Zufallszahlen: Statistik

Fabian, Marius, Nils, Tim

Institut für Physik, Humboldt-Universität zu Berlin

Maurers Test

Maurers universeller statistischer Test ist ein entropiebasierter statistischer Test, der es erlaubt schnell große Datenmengen zu prüfen. Wie jeder entropiebasierte Test misst er Gleichwahrscheinlichkeit und nicht echte Unvorhersehbarkeit. Er kann deshalb nicht zwischen Echten und Pseudozufallszahlen unterscheiden.

Besondere Eigenschaften:

- Erkennt eine generelle Klasse von statistischen Fehlern, die durch einen ergodischen, stationären Generator mit begrenztem Speicher modelliert werden kann
- Misst den Einfluss der Zufallszahlen auf ein Verschlüsselungssystems

Funktionsweise:

Der Algorithmus funktioniert, indem eine Sequenz von Zufallszahlen in Blöcke der Länge L aufgeteilt werden. Jeder Block enthält eine der 2^L möglichen Permutationen. Gleichung (1) berechnet aus der Indexdifferenz des Auftauchens eines Blockes (curr) und seinem letztmaligen Auftauchen (last) eine Größe für die pro Bit Entropie f.

$$f = \frac{1}{K} \sum_{n=Q}^{Q+K-1} \log_2(curr(n) - last(n)) \qquad (1)$$

Für $L = 8$ würde ein perfekter Zufallsgenerator einen Wert von $f = 7.1837$ liefern. Eine Sequenz von nur Nullen oder Einsen würde $f = 0$ liefern.

Beispiele:

Im folgendem werden ein positives und ein negatives Beispiel für 'guten' Zufall dargestellt.

- RANDU: $f \approx 6.7$
- 10^6 Nachkommastellen von π: $f \approx 7.18$

Bei RANDU handelt es sich um einen vormals eingesetzten Algorithmus zur Erzeugung von Zufallszahlen, der mit einer pro Bit Entropie von nur 6.7 schlechte Zufallszahlen liefert und so für viele Anwendungen (Verschlüsselungen, Monte-Carlo-Simulationen, etc.) ungeeignet ist. Die Nachkommastellen von π sind hingegen wie zu erwarten vollständig zufällig.

Spektraltest

Der Spektraltest (engl. spectral test) ist ein spezieller Test, der die Güte sogenannter Linear Congruential Generators (LCGs) überprüft. Diese bilden eine Klasse der Pseudo-Zufallszahlengeneratoren (pseudo random number generators, PRNGs) und sind über folgende Vorschrift iterativ definiert:

$$X_{n+1} = (aX_n + c) \bmod m \qquad (2)$$

Die Idee des Tests ist es, aufeinanderfolgende Zahlen einer Kette als Koordinaten eines n-dimensionalen Vektors aufzufassen und dann darzustellen.

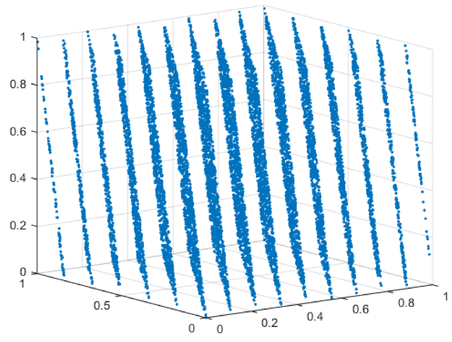

Abbildung 1:Dreidimensionale Darstellung von $N = 10000$ von $RANDU$ ($a = 65539$, $m = 2^{31}$, $c = 0$) generierte Zufallszahlen, gedreht

Die hier gezeigte Eigenschaft ist typisch für jeden LCG. Der gesamte Output verteilt sich auf eine endliche Zahl von Hyperebenen. Die Anzahl und der Abstand dieser Ebenen kann zur Bewertung des Generators genutzt werden. Die Hyperebenen sind dabei nicht eindeutig, es gibt oft mehrere solcher Familien von Ebenen, die alle Punkte überdecken.

Es scheint, als seinen LCGs durch diese enorme Regelmäßigkeit sehr ungeeignet für die Erzeugung von Zufallszahlen. Tatsächlich ist diese aber vernachlässigbar, wenn m nur groß genug gewählt wird. Für praktische Anwendungen benötigt man keine komplette Zufälligkeit, es genügt eine Unabhängigkeit der ersten paar Nachkommastellen, wenn die produzierten Zahlen gerundet und mit endlicher Präzision behandelt werden.

weitere Tests

- **BirthdaySpacings**
  - mithilfe der Zufallszahlen werden Tage innerhalb eines Jahres ausgewählt, also „Geburtstage „
  - Bestimmung der Abstände aufeinanderfolgender Geburtstage, welche einer Poisson-Verteilung genügen sollen
- **SimpPoker**
  - jeweils 5 Zufallszahlen werden zu einer Sequenz zusammen gefasst
  - Untersuchung, wie häufig „Pokerhände “ auftreten, also z.B. Paare, Drillinge, Full House, Straße
- **Collision Test**
  - für diesen Test werden Bälle in Urnen geworfen, wobei es deutlich mehr Urnen als Bälle gibt
  - es werden die „Kollisionen “ gezählt, wie oft also ein Ball in eine bereits belegte Urne geworfen wird
- **CouponCollector**
  - wie lang muss eine Zahlenfolge sein, damit man jede Ziffer „gesammelt “ hat?
  - vergleichbar mit einem Sammelalbum für Sticker