

TrÜbPI Münsingen – Kampfmittelräumung
Berechnungswerkzeug zu Testfelduntersuchungen
Stichprobenplanung und -ziehung
Fachkonzept

Mai/Juni 2003

Dr. Helmut Küchenhoff

Dr. Andreas Henking

Inhaltsverzeichnis

INHALTSVERZEICHNIS	2
1 EINLEITUNG UND ÜBERBLICK	3
2 MODUL STICHPROBENPLANUNG	3
2.1 Hilfsgrößen	3
2.2 Hilfsfunktionen	4
2.2.1 verarbeitung_vorwissen	4
2.2.2 berechne_nh	4
2.2.3 berechne_stdabw_schichtmittel	5
2.2.4 quantil	5
2.3 Varianzoptimal	6
2.4 vorgabe_genauigkeit	6
2.5 vorgabe_stichprobenumfänge	7
3 MODUL STICHPROBENZIEHUNG	7
3.1 Zufallszahlen	7
3.1.1 Zufallszahlengenerator	7
3.1.2 Zufallszahlenziehung	8
3.2 Sortierung	8
3.3 Bestimmung der Schichtgröße zweiter Ordnung	9
3.4 Stichprobenziehung	10
3.4.1 Bestimmung der Stichprobenumfänge	10
3.4.2 Testfeldauswahl	11
3.5 Ausgabe	12

1 Einleitung und Überblick

In diesem Fachkonzept wird die für das Berechnungswerkzeug zur „Studie zur Testfelduntersuchung“ nötige Rechenlogik beschrieben. Dabei wird eine implementierungsnaher Beschreibung, etwa durch Bildung von Hilfsfunktionen, gewählt.

Das Berechnungswerkzeug unterteilt sich in die zwei Module „Stichprobenplanung“ und „Stichprobenziehung“. In dem Modul Stichprobenplanung muss je nach Eingabe das Vorwissen über die Schichten umgerechnet werden und es können für drei Planungsmethoden Berechnungen angestoßen werden. In dem Modul Stichprobenziehung werden mit Hilfe eines Zufallszahlengenerators Stichproben gezogen.

2 Modul Stichprobenplanung

Die Schichtungsparameter werden als globale Variable behandelt und finden sich daher nicht bei den Eingaben der jeweiligen Berechnungsfunktionen wieder.

Programmabfolge:

1. Berechnung Hilfsgrößen
2. verarbeitung_vorwissen
3. Planungsfunktionsaufruf
 - a. varianzoptimal oder
 - b. vorgabe_genauigkeit oder
 - c. vorgabe_stichprobenumfang
4. Ergebnisausgabe

2.1 Hilfsgrößen

Die Hilfsgrößen sind solche, die in den Berechnungen häufig verwendet werden und nur von den globalen Schichtvariablen abhängen. Daher werden die Hilfsgrößen auch als globale Variable behandelt.

$$N = \sum_{h=1}^L N_h$$

$$W_h = \frac{N_h}{N}, \quad h = 1, \dots, L$$

$$H_1 = \left(\sum_{h=1}^L W_h \sigma_h \sqrt{c_h} \right)^2$$

$$H_2 = \frac{1}{N} \sum_{h=1}^L W_h \sigma_h^2$$

$$H_3 = \sum_{h=1}^L N_h \frac{\sigma_h}{\sqrt{c_h}}$$

2.2 Hilfsfunktionen

round(x) rundet x

ceil(x) rundet x auf die nächst größere ganze Zahl auf

2.2.1 verarbeitung_vorwissen

Diese Hilfsfunktion ist global. D.h. die Ergebnisse sind im gesamten Programm verfügbar.

Eingabe: *Vorwissen*

Ausgabe: $\sigma_h, h = 1, \dots, L$

Berechnung:

Falls *Vorwissen* = Spannweite

$$\sigma_h = \frac{y_{o_h} - y_{u_h}}{4}, \text{ für } h = 1, \dots, L$$

sonst

σ_h , für $h = 1, \dots, L$, bereits durch Nutzereingabe richtig belegt

2.2.2 berechne_nh

Eingabe: n

Ausgabe: n_h^*

Berechnung:

$$n_h^* = \frac{n \cdot N_h \cdot \frac{\sigma_h}{\sqrt{c_h}}}{H_3}$$

2.2.3 berechne_stdabw_schichtmittel

Eingabe: n_h

Ausgabe: $\sigma(\hat{Y}_h)$

Berechnung:

$$\sigma(\hat{Y}_h) = \sqrt{\frac{1}{n_h} \left(1 - \frac{n_h}{N}\right)} \cdot \sigma_h$$

2.2.4 quantil

Eingabe: $1 - \alpha$

Ausgabe: $z_{1 - \frac{\alpha}{2}}$

Berechnung:

$1 - \alpha$	$z_{1 - \frac{\alpha}{2}}$
0,9	1,64
0,95	1,96
0,99	2,58

2.3 Varianzoptimal

Eingabe: $n, 1-\alpha$

Ausgabe: n_h, G, G_h , für $h=1, \dots, L$

Berechnung:

$$n_h = \max(2, \text{round}(\text{berechne_nh}(n))), \text{ für } h=1, \dots, L$$

$$G_h = \text{quantil}(1-\alpha) \cdot \text{berechne_stdabw_schichtmittel}(n_h), \text{ für } h=1, \dots, L$$

$$G = \text{quantil}(1-\alpha) \cdot \sqrt{\sum_{h=1}^L W_h^2 \cdot \text{berechne_stdabw_schichtmittel}(n_h)^2}$$

2.4 vorgabe_genauigkeit

Eingabe: $G, 1-\alpha$

Ausgabe: n_h, n, G_h , für $h=1, \dots, L$

Berechnung:

Lokale Hilfsgrößen:

$$\sigma_{\text{genau}}^2 = \left(\frac{G}{\text{quantil}(1-\alpha)} \right)^2$$

$$C_{\text{opt}} = \frac{H_1}{\sigma_{\text{genau}}^2 + H_2}$$

$$H_4 = \sum_{h=1}^L N_h \sigma_h c_h$$

$$n = \text{ceil} \left(\frac{C_{\text{opt}} \cdot H_3}{H_4} \right)$$

$$n_h = \max(2, \text{ceil}(\text{berechne_nh}(n))), \text{ für } h=1, \dots, L$$

$$G_h = \text{quantil}(1-\alpha) \cdot \text{berechne_stdabw_schichtmittel}(n_h), \text{ für } h=1, \dots, L$$

2.5 vorgabe_stichprobenumfänge

Eingabe: n_h , für $h=1, \dots, L$, $1-\alpha$

Ausgabe: G, G_h , für $h=1, \dots, L$

Berechnung:

$$G_h = \text{quantil}(1-\alpha) \cdot \text{berechne_stdabw_schichtmittel}(n_h)$$

$$G = \text{quantil}(1-\alpha) \cdot \sqrt{\sum_{h=1}^L W_h^2 \cdot \text{berechne_stdabw_schichtmittel}(n_h)^2}$$

3 Modul Stichprobenziehung

Programmabfolge:

1. Einlesen der Daten und Übergabe der Parameter aus Planungsmodul
2. Ziehen einer Zufallszahl für jedes Testfeld
3. Sortieren der Testfelder
4. Bestimmung der Anzahl der Schichten zweiter Ordnung je Schicht erster Ordnung und Bestimmung der Anzahl Testfelder erster und zweiter Ordnung
5. Zuweisung der Stichprobe
6. Ausgabe

3.1 Zufallszahlen

3.1.1 Zufallszahlengenerator

Die (Pseudo-) Zufallszahlen werden mit einem linearen Kongruenzgenerator erzeugt. Dieser funktioniert nach folgendem Prinzip:

$$u_t = \frac{(a \cdot u_{t-1} + c) \bmod m}{m}$$

Der Modulooperator \bmod liefert als Ergebnis den Rest bei der Division durch m . Die Zufallszahlen liegen damit in $[0;1)$. Der Startwert (seed) ist u_0 und wird durch den Nutzer eingegeben. Bei gleichen

Startwerten liefert der Zufallszahlengenerator die gleichen Zahlen. Die Konstanten a, c und m sind so gewählt, dass die resultierenden Zahlen Zufallscharakter haben:

$$\begin{aligned}a &= 1366 \\c &= 150889 \\m &= 714025\end{aligned}$$

Mit diesen Konstanten sollte der Zufallszahlengenerator nur bei einer Feldlänge unter 2^{31} einen Überlauf erzeugen.

Die Funktion $\text{zufall}(u_0)$ liefert nach obigen Prinzip Zufallszahlen.

Wenn u_i als globale Variable vereinbart wird, muss an $\text{zufall}()$ kein Wert übergeben werden. Die Zuweisung $u_0 = \text{seed}$ kann dann auch global geschehen.

3.1.2 Zufallszahlenziehung

Nach Einlesen der $\text{Testfeld}_i[]$ arrays wird an diese eine vierte Variable, deren Wert zufällig bestimmt wird, angehängt. Das heißt, dass die Testfeldarrays vierdimensional vereinbart werden.

Eingabe: $\text{Testfeld}_i[Nr, h, k,], i = 1, \dots, N$

Ausgabe: $\text{Testfeld}_i[Nr, h, k, u], i = 1, \dots, N$

Berechnung:

For $i = 1, \dots, N$

$$\text{Testfeld}_i[Nr, h, k,] = \text{Testfeld}_i[Nr, h, k, \text{zufall} ()]$$

3.2 Sortierung

$\text{Testfeld}[]$ muss vor Verarbeitung sortiert werden. Die Sortierung erfolgt dreistufig: zunächst nach Schicht erster Ordnung, dann nach Schicht zweiter Ordnung und zuletzt nach den Zufallszahlen. Die Eingabedatei hat nach Sortierung folgende Form (die Zahlenbeispiele sind willkürlich gewählt):

Nr	1. Ordnung	2. Ordnung	Uni(0,1)
123	1	1	0,15
122	1	1	0,32
140	1	2	0,21
...
53	2	4	0,69
45	2	4	0,73
...
92	h	J	0,48
98	h	J	0,49

3.3 Bestimmung der Schichtgröße zweiter Ordnung

Für die Aufteilung der Stichprobenumfänge auf die Schichtung zweiter Ordnung, müssen die Schichtgrößen zweiter Ordnung für jede Schicht erster Ordnung ermittelt werden.

Eingabe: L, N_h , für $h=1, \dots, L$, $Testfeld_i[Nr, h, k, u]$, $i=1, \dots, N$

Ausgabe:

M_h Anzahl Schichten zweiter Ordnung in der h -ten Schicht erster Ordnung

N_{hj} Anzahl Testfelder der h -ten Schicht erster Ordnung und der j -ten Schicht zweiter Ordnung

Berechnung:

Alle Variablen werden vor Aufruf des Algorithmus mit 0 initialisiert.

```

for  $h=1, \dots, L$ 
   $j=1$ 
  for  $i = summe + 1, \dots, summe + N_h$ 
    if  $k$  aus  $Testfeld_i[Nr, h, k, u] = j$ 
       $N_{hj} = N_{hj} + 1$ 
    else
       $j = k$  aus  $Testfeld_i[Nr, h, k, u]$ 
       $N_{hj} = N_{hj} + 1$ 
    end if
  end for
   $M_h = j$ 

```

```

        summe = summe + Nh
    end for

```

3.4 Stichprobenziehung

Durch die Sortierung nach der Zufallszahlen ist die Reihenfolge der Testfelder in den Schichten erster und zweiter Ordnung zufällig. Zur Bestimmung der Testfelder, die in die Stichprobe gelangen, müssen nur die jeweils ersten einer Schicht ausgewählt werden.

3.4.1 Bestimmung der Stichprobenumfänge

Vom Planungsmodul werden die Stichprobenumfänge für die Schichtung erster Ordnung geliefert. Diese werden größenproportional auf die Schichtung zweiter Ordnung verteilt.

Stichprobenumfänge

Eingabe: $n_h, N_h, M_h, N_{hj}, h = 1, \dots, L, j = 1, \dots, M_h$

Ausgabe: $n_{hj}, h = 1, \dots, L, j = 1, \dots, M_h$ /* Stichprobenumfang je Schicht erster und zweiter Ordnung */

Berechnung:

```

For h = 1, ..., L
    For j = 1, ..., Mh
         $W_{hj} = \frac{N_{hj}}{N_h}$  /* Gewichte der Schichten zweiter Ordnung */
    end for
    if Mh > 1
        For j = 1, ..., Mh - 1
             $n_{hj} = \text{round}(n_h W_{hj})$  /* Stichprobenumfang muss ganzzahlig sein */
        end for
         $n_{hM_h} = n_h - \sum_{j=1}^{M_h-1} n_{hj}$ 
    else if Mh = 1
         $n_{hM_h} = n_h$ 
    end if
end for

```

Durch die Rundung, um ganzzahlige Stichprobenumfänge zu erhalten, kann es passieren, dass n_{hM_h} negativ wird. Dies muss abgefragt werden und ggf. behoben werden. Falls n_{hM_h} negativ ist, wird willkürlich die erste Schicht mit positiven Stichprobenumfang um diesen Wert reduziert.

Test_Stichprobenumfänge

Eingabe: n_{hj} , $h = 1, \dots, L$, $j = 1, \dots, M_h$

Ausgabe: n_{hj} , $h = 1, \dots, L$, $j = 1, \dots, M_h$

Berechnung:

```

For  $h = 1, \dots, L$ 
  If  $n_{hM_h} < 0$ 
     $a = 0$ 
     $j = 1$ 
     $k = M_h$ 
    while  $a = 0$ 
       $n_{hj} = n_{hj} + n_{hk}$ 
      if  $n_{hj} \geq 0$ 
         $a = 1$ 
      else
         $k = j$ 
         $j = j + 1$ 
      end if
    end while
  end if
end for

```

3.4.2 Testfeldauswahl

In dieser Prozedur werden die Testfeldnummern, der Testfelder die in die Stichprobe gelangen, bestimmt.

Eingabe: n_{hj} , M_h , N_{hj} , $h = 1, \dots, L$, $j = 1, \dots, M_h$

Ausgabe: $Stichprobe_i, i = 1, \dots, n$

Berechnung:

```
i = 1
t = 1
for h = 1, ..., L
  for j = 1, ..., Mh
    for k = 1, ..., nhj
      Stichprobei = Nr aus Testfeldi
      i = i + 1
      t = t + 1
    end for
    t = Nhj + 1
  end for
end for
```

3.5 Ausgabe

Zunächst wird der Vektor *Stichprobe* sortiert. Die Ausgabe der ausgewählten Testfelder *Stichprobe_i*, *i* = 1, ..., *n* erfolgt in eine Datei. Der Name und der Ort der Ausgabedatei kann vorgegeben werden. Die Voreinstellung für den Namen ist OFD_ziehung.txt und für den Ort das Arbeitsverzeichnis der aufgerufenen Anwendung. Bei der Berechnung wird die Datei neu angelegt bzw. überschrieben. Eine Warnung bei Überschreibung ist nicht vorgesehen. Die Ausgabedatei enthält die sortierten Nummern der ausgewählten Testfelder.