THE UNIVERSITY OF MINNESOTA

Statistics 5401 November 14, 2005

Example of Maximum Likelihood (MLE) Factor Estimation

This handout contains MacAnova output illustrating MLE factor analysis estimation. Two methods are used, the iterative method described in the handout on factor analysis estimation as implemented in **stepml()** and direct minimization of the ML criterion as implemented in **facanal()**.

The data are the chicken bone data in file **cbbones.txt**.

```
Cmd> bonedata <- read("","bonedata") # read cbbones.txt
bonedata      276      6 format labels
) Bone measurements on n = 276 outbred female chickens, all in mm.
) Col. 1:  skull length
) Col. 2:  skull breadth
) Col. 3:  femur length (leg bone)
) Col. 4:  tibia length (leg bone)
) Col. 5:  humerus length (wing bone)
) Col. 6:  ulna length (wing bone)
Read from file "TP1:Stat5401:Data:cbbones.txt"
```

```
Cmd> r <- cor(bonedata) # compute correlation matrix
```

Iteration using stepml()

stepml() is used similarly to **stepuls()** and **stepgls()**:

```
Cmd> result <- stepml(s,psi,m)
```

or

```
Cmd> result <- stepml(s,psi,m,print:T)
```

where **s** is a p by p variance matrix or correlation matrix, **psi** is a vector whose value is the current values of ψ_1, ..., ψ_p, the diagonal elements of $\mathbf{\Psi}$, **m** is the number of factors. With **print:T**, the new value of **psi** and a goodness of fit quantity will be printed. Argument **psi** can also be a structure whose first component is a vector containing the diagonal elements of $\mathbf{\Psi}$.

result is a structure with three components, **psi**, **loadings**, and **crit**, where **psi** is the vector of updated uniquenesses ψ_j, **loadings** is the updated matrix of loadings, and **crit** is a goodness of fit criterion. **crit** actually measures the goodness of the fit provided by the argument **psi** with non-updated loadings.

Because **stepml()** accepts a structure as second argument, you can use **result** in place of **psi** in the next iteration. In fact, a generic step of the iteration is

```
Cmd> psihat <- stepml(s,paihat,m [,print:T])
```

Besides returning a structure as values, **stepml()** creates variables **PSI**, **LOADINGS** and **CRITERION** as "side effects". These are identical to components

psi, **loadings** and **crit** of the returned value.

The goodness of fit criterion **crit** is proportional to a log likelihood ratio test statistic for the null hypothesis that **S** has structure satisfying the factor analytic model. Specifically, **crit** is

$$M(\hat{\boldsymbol{\Psi}}, \hat{\mathbf{L}}) = \mathrm{tr}(\hat{\boldsymbol{\Sigma}}^{-1}\mathbf{S} - \mathbf{I}_p) - \ln(\hat{\boldsymbol{\Sigma}}^{-1}\mathbf{S}) = \textstyle\sum_{m+1 \le i \le p}(\gamma_i - 1 - \ln\gamma_i),$$

where γ_j are the eigenvalues of **S** relative to the input value for $\hat{\boldsymbol{\Psi}}$. This quantity is reduced on each iteration, and its minimum is achieved at the maximum likelihood estimate. The value of $\sum_{m+1 \le i \le p}(\gamma_i - 1 - \ln\gamma_i)$ is the same whether **S** is the sample covariance or correlation matrix.

For large f_e = degrees of freedom in **S**, when H_0: ($\boldsymbol{\Sigma}$ is of factor analytic form) is true, the statistic

$$(f_e - (2p+5)/6 - 2m/3) \textstyle\sum_{m+1 \le i \le p}(\gamma_i - 1 - \ln\gamma_i),$$

is approximately χ_f^2, $f = \{(p-m)^2 - p - m\}/2$. You can use this to test goodness of fit to the factor analytic model.

Note The iterative method implemented in **stepml()** is *not* one you should use in practice. It may converges slowly, if at all. Moreover, it may abort without converging if $\gamma_i \le 1$ on any step. It is presented because it is an algorithm that can be readily implemented in MacAnova. In actual practice, you should use an algorithm that directly minimizes $\sum_{m+1 \le i \le p}(\gamma_i - 1 - \ln\gamma_i)$ as a function of $\psi_1, \ldots, \psi_p$, as is implemented in **facanal()** (see below).

```
Cmd> psi0 <- 1/diag(solve(r)) # compute initial values

Cmd> psihat <- stepml(r,psi0,2,print:T)  # 1 step of iteration
WARNING: searching for unrecognized macro stepml near psihat <-
stepml(
psi:    Printed because of print:T
   SklLngth   SklBrdth   FemLngth   TibLngth   HumLngth   UlnLngth
   0.514564   0.567289   0.102563   0.095886   0.086000   0.088397
criterion:
(1)   0.369982 Printed because of print:T
```

```
Cmd> print(psihat)
psihat:
component: psi
   SklLngth   SklBrdth   FemLngth   TibLngth   HumLngth   UlnLngth
   0.514564   0.567289   0.102563   0.095886   0.086000   0.088397
component: loadings   Factor loadings after first step
                (1)        (2)
SklLngth   0.642221   0.270163
SklBrdth   0.584576   0.301633
FemLngth   0.936884  -0.140307
TibLngth   0.944203  -0.112229
HumLngth   0.950657   0.101252
UlnLngth   0.953759   0.044118
component: crit   Goodness of fit criterion
(1)   0.369982

Cmd> print(PSI,LOADINGS,CRITERION) # print side effect variables
PSI:
   SklLngth   SklBrdth   FemLngth   TibLngth   HumLngth   UlnLngth
   0.514564   0.567289   0.102563   0.095886   0.086000   0.088397
LOADINGS:
                (1)        (2)
SklLngth   0.642221   0.270163
SklBrdth   0.584576   0.301633
FemLngth   0.936884  -0.140307
TibLngth   0.944203  -0.112229
HumLngth   0.950657   0.101252
UlnLngth   0.953759   0.044118
CRITERION:
(1)   0.369982

Cmd> iter <- 1 # iteration count

Cmd>  n <- 5; for(i,1,n){   # do 5 iterations
psihat <- stepml(r,psihat,2,print:T);iter <- iter+1;;}
psi:    After iteration 2
  SklLngth  SklBrdth  FemLngth  TibLngth  HumLngth  UlnLngth
  0.553695  0.618761  0.090000  0.084878  0.075262  0.081947
criterion:
(1)  0.303163
psi:    After iteration 3
  SklLngth  SklBrdth  FemLngth  TibLngth  HumLngth  UlnLngth
  0.578795  0.647014  0.083512  0.077794  0.067382  0.076348
criterion:
(1)  0.245295
psi:    After iteration 4
  SklLngth  SklBrdth  FemLngth  TibLngth  HumLngth  UlnLngth
  0.585843  0.654026  0.080714  0.073757  0.062290  0.073408
criterion:
(1)  0.218974
psi:    After iteration 5
  SklLngth  SklBrdth  FemLngth  TibLngth  HumLngth  UlnLngth
  0.588131  0.655894  0.079839  0.071277  0.058830  0.072374
criterion:
(1)  0.209569
```

```
psi:    After iteration 6
  SklLngth  SklBrdth  FemLngth  TibLngth  HumLngth  UlnLngth
  0.589119  0.656360  0.079985  0.069521  0.056237  0.072399
criterion:
(1)  0.205567

Cmd> n <- 100;for(i,run(n)){   # do 100 more iterations
        psihat <- stepml(r,psihat,2);iter <- iter+1;;}

Cmd> print(iter,PSI,CRITERION)
iter:
(1) 106.000000     Number of iterations so far
PSI:               Psihat after iteration 106
   SklLngth   SklBrdth   FemLngth   TibLngth   HumLngth   UlnLngth
   0.595579   0.653425   0.112429   0.020604   0.014485   0.090814
CRITERION:     Criterion
(1)   0.156574

Cmd> fe <- nrows(bonedata)-1;p <- ncols(bonedata)#S d.f., dimension

Cmd> m <- 2 # Number of factors

Cmd> const <- fe - (2*p +5)/6 - 2*m/3; const
(1) 270.833333  Factor to scale criterion to be chisquared

Cmd> const*CRITERION #or const*psihat$crit; approx chisq if converged
(1)  42.405587

Cmd> for(i,run(n)){   # do 100 more iterations
        psihat <- stepml(r,psihat,2);iter <- iter+1;;}

Cmd> print(iter,PSI,CRITERION)
iter:
(1) 206.000000
PSI:               Psihat after iteration 206
   SklLngth   SklBrdth   FemLngth   TibLngth   HumLngth   UlnLngth
   0.596551   0.653335   0.115372   0.014634   0.009733   0.093082
CRITERION:
(1)   0.153672

Cmd> const*CRITERION
(1)  41.619498
```

Similarly to the GLS iteration, $\hat{\psi}_5$ appears to be headed toward 0, although very slowly. Let's try a lot more (2000) iterations:

```
Cmd> n <- 2000;for(i,run(n)){   # do 2000 more iterations
        psihat <- stepml(r,psihat,2);iter <- iter+1;;}

Cmd> print(iter,PSI,CRITERION)
iter:
(1) 2206.000000
PSI:               Psihat after iteration 2206
   SklLngth   SklBrdth   FemLngth   TibLngth   HumLngth   UlnLngth
   0.598439   0.653410   0.120086   0.005327   0.002169   0.096945
CRITERION:
(1)   0.151060

Cmd> const*CRITERION
(1)  40.912184
```

```
Cmd> LOADINGS # unrotated ML loadings
                (1)        (2)
SklLngth   0.632828   0.033019
SklBrdth   0.588679  -0.006814
FemLngth   0.915590   0.203984
TibLngth   0.932914   0.352626
HumLngth   0.989141  -0.139395
UlnLngth   0.950064   0.020818
```

After a total of 15,000 iterations, the criterion reached 0.150826.

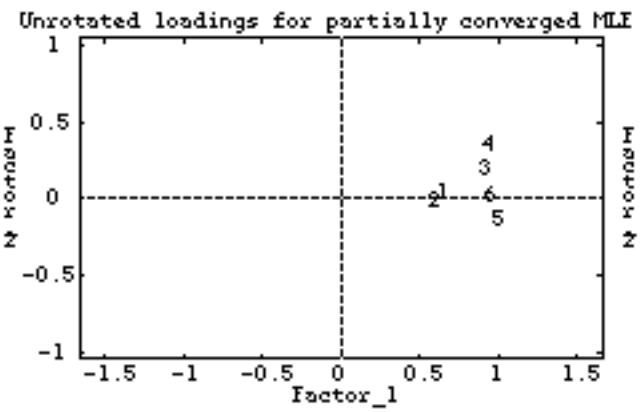
Compute more intuitive goodness of fit quantities, namely the sum of squares of the residuals between elements of **S** and $\hat{\mathbf{\Sigma}}$ and the absolute maximum residual.

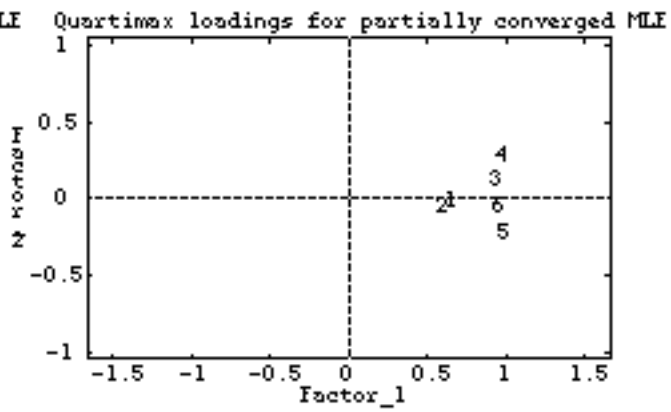
```
Cmd> # Compute estimated correlation matrix assuming factor model
Cmd> sigmahat <- LOADINGS %*% LOADINGS' + dmat(PSI)
Cmd> dev <- r - sigmahat # residuals of r from estimate
Cmd> vector(sum(vector(dev)^2),abs(max(vector(dev))))
(1)   0.092788   0.210702  SS and max(|resid|)
Cmd> sigmahat
          SklLngth  SklBrdth  FemLngth  TibLngth  HumLngth  UlnLngth
SklLngth  1.000000  0.372308  0.586146  0.602017  0.621353  0.601914
SklBrdth  0.372308  1.000000  0.537599  0.546784  0.583236  0.559141
FemLngth  0.586146  0.537599  1.000000  0.926096  0.877213  0.874115
TibLngth  0.602017  0.546784  0.926096  1.000000  0.873629  0.893669
HumLngth  0.621353  0.583236  0.877213  0.873629  1.000000  0.936845
UlnLngth  0.601914  0.559141  0.874115  0.893669  0.936845  1.000000
Cmd> dev # residuals from unrestricted correlation matrix
          SklLngth  SklBrdth  FemLngth  TibLngth  HumLngth  UlnLngth
SklLngth  0.000000  0.210702 -0.017035  0.000242 -0.000234  0.000419
SklBrdth  0.210702  0.000000 -0.022289  0.000815  0.000316 -0.034636
FemLngth -0.017035 -0.022289  0.000000  0.000008  0.000009  0.003338
TibLngth  0.000242  0.000815  0.000008  0.000000 -0.000001 -0.000058
HumLngth -0.000234  0.000316  0.000009 -0.000001  0.000000  0.000034
UlnLngth  0.000419 -0.034636  0.003338 -0.000058  0.000034  0.000000
```

Residuals are small except in row 2, column 1.

```
Cmd> # Rotate loadings using varimax rotation
Cmd> loadings_rot <- rotation(LOADINGS,reorder:F,kaiser:T) # varimax
Cmd> loadings_rot #all variables load high on both factors; not simple
                (1)        (2)
SklLngth   0.477676   0.416398
SklBrdth   0.467504   0.357812
FemLngth   0.594740   0.725395
TibLngth   0.516673   0.853066
HumLngth   0.864468   0.500526
UlnLngth   0.734873   0.602509
```

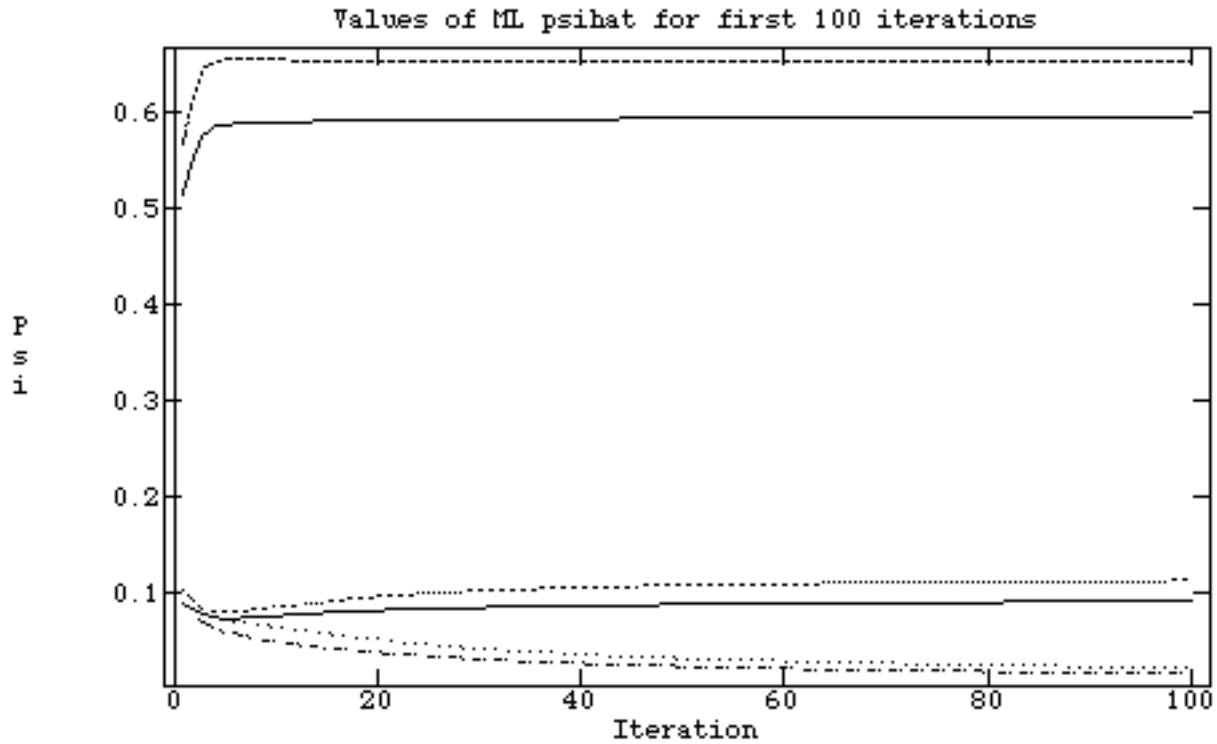
```
Cmd> loadings_rot <- rotation(LOADINGS,method:"quartimax",kaiser:T,\
       reorder:F) # quartimax rotation

Cmd> loadings_rot # quartimax rotated ML loadings
                 (1)        (2)
SklLngth   0.633477  -0.016363
SklBrdth   0.586361  -0.052636
FemLngth   0.928694   0.132064
TibLngth   0.957541   0.278904
HumLngth   0.975281  -0.216001
UlnLngth   0.948800  -0.053231

Cmd> # Scatter plots of unrotated & rotated loadings; values for xmin

Cmd> # selected to make horizontal and vertical scales about the same

Cmd> plot(Factor_1:psihat$loadings[,1],symbols:run(6),\
    Factor_2:psihat$loadings[,2],xmin:-1.6,xmax:1.6,ymin:-1,ymax:1,\
    title:"Unrotated loadings for partially converged MLE")

Cmd> plot(Factor_1:lrotate[,1],Factor_2:lrotate[,2],symbols:run(6),\
    xmin:-1.6,xmax:1.6,ymin:-1,ymax:1,\
    title:"Varimax loadings for partially converged MLE")
```



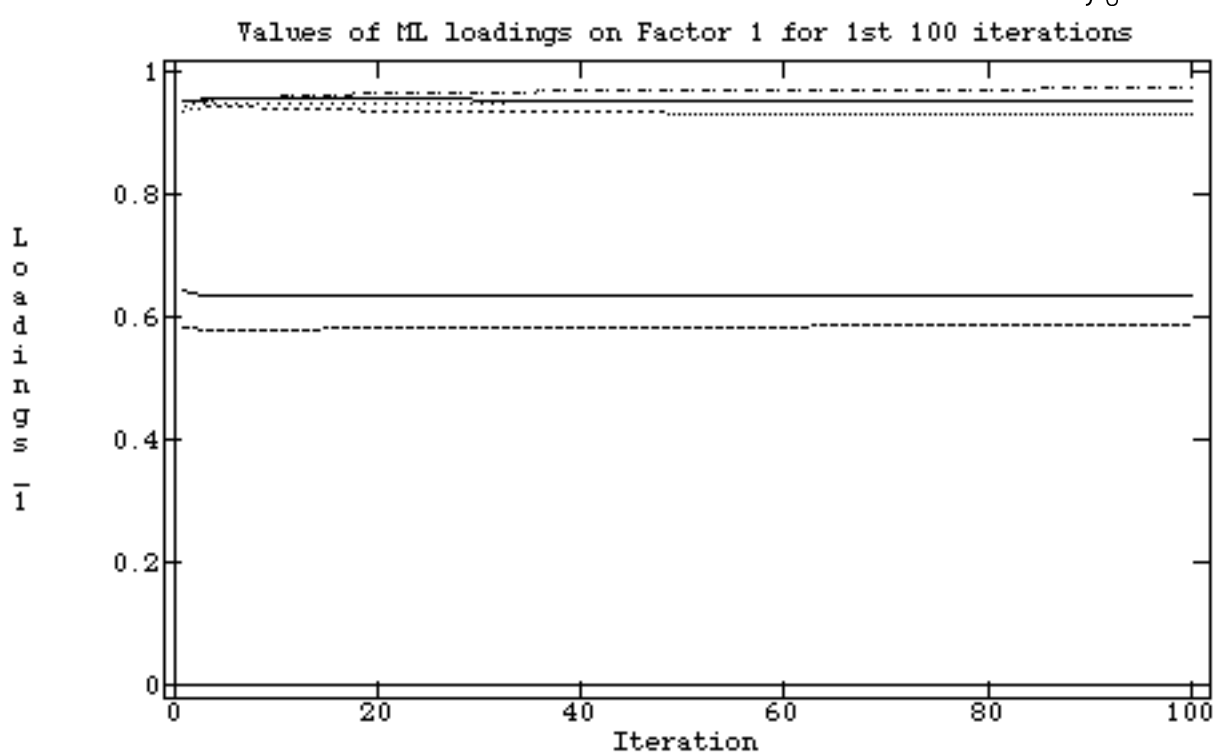


What happened during the iterations

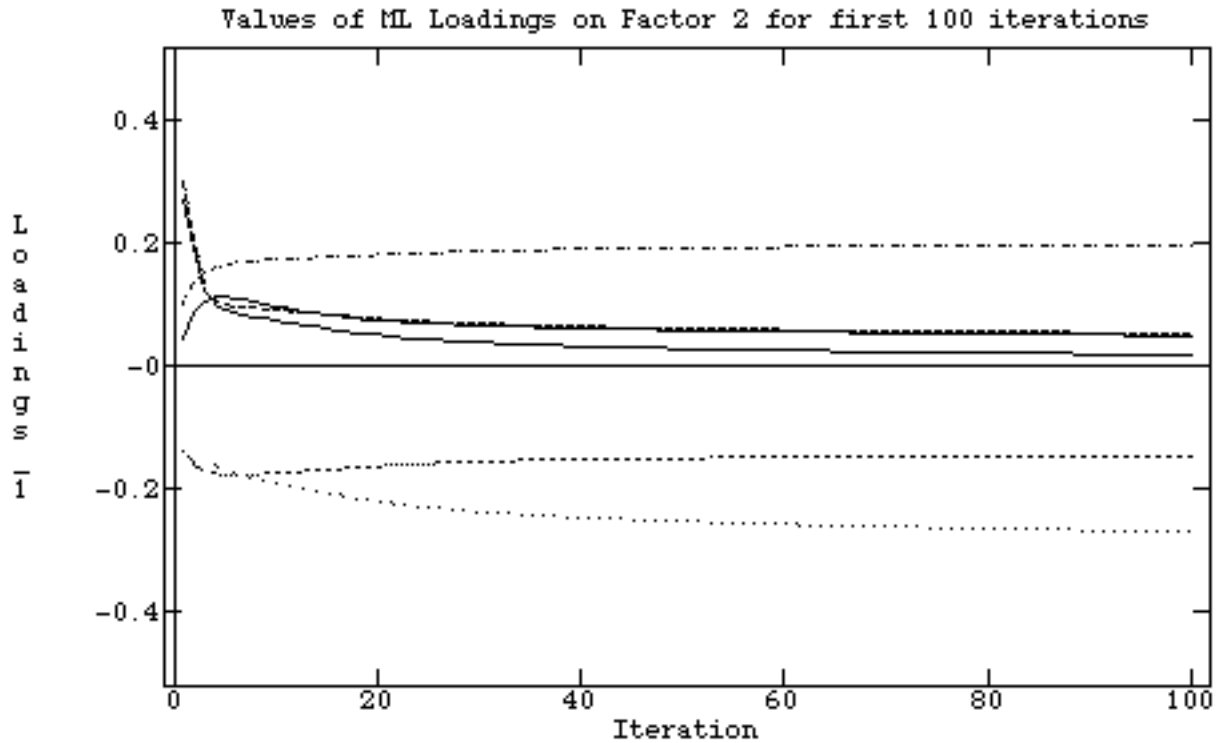
It is of some interest to follow the progress of this iterative scheme. Here are plots of the elements of $\hat{\Psi}$ and $\hat{L}$ versus iteration number.



Note that $\hat{\psi}_5$ and $\hat{\psi}_6$ are still decreasing. In fact, for these data it can be shown that the actual maximum of the likelihood occurs when $\hat{\psi}_6 = 0$.



The loadings on the first factor change very little after about 20 iterations.



The loadings on factor 2 still seem to be changing.

Direct minimization of MLE criterion

A better way to do the computations is direct minimizaiton of the MLE criterion. In MacAnova, **facanal()** does this, and can do rotation at the same time.

```
Cmd> results <- facanal(r,2,method:"ml",rotate:"quartimax")
WARNING: searching for unrecognized macro facanal near results <-
facanal(
Convergence in 20 iterations by criterion 2
estimated uniquenesses:
   SklLngth   SklBrdth   FemLngth   TibLngth   HumLngth   UlnLngth
   0.599025   0.653489   0.121379   0.002855   0.000160   0.098034
quartimax rotated estimated loadings:
           Factor 1   Factor 2
SklLngth   0.633033  -0.015713
SklBrdth   0.586343  -0.052132
FemLngth   0.928207   0.130584
TibLngth   0.958198   0.281072
HumLngth   0.975789  -0.218347
UlnLngth   0.948231  -0.053142
minimized ml criterion:
(1)   0.150793
```

After 20 **facanal()** iterations, the criterion is slightly smaller than after 15,000 **stepml()** iterations (0.150826).

```
Cmd> compnames(results) # components
(1) "psihat"
(2) "loadings"
(3) "criterion"
(4) "eigenvals"
(5) "gradient"
(6) "method"
(7) "rotation"
(8) "iter"
(9) "status"

Cmd> results # complete results
component: psihat
   SklLngth   SklBrdth   FemLngth   TibLngth   HumLngth   UlnLngth
   0.599025   0.653489   0.121379   0.002855   0.000160   0.098034
component: loadings
           Factor 1   Factor 2
SklLngth   0.633033  -0.015713
SklBrdth   0.586343  -0.052132
FemLngth   0.928207   0.130584
TibLngth   0.958198   0.281072
HumLngth   0.975789  -0.218347
UlnLngth   0.948231  -0.053142
component: criterion
(1)   0.150793
component: eigenvals
(1) 6555.030997  80.491996   1.390521   1.008627   0.966219   0.634653
component: gradient
(1)   0.000004   0.000006  -0.000012  -0.000027   0.000016  -0.000006
component: method
(1) "ml"
component: rotation
(1) "quartimax"
component: iter
(1)  20.000000
component: status
(1)   2.000000
```

The uniquenesses and rotated loadings are almost the same as after 2206 `stempml()` iterations.