

## Vertikalisierung und Horizontalisierung von Datasets

### Alternativen zu PROC TRANSPOSE

Heinrich Stürzl  
Dade Behring, Behring Diagnostics GmbH,  
Marburg

2. KSFE, Jena, 26./27. Februar 1998

---

#### Dade Behring Inc.

- Weltweit führender Anbieter von Labor-Diagnostika Systemen
- Vertrieb in 22 Ländern
- Umsatz 1,5 Milliarden US\$
- Mitarbeiter 8.700

#### Produktgruppen

- Klinische Chemie
- Immunologie
- Mikrobiologie
- Gerinnung
- Plasmaproteinanalytik
- Arzneispiegelüberwachung
- Drogennachweis

#### Kunden

- Ärzte
- Krankenhäuser
- Laborpraxen
- Blutbanken

#### Firmengeschichte

- 1904 Firmengründung in Marburg durch Prof. Dr. Emil von Behring
- 1929 Übernahme durch I.G. Farbenindustrie
- 1952 Behringwerke AG als 100% Tochter der Hoechst AG
- 1995 Erwerb von Syva, CA, USA
- 1.1.97 Behring Diagnostics GmbH
- 2.10.97 Fusion mit Dade International  
**Dade Behring Inc., IL, USA**

---

## Vertikalisierung und Horizontalisierung von Datasets

#### Anordnung der Daten

- „nebeneinander“ in einer Beobachtung
- „untereinander“ in einer Variablen

#### Problem

Je nach Fragestellung und Auswertung wird die eine oder andere Struktur benötigt

- PROC MEANS, UNIVARIATE, TTEST, ...
- ☑ **eine Variable**  
(vertikale Struktur)
- PROC FREQ, TABULATE, CORR, REG, ..., Vergleich im Data Step
- ☑ **eine Beobachtung**  
(horizontale Struktur)

#### Fragestellung

- Wie können die Daten von einer Struktur in die andere gebracht werden?
- Wie können Variablen und Beobachtungen in gewünschter Weise vertauscht werden?
- Wie kann ein Dataset ganz oder teilweise transponiert werden?

#### Lösungsansätze

- PROC TRANSPOSE
- Datenschnitt (DATA STEP)

### Fall 1: Vertikalisierung

- Daten, die „nebeneinander“ in verschiedenen Variablen stehen, sollen in verschiedene Beobachtungen gebracht werden.
- ☑ Zusätzliche Schlüsselvariablen zur Unterscheidung der Beobachtungen

#### Beispiel

| ID | Puls1 | Puls2 | Puls3 | Temp1 | Temp2 | Temp3 |
|----|-------|-------|-------|-------|-------|-------|
| A  | 72    | 85    | 110   | 36,3  | 36,7  | 37,2  |
| B  | 80    | 120   |       | 38,1  | 38,6  |       |
| C  | 65    | 70    | 85    | 36,3  | 36,6  | 37,0  |
| D  | 85    | 96    | 120   | 37,2  | 37,8  | 37,6  |

#### Ziel

- Pro Messung eine Beobachtung (alle Messungen „untereinander“)
- Neue Schlüsselvariable(n) für
  - Parameter: Puls, Temperatur
  - Zeitpunkt: 1, 2, 3

#### Vertikalisierung

##### Lösung PROC TRANSPOSE

```
proc sort data=werte;
  by id;
run;
proc transpose data=werte out=vert;
  var puls1-puls3 temp1-temp3;
  by id;
run;
```

#### Vertikalisierung

##### Data Step Lösung 1

```
data vert;
  set werte;
  param="Puls"; zeit=1; mess=puls1; output;
  param="Puls"; zeit=2; mess=puls2; output;
  param="Puls"; zeit=3; mess=puls3; output;
  param="Temp"; zeit=1; mess=temp1; output;
  param="Temp"; zeit=2; mess=temp2; output;
  param="Temp"; zeit=3; mess=temp3; output;
  drop puls1-puls3 temp1-temp3;
run;
```

#### Vertikalisierung

##### Data Step Lösung 2

```
data vert; set werte;
  array p [*] puls1-puls3;
  array t [*] temp1-temp3;

  do zeit=1 to 3;
    param="Puls"; mess=p[zeit]; output;
    param="Temp"; mess=t[zeit]; output;
  end;
  drop puls: temp:;
run;
```

### Fall 2: Horizontalisierung

- Daten, die „untereinander“ in verschiedenen Beobachtungen stehen, sollen in verschiedene Variablen gebracht werden.
- ☑ Schlüsselvariablen auflösen

#### Beispiel

| ID | Parameter | Zeit | Messwert |
|----|-----------|------|----------|
| A  | Puls      | 1    | 72.0     |
| A  | Puls      | 2    | 85.0     |
| A  | Puls      | 3    | 110.0    |
| A  | Temp      | 1    | 36.3     |
| A  | Temp      | 2    | 36.7     |
| A  | Temp      | 3    | 37.2     |
| B  | Puls      | 1    | 80.0     |
| B  | Puls      | 2    | 120.0    |
| B  | Puls      | 3    | .        |
| B  | Temp      | 1    | 38.1     |
| B  | Temp      | 2    | 38.6     |
| B  | Temp      | 3    | .        |
| .. | ..        | ..   | ..       |

## Horizontalisierung Ziel 1 (H1)

- Pro Messzeitpunkt eine Beobachtung (alle Messwerte eines Zeitpunktes „nebeneinander“)
- Schlüsselvariable „Parameter“ auflösen

### H1 Lösung PROC TRANSPOSE

```
proc sort data=vert;
  by id zeit param; * !!;
run;

proc transpose data=vert out=hor1 prefix=m;
  var mess;
  by id zeit;
run;
```

### H1 Data Step Lösung 1 (Retain+Output)

```
proc sort data=vert;
  by id zeit;
run;
data hor1;
  retain puls temp;
  set vert; by id zeit;
  if param="Puls" then puls=mess;
  if param="Temp" then temp=mess;
*..;
  if last.zeit then output;
  drop param mess;
run;
```

### H1 Data Step Lösung 2 (Do-Until Schleife)

```
proc sort data=vert;
  by id zeit;
run;

data hor1;
  do until (last.zeit);
    set vert; by id zeit;
    if param="Puls" then puls=mess;
    if param="Temp" then temp=mess;
  end;
*..;
  drop param mess;
run;
```

### H1 Data Step Lösung 3 (Do-Until Schleife, Array)

```
proc sort data=vert;
  by id zeit;
run;
proc format; value $paramf
  "Puls" = 1 "Temp" = 2;
run;

data hor1;
  array x [*] puls temp;
  do until (last.zeit);
    set vert; by id zeit;
    x[put(param, $paramf.)]=mess;
  end;
*..;
  drop param mess;
run;
```

## Horizontalisierung Ziel 2 (H2)

- Eine Beobachtung für jeden Probanden (alle Messwerte eines Probanden „nebeneinander“)
- Schlüsselvariablen „Parameter“ und „Zeit“ auflösen

### H2 Lösung PROC TRANSPOSE

```
proc sort data=vert;
  by id param zeit; * !!;
run;

proc transpose data=vert out=hor2;
  var mess;
  by id;
run;
```

### H2 Data Step Lösung

```
proc sort data=vert;
  by id;
run;

proc format; value $paramf
  "Puls" = 1 "Temp" = 2;
run;

data hor2;
  array x [2,3] puls1-puls3 temp1-temp3;
  do until (last.id);
    set vert; by id;
    x[put(param, $paramf.), zeit]=mess;
  end;
*..;
  drop param zeit mess;
run;
```

## Zusammenfassung

|                    | PROC TRANSPOSE | DATA STEP           |
|--------------------|----------------|---------------------|
|                    | Standardlösung | Individuelle Lösung |
| Programmieraufwand | gering         | hoch                |
| Flexibilität       | gering         | hoch                |

## Weiterführende Literatur

- SAS Language and Procedures, Usage2, Chap. 9, „Reshaping Data“
- Observations, Shelton, Wolfson, „Rediscovering the TRANSPOSE Procedure“, 2Q93
- SAS-Anwenderhandbuch im Netz <http://www.rzuser.uni-heidelberg.de/~x16/sas-Ah.html>