1. Konferenz für SAS-Benutzer in Forschung und Entwicklung (KSFE)
Humboldt-Universität zu Berlin, 20.-21. Februar 1997

Simulationsuntersuchungen zum Genauigkeitsverlust der besten linearen erwartungstreuen Vorhersage (BLUP) bei unbekannten Varianzkomponenten in gemischten linearen Modellen mit SAS

Armin TUCHSCHERER, Paul Eberhard RUDOLPH und Günter HERRENDÖRFER

Forschungsinstitut für die Biologie landwirtschaftlicher Nutztiere Dummerstorf-Rostock
Wilhelm-Stahl-Allee 2, D-18196 Dummerstorf

Abstract

Die von Henderson (1974) entwickelte BLUP-Methode (Best Linear Unbiased Prediction) für die Vorhersage der zufälligen Effekte $\underline{u}$ in gemischten linearen Modellen der Gestalt $\underline{Y} = X\beta + Z\underline{u} + \underline{e}$ erfordert die Kenntnis der Varianzkomponenten (Kovarianzmatrix) von $\underline{u}$ und $\underline{e}$ damit die Eigenschaften 'beste', 'lineare' und 'erwartungstreue' erfüllt sind.

In der Praxis steht in der Regel nur die Stichprobeninformation zur Verfügung. Um das Verfahren BLUP anwenden zu können, schätzt man die unbekannten Varianzkomponenten mit Hilfe der Stichprobeninformation und setzt diese anstelle der bekannten in die Vorhersage ein. Das resultierende Verfahren ist nicht mehr BLUP sondern nur noch eine 'geschätzte BLUP'. Eigenschaften wie „beste" und „lineare" gehen verloren, und die Berechnung der Vorhersagegenauigkeit (MSE) wird problematisch. Wie üblich wird auch in der SAS-Prozedur Mixed die Vorhersagegenauigkeit durch eine Näherung angegeben, wo man die geschätzten Varianzkomponenten anstelle der bekannten einsetzt und den MSE mit der Formel für bekannte Varianzkomponenten berechnet.

Zur Beurteilung des Genauigkeitsverlustes der 'geschätzten BLUP' im Vergleich zur BLUP sowie der Brauchbarkeit der Genauigkeitsapproximation für die 'geschätzten BLUP' wird eine Simulationsstudie mit SAS und den in der Prozedur Mixed enthaltenen Varianzkomponentenschätzverfahren für ausgewählte Modelle durchgeführt.

Modell und Voraussetzungen

Wir betrachten das gemischte lineare Modell

$$\underline{y} = X\beta + Z\underline{u} + \underline{e}, \qquad (1)$$

wobei $\underline{y}$ ein N-dimensionaler Vektor der Beobachtungen, β ein b-dimensionaler Vektor der festen Effekte, $\underline{u}$ ein a-dimensionaler Vektor der zufälligen Effekte, $\underline{e}$ ein N-dimensionaler Vektor der zufälligen Resteffekte sowie X und Z bekannte Designmatrizen mit den Elementen 0 und 1 seien, mit den Voraussetzungen

$$E(\underline{u}) = 0,\ E(\underline{e}) = 0,\ Var(\underline{u}) = G,\ Var(\underline{e}) = R,\ Cov(\underline{u}, \underline{e}') = 0 \qquad (2)$$

(d.h. $Var(\underline{y}) = V_y = ZGZ'+R$); außerdem seien G und R bekannt.

Unter den Voraussetzungen (2) hat die beste lineare erwartungstreue Vorhersage (BLUP) für $\underline{u}$ die Gestalt

$$\underline{\hat{u}} = GZV_y^{-1}\left(\underline{y} - X\,\underline{\hat{\beta}}\right), \tag{3}$$

wobei

$$\underline{\hat{\beta}} = \left(X'V_y^{-1}X\right)^{-} X'V_y^{-1}\underline{y} \tag{4}$$

die beste lineare erwartungstreue Schätzung (BLUE) für β ist.

Das gleiche Ergebnis für (3) und (4) erhält man auch durch die Lösung des Mixed-model-Gleichungssystems

$$\begin{bmatrix} X'R^{-1}X & X'R^{-1}Z \\ Z'R^{-1}X & Z'R^{-1}Z + G^{-1} \end{bmatrix}\begin{bmatrix} \hat{\beta} \\ \underline{\hat{u}} \end{bmatrix} = \begin{bmatrix} X'R^{-1}\underline{y} \\ Z'R^{-1}\underline{y} \end{bmatrix}. \tag{5}$$

Die Genauigkeit der Vorhersage (3) wird als MSE (mittlerer quadratischer Fehler) durch

$$MSE(\underline{\hat{u}}) = E(\underline{\hat{u}} - \underline{u})'(\underline{\hat{u}} - \underline{u}) = Sp\left[V(\underline{\hat{u}} - \underline{u})\right] + E(\underline{\hat{u}} - \underline{u})'E(\underline{\hat{u}} - \underline{u}) = Sp\left[V(\underline{\hat{u}} - \underline{u})\right] \tag{6}$$

bestimmt, falls die Voraussetzungen (2) erfüllt werden.

Situation in der Praxis

In der Praxis steht in der Regel nur die Stichprobeninformation zur Verfügung. Um das Verfahren BLUP anwenden zu können, schätzt man die unbekannten G und R mit Hilfe der Stichprobeninformation und setzt die Schätzungen $\hat{G}$ und $\hat{R}$ an deren Stelle in die Vorhersage ein. Das resultierende Verfahren ist nicht mehr BLUP sondern nur noch eine 'geschätzte BLUP'. Eigenschaften wie beste und lineare gehen verloren und die Berechnung der Vorhersagegenauigkeit (MSE) auf analytischem Wege ist kaum noch möglich. Es bietet sich eine geplante Simulation an.

Zielstellung für die Untersuchung:

1. Ist SAS für eine derartige Simulation geeignet?
2. Wie verhält sich der MSE der 'geschätzten BLUP' zum MSE der BLUP (Größe des Genauigkeitsverlustes)?
3. Wie ist die praktische MSE-Bestimmung (auch durch SAS), bei der geschätzte Varianzkomponenten in die MSE- oder Varianz-Formel eingesetzt werden, zu bewerten?

Simulationsexperiment mit SAS

Eine Simulation zur Bestimmung der Genauigkeit der Vorhersage (3) mit geschätzten (bzw. bekannten) Varianzkomponenten mit SAS läuft etwa nach folgendem Schema ab.

Erzeugung von Daten mit einer vorgegebenen Modellstruktur: Vorgeben der Verteilung von u und e (Zufallszahlengenerator) Vorgeben von G und R Vorgeben des Versuchsplanes	Siehe z.B. A1, A3 Ergebnis: temp. SAS-Datei 'Zufall'

Verarbeiten der erzeugten Daten mit **proc mixed:**
Entscheidung: bekannte G und R Schätzung (REML, ML, MIVQUE0)
Ausgabedatei: Vorhersage Standardfehler Vorhers.
	'proc mixed' für das einfachste Modell
	geschätzte VK:
	`proc mixed data=zufall method=REML maxiter=50 ;`
	`class i;`
	`model y = ;`
	`random i / s ;`
	`make 'solutionR' out=vorher(keep=est se_pred);`
	`run;`
	bekannte VK:
	`proc mixed data=zufall sigiter ;`
	`class i;`
	`model y = ;`
	`random i / s ;`
	`parms (&sigmau) (&sigmae) / noiter;`
	`make 'solutionR' out=vorher(keep= est se_pred);`
	`run;`
	Ergebnis:
	temp. SAS-Datei 'Vorher'

Zusammenführen von u und û: Reduktion der Datei Zufall u Zusammenfügen (merge) von 'Zufall' und 'Vorher' Bildung: $u_i - \hat{u}_i$ für BIAS-Schätzung $(u_i - \hat{u}_i)^2$ für MSE-Schätzung Summation der Simulationsergebnisse	Ergebnis: temp. SAS-Datei 'Vorher' Ergebnis: temp. SAS-Datei 'Erg'

== Makro '`%vorher`'

Durchführung von N_s Simulationsläufen	`%macro simu(ns);`
	`%let K=1;`
	`%do %while(&K<=&ns);`
	`%VORHER;`
	`%let K=%eval(&k+1);`
	`%end;`
	`%mend simu;`
	
	Ergebnis:
	temp. SAS-Datei 'Erg

— Makro '`%simu(ns)`'

Nachdem das Makro '`%vorher`' N_s mal durch das Makro '`%simu(ns)`' aufgerufen wurde kommt der letzte Schritt:

Division der Summen über die N_s Simulationsläufe durch Anzahl der Simulationen N_s	Ergebnis: perm. SAS-Datei 'Ergmit' u.a. mit Anzahl der Simulationen N_s mittlere SAS-MSE-Berechnung simulierter MSE*

*geschätzter MSE über N_S Simulationen

Genauigkeit der Simulation = Genauigkeit der Schätzung des MSE

Die Genauigkeit der Simulation wird durch den Simulationsumfang (N_S) bestimmt. Mit einer Versuchsplanung über das *Konfidenzintervall für den MSE* bei vorgegebenem Konfidenzkoeffizienten von $1-\alpha = 0{,}95$ und einer halben erwarteten Breite d von 2 % der Standardabweichung des geschätzten MSE erhält man mit Hilfe der Verfahrensbibliothek oder CADEMO einen notwendigen Simulationsumfang von $N_S = 9607$.

Wahl des Simulationsumfanges: $N_S = 10\,000$.

Besonderheiten

Ausfallwerte im proc mixed Output:
Ausfallwerte in der proc mixed Output-Datei sind auszuschließen, d.h. Simulationen mit Ausfallwerten in der proc mixed Output-Datei werden übersprungen und nicht mitgezählt.
Überlauf OUTPUT-Fenster:
Um einen Überlauf des OUTPUT-Fensters zu verhindern, erfolgt eine Umleitung des nicht zu unterdrückenden Outputs im OUTPUT-Fenster in eine Datei (Bestandteil des Makros '`%vorher`').
Überlauf LOG-Fenster:
Um einen Überlauf des LOG-Fensters zu verhindern, erfolgt die Unterdrückung des Schreibens der `NOTES` und `SOURCES` in das LOG-Fenster.

Simulationsergebnisse

Im folgenden werden nur einige wenige ausgewählte Ergebnisse zur Veranschaulichung dargestellt.
Der Genauigkeitsverlust der 'geschätzten BLUP' im Vergleich zur BLUP für das einfachste Modell in Abhängigkeit von der Balanciertheit des Versuchsplanes wird in Abb.1 und in Abhängigkeit vom Verhältnis der Varianzkomponenten in Abb.2 dargestellt. Die Genauigkeit der Simulation ist für bekannte Varianzkomponenten in beiden Abbildungen als Genauigkeitsverlust der simulierten Genauigkeit (BLUP-SIM) im Vergleich zur berechneten Genauigkeit (BLUP) dargestellt.

Tabelle 1: Versuchspläne (a; n_i; N=100) für ausgewählte Simulationsuntersuchungen

Versuchs-plan	a	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9	n_{10}
1	10	1	1	1	1	1	1	1	1	1	91
2	10	2	2	2	2	2	2	2	2	2	82
3	10	3	3	3	3	3	3	3	3	3	73
4	10	4	4	4	4	4	4	4	4	4	64
5	10	5	5	5	5	5	5	5	5	5	55
6	10	6	6	6	6	6	6	6	6	6	46
7	10	7	7	7	7	7	7	7	7	7	37
8	10	8	8	8	8	8	8	8	8	8	28
9	10	9	9	9	9	9	9	9	9	9	19
10	10	10	10	10	10	10	10	10	10	10	10

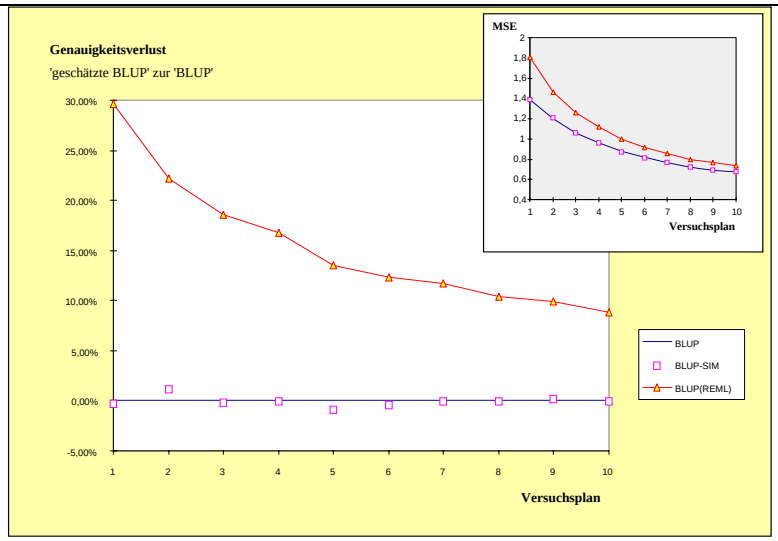

Abbildung 1: Genauigkeitsverlust (in Prozent) der ‘geschätzten BLUP’ (Varianzkomponenten mit REML geschätzt) im Vergleich zur BLUP für das Modell $\underline{y}_{ij} = \mu + \underline{u}_i + \underline{e}_{ij}$ mit $\sigma_u^2 = 0{,}175$ $\sigma_e^2 = 0{,}825$ in Abhängigkeit von der Balanciertheit des Versuchsplanes (Tabelle 1)

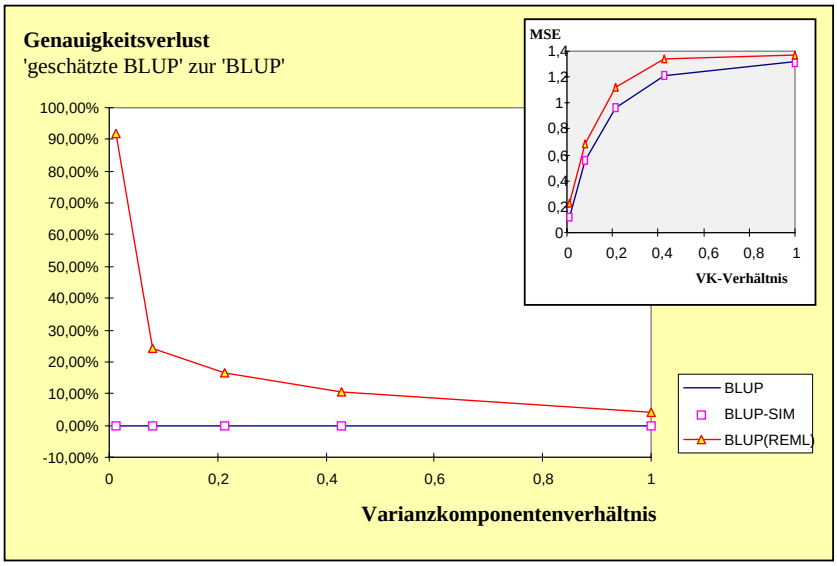

Abbildung 2: Genauigkeitsverlust (in Prozent) der ‘geschätzten BLUP’ (Varianzkomponenten mit REML geschätzt) im Vergleich zur BLUP für das Modell $\underline{y}_{ij} = \mu + \underline{u}_i + \underline{e}_{ij}$ für Versuchsplan 4 (vgl. Tabelle 1) in Abhängigkeit vom Verhältnis der Varianzkomponenten

Aus den Abbildungen 1 und 2 ist ersichtlich, daß beim praktikablen Verfahren ‘geschätzte BLUP’ in Abhängigkeit vom Versuchsplan und den Modellparametern zum Teil mit erheblichen Genauigkeitseinbußen gegenüber BLUP gerechnet werden muß.
In der Praxis geht man in der Regel davon aus, daß sich an der Genauigkeit der Vorhersage nicht viel ändert, wenn man von der BLUP zur ‘geschätzten BLUP’ übergeht. Warum dies so ist, läßt sich leicht aus der in der Praxis sehr verbreiteten Genauigkeitbestimmung, wo man die

geschätzten Varianzkomponenten in die MSE-Formel der BLUP einsetzt, erklären. Betrachtet man Tabelle 2 und Abbildung 3, so wird man leicht feststellen können, daß durch diese Genauigkeitsangabe die tatsächliche Genauigkeit beachtlich überschätzt wird, d.h. die ausgewiesene Genauigkeit in der Nähe der BLUP-Genauigkeit liegt.

Tabelle 2: Beispiel für die Vorhersage-Genauigkeitsüberschätzung durch SAS (geschätzte Varianzkomponenten, 10000 Simulationen)**:**

Modell 1: $\underline{y}_{ij} = \mu + \underline{u}_i + \underline{e}_{ij}$

Modell 2: $\underline{y}_{kij} = \mu + b_k + \underline{u}_i + \underline{e}_{kij}$

Modell 3: $\underline{y}_{kij} = \mu + b_k + \underline{w}_{ki} + \underline{u}_i + \underline{e}_{kij}$

Modell	MSE-SAS:	MSE-SIM:	Genauigkeits-überschätzung durch SAS	N_s
1	0,77710200	0,85583879	9,20 %	9956
1-Schiefe-$\underline{u}$	0,77815910	0,97782829	20,42 %	9937
2	0,87419517	0,97274968	10,13 %	9891
3	0,94648065	1,08069794	12,42 %	9829

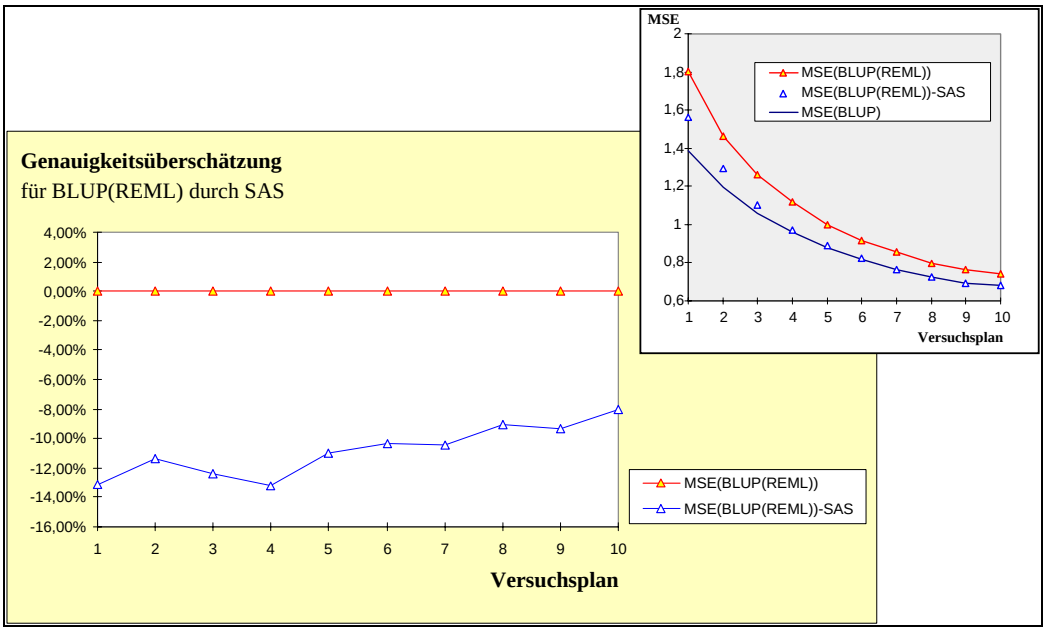

Abbildung 3: Genauigkeitsüberschätzung (in Prozent) der ‘geschätzten BLUP’ (Varianzkomponenten mit REML geschätzt) durch SAS im Vergleich zur tatsächlichen (simulierten Genauigkeit) für das Modell $\underline{y}_{ij} = \mu + \underline{u}_i + \underline{e}_{ij}$ mit $\sigma_u^2 = 0{,}175$ $\sigma_e^2 = 0{,}825$ in Abhängigkeit von der Balanciertheit des Versuchsplanes (Tabelle 1)

Zusammenfassung - Schlußfolgerungen

1. Nutzung von SAS für Simulationen:

   Vorteil:
   Das Durchführen von Simulationsuntersuchungen mit SAS ist ohne großen Programmieraufwand möglich.

   Aufwendiges Austesten der Programme entfällt, da geprüfte SAS-Prozeduren (wie z.B. proc mixed) zur Verfügung stehen.

   Nachteil:
   Durch die Komplexität der SAS-Prozeduren und die für aufwendige Simulationen nicht optimale Verarbeitung muß man einen entsprechenden Schnelligkeitsnachteil gegenüber herkömmlichen Fortran-Simulationsprogrammen in Kauf nehmen.

2. Genauigkeitsverlust der ‘geschätzten BLUP’ gegenüber der BLUP:

   Der Genauigkeitsverlust des praktikablen Verfahrens ‘geschätzte BLUP’ ist im Vergleich zur BLUP beachtlich (starke Abhängigkeit von Modell, Modellparametern und Versuchsplan).

   Vorsicht! Die von SAS ausgewiesene Vorhersagegenauigkeit gibt den tatsächlichen Genauigkeitsverlust nicht richtig wieder. Durch die Überschätzung der Genauigkeit der ‘geschätzten BLUP’ fällt der ausgewiesene Genauigkeitsverlust z.T. erheblich zu niedrig aus.

3. Vorhersagegenauigkeit von SAS für ‘geschätzte BLUP’:

   Vorsicht! Die von SAS (proc mixed) berechnete Genauigkeit für die ‘geschätzte BLUP’ überschätzt die tatsächliche Genauigkeit der Vorhersage z.T. erheblich.

   Mit steigendem Kompliziertheitsgrad der Modelle steigt auch die Überschätzung der Genauigkeit der ‘geschätzten BLUP’.

4. Weitere Nutzungsmöglichkeiten derartiger SAS Simulationsprogramme:

   - Beurteilung von Modellen (Modellwahl)
   - Beurteilung von Versuchsplänen (Versuchsplanung)
   - Beurteilung von Varianzkomponentenschätzern

   usw.

Literatur

Fleishman, A.L. (1978): A method for simulating non-normal distributions. Psychometrika 43, 521-532

Henderson, C.R. (1974): General flexibility of linear model techniques for sire evaluation. J. Dairy Sci. 57, 963-972

Herrendörfer, G.; Tuchscherer, A. (1996): Selection and breeding. Journal of Statisical Planning and Inference vol. 54, N° 3, 307 - 321

Tuchscherer, A.; Herrendörfer, G.; Tuchscherer, M. (1996): Investigations on Robustness of the Prediction in Mixed Linear Models. Abstract, XVIIIth International Biometric Conference, 01.-05. 07. 1996, Amsterdam, The Netherlands, p. 215.

Tuchscherer, A. (1988): Some remarks on the prediction of breeding values by BLUP. in: Rasch; D.; Pirchner, F.; Adam, J.(eds): Proceedings of the International Conference on Population Mathematics in Schwerin, Part II, Dummerstorf; AdL Berlin, FZ für Tierproduktion, S. 120 - 133 (Schriftenreihe: Probleme der angewandten Statistik, Heft 25)

Tuchscherer, A. 1987: Die beste lineare erwartungstreue Vorhersage und ihre Anwendung für die Zuchtwertvorhersage. Dummerstorf; AdL Berlin, FZ für Tierproduktion, (Schriftenreihe: Probleme der angewandten Staistik, Heft 23)

Anhang

A1 Beispiel für ein SAS-Simulationsprogramm (10000 Simulationswiederholungen):

Modell:

$$\underline{y}_{kij} = \mu + \beta_k + \underline{w}_{ki} + \underline{u}_i + \underline{e}_{kij}$$

Voraussetzungen:

$$\underline{u}_i \sim N\left(0, \sigma_u^2\right)$$

$$\underline{e}_{kij} \sim N\left(0, \sigma_e^2\right) \qquad k = 1,\ldots,b;\ i = 1,\ldots,a;\ j = 1,\ldots,n_{ki}$$

$$\underline{w}_{ki} \sim N\left(0, \sigma_w^2\right)$$

$$\sum_{k=1}^{b} b_k = 0$$

Parameter:

β : fest=(1.3, 0.5, 0.2, -0.8, -1.2)’

μ : mue=20

σ_u^2 : sigmau=0.3

σ_e^2 : sigmae=0.5

σ_w^2 : sigmaw=0.2

Versuchsplan:

b=5

a=10

n_{ki}		i									
		1	2	3	4	5	6	7	8	9	10
k	1	1	5	5	1	2	2	1	1	1	1
	2	2	5	5	1	2	2	1	1	1	1
	3	3	5	5	1	2	2	1	1	1	1
	4	4	5	5	1	2	2	1	1	1	1
	5	5	5	5	1	2	2	1	1	1	1

SAS-Programm:

```
%let b=5;
%let a=10;
```

```
%let sigmau=0.3;
%let sigmae=0.5;
%let sigmaw=0.2;
%let mue=20;
%MACRO VORHER;
/*-------------------------------------------------------------*
 |                   Modellerzeugung                           |
 *-------------------------------------------------------------*/
data zufall;
  array Anz{5,10}(1, 5, 5, 1, 2, 2, 1, 1, 1, 1
                  2, 5, 5, 1, 2, 2, 1, 1, 1, 1
                  3, 5, 5, 1, 2, 2, 1, 1, 1, 1
                  4, 5, 5, 1, 2, 2, 1, 1, 1, 1
                  5, 5, 5, 1, 2, 2, 1, 1, 1, 1);

  array fest{5}(1.3, 0.5, 0.2, -0.8, -1.2);

   t=time(); h=hour(t); m=minute(t); s=second(t); s=int(s);
    Start1=h*10000+m*100+s;
    Start2=s*10000+m*100+h;
    Start3=m*10000+s*100+h;
do i=1 to &a;
   format x 20.8;
   call rannor(start1, x);
   do k=1 to &b;
     format festeff w 20.8;
     festeff=fest{k};
     call rannor(start3, w);
     do j=1 to Anz{k,i};
      format eps x1 eps1 w1 y 20.8;
      call rannor(start2, eps);
      x1=sqrt(&sigmau)*x;
      w1=sqrt(&sigmaw)*w;
      eps1=sqrt(&sigmae)*eps;
      y=&mue+festeff+w1+x1+eps1;
      output;
     end;
  end;
end;
run;
/*-------------------------------------------------------------*
 |  Umleitung des OUTPUT in Datei ergebn im SAS-Pfad           |
 *-------------------------------------------------------------*/
proc printto file='ergebn' new;
run;
```

```
/*-------------------------------------------------------------*
 |   Vorhersage für die zufälligen Effekte mit der Prozedur    |
 |   'proc mixed' bei Verwendung der erzeugten Daten aus der   |
 |    Datei 'zufall'                                           |
```

```
 *-------------------------------------------------------------*/
proc mixed data=zufall method=REML CONVH=1E-8 maxiter=50 ;
  class i k;
  model y = k;
  random i i*k / s;
  make 'solutionR' out=vorher(keep=est se_pred);
run;

/*-------------------------------------------------------------*
 |     Im Falle bekannter Varianzkomponenten verwendet man:    |
 |     proc mixed data=zufall sigiter ;                        |
 |       class i k;                                            |
 |       model y = k  ;                                        |
 |       random i i*k / s ;                                    |
 |       parms (&sigmau) (&sigmaw) (&sigmae) / noiter;         |
 |       make 'solutionR' out=vorher(keep= est se_pred);       |
 |     run;                                                    |
 *-------------------------------------------------------------*/

/*-------------------------------------------------------------*
 |    Reduktion der Zufallszahlendateien 'Zufall'              |
 |    auf die benötigten Größen:                               |
 |     Stufen i,  zufällige Effekte x1                         |
 *-------------------------------------------------------------*/
data zufall;
  set zufall(keep=i j k x1);
  if j=1 and k=1; drop j k;
run;

data vorher;
  set vorher (obs=&a);
  format mse_p 20.8;
  mse_p=se_pred*se_pred;
  drop se_pred;
run;

data vorher;
  merge zufall vorher;
  format diff diffq 20.8;
  diff=x1-est; diffq=diff*diff;
run;
```

```
/*-------------------------------------------------------------*
 |   Summation der Simulationsergebnisse                       |
 *-------------------------------------------------------------*/
data erg;
  merge vorher erg;
```

```
   if mse_p^=. then do;
    x1e=x1e+x1;
    este=este+est;
    mse_pe=mse_pe+mse_p;
    diffe=diffe+diff;
    diffqe=diffqe+diffq;
    simnr=simnr+1;
   end;
  drop i x1 est mse_p diff diffq;
run;
%MEND VORHER;
/*-------------------------------------------------------------*
 |   Makro zum ns-fachen Aufruf des Makros %VORHER             |
 *-------------------------------------------------------------*/
%macro simu(ns);
 %let K=1;
   %do %while(&K<=&ns);
       %VORHER;
       %let K=%eval(&k+1);
  %end;
%mend simu;
/*-------------------------------------------------------------*
 |   Nullsetzen der Ergebnisdatei                               |
 *-------------------------------------------------------------*/
data erg;
    do i=1 to &a;
      simnr=0;
      format x1e este mse_pe diffe diffqe 20.8 ;
      x1e=0;
      este=0;
      mse_pe=0;
      diffe=0;
      diffqe=0;
      output;
   end; drop i;
run;

%simu(10000);

data sasuser.ergmit;
     set erg;
      x1e=x1e/simnr;
      este=este/simnr;
      mse_pe=mse_pe/simnr;
      diffe=diffe/simnr;
      diffqe=diffqe/simnr;
run;
```

A2 Zufallszahlenerzeugung in SAS:

NORMAL	generates a normally distributed pseudo-random variate
RANBIN	generates an observation from a binomial distribution
RANCAU	generates an observation from a Cauchy distribution

RANEXP	generates an observation from an exponential distribution
RANGAM	generates an observation from a gamma distribution
RANNOR	generates an observation from a normal distribution
RANPOI	generates an observation from a Poisson distribution
RANTBL	generates an observation from a tabled probability mass function
RANTRI	generates an observation from a triangular distribution
RANUNI	generates an observation from a uniform distribution
UNIFORM	generates a pseudo-random variate uniformly distributed on the interval (0,1)

A3 Verteilung von u mit Schiefe für das einfachste Modell bei balanciertem Versuchsplan

```
%let a=10;
%let sigmau=0.3;
%let sigmae=0.7;
%let mue=20;
/*-------------------------------------------------------------*
 | Fleishman-Koeffizienten zur Erzeugung von Schiefe u. Exzeß  |
 *-------------------------------------------------------------*/
%let ub=-1.1368565847;
%let uc=0.20211769874;      /*  Schiefe: 0.8; Exzeß:0.0  */
%let ud=0.063802837053;

%let eb=1;
%let ec=0;
%let ed=0;
/*-------------------------------------------------------------*
 |                  Modellerzeugung                            |
 *-------------------------------------------------------------*/
data zufall;
  array Anz{10}(10, 10, 10, 10, 10, 10, 10, 10, 10, 10);
  t=time(); h=hour(t); m=minute(t); s=second(t); s=int(s);
    Start1=h*10000+m*100+s;
    Start2=s*10000+m*100+h;
  do i=1 to &a;
     format x 20.8;
     call rannor(start1, x);
     do j=1 to Anz{i};
      format eps x1 eps1 y 20.8;
      call rannor(start2, eps);
      x1=sqrt(&sigmau)*(&ub*x+&uc*x*x+&ud*x*x*x);
      eps1=sqrt(&sigmae)*(&eb*eps+&ec*eps*eps+&ed*eps*eps*eps);
      y=&mue+x1+eps1;
      output;
     end;
  end;
run;
```