

1. Write a program implementing a linear congruential generator (LCG). Be sure that the program works correctly using only integer representation.
   (a) Generate 10.000 (pseudo-) random numbers and present these numbers in a histogramme (e.g. 10 classes).
   (b) Evaluate the quality of the generator by graphical descriptive statistics (histogrammes, scatter plots) and statistical tests - χ^2,Kolmogorov-Smirnov, run-tests, and correlation test.
   (c) Repeat (a) and (b) by experimenting with different values of "a", "b" and "M". In the end you should have a decent generator. Report at least one bad and your final choice.

2. Apply a system available generator and perform the various statistical tests you did under Part 1 point (b) for this generator too.

3. You were asked to simulate one sample and perform tests on this sample. Discuss the sufficiency of this approach and take action, if needed.