Wilfried Rohm wilfried.rohm@schule.at

Testen von Pseudozufallszahlen (Poker-Test)

Kopfregion

- Mathematische / Fachliche Inhalte in Stichworten:

  Simulation, Zufall, Zufallszahlen, Pseudozufallszahlen, Lineare Kongruenzmethode, Chiquadrattest, Wahrscheinlichkeitsrechnung, Programmieren mit Mathcad, Statistische Tests

- Kurzzusammenfassung

  Der Pokertest ist ein Testverfahren, das - angelehnt an Ereignisse, welche vom "Pokern" her bekannt sind - künstlich erzeugte "Zufallsziffern" (0,1,2,3,4,5,6,7,8,9) statistisch auf Zufälligkeit überprüft.
  Die Erzeugung der Pseudozufallszahlen geschieht hier auf 2 verschiedene Arten mit unterschiedlichen Ergebnissen bzw. Interpretationsmöglichkeiten: Zunächst wird der in Mathcad eingebaute Generator verwendet, anschließend wird eine spezielle Version des linearen Kongruenzgenerators, welche im Taschenrechner TI-59 eingebaut wurde, verwendet und auf "Zufälligkeit" überprüft.

- Didaktische Überlegungen:

  Der Pokertest eignet sich für die Berechnung interessanter elementarer Wahrscheinlichkeiten (wie sie eben auch beim "Pokern" auftreten"). In Zusammenarbeit mit dem Informatikunterricht handelt es sich auch um eine interessante und für Anfänger anspruchsvolle Programmieraufgabe, welche mit jeder Programmsprache auf der Basis von Feldern bzw. Arrays gelöst werden kann.

- Lehrplanbezug (bzw. Gegenstand / Abteilung / Jahrgang):

  Angewandte Mathematik, Einsatz ab den 2. Jahrgängen und in allen Abteilungen möglich

- Literatur bzw. Links über Zufallszahlen:

  Peter Hellekalek, Universität Salzburg: Server über Zufallszahlen http://random.mat.sbg.ac.at/

  Peter Hellekalek: Vorsicht! Zufallszahlen (1999)
  http://random.mat.sbg.ac.at/results/peter/imn99.pdf

  Zufallszahlen und ihre Erzeugung:
  http://www.math.uni-frankfurt.de/~numerik/lehre/Vorlesungen/Comp_Fin09/skript/n-shell3.pdf

  Thomas Morgenstern: Zufallszahlen in Forschung und Lehre

  http://www.home.hs-karlsruhe.de/~moth0001/forschung/Zufallszahlen%20in%20Forschung%20und%20Lehre%20Festschrift%20HS%20Harz.pdf

- Mathcad-Version:

  erstellt mit Mathcad 15 und Mathcad 11

Kopfregion

0) Pseudozufallszahlen

Zufallszahlen spielen in der Forschung und in vielen Bereichen der Angewandten Mathematik (Statistik, Kryptologie, ...) eine wichtige Rolle. Im Wesentlichen werden 2 Verfahren zur Erzeugung von Zufallszahlen unterschieden:

- **"Echte Zufallszahlen"** werden mit physikalischen Prozessen (z.B. radioaktiver Zerfall, Elektronisches Rauschen, quantenphysikalische Effekte) erzeugt. Derartige Verfahren sind zeitlich und technisch aufwändig, werden aber sehr wohl bei heiklen Anwendungen wie der Erzeugung kryptographischer Schlüssel verwendet.

- **"Pseudozufallszahlen"** werden hingegen nach einem vorgegebenen Algorithmus erzeugt, sind also "errechnete" Zahlen, welche nie wirklich "zufällig" sein können.

In der Regel werden ausgehend von einem Startwert nach einem iterativen Verfahren die weiteren Pseudozufallszahlen errechnet. Ein sehr häufig verwendeter Algorithmus ist die "Lineare Kongruenzmethode", welche weiter unten vorgestellt und in einer speziellen Version verwendet wird.

Auch wenn diese Pseudozufallszahlenfolgen nicht "zufällig" sind, haben gute Pseudzufallszahlenfolgen ähnliche **STATISTISCHE** Eigenschaften wie "echte Zufallszahlen". Für einen gegebenen Pseudozufallszahlengenerator muss dies mit verschiedenen statistischen Testverfahren überprüft werden. Eines dieser Testverfahren - der Pokertest - wird hier vorgestellt.
In vielen realen Anwendungen genügen Pseudozufallszahlen den Anforderungen und finden daher auch vielfach Anwendung. Deshalb sind in allen Mathematikprogrammen entsprechende Pseudozufallszahlengeneratoren fix eingebaut.
In der Praxis werden Pseudozufallszahlen mit verschiedenen Testverfahren überprüft, der Pokertest ist hierfür nur eine Variante. Eine andere Variante ist der "**Run-Test**", zu dem es einen eigenen Artikel von mir auf **www.math-tech.at** gibt.

In diesem Artikel werden zwei verschiedene Zufallszahlengenratoren untersucht.

- Der in Mathcad eingebaute Pseudozufallszahlengenrator zur Erzeugung gleichverteilter Zufallszahlen mit Hilfe der Funktion **`rnd`**. Es wird sich zeigen, dass gemäß dem durchgeführten Pokertest dieser zufriedenstellende Ergebnisse liefert!
- Der in einem früheren Spitzenmodell von Texas Instruments (Taschenrechner TI59 aus dem Jahr 1977) eingebaute Pseudozufallszahlengenerator nach der Linearen Kongrumenzmethode war - wie sich herausstellen wird - noch ziemlich schlecht und liefert gemäß dem vorgestellten Pokertest kein zufiredenstellendes Ergebnis!

1) Charakterisierung des Pokertests

Der Pokertest entspricht ideenmäßig dem Pokern mit 5 Würfeln. Dort werden mit Hilfe der 5 Würfel fünf Zufallszahlen, jeweils im Bereich 1..6, erzeugt. Und dann geht es bei diesem Spiel (meist) darum, durch nochmaliges Würfeln frei wegzunehmender Würfel ein "besseres" bzw. "wertvolleres" Ergebnis zu erzielen. Dieser Vorgang darf anschließend nochmals wiederholt werden - also sind (üblicher Weise) maximal 3 Würfe zulässig.
Vom Ergebnis her unterschiedet man:

alle verschieden:	zum Beispiel 13562
1 Paar :	zum Beispiel 35441
2 Paare:	zum Beispiel 16363
3 gleiche:	zum Beispiel 42212
Paar + 3 gleiche:	zum Beispiel 61166 (auch "**Full**" genannt)
4 gleiche:	zum Beispiel 51111 (auch "**Poker**" genannt)
5 gleiche:	zum Beispiel 55555 (auch "**Grande**" genannt)

Anmerkung: Auch sogenannte "Strassen" (nach Sortierung etwa 23456 oder 12345) spielen eine Rolle beim Würfelpokern, sie bleiben hier aber unberücksichtigt (sie werden hier als Teil von "alle verschieden" aufgefasst.)

Beim **Pokertest für Zufallszahlen** erfolgen leichte Veränderungen. Man wirft (gedanklich) 5 "zehnseitige" Würfel, die Würfelseiten enthalten (mit theoretisch gleicher Wahrscheinlichkeit) die Ziffern 0,1,2,3,4,5,6,7,8,9
Allerdings werden nach jedem Wurf wieder 5 neue Zufallszahlen im Bereich 0..9 erzeugt (es gibt also kein "Aussortieren" von Würfeln bzw. Zufallszahlen) - im Sprachjargon des Würfelpokers würde man sagen: "Es wird jedes Mal neu serviert".

Jeder Wurf wird nun nach dem Schema der folgenden Tabelle ausgewertet. Diese Tabelle enthält auch die theoretischen Wahrscheinlichkeiten, deren Formeln weiter unten erklärt werden.

Form des Blockes	Bezeichnung	Beispiel	theoretische Wahrscheinlichkeit
abcde	alle verschieden (*all different*)	58241	$\frac{10\cdot9\cdot8\cdot7\cdot6}{10^5}=0{,}3024$
aabcd	ein Paar gleicher Ziffern (*one pair*)	53903	$\frac{\binom{5}{2}\cdot10\cdot9\cdot8\cdot7}{10^5}=0{,}5040$
aabbc	zwei Paare gleicher Ziffern (*two pairs*)	67726	$\frac{\binom{5}{2}\cdot\binom{3}{2}\cdot\frac{1}{2!}\cdot10\cdot9\cdot8}{10^5}=0{,}1080$
aaabc	drei gleiche Ziffern (*drilling*)	90499	$\frac{\binom{5}{3}\cdot10\cdot9\cdot8}{10^5}=0{,}0720$
aaabb	drei + zwei gleiche Ziffern (*full house*)	11166	$\frac{\binom{5}{3}\cdot10\cdot9}{10^5}=0{,}0090$
aaaab	vier gleiche Ziffern (*poker*)	37333	$\frac{\binom{5}{1}\cdot10\cdot9}{10^5}=0{,}0045$
aaaaa	fünf gleiche Ziffern (*grande*)	55555	$\frac{10}{10^5}=0{,}0001$

2) Berechnung der theoretischen Wahrscheinlichkeiten beim Pokertest

Grundsätzlich kommt hier die elementare Wahrscheinlichkeitsrechnung zum Zug - wobei dies hier auf zwei leicht unterschiedliche Arten erklärt werden soll.
Man kann nämlich einerseits konsequent nach der **Laplacedefinition für Wahrscheinlichkeiten**

$$\text{"Wahrscheinlichkeit des Ereignisses E"} = \frac{\text{"Anzahl der für E günstigen Fälle"}}{\text{"Anzahl aller möglichen Fälle"}}$$

vorgehen - andererseits erhält man das gleiche Ergebnis durch fortgesetztes **Anwenden des Multiplikationssatzes für voneinander unabhängige Ereignisse:**

$$P(E_1 \wedge E_2) = P(E_1)\cdot P(E_2)$$

Die Wahrscheinlichkeiten werden auch gleich in einen Vektor geschrieben (der Name "p" orientiert sich an "probability"). Dieser Vektor wird nämlich weiter unten beim optischen Vergleich sowie beim verwendeten statistischen Test benötigt.

Wahrscheinlichkeit, dass alle Ziffern verschieden sind (z.B: 70142)

Die 1.Ziffer kann eine von 10 möglichen sein, für die zweite Ziffer gibt es dann nur mehr 9 "günstige" Möglichkeiten - und so weiter, da ja keine Ziffer doppelt vorkommen darf: Man kann dies auch als "**Variation ohne Wiederholung**" bezeichnen.

Alle Möglichkeiten sind durch die Zahl 10^5 gegeben ("**Variation mit Wiederholung**").
Daher erhalten wir

$$p_0 := \frac{10\cdot9\cdot8\cdot7\cdot6}{10^5} = 0.302$$

Alternative Überlegung (Multiplikationssatz):

$$p_0 := 1 \cdot \frac{9}{10} \cdot \frac{8}{10} \cdot \frac{7}{10} \cdot \frac{6}{10} = 0.302$$

Die erste Ziffer ist mit Wahrscheinlichkeit $\frac{10}{10} = 1$ irgendeine der 10 Ziffern 0..9. Dass die zweite Ziffer jedenfalls anders ist - diese Wahrscheinlichkeit beträgt $\frac{9}{10}$ und muss multipliziert werden, weil man ja formulieren kann ("und"):
"Wie groß ist die Wahrscheinlichkeit, dass die erste Ziffer irgendeine ist **UND** die zweite Ziffer jedenfalls anders ...."
Dies gilt nun analog für die restlichen Ziffern. (**Multiplikationssatz für voneinander unabhängige Ereignisse**)

Wahrscheinlichkeit für ein Paar gleicher Ziffern (z.B. 66941)

Die erste Ziffer (hier im Beispiel 6) kann jede der 10 Möglichkeiten sein. Die zweite Ziffer -welche das Paar ergänzt - kann nur eine der 10 möglichen sein (daher: "1" in der folgenden Formel). Für die nächste Ziffer gibt es wiederum
9 Möglichkeiten (sie muss nur anders als die 1.Ziffer sein), für die nächstfolgende Ziffer gibt es 8 Möglichkeiten - usw....
Nun sind aber Fälle wie 69641 oder 91646 oder ....in ihrer Bedeutung gleichwertig, daher müssen noch die Anzahl der möglichen Kombinationen $\binom{5}{2} = \text{combin}(5,2) = 10$ berücksichtigt werden.Diese Zahl gibt an, wieviel derartige "bedeutungsgleiche" Ergebnisse möglich sind - anschaulich: Man die 2 Plätze für das Paar aus den 5 Ziffern aus.
Die möglichen Fälle bleiben natürlich gleich wie im ersten Fall, also 10^5. Damit ergibt sich:

$$p_1 := \frac{\text{combin}(5,2) \cdot 10 \cdot 1 \cdot 9 \cdot 8 \cdot 7}{10^5} = 0.504$$

Alternative Überlegung (Multiplikationssatz)

$$p_1 := \text{combin}(5,2) \cdot 1 \cdot \frac{1}{10} \cdot \frac{9}{10} \cdot \frac{8}{10} \cdot \frac{7}{10} = 0.504$$

Die erste Ziffer ist wieder mit Wahrscheinlichkeit $\frac{10}{10} = 1$ irgendeine der 10 Ziffern 0..9. Die zweite Ziffer ist mit Wahrscheinlichkeit $\frac{1}{10}$ gleich der ersten Ziffer. Die nächste Ziffer ist mit Wahrscheinlichkeit $\frac{9}{10}$ anders als die ersten beiden Ziffern usw. Genauso wie oben müssen aber auch noch die Anzahl der möglichen Kombinationen berücksichtigt werden.

Anmerkung : Interessant - weil wohl eher nicht "erwartet" - erscheint die Tatsache, dass der Fall "1 Paar" deutlich wahrscheinlicher als der Fall ist, dass alle Ziffern verschieden sind!

Wahrscheinlichkeit für zwei Paare gleicher Ziffern (z.B. 66001)

$$p_2 := \frac{\frac{\text{combin}(5,2) \cdot \text{combin}(3,2)}{2!} \cdot 10 \cdot 1 \cdot 9 \cdot 1 \cdot 8}{10^5} = 0.108$$

Alternative Überlegung:

$$p_2 := \text{combin}(5,2) \cdot \text{combin}(3,2) \cdot \frac{1}{2!} \cdot 1 \cdot \frac{1}{10} \cdot \frac{9}{10} \cdot \frac{1}{10} \cdot \frac{8}{10} = 0.108$$

Die Überlegungen sind analog zu jenen bei einem Paar. Da ein Vertauschen der Pärchen nichts am "Ergebnis" ändert, muß durch die Anzahl der (jeweils) möglichen Vertauschungen (das ist eben $2!$) noch dividiert werden

Wahrscheinlichkeit für drei gleiche Ziffern (z.B. 66601)

$$p_3 := \frac{\text{combin}(5,3)\cdot 10\cdot 1\cdot 1\cdot 9\cdot 8}{10^5} = 0.072$$

Alternative Überlegung:

$$p_3 := \text{combin}(5,3)\cdot 1\cdot \frac{1}{10}\cdot\frac{1}{10}\cdot\frac{9}{10}\cdot\frac{8}{10} = 0.072$$

Wahrscheinlichkeit für 1 Paar und drei gleiche Ziffern - "Full" (z.B. 66600)

$$p_4 := \frac{\text{combin}(5,3)\cdot \text{combin}(2,2)\cdot 10\cdot 1\cdot 1\cdot 9\cdot 1}{10^5} = 0.009$$

bzw .

$$p_4 := \text{combin}(5,3)\cdot 1\cdot \frac{1}{10}\cdot\frac{1}{10}\cdot\frac{9}{10}\cdot\frac{1}{10} = 0.009$$

Wahrscheinlichkeit für 4 gleiche Ziffern - "Poker" (z.B. 66660)

$$p_5 := \frac{\text{combin}(5,4)\cdot 10\cdot 1\cdot 1\cdot 1\cdot 9}{10^5} = 0.0045$$

bzw .

$$p_5 := \text{combin}(5,1)\cdot 1\cdot \frac{1}{10}\cdot\frac{1}{10}\cdot\frac{1}{10}\cdot\frac{9}{10} = 0.0045$$

Wahrscheinlichkeit für 5 gleiche Ziffern - "Grande" (z.B. 66666)

$$p_6 := \frac{10}{10^5} = 0.0001$$ bzw. $$p_6 := 1\cdot \frac{1}{10}\cdot\frac{1}{10}\cdot\frac{1}{10}\cdot\frac{1}{10} = 0.0001$$

Damit haben wir einen Vektor mit den theoretischen Wahrscheinlichkeiten für die einzelnen möglichen Fälle beim Pokertest definiert:

$$p = \begin{pmatrix} 0.3024 \\ 0.504 \\ 0.108 \\ 0.072 \\ 0.009 \\ 0.0045 \\ 0.0001 \end{pmatrix} \begin{array}{l} \text{Wahrscheinlichkeit für "alle verschieden"} \\ \text{1 Paar} \\ \text{2 Paare} \\ \text{Drilling} \\ \text{Full} \\ \text{Poker} \\ \text{Grande} \end{array}$$

3) Simulation des Pokertests mit dem in Mathcad eingebauten Zufallszahlengenerators

$N = 1000$ Anzahl der Simulationen, welche weiter unten fetgelegt wird.

Für programmiertechnisch weniger geübte Personen dürfte das folgende Programm - obwohl recht kurz - nicht gar so leicht verständlich sein. Die folgenden Kommentare erklären aber hoffentlich einigermaßen die Vorgangsweise:

Die verwendeten Variablen und ihre Bedeutung:

H .... Ein Vektor mit 11 Zeilen, welcher die absoluten Häufigkeiten der einzelenen "Pokerereignisse" speichert - so steht beispielsweise H_0 für die Anzahl der Ergebnisse mit lauter verschiedenen Ziffern, H_1 für die Anzahl der Ergebnisse mit einem Paar, usw....
Einzelne Zeilen (H_5, H_7, H_8, H_9) bleiben dabei leer, was weiter unten erklärt wird.

i,j,n ... Laufvariablen für die jeweiligen "for-Schleifen"

N ... Anzahl der Simulationen - so oft werden jeweils 5 Zufallsziffern erzeigt und gemäß dem Pokertest ausgewertet. (wird weiter unten bei der Grafik global festgelegt)

p ... Speichert für jede Simulation (jeden n-ten Durchlauf) die Anzahl der gleichen PAARE unter den 5 Zufallsziffern
Der Wert dieser Variablen bestimmt **EINDEUTIG**, welches der "Pokerergebnisse" vorliegt, denn:

lauter verschiedene Ziffern:	**p=0**
1 gleiches Paar	**p=1**
2 gleiche Paare	**p=2**
3 gleiche ("Drilling")	**p=3**
Paar und drei gleiche ("Full")	**p=4**
4 gleiche ("Poker"	**p=6**
alle gleich ("Grande")	**p=10**

Um zu verstehen, dass etwa beim Grande p=10 ist, muss man nur jede der 5 Ziffern (z.B. "77777") mit jeder vergleichen - dazu gibt es 10 Möglichkeiten.
Analog ergebn sich die anderen Werte für p.

Es sollte auffallen, dass p=5, p=7, p=8 und p=9 nicht vorkommen können. Deswegen bleiben wie oben erwähnt H_5, H_7, H_8, H_9 leer, denn der jeweilige Wert für p wird als Index für die absoluten Häufigkeiten verwendet.
Deswegen steht
H_0 für die Anzahl der Ergebnisse mit lauter verschiedenen Ziffern (p=0)
H_1 für die Anzahl der Ergebnisse mit 1 Paar (p=1)
H_2 für die Anzahl der Ergebnisse mit 2 Paaren (p=2)
H_3 für die Anzahl der "Drillinge" (p=3)
H_4 für die Anzahl der "Fulls" (p=4)
H_6 für die Anzahl der "Poker" (p=6)
H_{10} für die Anzahl der "Grande" (p=10)

Nach diesen Vorüberlegungen nun zum eigentlichen Programm zur Simulation des Pokertests:

```
pokertest := | for j ∈ 0 .. 10
             |    H_j ← 0
             | for n ∈ 1 .. N
             |    | p ← 0
             |    | for i ∈ 0 .. 4
             |    |    x_i ← floor(rnd(10))
             |    | for i ∈ 0 .. 3
             |    |    for j ∈ i + 1 .. 4
             |    |       p ← p + 1  if  x_i = x_j
             |    | H_p ← H_p + 1
             | return H
```

In der 1.For-Schleife werden alle absoluten Häufigkeiten H_j auf 0 gesetzt - dies ist notwendig für die Summation

For-Schleife für die Anzahl der Simulationen (N)

Anzahl der Paare (p) auf 0 setzen (neue Simulation!)

Erzeugung von 5 "Zufallsziffern" mit dem internen Zufallszahlengenerator im Bereich 0..9

In der i-j-Doppelschleife wird die Anzahl p der gleichen Paare unter den 5 Zufallsziffern ermittelt, indem jede Ziffer mit allen anderen Ziffern aufsteigend verglichen wird.

in Abhängigkeit von der aktuellen Anzahl der Paare (p) wird die jeweilige Häufigkeit hinaufgezählt

Rückgabe des Vektors mit den absoluten Häufigkeiten

pokertest =

	0
0	302
1	505
2	118
3	57
4	11
5	0
6	7
7	0
8	0
9	0
10	0

Hier sieht man das Ergebnis des Programms "pokertest":
Ein Vektor mit den absoluten Häufigkeiten der verschiedenen Poker-Ergebnisse; die erste Spalte ist einerseits der Index des Vektors, andererseits gleich p (Anzahl der gleichen Paare) - daher:

Index 0: Absolute Häufigkeit für "Alle verschieden" (p=0)
Index 1: Absolute Häufigkeit für "1 Paar" (p=1)
....
Index 10: Absolute Häufigkeit für "Grande" (p=10)

Wie oben besprochen bleiben die absoluten Häufigkeiten für p=5,7,8,9 immer leer (d.h. gleich 0), weil diese Fälle nicht auftreten können.

Im Folgenden erfolgt einen "Umdefinition" des eben erhaltenen Vektors, um graphisch und rechnerisch die in der Simulation **beobachteten Werte B_i (absoluten Häufigkeiten)** mit den theoretisch **zu erwartenden Werten E_i** besser vergleichen zu können:

$i := 0 .. 4$

$B_i := pokertest_i \qquad B_5 := pokertest_6 \qquad B_6 := pokertest_{10}$

$$B = \begin{pmatrix} 302 \\ 505 \\ 118 \\ 57 \\ 11 \\ 7 \\ 0 \end{pmatrix} \qquad E := N \cdot p \qquad E = \begin{pmatrix} 302.4 \\ 504 \\ 108 \\ 72 \\ 9 \\ 4.5 \\ 0.1 \end{pmatrix}$$

Graphische Gegenüberstellung: Man kann N variieren (blaues Feld) und erkennt damit auch intuitiv das **Gesetz der großen Zahlen**: Für größer werdenendes N streben die relativen Häufigkeiten (rote Ordner) gegen die theoretischen Wahrscheinlichkeiten (blaue Säulen)

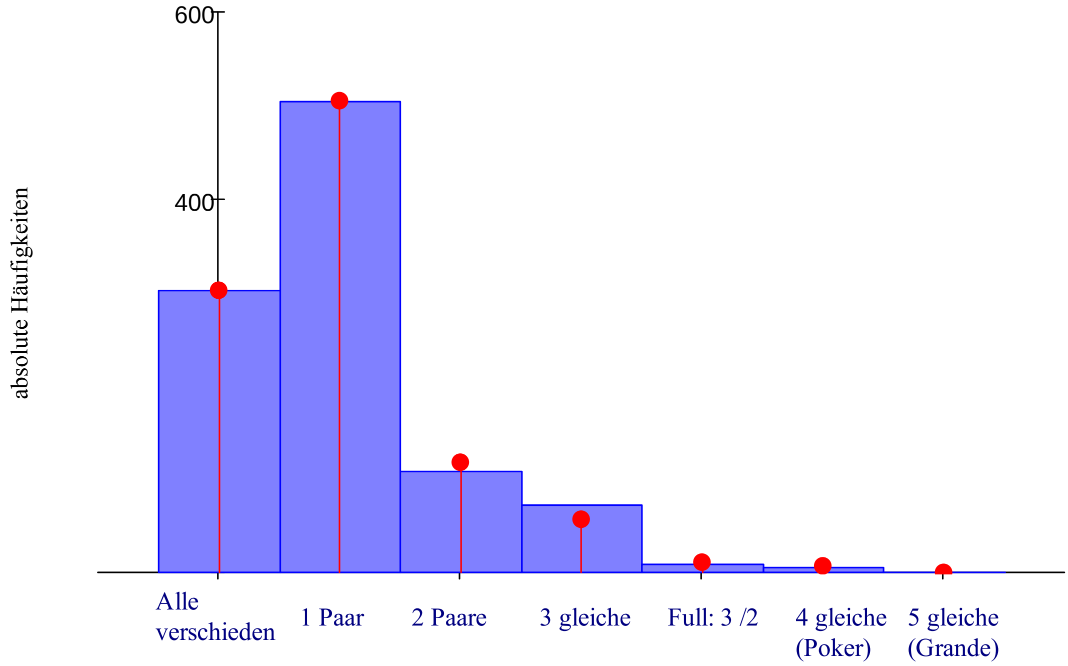
Achtung : Wegen der korrekten Durchführung des Chiquadrattests darf N nicht kleiner als 100 gewählt werden!

$N \equiv 1000$

N variieren / empfohlen:
N=100; N=1000; N=10000 ;
N=100000

Erneute Simulation: **STRG-F9** drücken!

$i := 0..6$



Ergebnis und Interpretation:

- Bei N=100 erkennt man (optisch) meist relativ große Unterschiede zwischen den beobachteten Werten (rot) und den theoretischen Werten (gemäß den errechneten Wahrscheinlichkeiten)
- Bei N=1000 sind diese Unterschiede deutlich geringer, aber der rote Stab für die beobachteten Werte verändert sich etwas bei erneuten Simulationen mit STRG-F9
- Bei N=10000 oder gar N=100000 sind die beobachteten Werte praktisch gleich den theoretischen Werten - es kommt kaum zu Veränderungen bei erneuter Simulation.

FAZIT: Durch diesen optischen Vergleich hat man den Eindruck, dass sich die generierten Pseudozufallszahlen statistisch beim Pokertest wie echte Zufallszahlen verhalten.

Das soll aber nun noch rechnerisch mittels Chiquadrattest überprüft werden.

4) Auswertung mit dem Chiquadrattest

Beim Chiquadrat-Anpassungstest wird zunächste eine bestimmte Verteilung "angenommen" - die Gültigkeit dieses Verteilungsmodells entspricht der Nullhypothese H0. Dann wird über die Prüfgröße bestimmt, ob die normierten Abweichungsquadrate zwischen beobachteten Werten B_i und den unter H0 zu erwartenden Werten E_i als "zufällig" eingestuft werden können - oder ob diese Abweichungen so groß sind, dass (mit einer gewissen Irrtumswahrscheinlichkeit α) davon auszugehen ist, dass das Verteilungsmodell gemäß H0 nicht zu den beobachteten Werten passt.

Es müssen ferner gewisse Voraussetzungen für die erwarteten Werte E_i erfüllt sein, nämlich:

- Alle $E_i \geq 1$
- Höchstens 20% der Werte E_i dürfen kleiner 5 sein.

Diese beiden Voraussetzungen werden bei uns erfüllt, indem wir fordern:

- **$N \geq 100$**
- **Zusammenfassung von Klassen mit kleinen E_i-Werten**

Durchführung des Chiquadrattests:

H0 : Die Verteilung unter Berücksichtigung der Pokerwahrscheinlichkeiten passt zu den beobachteten Werten das bedeutet: Die Pseudozufallszahlen verhalten sich statistisch gesehen gemäß diesem Test wie echte Zufallszahlen oder kurz gesagt: **Der Zufallszahlengenerator ist "GUT"**.

H1 : Die beiden Verteilungen sind unterschiedlich - oder kurz gesagt: **Der Zufallszahlengenerator ist "SCHLECHT"!**

Bevor die Prüfgröße gebildet wird, wird die Anzahl der "Klassen" bestimmt (bzw. die Anzahl der Summanden in der Prüfgröße)

$k := \text{wenn}(N < 1000, 5, 6)$

$k = 6$

Wenn N<1000 ist, fassen wir die letzten 3 Klassen (also Full, Poker und Grande) zu einer Klasse zusammen, sodass die Anzahl der Klassen dann 5 beträgt.
Ist $N \geq 1000$, reicht es aus, nur die beiden letzten Klassen zusammenzufassen, wodurch dann die Klassenanzahl 6 beträgt.

$$B_{k-1} := \sum_{i=k-1}^{6} B_i$$

$f := k - 1$

Der Freiheitsgrad der χ^2-Verteilung ist die um eins verringerte Klassenanzahl

$$Chiq_{pr} := \sum_{i=0}^{k-1} \frac{(B_i - E_i)^2}{E_i} = 5.887$$

Ermittlung der χ^2 - Prüfgröße durch Summation der normierten quadratischen Abweichungen zwischen beobachteten und unter H0 erwarteten Werten (k Summanden)

$chiq\_krit_{0.95} := \text{qchisq}(0.95, f) = 11.07$

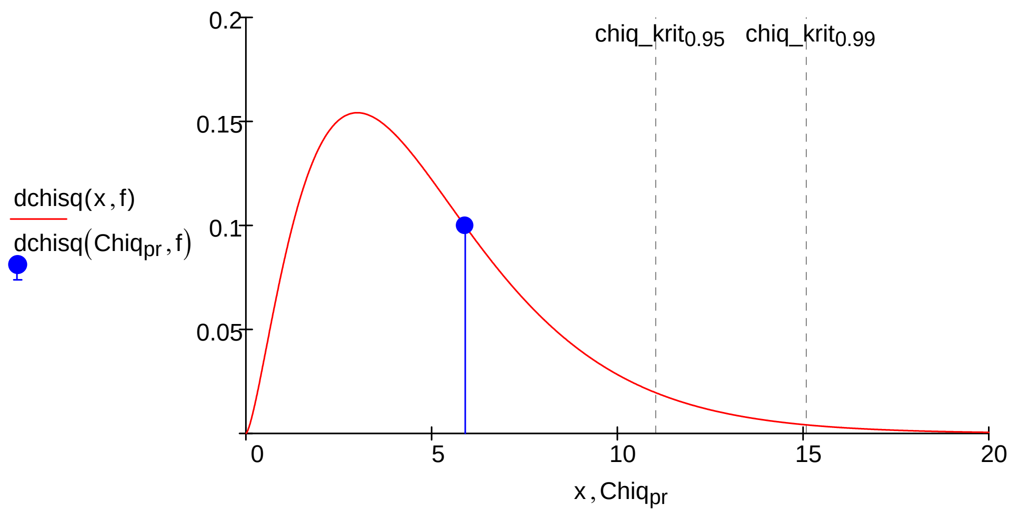
$chiq\_krit_{0.99} := \text{qchisq}(0.99, f) = 15.086$

$chiq\_krit_{0.999} := \text{qchisq}(0.999, f) = 20.515$

Ermittlung drei verschiedener "kritischer" Werte mit den Irrtumswahrscheinlichkeiten
$\alpha = 5\%$, $\alpha = 1\%$ und $\alpha = 0{,}1\%$

Grafische Darstellung des Testergebnisses **Erneute Simulation: STRG-F9** drücken!

$x := 0, 0.01 .. 4 \cdot f$



Ergebnis und Interpretation:

In der Grafik ist die Dichtefunktion der Chiquadratverteilung mit dem Freiheitsgrad f = k-1 gezeichnet. Prüfwert und kritische Werte werden auf der Abszisse aufgetragen - sind aber zur besseren Übersicht als senkrechte Ordner eingetragen.
Unabhängig von der speziellen Wahl für N (siehe oben) sollte Folgendes bei mehrmaliger Simulation (mit STRG-F9) zu beobachten sein:

- "Meist" ist die Prüfgröße $Chiq_{pr}$ (blauer Stamm mit Punkt) kleiner als der kritische Werte $chiq\_krit_{0.95}$. In diesem Fall lautet das Testergebnis, dass H0 nicht abgelehnt wird, d.h. der Pseudozufallsgenerator kann als "gut" angenommen werden.
- Doch: Auch wenn der Zufallszahlengenerator von Mathcad tatsächlich "gut" ist, wird der kritische Wert $chiq\_krit_{0.95}$ immer wieder überschritten - im Schnitt etwa jedes zwanzigste Mal!!! Die gewählte Irrtumswahrscheinlichkeit von α=5% bedeutet nämlich genau das !!
- Im Schnitt jedes hunderste Mal wird $chiq\_krit_{0.99}$ (also α=1%) überschritten

5) Der Zufallszahlengenerator eines Taschenrechners (Texas Instruments TI59)

Alternativ zum internen Zufallszahlengenerator von Mathcad soll ein anderer Zufallszahlengenerator verwendet werden. Dazu habe ich mir den Zufallsgenerator des Rechners TI59 ausgesucht, das "Spitzenmodell" von Texas Instruments aus dem Jahr 1977. Mit diesem Rechner habe ich seinerzeit erstmals im Unterricht den Pokertest durchgeführt (und damit schon damals festgestellt, dass der Zufallszahlengenerator "schlecht" ist). Hier soll gezeigt werden, wie dieser Pseudozufallszahlengenerator mit dem Pokertest als "schlecht" erkannt wird.

Der Zufallszahlengenerator arbeitet nach dem bekannten und viel verwendeten Prinzip der "**Linearen Kongruenzmethode**", das auf LEHMER (1948) zurückgeht. Das Verfahren liefert Zufallszahlen im Bereich 0.. m-1 nach folgender Vorschrift:

x_0 .... Startwert aus dem Bereich [0,m-1)

Iterationsformel : $x_k = (a \cdot x_{k-1} + b) \bmod (m)$

Die Qualität des Zufallszahlengenerators hängt von den speziellen Werten für a, b und m ab (siehe z.b. der Artikel "Zufallszahlen und ihre Erzeugung" -> Link dazu in der Kopfregion!

Für den TI-59 gelten folgende Werte:

$a := 24298$

$b := 99991$

$m := 199017$

Die Umsetzung mit Mathcad könnte so aussehen:

$x_0 := \text{floor}(\text{rnd}(m)) = 35773$

Den Startwert x_0 bestimme ich mit Hilfe des Mathcad-Zufallszahlengenerators. Die Funktion "`floor`" ergibt die "größte ganze Zahl kleiner oder gleich dem Wert `rnd(m)`". `rnd(m)` erzeugt eine gleichmäßig verteilte Fließkomma - Zufallzahl im Bereich 0-m

$k := 1 .. 5$

$x_k := \text{mod}(a \cdot x_{k-1} + b, m)$

Umsetzung der Iterationsformel von oben in Mathcad mit der Modulofunktion

$$x = \begin{pmatrix} 35773 \\ 6089 \\ 180882 \\ 79399 \\ 66095 \\ 9111 \end{pmatrix}$$

Inklusive dem Startwert x_0 wurden damit 6 Pseudo-Zufallszahlen im Bereich 0-m erzeugt.

$$\frac{x}{m} = \begin{pmatrix} 0.18 \\ 0.031 \\ 0.909 \\ 0.399 \\ 0.332 \\ 0.046 \end{pmatrix}$$

Durch die Division durch m erhalten wir (die gewünschten) Pseudozufallszahlen im Bereich (0,1)

Mit Hilfe dieses Zufallszahlengenerators wird nun die gleiche Simulation des Pokertests wie weiter oben durchgeführt. Sie unterscheidet sich von oben ausschließlich durch den anderen Zufallszahlengenerator und die Anzahl der Simulationen (welche durch den Parameter **N2** festgelegt wird - und zwar weiter unten neben der Grafik!)

```
pokertest2 := | for  j ∈ 0 .. 10
              |   H_j ← 0
              | a ← 24298                      Werte für a,c und m gemäß der Linearen
              | c ← 99991                      Kongruenzmethode festlegen
              | m ← 199017
              | y_0 ← floor(rnd(m − 1))        Startwert für die Iteration festlegen
              | for  n ∈ 1 .. N2               Simulation (im Prinzip gleich wie oben)
              |   | p ← 0                      N2 ... Anzahl der Simulationen
              |   | for  i ∈ 1 .. 5
              |   |   y_i ← mod(a · y_(i−1) + c , m)
              |   | y_0 ← y_5                  Der letzte Wert (y5) wird der "Startwert" für die Iteration
              |   |                            der nächsten 5 Zufallsziffern!
              |   | for  i ∈ 0 .. 4
              |   |   x_i ← floor(y_i/m · 10)  5 Zufallsziffern im Bereich 0,1,2,..9 werden im Vektor x
              |   |                            gespeichert.
              |   | for  i ∈ 0 .. 3
              |   |   for  j ∈ i + 1 .. 4
              |   |     p ← p + 1  if  x_i = x_j
              |   | H_p ← H_p + 1
              | return  H
```

so wie oben liefert die Funktion einen Vektor mit den absoluten Häufigkeiten der einzelnen Pokerereignisse

$pokertest2^T =$

	0	1	2	3	4	5	6	7	8	9	10
0	276	522	154	39	9	0	0	0	0	0	0

$i := 0..4$ $\quad B2_i := pokertest2_i$ $\quad B2_5 := pokertest2_6$ $\quad B2_6 := pokertest2_{10}$

Gleiche Vorgangsweise wie oben zur Bestimmung der "beobachteten Werte" $B2_i$ und der erwarteten Werte E_i

$$B2 = \begin{pmatrix} 276 \\ 522 \\ 154 \\ 39 \\ 9 \\ 0 \\ 0 \end{pmatrix} \qquad E := N2 \cdot p \qquad E = \begin{pmatrix} 302.4 \\ 504 \\ 108 \\ 72 \\ 9 \\ 4.5 \\ 0.1 \end{pmatrix}$$

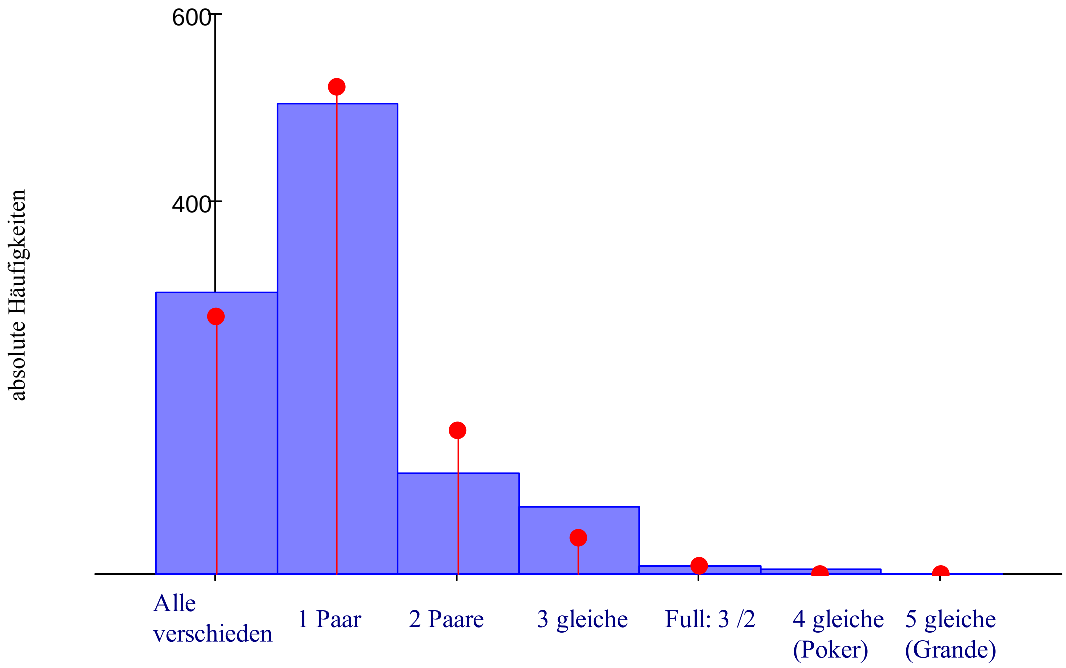
Graphische Gegenüberstellung:

Achtung : Wegen der korrekten Durchführung des Chiquadrattests darf N2 nicht kleiner als 100 gewählt werden!

$i := 0 .. 6$

$N2 \equiv 1000$

N2 variieren / empfohlen:
N2=100; N2=1000; N2=10000 ;
N2=100000
Erneute Simulation: **STRG-F9** drücken!



Auswertung mit dem Chiquadrattest (analog wie oben)

$k := \text{wenn}(N2 < 1000, 5, 6) \qquad k = 6 \qquad B2_{k-1} := \sum_{i=k-1}^{6} B2_i$

$f := k - 1$

$$Chiq_{pr} := \sum_{i=0}^{k-1} \frac{(B2_i - E_i)^2}{E_i} = 42.165$$

Prüfwert und kritische Werte analog zu oben bestimmen

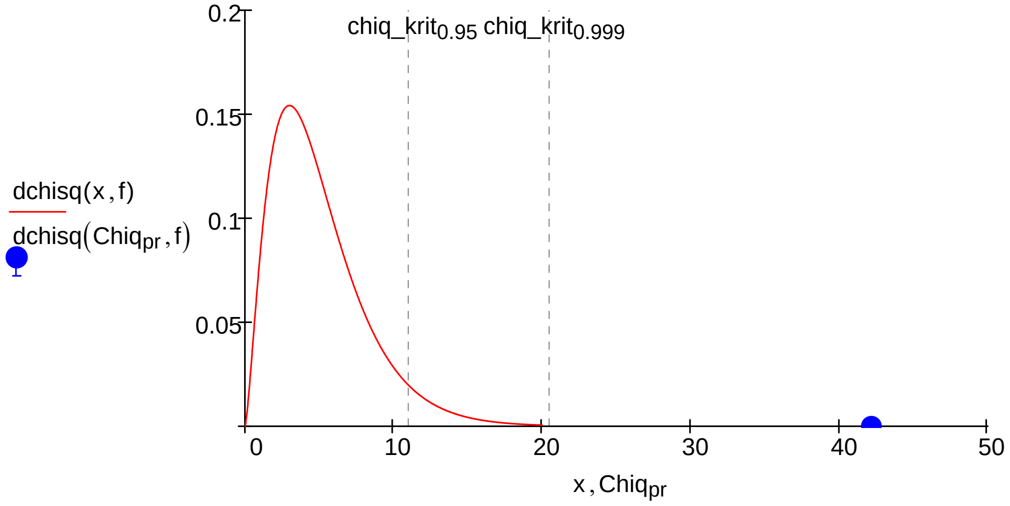
$chiq\_krit_{0.95} := \text{qchisq}(0.95, f) = 11.07$

$chiq\_krit_{0.99} := \text{qchisq}(0.99, f) = 15.086$

$chiq\_krit_{0.999} := \text{qchisq}(0.999, f) = 20.515$

Grafische Darstellung des Testergebnisses

$x := 0, 0.01 .. 4 \cdot f$



Ergebnis / Interpretation:

- Bei N2=1000 Simulationen kommt es praktisch immer zu einer deutlichen **Ablehnung der Nullhypothese** (H0: "die Zufallszahlen sind gut"), mit einer Irrtumswahrscheinlichkeit von α deutlich kleiner als 0,1%.

  Das bedeutet: Mit an Sicherheit grenzender Wahrscheinlichkeit ist dieser Zufallszahlengenerator **SCHLECHT** - der Verteilungsvergleich mit den Pokerwahrscheinlichkeiten fällt negativ aus.

- Dies erkennt man - im Vergleich zum Mathcad-Zufallszahlengenrator - auch optisch beim Häufigkeitsdiagramm, wenn man etwa N2=10000 setzt. **Die theoretischen Pokerwahrscheinlichkeiten werden durch die Simulation nicht ausreichend gut angenähert.** Der Chiquadrattest liefert also bloß eine rechnerische "Bestätigung" für das, was man auch mit "freiem Auge" sehen kann.

- Bei N2=100 Simulationen fällt auf, dass die Nullhypothese ("die Zufallszahlen sind gut") relativ häufig (auch auf dem Niveau α=5% **NICHT** abgelehnt wird.
  Das bedeutet: Der Fehler 2.Art ("irrtümliche Annahme der Nullhypothese") ist hier ziemlich groß.
  Dies deckt sich mit der allgemein gültigen Erkenntnis, dass der Fehler 2.Art über die "Versuchszahl" (=hier die Anzahl der Simulationen) unter Kontrolle gehalten wird.
  Für N=100 ist die Versuchszahl also so klein, dass der Fehler 2.Art sehr groß wird!
  Für größeres N2 (z:B: N=1000) ist hingegen der Fehler 2.Art sehr klein - es kommt so gut wie nie zu einer Annahme der Nullhypothese.