

Zufallszahlenüberprüfung

Dickbauer Y., Moser P., Perner M.

PS Computergestützte Modellierung, WS 2016/17

January 20, 2017

Outline

- 1 Aufgabenstellung
- 2 Erklärung
- 3 Flow Chart

Aufgabenstellung

Erzeugen Sie mit einem gemischten Kongruenzgenerator für verschiedene Parameter 100 Zufallszahlen zwischen 0 und 1. Die Parameter für den gemischten Kongruenzgenerator sollen hierbei flexibel eingegeben werden können oder automatisch in vorgegebenen Intervallen untersucht werden. Die resultierenden Zufallszahlen sollen mittels χ^2 – *Anpassungstest* und Runtest auf Unabhängigkeit geprüft werden (siehe Vorlesungs-Folien).

Die Berechnung der Häufigkeiten (χ^2 – *Anpassungstest*) und Run-Länge, der zugehörigen χ^2 -Testgröße sowie der Vergleich mit dem korrespondierenden Wert aus der χ^2 – *Tabelle* (z.B. zum 95% Signifikanzniveau) soll dabei automatisch erfolgen, wobei die Werte der χ^2 – *Tabelle* hard-codiert werden können.

Aufgabenstellung

- Eingabe: Parameter für gemischten Kongruenzgenerator
- Output: Zufallszahlen, Anzahl an Werten je Bereich, Annahme oder Ablehnung gemäß Tests

χ^2 – Anpassungstest - gemischte Kongruenzmethode

- 1 Input n ... Number of Random Numbers generated
- 2 Classification into \sqrt{n} classes of uniform size
- 3 Calculate the frequency of random number in each class
- 4 Calculate test statistic as the sum over all classes $\frac{(n_i - np)^2}{np}$
- 5 Compare test statistic with rejection value from χ^2 Table
- 6 If X_0 (test statistic) \leftarrow reject value
- 7 random numbers are significantly independent

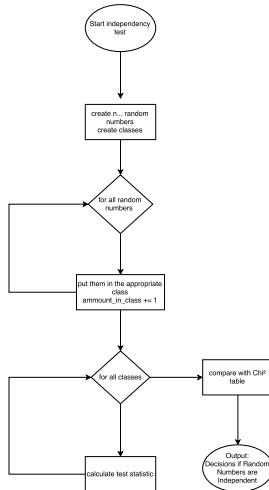
Beispiel

- $n = 100$
- $r = \sqrt{100} = 10 \text{ classes}$
- $u_i = 0.34 \Rightarrow$ is being counted to class 4 ($0.34 - 0.4$)
- test statistic with all classes

Runtest

- ① Input: n = number of Runs
- ② a run is defined as u_i (random numbers) that follow the order of being greater than the u_{i-1}
 - e.g. $u_1 = 0.15, u_2 = 0.3, u_3 = 0.25$
- ③ Run in this case is 2
- ④ Create classes according to length of run (here create class 2)
- ⑤ Count number of run occurrences in each class
- ⑥ Calculate $\chi_0 = \frac{1}{len(classes)!} - \frac{1}{(len(classes)+1)!}$
- ⑦ Calculate $n * \chi$
- ⑧ for test statistic calculate sum over all classes (length of class – np)/np
- ⑨ Test it the same way as before

Flow Chart - Independency Test



Flow Chart - Runtest

