

Seite 354

Detaillierte Lösungen für SHARP EL-9900G

Seite 354 Lehrtext: Sigma - Regeln

Dieses Beispiel lässt sich nicht so ohne Weiteres auf dem EL-9900G umsetzen. Mit einem kleinen Programm kann man dennoch die Screenshots aus dem Buch erhalten. Dieses Programm benutzt den Befehl *cdfnorm*, den die SchülerInnen an dieser Stelle noch nicht kennen und von dem sie auch noch nicht wissen, warum er für große Werte von *n* statt *cdfbin* benutzt werden darf.

Das Programm *sigmareg.g4p* ist unter http://www.sharp.de/cps/rde/xbcr/documents/documents/Service_Information/Software/Sigma_Regeln.zip herunterzuladen und mit dem PC-Link auf den GTR zu übertragen. Ist das Programm auf den EL-9900G überspielt, kann es mit **PRGM** **A** aufgerufen werden.



Nach Aufruf von *SIGMAREG* erscheint ein Bildschirm, in dem kurz erklärt wird, was das Programm berechnet: Es bestimmt für verschiedene *n* die Wahrscheinlichkeiten, dass eine binomialverteilte Zufallsvariable in einem 1σ -, 2σ - oder 3σ -Intervall liegt, also $P(X \in [\mu - \sigma, \mu + \sigma])$, $P(X \in [\mu - 2\sigma, \mu + 2\sigma])$ und $P(X \in [\mu - 3\sigma, \mu + 3\sigma])$.

SIGMAREG
DAS PROGRAMM
BERECHNET FUER
VERSCHIEDENE N
DIE WAHRSCHEINLICH-
KEITEN FUER EIN
1, 2, 3-SIGMA-INTERVALL

Bestätigt man mit **ENTER**, wird man nach einigen Variablen gefragt: der Wahrscheinlichkeit *p*, der Anzahl der Berechnungen *B* und der Schrittweite *D*. Geben wir *B*=10 und *D*=200 ein, werden die Wahrscheinlichkeiten für *n*=200, 400, 600, ..., 2000 berechnet.

P=
.4
ANZAHL D. BERECHNUNG
B=
10
SCHRITTWEITE
D=
200

Wir bestätigen mit **ENTER** und erhalten die Information, dass wir die Ergebnisse im Listen-Editor anschauen sollen, wobei *L1* die Werte für *n*, *L2* $P(X \in [\mu - \sigma, \mu + \sigma])$, *L3* $P(X \in [\mu - 2\sigma, \mu + 2\sigma])$ und *L4* $P(X \in [\mu - 3\sigma, \mu + 3\sigma])$ enthalten.

ERGEBNISSE BITTE IM
LISTEN-EDITOR
ANSCHAUEN:
L1 N
L2 1-SIGMA-,
L3 2-SIGMA-,
L4 3-SIGMA-INTERVALL

Wir gehen also in den Listen-Editor (**STAT** **A** **ENTER**) und schauen uns die Ergebnisse an. Bis auf geringe Abweichungen gleichen diese den Werten der Screenshots im Buch.

No	1: L1	2: L2	3: L3
1	200	0.68143	0.953937
2	400	0.68206	0.954218
3	600	0.68227	0.954312
4	800	0.682374	0.954359
5	1000	0.682457	0.954387
6	1200	0.682479	0.954406
7	1400	0.682479	0.954406
8	1600	0.682479	0.954406
9	1800	0.682479	0.954406
10	2000	0.682479	0.954406

No	4: L4	5: L5	6: L6
1	0.997231	-----	-----
2	0.997266	-----	-----
3	0.997277	-----	-----
4	0.997283	-----	-----
5	0.997286	-----	-----
6	0.997289	-----	-----

No	1: L1	2: L2	3: L3
6	1200	0.682479	0.954406
7	1400	0.682509	0.954419
8	1600	0.682532	0.954429
9	1800	0.682549	0.954437
10	2000	0.682563	0.954443
11	-----	-----	-----

No	4: L4	5: L5	6: L6
6	0.997289	-----	-----
7	0.99729	-----	-----
8	0.997292	-----	-----
9	0.997293	-----	-----
10	0.997293	-----	-----
11	-----	-----	-----

Seite 354 Beispiel

Mit dem Befehl *cdfbin*(*n, p, k*) ist Vorsicht geboten. Falls die Parameter für *n* und *k* zu groß sind, gibt der EL-9900G ohne Fehlermeldung als Ergebnis 0 aus.

Das Ergebnis 0 muss man bei diesem Befehl also immer hinterfragen.

In diesem Beispiel haben wir jedoch Glück:

Wir geben den Befehl *cdfbin* (**STAT** **F** **1** **1**), gefolgt von den drei Parametern ein. Es funktioniert: Bei beiden Berechnungen erhält man Ergebnisse.

$cdfbin\left(600, \frac{1}{6}, 118\right)$
0.976818855
 $cdfbin\left(600, \frac{1}{6}, 81\right)$
0.019298101

Also kann man die Befehle wie in Fig. 4 in einer Zeile eingeben:

$cdfbin\left(600, \frac{1}{6}, 118\right) - cdfbin\left(600, \frac{1}{6}, 81\right)$
0.957520753