Benford-Verteilung

Benford-Verteilung Haftendorn 2011

siehe Erklärungen "Benford-Gesetz-Beweis.pdf" auf www.mathematik-verstehen.de Bereich Stochastik -> Verteilungen -> Diskrete Verteilungen.

Auf diesen Seiten hier werden die Benford-Wahrscheinlichkeiten berechnet und dargestellt.

Auf Seite 3 wird das Wachstum mit der e-Funktion simuliert und die führenden Ziffern werden untersucht.

Sie sind Benford-verteilt. Das wird noch mit einem Chiquadrattest untermauert.

Theoretische Benford-Verteilung

ben(x)=P(x ist führende Ziffer)

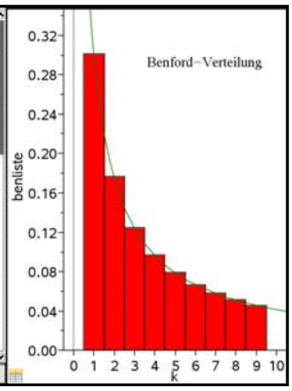
ben$(x):=\log_{10}\left(1+\frac{1}{x}\right)$ ▸ *Fertig*

benliste:=approx(seq(**ben**(x),x,1,9))
▸ {0.30103,0.176091,0.124939,0.09691,0.079181,0.066947,0.057992,0.051153,0.045757}

1.1

	A k	B benliste	C	D
◆	=seq(i,i,1,!			
1	1	0.30103		
2	2	0.176091		
3	3	0.124939		
4	4	0.09691		
5	5	0.079181		
6	6	0.066947		
7	7	0.057992		
8	8	0.051153		
9	9	0.045757		

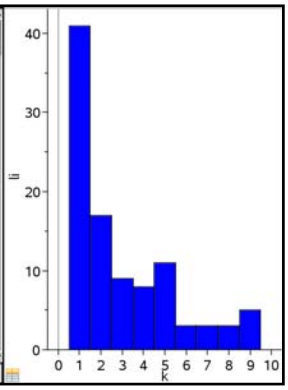
B1 =0.30102999566399



1.2

	:ufall	C ew	D fu...	E k	F li
◆		=e^(zufall*1.ε			
1	627	1.06471	1.	1	41
2	784	1.08156	1.	2	17
3	1244	1.13247	1.	3	9
4	1257	1.13394	1.	4	8
5	1698	1.18507	1.	5	11
6	2882	1.33402	1.	6	3
7	4280	1.53419	1.	7	3
8	4400	1.55271	1.	8	3
9	5259	1.69198	1.	9	5
10	7719	2.16387	2.		
11	8574	2.35702	2.		
12	10802	2.94527	2.		
13	11385	3.12208	3.		
14	11704	3.22328	3.		

F1 =41



1.3

Für die Simulation wird zunächste ein Zufallsliste mit 100 Werten zwischen 1 und 10000 erzeugt.

zufall:=seq(randint(1,10000−1)) Mit sortA **zufall** **wird diese Liste sortiert.**

Anmerkung: sortA gibt die genannte Liste sortiert wieder zurück, man kann also die ursprüngliche Liste nicht bewahren. Dieses Verhalten ist didaktisch höchst ungeschickt. Dabei wird auch die ursprüngliche Definition der Liste gelöscht.

Mit **ew**:=$e^{0.0001\cdot \mathbf{zufall}}$ werden e-Funktionswerte zwischen 1 und etwa 2500 erzeugt.

Die auf der letzten Seite definierte Funktion **fzif(x)** gibt die führenden Ziffern dieser Werte zuück. Sie werden in der Liste **fuehrend** abgelegt.

Leider wirkt meine Funktion fzif(ew) nicht, ist also nicht "listable".

daher ist fzif(c1) nach unten gezogen. Mit seq wäre es auch gegangen.

Schließlich ist mit **li**:=seq(countIf(**fuehrend**,j·1),j,1,9)

die Häufigkeitsliste der Ziffern erzeugt.

Mit Re-Maus Ergebnisdiagramm ist sie als Histogramm dargestellt.

1.4

Chi-Quadrattest

Es bietet sich an, hier einen Chi-Quadrattest durchzuführen.

Die Rechnungen(lt. meinem Statistigheft Seite 30) sind auf der nächsten Seite durchgeführt.

Es ergibt sich **chiq** ▸ 10.7415 und damit χ^2Cdf(**chiq**,1000,8) ▸ 0.21678

Die Simulation ist also mit der Benford-Verteilung verträglich, was nicht überrascht.

χ^2GOF **li**,**hi**,8: **stat.results** ▸ $\begin{bmatrix} \text{"Titel"} & \text{"}\chi^2\text{ GOF"} \\ \text{"}\chi^2\text{"} & 10.7415 \\ \text{"PVal"} & 0.21678 \\ \text{"df"} & 8. \\ \text{"CompList"} & \text{"\{...\}"} \end{bmatrix}$

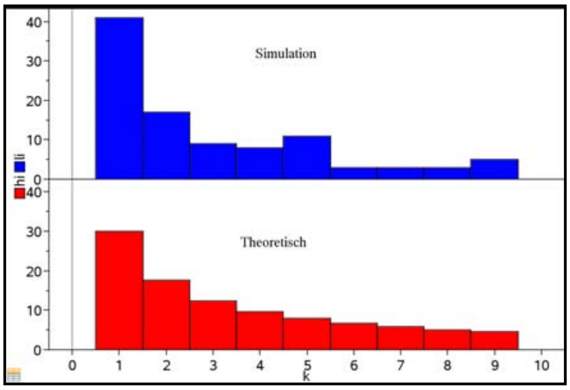
Also die Auswertung geht auch in einem Rutsch.

1.5

	A k	B li	C benliste	D hi	E delta	F deltaq	G deltaqre	H
◆				=benliste*	=hi-li	=delta^2	=deltaq/hi	
1	1	41	0.30103	30.103	-10.897	118.745	3.94461	
2	2	17	0.176091	17.6091	0.609126	0.371034	0.021071	
3	3	9	0.124939	12.4939	3.49387	12.2072	0.977051	
4	4	8	0.09691	9.691	1.691	2.85949	0.295066	
5	5	11	0.079181	7.91812	-3.08188	9.49796	1.19952	chiq
6	6	3	0.066947	6.69468	3.69468	13.6507	2.03903	**10.7415**
7	7	3	0.057992	5.79919	2.79919	7.83549	1.35113	0.21678
8	8	3	0.051153	5.11525	2.11525	4.47429	0.874696	
9	9	5	0.045757	4.57575	-0.424251	0.179989	0.039335	

H7 =χ^2cdf(h6,1000,8)

1.6



1.7

"fzif" erfolgreich gespeichert

```
Define fzif(x)=
Func
© bestimmt die führende Ziffer
Local z,xx
  xx:=|x|
  While xx>10
   xx:=xx/10
  EndWhile
  If xx≥1 Then
   z:=int(xx)
  ElseIf xx=0 Then
   z:=0
  Else
   While xx<1
    xx:=xx·10
   EndWhile
   z:=int(xx)
  EndIf
```

fzif(78.34) ▸ 7.
fzif(678686876876) ▸ 6
fzif(2.E-6) ▸ 2.
fzif(1) ▸ 1
fzif(0) ▸ 0
fzif(-654) ▸ 6

1.8