g-CALIB Release 2.0

SPSS based software for calibration

by **Camille VANDERHOEFT**

METHODOLOGIST AT STATISTICS BELGIUM

camille.vanderhoeft@statbel.mineco.fgov.be
http://statbel.fgov.be/studies/cal_en.asp

Workshop: Calibration tools for Survey Statisticians

Swiss Statistical Society & Université de Neuchâtel

Neuchâtel, September 8-9, 2005

1. *A standard calibration problem in g-Calib 2.0*

1.1. *Step by step creation of a job*

- (Re)set the working directory, if necessary
  @WORKDIR : .sav's, .gc2, .svo, .spo
- Specify at least one survey data input file (.sav)
- Name the numeric output file (.svo)
- Define the special variables
- Define the constraints, including totals
- Choose the calibration level (or type)
- (Re)set the calibration method (incl. limits on g-weights)
- Set the (initial / fixed) scale parameter

- Check *Run Calibration*
- Optional: name and save the job (.gc2)

1.2. *Loading a job*

- Can open a .gc2 from @WORKDIR
- .gc2 can be complete or incomplete
- .gc2 can be the start for a new .gc2

1.3. *Running a job*

- *Run* button or menu
- Before actually running the job, creates *Temp.spp* and *Temp_Design.sps*

1.3.1. *Temp.spp*

- First part of .gc2 (slightly modified)

- Initialises internal parameters (@WORKDIR, …)
- Lists relevant syntaxes (.sps)
- Automatically executed from the interface, by calling *spssprod.exe*
- Available afterwards

1.3.2. *Temp_Design.sps*

- Translated second part of .gc2
- Creates design matrix from survey data
- Automatically executed from SPSS Production, by calling *spsswin.exe*
- Available afterwards

2. *Input and output (files)*

2.1. *Input*

2.1.1. *Element level survey data*

- Internal parameter: @XDATA
- Any (survey) variable
- Special variables
  - Element identification (numeric)
  - Initial weight
  - Calibration stratum (1, 2, …)
  - Q-weight
  - Optional: cluster (identification; numeric)

2.1.2. *Cluster level survey data*

- Internal parameter: @ZDATA
- Any (survey) variable
- Special variables
  - Cluster identification (numeric)
  - Initial weight
  - Calibration stratum (1, 2, …)
  - Q-weight

2.1.3. *Totals*

- To be typed …
- Future: from file (register, …) ??

2.2. *Output*

2.2.1. *Numerical output*

- Internal parameter: @OUTWEI
- .svo, named by the user
- Variables
  - Indentification (element or cluster)
  - Initial weight (element or cluster)
  - Calibration stratum
  - Q-weight
  - Design matrix
  - CLSIZE__ (if calibration at cluster level)

  - SCALE : scale parameter (fixed / estimated)
  - SCAWEI : scaled initial weight
  - CALWEI : final calibrated weight
  - G_WEIG : final g-weight

2.2.2. *Informative output*

- *Temp.spo*
- See manual (rel. 1.0) section(s) 3.1.3.5 (and 3.1.3.6)
- Includes
  - Model specifications
  - Summary of survey data
  - History of entire job

- History of iteration
- Different totals
- Tables and graphs about distribution of SCAWEI, CALWEI, G-WEIG
- Output from user-defined syntaxes
  - Before calibration syntax(es)
  - After calibration syntax(es)

2.3. *Intermediate data files*

- Transformed survey data (collapsed, design matrix, merged, …)
- Fixed totals
- Useful for further development

3. *Defining constraints*

3.1. *From easy …*

$$\sum_{k=1}^{n} w_k x_k = t_x$$

- In g-Calib :

Active?	Name	Condition	Rescale	(Count) or quant. variable	Stratum1
x	Xvar		1	X	t_x

- In *Temp_Design.sps* :

```
COMPUTE Xvar = 1 * (X).
```

3.2. *… or easy and common …*

$$\sum_{k=1}^{n} w_k = N$$

- In g-Calib :

	Active?	Name	Condition	Rescale	(Count) or quant. variable	Stratum1
	x	Popul		0.5	(count)	*N*
Or :	x	Popul		0.5	1	*N*

- In *Temp_Design.sps* :

```
COMPUTE Popul = 0.5 * (1).
```

3.3. *… to more complex …*

$$\sum_{k|y_k=expr} w_k x_k = t_x$$

- In g-Calib :

	Active?	Name	Condition	Rescale	(Count) or quant. variable	Stratum1
	x	Xvar1	Y=*expr*	1	X	t_x
Or :	x	Xvar1		1	X * (Y= *expr*)	t_x

- In *Temp_Design.sps* :

```
COMPUTE Xvar1 = 0.
IF (Y=expr) Xvar1 = 1 * (X).
```

```
Or: COMPUTE Xvar1 = 1 * (X * (Y=expr)).
```

3.4. *… or with stratification*

$$\sum_{k|k\in h} w_{hk} x_{hk} = t_{xh} \quad (h = \text{calibration stratum})$$

- In g-Calib :

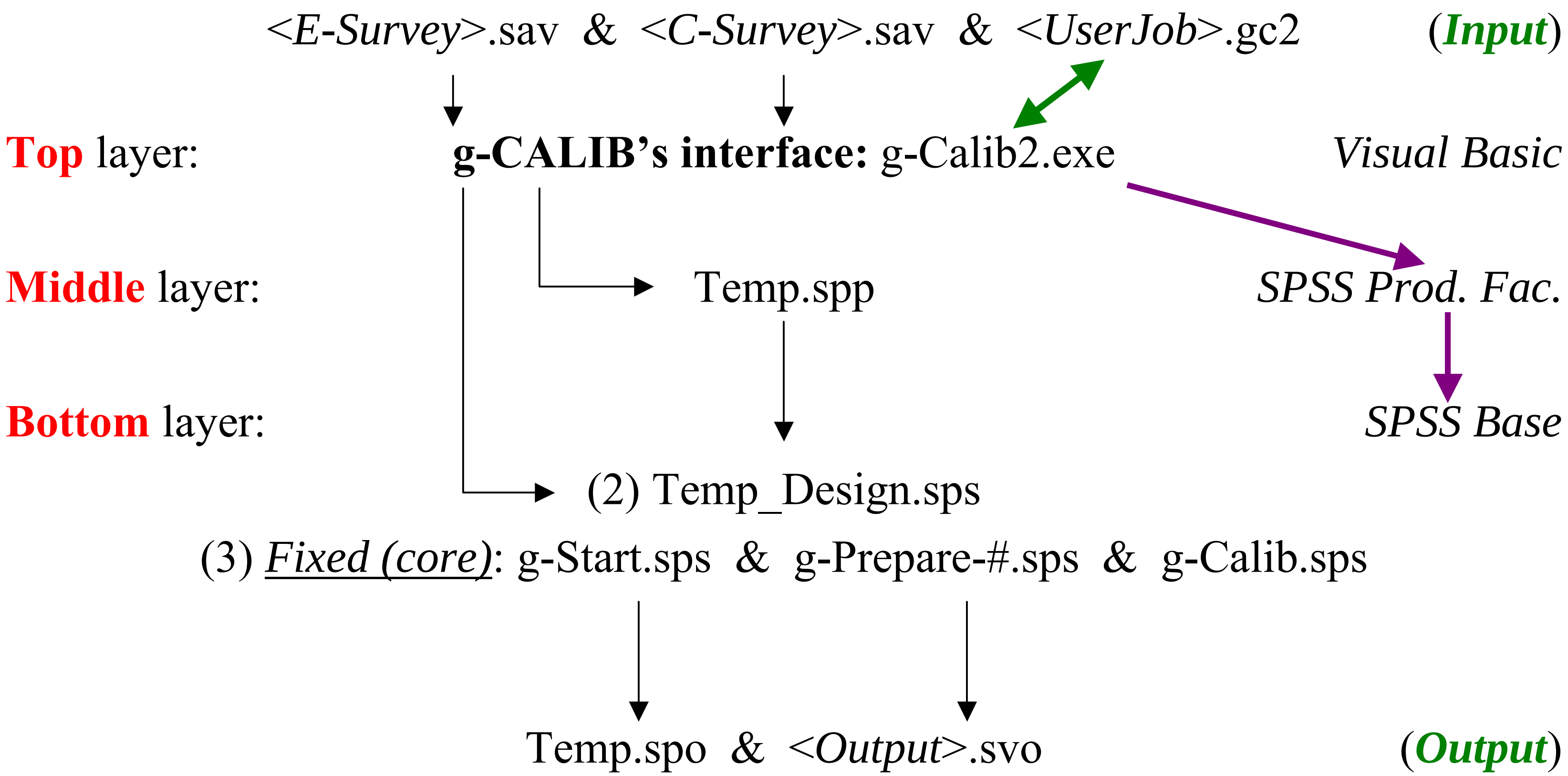
Active?	Name	Condition	Rescale	(Count) or quant. variable	Stratum1	Stratum2	…
x	Xvar2		1	X	t_{x1}	t_{x1}	…

- In *Temp_Design.sps* :

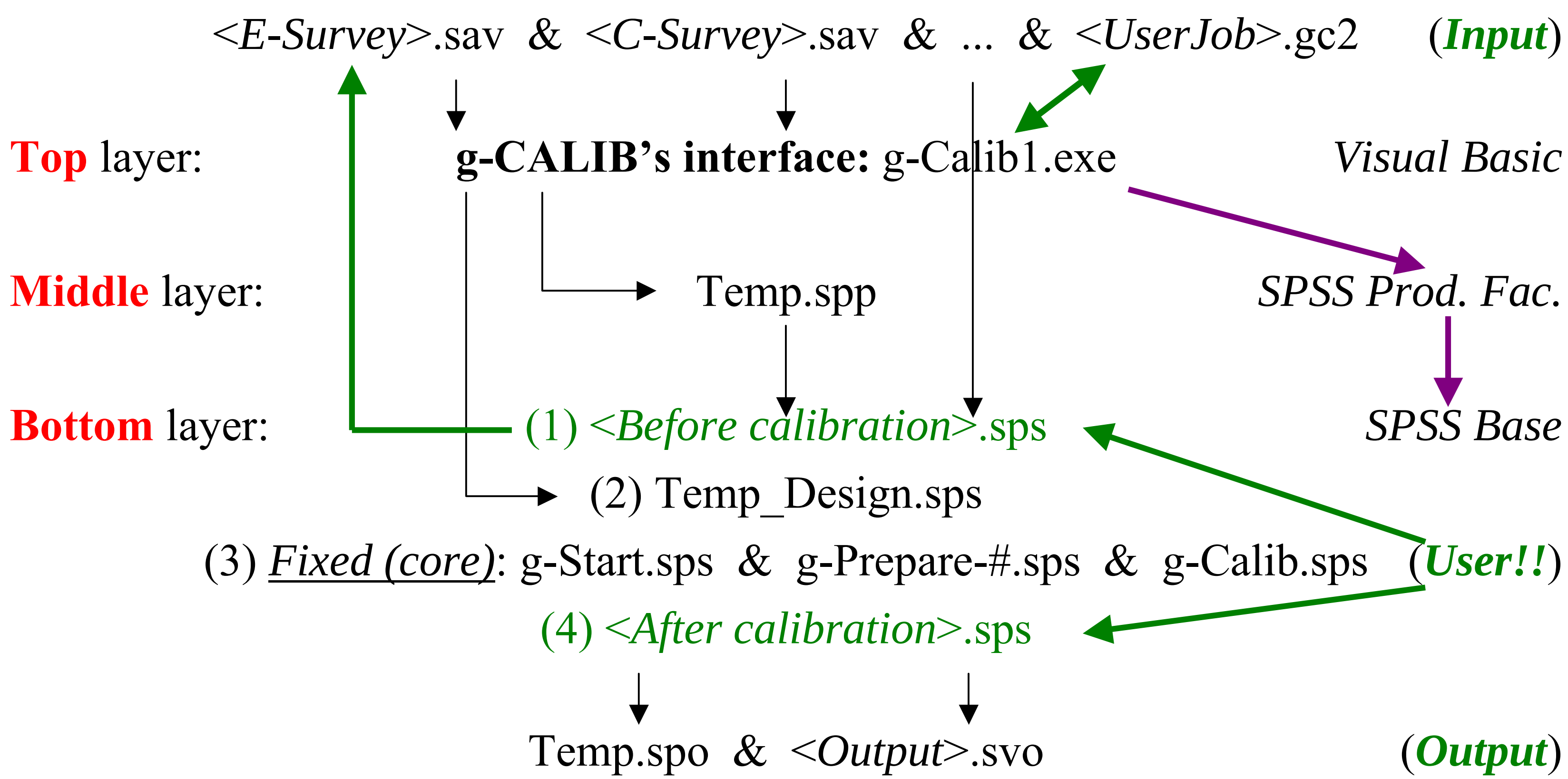
```
COMPUTE Xvar2 = 1 * (X).
```

4. *Implementation: 3 layers*

4.1. *Standard calibration job*



4.2. *Extended 'calibration' job*



5. *Special features*

5.1. *Calibration strata*

Design matrix **X**

- "linear structure", e.g. PROV * AGECAT * SEX (*post-stratification*)
  PROV + AGECAT + SEX (*raking*)
  PROV + AGECAT + SEX*SALARY
- factorisation, e.g. PROV * (AGECAT + SEX) (*raking* in PROVs)
  PROV * AGECAT * (SEX + SALARY + A*B + …)
- calibration stratification variable: e.g.
  PROV = 1, 2, …
  PROV * AGECAT *recoded into* PROV_AGE = 1, 2, …
- **X** is block-diagonal: $diag(\mathbf{X}_h)$ where h = calibration stratum
- **g-CALIB 2.0** is processing h separately $\Rightarrow$ faster

5.2. *Collapsing*

- Property of $\min\left\{\mathbf{d}^T\mathbf{Q}^{-1}G(\mathbf{g});\mathbf{X}^T\mathbf{D}\mathbf{g}=\mathbf{t};\mathbf{g}\in\Omega_B\right\}$

  If $\left(q_k,\mathbf{x}_k^T\right)=\left(q_{k'},\mathbf{x}_{k'}^T\right)$ (units k,k') $\Rightarrow$ $g_k=F\left(q_k\mathbf{x}_k^T\boldsymbol{\lambda}\right)=F\left(q_{k'}\mathbf{x}_{k'}^T\boldsymbol{\lambda}\right)=g_{k'}$

- Define: *Collapsing cell* = collection of units with same *qx*-vector

- Introduce / compute
  - $c(k)$ = collapsing cell to which *k* belongs
  - $\left(q_k,\mathbf{x}_k^T\right)=\left(\tilde{q}_c,\tilde{\mathbf{x}}_c^T\right)$ if $k\in c$ $\Rightarrow$ $\left(\tilde{\mathbf{q}}\middle|\tilde{\mathbf{X}}\right)$ = matrix of *different* *qx*-vectors
  - $g_k=\tilde{g}_c$ if $k\in c$
  - $d_c^+=\sum_{k\in c}d_k$ = total initial weight for collapsing cell *c*

$\Rightarrow$ Reduced problem: $\min\left\{\mathbf{d}^{+T}\mathbf{Q}^{-1}G(\tilde{\mathbf{g}});\tilde{\mathbf{X}}^T\mathbf{D}^+\tilde{\mathbf{g}}=\mathbf{t};\tilde{\mathbf{g}}\in\tilde{\Omega}_B\right\}$ but equivalent !

$\Rightarrow$ g-Calib 2.0 generally runs faster

5.3. *Scale parameter* ϕ

	Weights	Totals (j = 1 … m)
Fixed:	-	t_j
Initial (sampling):	$\mathbf{d}$	$\hat{t}_j^{INI} = \mathbf{d}^T \mathbf{X}_j$
Scaled:	$\mathbf{d}_\phi = \phi \cdot \mathbf{d}$	$\hat{t}_j^{SCA} = \mathbf{d}_\phi^T \mathbf{X}_j = \phi \hat{t}_j^{INI}$
Calibrated:	$\mathbf{w} = \phi \cdot \mathbf{Dg} = \mathbf{D}_\phi \mathbf{g}$	$\hat{t}_j^{CAL} = \mathbf{w}^T \mathbf{X}_j = \mathbf{g}^T \mathbf{D}_\phi \mathbf{X}_j = t_j$

g-CALIB 2.0 :

- *either* fix ϕ (one value for all strata)
- *or* estimate ϕ (separately for each stratum) as follows:

  g-Calib chooses the "first" constraint, j=1, and computes $\hat{\phi} = t_1 / \hat{t}_1^{INI}$

Remarks: ↳ "first" constraint (j=1) should involve all elements or clusters
↳ g-weights are relative to scaled weights

5.4. *Rescaling constraints*

(Iterative) algorithm must compute $\left(\mathbf{X}^T\mathbf{V}(\boldsymbol{\lambda})\mathbf{X}\right)^-$

$\rightarrow$ $\mathbf{X}^T\mathbf{V}(\boldsymbol{\lambda})\mathbf{X}$ should not be ill-conditioned

$\rightarrow$ Columns of $\left(\mathbf{X}^T|\mathbf{t}\right)^T$, i.e. $\begin{pmatrix}\mathbf{X}_j\\ t_j\end{pmatrix}$, should have ± same order of magnitude

$\rightarrow$ **g-CALIB 2.0** : can set *rescale factors* ρ_j $\rightarrow$ $\sum_{k\in s} g_k d_k\left(\rho_j x_{kj}\right)=\rho_j t_j$

$\rightarrow$ $\mathbf{R}=diag\left(\rho_j\right)$ transforms **X** resp. **t** into **XR** resp. **Rt**

$\rightarrow$ $\mathbf{R}^T\mathbf{X}^T\mathbf{V}(\boldsymbol{\lambda})\mathbf{XR}$ should allow better performance of g-CALIB 2.0

5.5. *Internal parameters*

5.5.1. *System parameters*

- Fixed
- `@WORKDIR, @LIBDIR, @XDATA, @OUTWEI, @STRAT, @IDENT, …`
- Available → user's syntaxes
- Example: `Include File = @LIBDIR + "Estimates.sps".`
- Example: in a user syntax, you can write

```
Get File = @WORKDIR + @XDATA.
Sort Data By @STRAT @IDENT.
Match Files File = *
           /Tables = @WORKDIR + @OUTWEI
           /By @STRAT @IDENT.
```

- Developing *before* and *after calibration* syntaxes …

5.5.2. *User parameters*

- Problem dependent
- Defined through the interface
- A set for before calibration syntax
- A set for after calibration syntax
- g-Calib overwrites user parameters by system parameters …

5.6. *General remarks*

- Understanding special features
  → helps interpreting output
  → helps understanding failures

- Knowing syntax language (incl. macro's, matrix language)
  → helps integrating g-Galib in the survey process

6. *Extended calibration jobs: examples*

6.1. *Many strata*

- Limitations in g-Calib :
  - Max. 62 active calibration constraints/variables (short names!)
  - Max. 94 calibration strata
  - Max. 5828 calibration totals, but …
- Stratification variables ARE calibration variables
  → to be explored
- Example: traffic accidents
  - 196 police DISTRICTs
  - Models of interest:
    (1) **DISTRICT**
    (2) DISTRICT + DISTRICT * DEATH = **DISTRICT** * (1 + DEATH)
    (3) **DISTRICT** * DEATH

- Solution: grouping districts

  DISTRICT = **NEW_STR** * GRP_DIST
  where **NEW_STR** = groups of districts (e.g. **49**) = **strata**
  GRP_DIST = numbering of districts within groups (e.g. 4 in each)

  Notice: 196 = **49** × 4 → **49** strata

  Alternatives: 196 = **94** × 3 – 86 → **94** strata (max!)
  196 = **20** × 10 – 4 → **20** strata; 10 rows; 4 cells ‘unused’

- Equivalent models:
  (1’) **NEW_STR** * GRP_DIST
  (2’) **NEW_STR** * GRP_DIST + NEW_STR * GRP_DIST * DEATH
  = **NEW_STR** * GRP_DIST * (1 + DEATH)
  (3’) **NEW_STR** * GRP_DIST * DEATH

- Demo / User Forum
- Still not possible to apply all models!

6.2. *Adjusting cross-tabulations*

6.2.1. *Adjusting to fixed margins*

From Bishop, Fienberg & Hollander (1976) *Discrete Multivariate Analysis*, p. 98

	Observed cross-tabulation (1957): Marital status (3 cl)				Adjusted cross-tabulation (1958): Marital status (3 cl)			
	single	married	widowed or divorced	Group Total	single	married	widowed or divorced	Group Total
	Nbr (x1000) of women 1957 Sum	Nbr (x1000) of women 1957 Sum	Nbr (x1000) of women 1957 Sum	Sum	CALWEI Sum	CALWEI Sum	CALWEI Sum	Sum t_i^{row}
Age cl - 5 classes (8 Yr) 15-19	**1306.00**	**83.00**	**.**	1389.00	1325.27	86.73	.	**1412.00**
20-24	**619.00**	**765.00**	**3.00**	1387.00	615.56	783.39	3.05	**1402.00**
25-29	**263.00**	**1194.00**	**9.00**	1466.00	253.94	1187.18	8.88	**1450.00**
30-34	**173.00**	**1372.00**	**28.00**	1573.00	165.13	1348.55	27.32	**1541.00**
35-39	**171.00**	**1393.00**	**51.00**	1615.00	173.41	1454.71	52.87	**1681.00**
40-44	**159.00**	**1372.00**	**81.00**	1612.00	147.21	1308.12	76.67	**1532.00**
45-49	**208.00**	**1350.00**	**108.00**	1666.00	202.33	1352.28	107.40	**1662.00**
50+	**1116.00**	**4100.00**	**2329.00**	7545.00	1105.16	4181.04	2357.81	**7644.00**
Group Total	4015.00	11629.00	2609.00	18253.00	t_j^{col} **3988.00**	**11702.00**	**2634.00**	**18324.00**

BFH: Section 3.6 "Classical uses of iterative proportional fitting"
Example 3.6-1 "Marriage status: updated margins for an old two-way table"

Calibration problem: observed freq. = $d_{ij} \rightarrow w_{ij} = d_{ij} \cdot g_{ij}$ = adjusted freq.

$$\min \sum_{i,j} d_{ij} \left(g_{ij} \log g_{ij} - g_{ij} + 1 \right) \qquad \text{or} \qquad \min \sum_{k} d_{k} \left(g_{k} \log g_{k} - g_{k} + 1 \right)$$

$$s.t. \quad \sum_j w_{ij} = t_i^{row} \quad \forall i \qquad \qquad s.t. \quad \sum_k w_k x_{ik}^{row} = t_i^{row} \quad \forall i$$

$$\sum_i w_{ij} = t_j^{col} \quad \forall j \qquad \qquad \sum_k w_k x_{jk}^{col} = t_j^{col} \quad \forall j$$

where: $x_{ik}^{row} = 1$ if cell $k \in$ row i, $= 0$ otherwise and $x_{jk}^{col} = 1$ if cell $k \in$ col j, $= 0$ otherwise

$\rightarrow$ [appropriate *before calibration syntax* (*BCS*): from 'table' to 'column format']
$\rightarrow$ appropriate *after calibration syntax* (*ACS*): show results in table format
$\rightarrow$ see old job (.gc1)

6.2.2. *Contrast variables*

From BFH (1976) *Discrete Multivariate Analysis*, p. 99-100

Observed cross-tabulation

		Congressmen's votes (3cl) yes Number of votes Sum	abstained Number of votes Sum	no Number of votes Sum	Group Total Sum
Section represented (3 cl)	north	**61.00**	**12.00**	**60.00**	133.00
	border	**17.00**	**6.00**	**1.00**	24.00
	south	**39.00**	**22.00**	**7.00**	68.00
Group Total		117.00	40.00	68.00	225.00

Adjusted cross-tabulation

		Congressmen's votes (3cl) yes Sum	abstained Sum	no Sum	Group Total Sum
Section represented (3 cl)	north	20.15	10.16	69.69	**100.00**
	border	47.36	42.85	9.80	100.00
	south	32.50	46.99	20.51	100.00
Group Total		100.00	100.00	100.00	300.00

"*These raw data are difficult to interpret because the margins contain unequal numbers*."

"*… the North-Yea and the South-Nay both show negative interactions of about the same magnitude, and on further inspection we observe a symmetry in the table about the diagonal running from South-Yea to Noth-Nay which was not apparent previously*."

BFH: Section 3.6 "Classical uses of iterative proportional fitting"
Example 3.6-2 "Congressmen's votes: detecting a diagonal trend by adjusting margins of a single two-way table"

Calibration problem: $d_{ij} \rightarrow w_{ij} = d_{ij} \cdot g_{ij}$

$$\begin{aligned} &\min \sum_{i,j} d_{ij} \left(g_{ij} \log g_{ij} - g_{ij} + 1 \right) \\ &s.t. \quad \sum_j w_{ij} = 100 \quad \forall i \\ &\qquad \sum_i w_{ij} = 100 \quad \forall j \end{aligned}$$

or

$$\begin{aligned} &\min \sum_{i,j} d_{ij} \left(g_{ij} \log g_{ij} - g_{ij} + 1 \right) \\ &s.t. \quad \sum_j w_{1j} = 100 \qquad (optional!) \\ &\qquad \sum_j w_{ij} = \sum_j w_{1j} \quad \forall i > 1 \\ &\qquad \sum_j w_{ij} = \sum_j w_{ji} \quad \forall i \end{aligned}$$

g-Calib job: Congressmen's votes.gc2 $\rightarrow$ working with "*contrast variables*"

6.2.3. *Updating a SAM*

From Robinson, Cataneo & El-Said (2001) Updating and Estimating a Social Accounting Matrix (SAM) Using Cross-Entropy Methods, In: *Economic Systems Research, Journal of the International Input-Output Association*. Vol. 13, No. 1, pp. 47-64.

$\bar{\mathbf{T}} = \left(\bar{t}_{ij}\right)$: (square) matrix of input-output flows (or "transactions")

column totals: $y_j = \sum_i \bar{t}_{ij}$

$\bar{\mathbf{A}} = \left(\bar{a}_{ij}\right)$: (square) matrix of coefficients (conditional "probability" distributions)

transformation: $\bar{a}_{ij} = \bar{t}_{ij} / y_j$ hence: $\sum_i \bar{a}_{ij} = 1$

Fixed marginal flows: y_j^* for *j*-th row & column

Calibration problems: (a) $\bar{\mathbf{T}} = \left(\bar{t}_{ij}\right) \rightarrow \mathbf{T} = \left(t_{ij}\right)$ s.t. (fixed totals, contrasts)

(b) $\bar{\mathbf{A}} = \left(\bar{a}_{ij}\right) \rightarrow \mathbf{A} = \left(a_{ij}\right)$ s.t. … $\rightarrow \mathbf{T} = \left(t_{ij} = a_{ij} y_j^*\right)$

Distance (for (b))

→ Information theory: Kullback-Leibler measure of cross-entropy distance between probability distributions

$$\max\left(-\sum_{i,j} a_{ij} \log \frac{a_{ij}}{\bar{a}_{ij}}\right)$$

→ exponential method in generalised calibration (given constraints):

$$\min \sum_{i,j} a_{ij} \log \frac{a_{ij}}{\bar{a}_{ij}} = \sum_{i,j} \bar{a}_{ij} \left(\frac{a_{ij}}{\bar{a}_{ij}} \log \frac{a_{ij}}{\bar{a}_{ij}} - \frac{a_{ij}}{\bar{a}_{ij}} + 1\right) = \sum_{i,j} d_{ij} \left(g_{ij} \log g_{ij} - g_{ij} + 1\right)$$

Constraints (for (b))

$$\begin{cases} \sum_j a_{ij} y_j^* = y_i^* & \forall i \\ \sum_i a_{ij} = 1 & \forall j \\ \left(0 \le a_{ij} \le 1 \right. & \left. \forall i, j \right) \end{cases} \Leftrightarrow \begin{cases} \sum_{i',j} a_{i'j} x_{i'j}^{(i)} = y_i^* \quad \forall i \quad \text{where} \quad x_{i'j}^{(i)} = y_j^* \quad \text{if} \quad i' = i \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad = 0 \quad \text{if} \quad i' \ne i \\ \sum_{i,j'} a_{ij'} x_{ij'}^{(j)} = 1 \quad \forall j \quad \text{where} \quad x_{ij'}^{(j)} = 1 \quad \text{if} \quad j' = j \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad = 0 \quad \text{if} \quad j' \ne j \end{cases}$$

Not straightforward:

1. Fixed totals also appear in calibration variables, and vary across cells
2. Data usually come in table format: Mozambik.sav

→ General syntax for transformation of data: Moz.sps
→ Extended job: Mozambik.gc2

Remark:

$$
\begin{aligned}
a_{ij} &= \bar{a}_{ij} \exp\left(\sum_{i'} \lambda_{i'}^{(1)} x_{ij}^{(i')} + \sum_{j'} \lambda_{j'}^{(2)} x_{ij}^{(j')} \right) \\
&= \bar{a}_{ij} \exp\left(\lambda_i^{(1)} y_j^* + \lambda_j^{(2)} \right) \\
&= \frac{\bar{a}_{ij} \exp\left(\lambda_i^{(1)} y_j^* \right)}{\sum_{i'} \bar{a}_{i'j} \exp\left(\lambda_{i'}^{(1)} y_j^* \right)} = \frac{(\text{prior}) \times (\text{info from data})}{\text{normalisation factor}}
\end{aligned}
$$

→ Specialised literature indicates how to transform / interpret general expressions from generalised calibration methodology

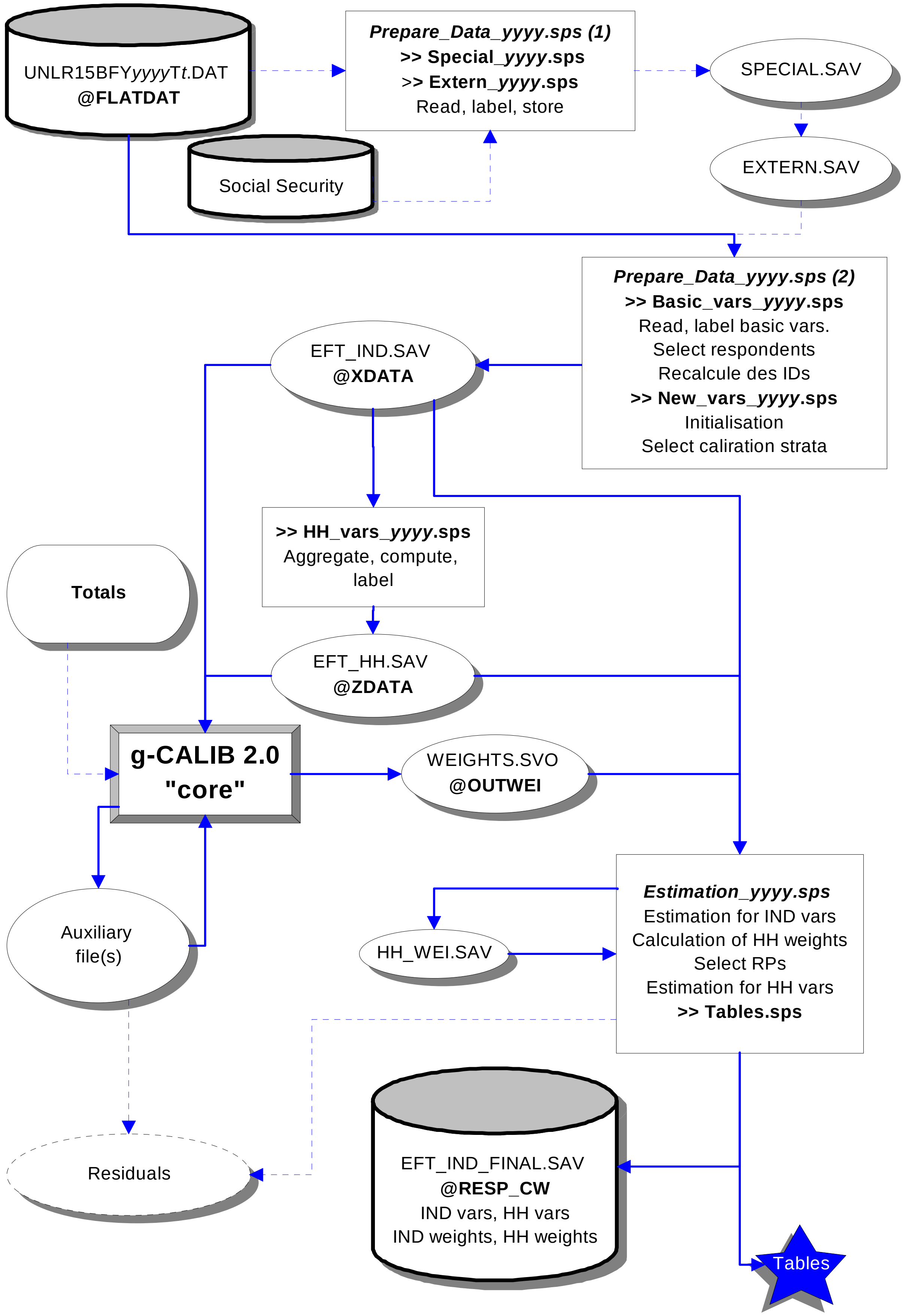
6.3. *Belgian LFS*

6.3.1. *Extended job*

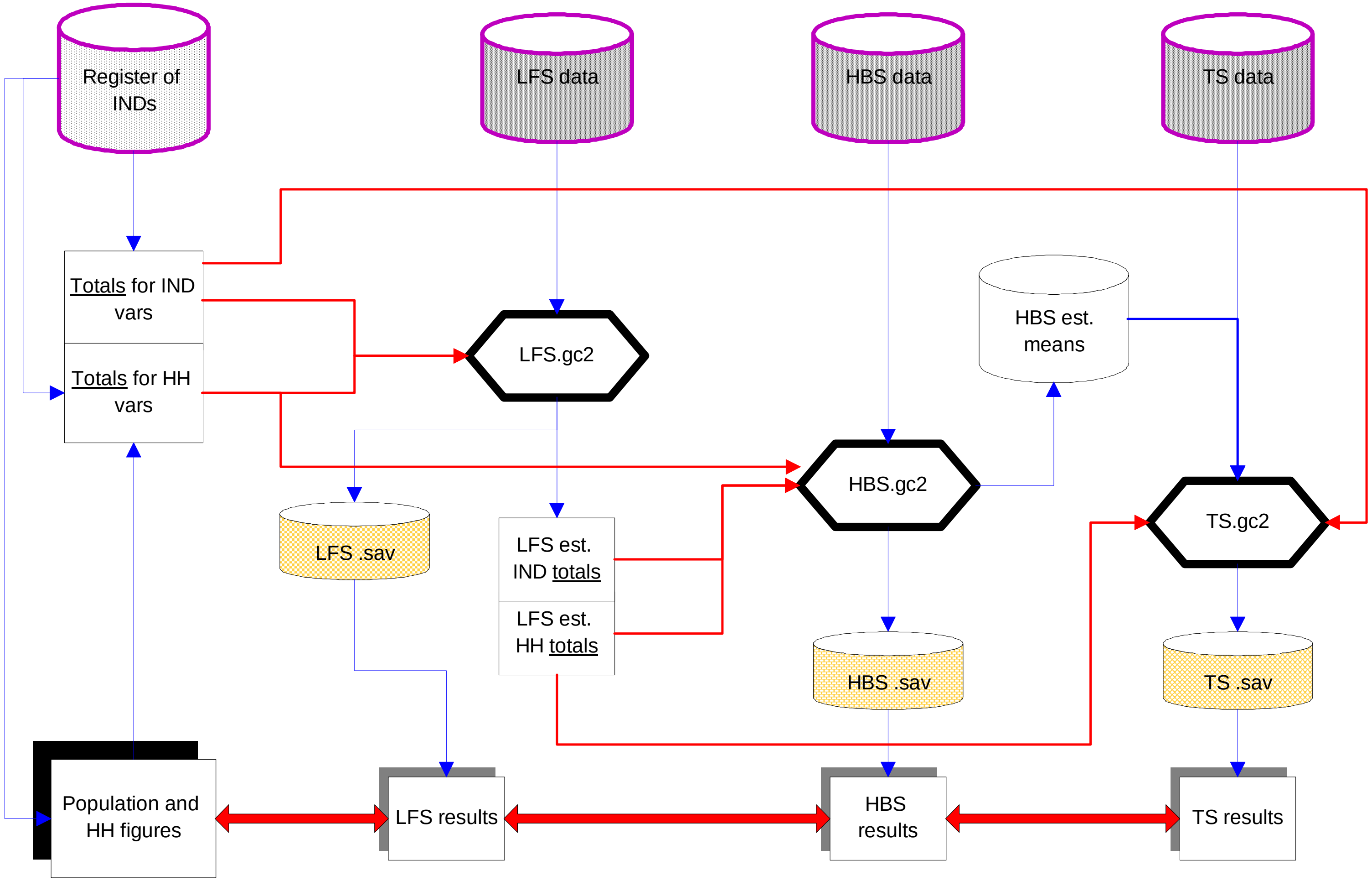
- *BFC* to transform data (from different sources) into SPSS format
- *BFC* to impute
- *BFC* to create household (cluster) data from individuals (element) data

- *AFS* to compute estimates at individual and household level
- *AFS* to create tables
- *AFS* to produce final output files (in different formats)

$\Rightarrow$ *g-Calib can be efficiently integrated in the survey process*


UNLR15BFYyyyyTt.DAT
@FLATDAT
Prepare_Data_yyyy.sps (1)
>> Special_yyyy.sps
>> Extern_yyyy.sps
Read, label, store
SPECIAL.SAV
EXTERN.SAV
Social Security
Prepare_Data_yyyy.sps (2)
>> Basic_vars_yyyy.sps
Read, label basic vars.
Select respondents
Recalcule des IDs
>> New_vars_yyyy.sps
Initialisation
Select caliration strata
EFT_IND.SAV
@XDATA
>> HH_vars_yyyy.sps
Aggregate, compute,
label
Totals
EFT_HH.SAV
@ZDATA
g-CALIB 2.0
"core"
WEIGHTS.SVO
@OUTWEI
Auxiliary
file(s)
Estimation_yyyy.sps
Estimation for IND vars
Calculation of HH weights
Select RPs
Estimation for HH vars
>> Tables.sps
HH_WEI.SAV
Residuals
EFT_IND_FINAL.SAV
@RESP_CW
IND vars, HH vars
IND weights, HH weights
Tables

6.3.2. *Project: higher consistency*



6.4. *Further developments*

6.4.1. *Calculation of residuals*

- Have created 7 general syntaxes: g-Resid-1E.sps, … g-Resid-4C.sps
- Estimation of totals (per stratum) + residuals for list of element/cluster survey variables
- Developed without changing (yet) g-Calib's interface
- Explored intermediate data files
- New internal system parameters are ready
- *BFC* to call the right syntax
- Hence: analysis is done
- To be used entirely through current interface
- Remains to redesign/adapt the interface

6.4.2. *Generalising type 4 calibration*

- g-Calib 2.0
  - Uses initial cluster weights $\tilde{d}_l$ only
  - Justification: $d_k = \tilde{d}_l$ for all elements *k* in cluster *l*
- g-Calib 2.0.1
  - Uses d_k and $\tilde{d}_l$
  - $d_k = \tilde{d}_l \underset{\sim}{d}_k$ for all *k* in *l* (often) makes sense: $\pi_k = \tilde{\pi}_l \underset{\sim}{\pi}_k$, π^*-estimator
  - Thanks to development of residual calculation
  - Calibration equations: $\begin{pmatrix} \mathbf{s} \\ \mathbf{t} \end{pmatrix} = \begin{pmatrix} \mathbf{Z}' \\ \widehat{\mathbf{X}}'_{\underset{\sim}{d}} \end{pmatrix} \tilde{\mathbf{D}} \tilde{\mathbf{g}} = \begin{pmatrix} \mathbf{Z}' \\ \widehat{\mathbf{X}}'_{\underset{\sim}{d}} \end{pmatrix} \tilde{\mathbf{D}} F\left(\tilde{\mathbf{Q}}\left(\mathbf{Z} \middle| \widehat{\mathbf{X}}_{\underset{\sim}{d}}\right)\boldsymbol{\lambda}\right)$
  - Soon downloadable

6.4.3. *Variance estimation*

- Next step after residual calculation
- Development again through *BCS* and extended jobs
- Testing …
- … adapting interface

6.4.4. *Getting totals from databases*

- Individual data
- Aggregated data
- Similar to Temp_Design.sps: generated by the system
- Aggregate once for maximal model
- Then ready for sub-models, just further aggregation