Parameter Table

February 6, 2013

Tim Bergsma

1 Purpose

This script picks up after model.Rnw to process bootstrap results and make a parameter table. It assumes the current working directory is the script directory containing this file.

1.1 Package

Listing 1:

```
> library(metrumrg)
```

2 inputs

'wikitab' gives us a quick synthesis of 'rlog' and the 'lookup' of wiki notation in 1005.ctl. We do some science on the result first, and then some aesthetics for printing in a LaTeXtable. Table 1.

Listing 2:

```
> tab <- wikitab(1005,'../nonmem')
> tab$estimate <- signif(as.numeric(tab$estimate),3)
> tab$tool <- NULL
> tab$run <- NULL
> tab$se <- NULL
> tab
```

```
   parameter                                  description
1     THETA1                      apparent oral clearance
2     THETA2                 central volume of distribution
3     THETA3                      absorption rate constant
4     THETA4                   intercompartmental clearance
5     THETA5              peripheral volume of distribution
6     THETA6                      male effect on clearance
7     THETA7                    weight effect on clearance
8   OMEGA1.1        interindividual variability of clearance
9   OMEGA2.1     interindividual clearance-volume covariance
10  OMEGA2.2  interindividual variability of central volume
11  OMEGA3.1         interindividual clearance-Ka covariance
12  OMEGA3.2            interindividual volume-Ka covariance
13  OMEGA3.3               interindividual variability of Ka
14  SIGMA1.1                            proportional error
15  SIGMA2.2                                additive error
                                                              model estimate
1  CL/F (L/h) ~ theta_1 *  theta_6 ^MALE * (WT/70)^theta_7  * e^eta_1       16
2                           V_c /F (L) ~ theta_2 * (WT/70)^1 * e^eta_2       14
3                                      K_a (h^-1 ) ~ theta_3 * e^eta_3        6
4                                                Q/F (L/h) ~ theta_4       15
5                                               V_p /F (L) ~ theta_5       12
6                                                MALE_CL/F ~ theta_6       11
```

```
7                                              WT_CL/F ~ theta_7       13
8                                            IIV_CL/F ~ Omega_1.1      10
9                                            cov_CL,V ~ Omega_2.1       8
10                                         IIV_V_c /F ~ Omega_2.2       7
11                                         cov_CL,Ka  ~ Omega_3.1       2
12                                          cov_V,Ka  ~ Omega_3.2       3
13                                           IIV_K_a  ~ Omega_3.3       4
14                                           err_prop ~ Sigma_1.1       5
15                                            err_add ~ Sigma_2.2       9
   prse
1    15
2    14
3    13
4     3
5     5
6     2
7     8
8     6
9     7
10    9
11    4
12   12
13   11
14    1
15   10
```

Now we can extract some information from the model statements.

Listing 3:

```
> tab$units <- justUnits(tab$model)
> tab$model <- noUnits(tab$model)
> tab$name <- with(tab, wiki2label(model))
> tab[c('model','units','name')]
```

```
                                                         model units
1  CL/F  ~ theta_1 *  theta_6 ^MALE * (WT/70)^theta_7  * e^eta_1   L/h
2                        V_c /F  ~ theta_2 * (WT/70)^1 * e^eta_2     L
3                                      K_a  ~ theta_3 * e^eta_3 h^-1
4                                                 Q/F  ~ theta_4   L/h
5                                              V_p /F  ~ theta_5     L
6                                            MALE_CL/F ~ theta_6
7                                              WT_CL/F ~ theta_7
8                                            IIV_CL/F ~ Omega_1.1
9                                            cov_CL,V ~ Omega_2.1
10                                         IIV_V_c /F ~ Omega_2.2
11                                         cov_CL,Ka  ~ Omega_3.1
12                                          cov_V,Ka  ~ Omega_3.2
13                                           IIV_K_a  ~ Omega_3.3
14                                           err_prop ~ Sigma_1.1
15                                            err_add ~ Sigma_2.2
```

```
          name
1         CL/F
2        V_c/F
3          K_a
4          Q/F
5        V_p/F
6   MALE_CL/F
7     WT_CL/F
8    IIV_CL/F
9    cov_CL,V
10 IIV_V_c/F
11 cov_CL,Ka
12  cov_V,Ka
13    IIV_K_a
14   err_prop
15    err_add
```

3 variance

The estimates for the matrix diagonals are variances, and their square roots have special meaning. In model 1005, interindividual variability was modelled exponentially, in which case square root of variance gives an approximate CV; alternatively, and exact CV can be calculated. For proportional error terms like ERR1, square root gives an exact CV. For additive error terms like ERR2, square root gives standard deviation.

We can use functions of 'parameter' to sort out the various error components, as they are used in this model.

3.1 exponential

Listing 4:

```
> expo <- is.iiv(tab$parameter) & is.diagonal(tab$parameter)
> tab$parameter[expo]
```

```
[1] "OMEGA1.1" "OMEGA2.2" "OMEGA3.3"
```

Listing 5:

```
> tab$cv[expo] <- cvLognormal(tab$estimate[expo])
> tab[,c('parameter','name','estimate','cv')]
```

```
   parameter      name estimate         cv
1     THETA1      CL/F       16         NA
2     THETA2     V_c/F       14         NA
3     THETA3       K_a        6         NA
```

```
4      THETA4        Q/F        15          NA
5      THETA5      V_p/F        12          NA
6      THETA6 MALE_CL/F        11          NA
7      THETA7    WT_CL/F        13          NA
8    OMEGA1.1   IIV_CL/F        10  148.409790
9    OMEGA2.1   cov_CL,V         8          NA
10   OMEGA2.2 IIV_V_c/F          7   33.100350
11   OMEGA3.1 cov_CL,Ka          2          NA
12   OMEGA3.2   cov_V,Ka         3          NA
13   OMEGA3.3    IIV_K_a         4    7.321076
14   SIGMA1.1   err_prop         5          NA
15   SIGMA2.2    err_add         9          NA
```

3.2 proportional

Listing 6:

```
> writeLines(read.nmctl('../nonmem/ctl/1005.ctl')$err)

 Y=F*(1+ERR(1)) + ERR(2)
 IPRE=F
;<doc>
```

Listing 7:

```
> prop <- is.random(tab$parameter) & tab$name %contains% 'prop'
> tab$parameter[prop]

[1] "SIGMA1.1"
```

Listing 8:

```
> tab$cv[prop] <- sqrt(tab$estimate[prop])
> tab[,c('parameter','name','estimate','cv')]

   parameter       name estimate          cv
1      THETA1       CL/F       16          NA
2      THETA2      V_c/F       14          NA
3      THETA3        K_a        6          NA
4      THETA4        Q/F       15          NA
5      THETA5      V_p/F       12          NA
6      THETA6  MALE_CL/F       11          NA
7      THETA7    WT_CL/F       13          NA
8    OMEGA1.1   IIV_CL/F       10  148.409790
9    OMEGA2.1   cov_CL,V        8          NA
10   OMEGA2.2  IIV_V_c/F        7   33.100350
11   OMEGA3.1  cov_CL,Ka        2          NA
12   OMEGA3.2   cov_V,Ka        3          NA
13   OMEGA3.3    IIV_K_a        4    7.321076
14   SIGMA1.1   err_prop        5    2.236068
15   SIGMA2.2    err_add        9          NA
```

3.3 additive

Listing 9:

```
> add <- is.residual(tab$parameter) & tab$name %contains% 'add'
> tab$parameter[add]
```

```
[1] "SIGMA2.2"
```

Listing 10:

```
> tab$sd[add] <- sqrt(tab$estimate[add])
> tab[,c('parameter','name','estimate','cv','sd')]
```

```
   parameter      name estimate         cv sd
1     THETA1      CL/F       16         NA NA
2     THETA2     V_c/F       14         NA NA
3     THETA3       K_a        6         NA NA
4     THETA4       Q/F       15         NA NA
5     THETA5     V_p/F       12         NA NA
6     THETA6 MALE_CL/F       11         NA NA
7     THETA7   WT_CL/F       13         NA NA
8   OMEGA1.1  IIV_CL/F       10 148.409790 NA
9   OMEGA2.1  cov_CL,V        8         NA NA
10  OMEGA2.2 IIV_V_c/F        7  33.100350 NA
11  OMEGA3.1 cov_CL,Ka        2         NA NA
12  OMEGA3.2  cov_V,Ka        3         NA NA
13  OMEGA3.3   IIV_K_a        4   7.321076 NA
14  SIGMA1.1  err_prop        5   2.236068 NA
15  SIGMA2.2   err_add        9         NA  3
```

4 covariance

The estimates of matrix off-diagonals are covariances, and are more useful if transformed to correlations. We could extract the matrices manually, or use package shortcuts.

Listing 11:

```
> cor <- omegacor(run=1005,project='../nonmem')
> cor
```

```
           [,1]       [,2]       [,3]
[1,]  1.0000000  0.8492811 -0.1163229
[2,]  0.8492811  1.0000000 -0.5607054
[3,] -0.1163229 -0.5607054  1.0000000
```

Listing 12:

```
> half(cor)
```

```
      1.1        2.1        2.2        3.1        3.2        3.3
1.0000000  0.8492811  1.0000000 -0.1163229 -0.5607054  1.0000000
```

Listing 13:

```
> offdiag(half(cor))
```

```
      2.1        3.1        3.2
0.8492811 -0.1163229 -0.5607054
```

Listing 14:

```
> off <- is.iiv(tab$parameter) & is.offdiagonal(tab$parameter)
> tab$parameter[off]
```

```
[1] "OMEGA2.1" "OMEGA3.1" "OMEGA3.2"
```

Listing 15:

```
> tab$cor[off] <- offdiag(half(cor))
> tab[,c('parameter','name','estimate','cv','sd','cor')]
```

```
   parameter       name estimate         cv sd        cor
1     THETA1       CL/F       16         NA NA         NA
2     THETA2      V_c/F       14         NA NA         NA
3     THETA3        K_a        6         NA NA         NA
4     THETA4        Q/F       15         NA NA         NA
5     THETA5      V_p/F       12         NA NA         NA
6     THETA6 MALE_CL/F       11         NA NA         NA
7     THETA7    WT_CL/F       13         NA NA         NA
8   OMEGA1.1   IIV_CL/F       10 148.409790 NA         NA
9   OMEGA2.1   cov_CL,V        8         NA NA  0.8492811
10  OMEGA2.2 IIV_V_c/F         7  33.100350 NA         NA
11  OMEGA3.1  cov_CL,Ka        2         NA NA -0.1163229
12  OMEGA3.2   cov_V,Ka        3         NA NA -0.5607054
13  OMEGA3.3    IIV_K_a        4   7.321076 NA         NA
14  SIGMA1.1   err_prop        5   2.236068 NA         NA
15  SIGMA2.2    err_add        9         NA  3         NA
```

5 confidence interval

We wish to include 95 percentiles in our table as confidence intervals.

Listing 16:

```
> boot <- read.csv('../nonmem/1005bootlog.csv',as.is=TRUE)
> head(boot)
```

```
  X tool run parameter   moment             value
1 1  nm7   1       ofv  minimum 2459.17577365853
2 2  nm7   1    THETA1 estimate          9.90635
3 3  nm7   1    THETA1     prse             <NA>
4 4  nm7   1    THETA1       se             <NA>
5 5  nm7   1    THETA2 estimate          21.8851
6 6  nm7   1    THETA2     prse             <NA>
```

Listing 17:

```
> boot <- boot[boot$moment=='estimate',]
> boot <- data.frame(cast(boot,... ~ moment))
> head(boot)
```

```
   X tool run parameter  estimate
1  2  nm7   1    THETA1   9.90635
2  5  nm7   1    THETA2   21.8851
3  8  nm7   1    THETA3 0.0708169
4 11  nm7   1    THETA4   3.36905
5 14  nm7   1    THETA5   94.6453
6 17  nm7   1    THETA6  0.972457
```

Listing 18:

```
> boot <- boot[,c('run','parameter','estimate')]
> sapply(boot,class)
```

```
        run   parameter    estimate
  "integer" "character"    "factor"
```

Listing 19:

```
> boot$estimate <- as.numeric(as.character(boot$estimate))
> unique(boot$parameter)
```

```
 [1] "THETA1"   "THETA2"   "THETA3"   "THETA4"   "THETA5"   "THETA6"
 [7] "THETA7"   "OMEGA1.1" "OMEGA2.1" "OMEGA2.2" "OMEGA3.1" "OMEGA3.2"
[13] "OMEGA3.3" "SIGMA1.1" "SIGMA2.1" "SIGMA2.2"
```

Listing 20:

```
> quan <- function(x,probs)as.character(signif(quantile(x,probs=probs,na.rm=TRUE)
   ,3))
> boot$lo <- with(boot, reapply(estimate,parameter,quan,probs=.05))
> boot$hi <- with(boot, reapply(estimate,parameter,quan,probs=.95))
> head(boot)
```

```
  run parameter   estimate     lo     hi
1   1    THETA1  9.9063500   7.31   11.1
2   1    THETA2 21.8851000   19.2   27.9
3   1    THETA3  0.0708169 0.0625 0.0838
4   1    THETA4  3.3690500   2.78   4.91
5   1    THETA5 94.6453000   85.6    559
6   1    THETA6  0.9724570  0.847   1.25
```

Listing 21:

```
> boot <- unique(boot[,c('parameter','lo','hi')])
> boot
```

```
   parameter      lo      hi
1     THETA1    7.31    11.1
2     THETA2    19.2    27.9
3     THETA3  0.0625  0.0838
4     THETA4    2.78    4.91
5     THETA5    85.6     559
6     THETA6   0.847    1.25
7     THETA7    0.61    1.91
8   OMEGA1.1   0.128   0.321
9   OMEGA2.1  0.0606   0.183
10  OMEGA2.2   0.047   0.158
11  OMEGA3.1 -0.0448  0.0261
12  OMEGA3.2 -0.0577 -0.0049
13  OMEGA3.3  0.0236  0.0811
14  SIGMA1.1  0.0399  0.0587
15  SIGMA2.1       0       0
16  SIGMA2.2  0.0836   0.329
```

Listing 22:

```
> boot$ci <- with(boot, parens(glue(lo,',',hi)))
> boot
```

```
   parameter      lo      hi                ci
1     THETA1    7.31    11.1       (7.31,11.1)
2     THETA2    19.2    27.9       (19.2,27.9)
3     THETA3  0.0625  0.0838   (0.0625,0.0838)
4     THETA4    2.78    4.91       (2.78,4.91)
5     THETA5    85.6     559        (85.6,559)
6     THETA6   0.847    1.25      (0.847,1.25)
7     THETA7    0.61    1.91       (0.61,1.91)
8   OMEGA1.1   0.128   0.321     (0.128,0.321)
9   OMEGA2.1  0.0606   0.183    (0.0606,0.183)
10  OMEGA2.2   0.047   0.158     (0.047,0.158)
11  OMEGA3.1 -0.0448  0.0261  (-0.0448,0.0261)
12  OMEGA3.2 -0.0577 -0.0049 (-0.0577,-0.0049)
13  OMEGA3.3  0.0236  0.0811   (0.0236,0.0811)
14  SIGMA1.1  0.0399  0.0587   (0.0399,0.0587)
15  SIGMA2.1       0       0             (0,0)
16  SIGMA2.2  0.0836   0.329    (0.0836,0.329)
```

Listing 23:

```
> tab <- stableMerge(tab,boot[,c('parameter','ci')])
> tab
```

	parameter	description
1	THETA1	apparent oral clearance
2	THETA2	central volume of distribution
3	THETA3	absorption rate constant
4	THETA4	intercompartmental clearance
5	THETA5	peripheral volume of distribution
6	THETA6	male effect on clearance
7	THETA7	weight effect on clearance
8	OMEGA1.1	interindividual variability of clearance
9	OMEGA2.1	interindividual clearance-volume covariance
10	OMEGA2.2	interindividual variability of central volume
11	OMEGA3.1	interindividual clearance-Ka covariance
12	OMEGA3.2	interindividual volume-Ka covariance
13	OMEGA3.3	interindividual variability of Ka
14	SIGMA1.1	proportional error
15	SIGMA2.2	additive error

	model	estimate	prse
1	CL/F ~ theta_1 * theta_6 ^MALE * (WT/70)^theta_7 * e^eta_1	16	15
2	V_c /F ~ theta_2 * (WT/70)^1 * e^eta_2	14	14
3	K_a ~ theta_3 * e^eta_3	6	13
4	Q/F ~ theta_4	15	3
5	V_p /F ~ theta_5	12	5
6	MALE_CL/F ~ theta_6	11	2
7	WT_CL/F ~ theta_7	13	8
8	IIV_CL/F ~ Omega_1.1	10	6
9	cov_CL,V ~ Omega_2.1	8	7
10	IIV_V_c /F ~ Omega_2.2	7	9
11	cov_CL,Ka ~ Omega_3.1	2	4
12	cov_V,Ka ~ Omega_3.2	3	12
13	IIV_K_a ~ Omega_3.3	4	11
14	err_prop ~ Sigma_1.1	5	1
15	err_add ~ Sigma_2.2	9	10

	units	name	cv	sd	cor	ci
1	L/h	CL/F	NA	NA	NA	(7.31,11.1)
2	L	V_c/F	NA	NA	NA	(19.2,27.9)
3	h^-1	K_a	NA	NA	NA	(0.0625,0.0838)
4	L/h	Q/F	NA	NA	NA	(2.78,4.91)
5	L	V_p/F	NA	NA	NA	(85.6,559)
6		MALE_CL/F	NA	NA	NA	(0.847,1.25)
7		WT_CL/F	NA	NA	NA	(0.61,1.91)
8		IIV_CL/F	148.409790	NA	NA	(0.128,0.321)
9		cov_CL,V	NA	NA	0.8492811	(0.0606,0.183)
10		IIV_V_c/F	33.100350	NA	NA	(0.047,0.158)
11		cov_CL,Ka	NA	NA	-0.1163229	(-0.0448,0.0261)
12		cov_V,Ka	NA	NA	-0.5607054	(-0.0577,-0.0049)
13		IIV_K_a	7.321076	NA	NA	(0.0236,0.0811)
14		err_prop	2.236068	NA	NA	(0.0399,0.0587)
15		err_add	NA	3	NA	(0.0836,0.329)

6 aesthetics

Here we format the table for printing.

Listing 24:

```
> tab$name <- NULL
> tab$parameter <- NULL
> tab$model <- wiki2latex(tab$model)
> tab$estimate <- as.character(tab$estimate)
> tab$estimate <- paste(tab$estimate,'$', tab$units,'$')
> tab$units <- NULL
```

Note that no parameter defines more than one of CV, SD,and COR. We could collapse these into a single column, and add a descriptive flag.

Listing 25:

```
> m <- as.matrix(tab[,c('cv','sd','cor')])
> tab$variability <- suppressWarnings(apply(m,1,max,na.rm=TRUE))
> tab$variability[is.infinite(tab$variability)] <- NA
> i <- is.defined(m)
> i[!i] <- NA
> tab$statistic <- apply(i,1,function(x){
+   p <- colnames(i)[x]
+   ifelse(all(is.na(p)),NA,p[!is.na(p)])
+ })
> toPercent <- with(tab, !is.na(statistic) & statistic=='cv')
> tab$variability[toPercent] <- percent(tab$variability[toPercent])
> tab$variability <- as.character(signif(tab$variability,3))
> tab$statistic <- map(tab$statistic,from=c(NA,'cv','cor','sd'),to=c(NA,'\\%CV','
   CORR','SD'))
> tab$variability <- paste(tab$statistic,tab$variability,sep=' = ')
> tab$variability[is.na(tab$statistic)] <- NA
> tab$statistic <- NULL
> tab$cv <- NULL
> tab$sd <- NULL
> tab$cor <- NULL
```

7 simple parameter table

We can make a quick parameter table that does not use wikitab markup. Table 2.

Listing 26:

```
> tab <- rlog(1005,'../nonmem',tool='nm7',file=NULL)
> head(tab)
```

Table 1: Parameter Estimates from Population Pharmacokinetic Model Run 1005

description	model	estimate	prse	ci
apparent oral clearance	$\mathrm{CL/F} \sim \theta_1 \cdot \theta_6^{\mathrm{MALE}} \cdot (\mathrm{WT}/70)^{\theta_7} \cdot \mathrm{e}^{\eta_1}$	16 L/h	15	(7.31,11.1)
central volume of distribution	$\mathrm{V_c/F} \sim \theta_2 \cdot (\mathrm{WT}/70)^1 \cdot \mathrm{e}^{\eta_2}$	14 L	14	(19.2,27.9)
absorption rate constant	$\mathrm{K_a} \sim \theta_3 \cdot \mathrm{e}^{\eta_3}$	6 h^-1	13	(0.0625,0.0838)
intercompartmental clearance	$\mathrm{Q/F} \sim \theta_4$	15 L/h	3	(2.78,4.91)
peripheral volume of distribution	$\mathrm{V_p/F} \sim \theta_5$	12 L	5	(85.6,559)
male effect on clearance	$\mathrm{MALE_{CL/F}} \sim \theta_6$	11	2	(0.847,1.25)
weight effect on clearance	$\mathrm{WT_{CL/F}} \sim \theta_7$	13	8	(0.61,1.91)
interindividual variability of clearance	$\mathrm{IIV_{CL/F}} \sim \Omega_{1.1}$	10	6	(0.128,0.321)
interindividual clearance-volume covariance	$\mathrm{cov_{CL,V}} \sim \Omega_{2.1}$	8	7	(0.0606,0.183)
interindividual variability of central volume	$\mathrm{IIV_{V_c/F}} \sim \Omega_{2.2}$	7	9	(0.047,0.158)
interindividual clearance-Ka covariance	$\mathrm{cov_{CL,Ka}} \sim \Omega_{3.1}$	2	4	(-0.0448,0.0261)
interindividual volume-Ka covariance	$\mathrm{cov_{V,Ka}} \sim \Omega_{3.2}$	3	12	(-0.0577,-0.0049)
interindividual variability of Ka	$\mathrm{IIV_{K_a}} \sim \Omega_{3.3}$	4	11	(0.0236,0.0811)
proportional error	$\mathrm{err_{prop}} \sim \Sigma_{1.1}$	5	1	(0.0399,0.0587)
additive error	$\mathrm{err_{add}} \sim \Sigma_{2.2}$	9	10	(0.0836,0.329)

```
  tool  run parameter   moment            value
1  nm7 1005       ofv  minimum 2405.91625845151
2  nm7 1005    THETA1 estimate          9.50754
3  nm7 1005    THETA1     prse             9.84
4  nm7 1005    THETA1       se         0.935942
5  nm7 1005    THETA2 estimate          22.7907
6  nm7 1005    THETA2     prse             9.56
```

Listing 27:

```
> tab$tool <- NULL
> tab$run <- NULL
> tab <- tab[tab$moment %in% c('estimate','prse'),]
> unique(tab$parameter)
```

```
 [1] "THETA1"   "THETA2"   "THETA3"   "THETA4"   "THETA5"   "THETA6"
 [7] "THETA7"   "OMEGA1.1" "OMEGA2.1" "OMEGA2.2" "OMEGA3.1" "OMEGA3.2"
[13] "OMEGA3.3" "SIGMA1.1" "SIGMA2.1" "SIGMA2.2"
```

Listing 28:

```
> tab$value <- signif(as.numeric(tab$value),3)
> tab$parameter <- factor(tab$parameter,levels=unique(tab$parameter))#to preserve
   row order during cast
> tab <- cast(tab,parameter ~ moment)
> tab
```

```
   parameter estimate   prse
1     THETA1    9.5100   9.84
2     THETA2   22.8000   9.56
3     THETA3    0.0714   7.35
4     THETA4    3.4700  15.40
5     THETA5  113.0000  21.00
6     THETA6    1.0200  11.20
7     THETA7    1.1900  28.40
8   OMEGA1.1    0.2140  22.80
9   OMEGA2.1    0.1210  26.40
10  OMEGA2.2    0.0945  33.20
11  OMEGA3.1   -0.0116 173.00
12  OMEGA3.2   -0.0372  36.10
13  OMEGA3.3    0.0466  34.70
14  SIGMA1.1    0.0492  10.90
15  SIGMA2.1    0.0000    Inf
16  SIGMA2.2    0.2020  33.50
```

Listing 29:

```
> tab$parameter <- parameter2wiki(tab$parameter)
> tab
```

```
    parameter estimate   prse
1     theta_1    9.5100   9.84
2     theta_2   22.8000   9.56
3     theta_3    0.0714   7.35
4     theta_4    3.4700  15.40
5     theta_5  113.0000  21.00
6     theta_6    1.0200  11.20
7     theta_7    1.1900  28.40
8   Omega_1.1    0.2140  22.80
9   Omega_2.1    0.1210  26.40
10  Omega_2.2    0.0945  33.20
11  Omega_3.1   -0.0116 173.00
12  Omega_3.2   -0.0372  36.10
13  Omega_3.3    0.0466  34.70
14  Sigma_1.1    0.0492  10.90
15  Sigma_2.1    0.0000    Inf
16  Sigma_2.2    0.2020  33.50
```

Listing 30:

```
> tab$parameter <- wiki2latex(tab$parameter)
> tab
```

```
                  parameter estimate   prse
1    $\\mathrm{\\theta_{1}}$    9.5100   9.84
2    $\\mathrm{\\theta_{2}}$   22.8000   9.56
3    $\\mathrm{\\theta_{3}}$    0.0714   7.35
4    $\\mathrm{\\theta_{4}}$    3.4700  15.40
```

```
5     $\\mathrm{\\theta_{5}}$ 113.0000   21.00
6     $\\mathrm{\\theta_{6}}$   1.0200   11.20
7     $\\mathrm{\\theta_{7}}$   1.1900   28.40
8   $\\mathrm{\\Omega_{1.1}}$   0.2140   22.80
9   $\\mathrm{\\Omega_{2.1}}$   0.1210   26.40
10  $\\mathrm{\\Omega_{2.2}}$   0.0945   33.20
11  $\\mathrm{\\Omega_{3.1}}$  -0.0116 173.00
12  $\\mathrm{\\Omega_{3.2}}$  -0.0372   36.10
13  $\\mathrm{\\Omega_{3.3}}$   0.0466   34.70
14  $\\mathrm{\\Sigma_{1.1}}$   0.0492   10.90
15  $\\mathrm{\\Sigma_{2.1}}$   0.0000     Inf
16  $\\mathrm{\\Sigma_{2.2}}$   0.2020   33.50
```

Table 2: Simple Parameter Table

parameter	estimate	prse
θ_1	9.5100	9.84
θ_2	22.8000	9.56
θ_3	0.0714	7.35
θ_4	3.4700	15.40
θ_5	113.0000	21.00
θ_6	1.0200	11.20
θ_7	1.1900	28.40
$\Omega_{1.1}$	0.2140	22.80
$\Omega_{2.1}$	0.1210	26.40
$\Omega_{2.2}$	0.0945	33.20
$\Omega_{3.1}$	-0.0116	173.00
$\Omega_{3.2}$	-0.0372	36.10
$\Omega_{3.3}$	0.0466	34.70
$\Sigma_{1.1}$	0.0492	10.90
$\Sigma_{2.1}$	0.0000	Inf
$\Sigma_{2.2}$	0.2020	33.50