betaMC: Internal Tests

Ivan Jacob Agaloos Pesigan

Tests

```
#> test-betaMC-beta-mc-est-mi
#> Call:
#> BetaMC(object = mc)
#>
#> Standardized regression slopes
#> type = "mvn"
#>            est     se R  0.05%    0.5%    2.5%  97.5%   99.5% 99.95%
#> NARTIC  0.4951 0.0396 5 0.5099 0.5099 0.5099 0.5958 0.5984 0.5990
#> PCTGRT  0.3915 0.0341 5 0.3009 0.3011 0.3021 0.3754 0.3760 0.3761
#> PCTSUPP 0.2632 0.0667 5 0.2159 0.2160 0.2168 0.3707 0.3766 0.3779
#> Call:
#> BetaMC(object = mc)
#>
#> Standardized regression slopes
#> type = "mvn"
#> Test passed
#> Call:
#> BetaMC(object = mc)
#>
#> Standardized regression slopes
#> type = "mvn"
#>          est     se R  0.05%    0.5%    2.5%  97.5%   99.5% 99.95%
#> NARTIC 0.7622 0.0533 5 0.7302 0.7302 0.7303 0.8462 0.8543 0.8561
#> Call:
#> BetaMC(object = mc)
#>
#> Standardized regression slopes
#> type = "mvn"

#> test-betaMC-beta-mc-est

#> Call:
#> BetaMC(object = mc)
#>
#> Standardized regression slopes
```

```
#> type = "mvn"
#>             est      se R  0.05%   0.5%    2.5%  97.5%   99.5% 99.95%
#> NARTIC  0.4951 0.0536 5 0.4460 0.4461 0.4462 0.5572 0.5580 0.5582
#> PCTGRT  0.3915 0.0470 5 0.3247 0.3251 0.3271 0.4391 0.4421 0.4427
#> PCTSUPP 0.2632 0.0953 5 0.1508 0.1511 0.1525 0.3649 0.3700 0.3711
#> Call:
#> BetaMC(object = mc)
#>
#> Standardized regression slopes
#> type = "mvn"
#> Test passed
#> Call:
#> BetaMC(object = mc)
#>
#> Standardized regression slopes
#> type = "mvn"
#>            est      se R 0.05%   0.5%    2.5%  97.5%   99.5% 99.95%
#> NARTIC 0.7622 0.0515 5 0.639 0.6392 0.6402 0.7582 0.7649 0.7664
#> Call:
#> BetaMC(object = mc)
#>
#> Standardized regression slopes
#> type = "mvn"
```

#> *test-betaMC-delta-r-sq-mc-est-mi*

```
#> Call:
#> DeltaRSqMC(object = mc)
#>
#> Improvement in R-squared
#> type = "mvn"
#>             est      se R  0.05%   0.5%    2.5%  97.5%   99.5% 99.95%
#> NARTIC  0.1859 0.1018 5 0.0830 0.0848 0.0925 0.3510 0.3626 0.3652
#> PCTGRT  0.1177 0.0661 5 0.0937 0.0938 0.0942 0.2231 0.2240 0.2242
#> PCTSUPP 0.0569 0.0405 5 0.0117 0.0121 0.0138 0.1038 0.1043 0.1044
#> Call:
#> DeltaRSqMC(object = mc)
#>
#> Improvement in R-squared
#> type = "mvn"
#> Test passed
#> Test passed
```

#> *test-betaMC-delta-r-sq-mc-est*

```
#> Call:
#> DeltaRSqMC(object = mc)
```

```
#>
#> Improvement in R-squared
#> type = "mvn"
#>           est      se R   0.05%    0.5%    2.5%  97.5%  99.5% 99.95%
#> NARTIC  0.1859 0.0347 5 0.1528 0.1529 0.1537 0.2257 0.2258 0.2258
#> PCTGRT  0.1177 0.0151 5 0.0882 0.0882 0.0883 0.1185 0.1187 0.1187
#> PCTSUPP 0.0569 0.0574 5 0.0178 0.0183 0.0203 0.1521 0.1546 0.1552
#> Call:
#> DeltaRSqMC(object = mc)
#>
#> Improvement in R-squared
#> type = "mvn"
#> Test passed
#> Test passed

#> test-betaMC-diff-beta-mc-est-mi

#> Call:
#> DiffBetaMC(object = mc)
#>
#> Differences of standardized regression slopes
#> type = "mvn"
#>                  est     se R    0.05%     0.5%     2.5%  97.5%  99.5% 99.95%
#> NARTIC-PCTGRT  0.1037 0.1565 5 -0.0955 -0.0924 -0.0785 0.3202 0.3340 0.3372
#> NARTIC-PCTSUPP 0.2319 0.1692 5  0.0025  0.0063  0.0234 0.4490 0.4593 0.4616
#> PCTGRT-PCTSUPP 0.1282 0.1117 5 -0.0043 -0.0025  0.0057 0.2859 0.2951 0.2972
#> Call:
#> DiffBetaMC(object = mc)
#>
#> Differences of standardized regression slopes
#> type = "mvn"
#> Test passed
#> Test passed

#> test-betaMC-diff-beta-mc-est

#> Call:
#> DiffBetaMC(object = mc)
#>
#> Differences of standardized regression slopes
#> type = "mvn"
#>                  est     se R    0.05%     0.5%     2.5%  97.5%  99.5% 99.95%
#> NARTIC-PCTGRT  0.1037 0.0575 5  0.1451  0.1453  0.1460 0.2806 0.2862 0.2875
#> NARTIC-PCTSUPP 0.2319 0.1266 5  0.0948  0.0955  0.0989 0.4033 0.4149 0.4175
#> PCTGRT-PCTSUPP 0.1282 0.0707 5 -0.0503 -0.0497 -0.0471 0.1230 0.1287 0.1300
#> Call:
#> DiffBetaMC(object = mc)
```

3

```
#>
#> Differences of standardized regression slopes
#> type = "mvn"
#> Test passed
#> Test passed

#> test-betaMC-mc-fixed-x-mi

#> MCMI(object = object, mi = mi, R = R, type = "mvn", fixed_x = TRUE)
#> MCMI(object = object, mi = mi, R = R, type = "adf", fixed_x = TRUE)
#> MCMI(object = object, mi = mi, R = R, type = "hc0", fixed_x = TRUE)
#> MCMI(object = object, mi = mi, R = R, type = "hc1", fixed_x = TRUE)
#> MCMI(object = object, mi = mi, R = R, type = "hc2", fixed_x = TRUE)
#> MCMI(object = object, mi = mi, R = R, type = "hc3", fixed_x = TRUE)
#> MCMI(object = object, mi = mi, R = R, type = "hc4", fixed_x = TRUE)
#> MCMI(object = object, mi = mi, R = R, type = "hc4m", fixed_x = TRUE)
#> MCMI(object = object, mi = mi, R = R, type = "hc5", fixed_x = TRUE)
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed

#> test-betaMC-mc-fixed-x

#> MC(object = object, R = R, type = "mvn", fixed_x = TRUE)
#> MC(object = object, R = R, type = "adf", fixed_x = TRUE)
#> MC(object = object, R = R, type = "hc0", fixed_x = TRUE)
#> MC(object = object, R = R, type = "hc1", fixed_x = TRUE)
#> MC(object = object, R = R, type = "hc2", fixed_x = TRUE)
#> MC(object = object, R = R, type = "hc3", fixed_x = TRUE)
#> MC(object = object, R = R, type = "hc4", fixed_x = TRUE)
#> MC(object = object, R = R, type = "hc4m", fixed_x = TRUE)
#> MC(object = object, R = R, type = "hc5", fixed_x = TRUE)
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
```

4

```
#> Test passed
#> Call:
#> MC(object = object, R = 5L, decomposition = "chol", fixed_x = TRUE)
#> The first set of simulated parameter estimates
#> and model-implied covariance matrix.
#>
#> $coef
#> [1] 0.4631859 0.5028178
#>
#> $sigmasq
#> [1] 0.5829794
#>
#> $vechsigmacapx
#> [1] 1.000000e+00 2.496804e-16 1.000000e+00
#>
#> $sigmacapx
#>              [,1]         [,2]
#> [1,] 1.000000e+00 2.496804e-16
#> [2,] 2.496804e-16 1.000000e+00
#>
#> $sigmaysq
#> [1] 1.050346
#>
#> $sigmayx
#> [1] 0.4631859 0.5028178
#>
#> $sigmacap
#>            [,1]         [,2]         [,3]
#> [1,] 1.0503464 4.631859e-01 5.028178e-01
#> [2,] 0.4631859 1.000000e+00 2.496804e-16
#> [3,] 0.5028178 2.496804e-16 1.000000e+00
#>
#> $pd
#> [1] TRUE
#>
#> Call:
#> MC(object = object, R = 5L, decomposition = "svd", fixed_x = TRUE)
#> The first set of simulated parameter estimates
#> and model-implied covariance matrix.
#>
#> $coef
#> [1] 0.4719623 0.4979797
#>
#> $sigmasq
#> [1] 0.5270782
#>
```

```
#> $vechsigmacapx
#> [1] 1.000000e+00 2.496804e-16 1.000000e+00
#>
#> $sigmacapx
#>              [,1]          [,2]
#> [1,] 1.000000e+00 2.496804e-16
#> [2,] 2.496804e-16 1.000000e+00
#>
#> $sigmaysq
#> [1] 0.9978104
#>
#> $sigmayx
#> [1] 0.4719623 0.4979797
#>
#> $sigmacap
#>           [,1]          [,2]          [,3]
#> [1,] 0.9978104 4.719623e-01 4.979797e-01
#> [2,] 0.4719623 1.000000e+00 2.496804e-16
#> [3,] 0.4979797 2.496804e-16 1.000000e+00
#>
#> $pd
#> [1] TRUE

#> test-betaMC-mc-mi

#> MCMI(object = object, mi = mi, R = R, type = "mvn")
#> MCMI(object = object, mi = mi, R = R, type = "adf")
#> MCMI(object = object, mi = mi, R = R, type = "hc0")
#> MCMI(object = object, mi = mi, R = R, type = "hc1")
#> MCMI(object = object, mi = mi, R = R, type = "hc2")
#> MCMI(object = object, mi = mi, R = R, type = "hc3")
#> MCMI(object = object, mi = mi, R = R, type = "hc4")
#> MCMI(object = object, mi = mi, R = R, type = "hc4m")
#> MCMI(object = object, mi = mi, R = R, type = "hc5")
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed

#> test-betaMC-mc

#> MC(object = object, R = R, type = "mvn")
```

```
#> MC(object = object, R = R, type = "adf")
#> MC(object = object, R = R, type = "hc0")
#> MC(object = object, R = R, type = "hc1")
#> MC(object = object, R = R, type = "hc2")
#> MC(object = object, R = R, type = "hc3")
#> MC(object = object, R = R, type = "hc4")
#> MC(object = object, R = R, type = "hc4m")
#> MC(object = object, R = R, type = "hc5")
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Test passed
#> Call:
#> MC(object = object, R = 5L, decomposition = "chol")
#> The first set of simulated parameter estimates
#> and model-implied covariance matrix.
#>
#> $coef
#> [1] 0.4681243 0.5077419
#>
#> $sigmasq
#> [1] 0.5336115
#>
#> $vechsigmacapx
#> [1] 1.024900520 0.005684992 1.014339591
#>
#> $sigmacapx
#>              [,1]         [,2]
#> [1,] 1.024900520 0.005684992
#> [2,] 0.005684992 1.014339591
#>
#> $sigmaysq
#> [1] 1.02241
#>
#> $sigmayx
#> [1] 0.4826673 0.5176840
#>
#> $sigmacap
#>           [,1]         [,2]         [,3]
#> [1,] 1.0224096 0.482667296 0.517683980
```

```
#> [2,] 0.4826673 1.024900520 0.005684992
#> [3,] 0.5176840 0.005684992 1.014339591
#>
#> $pd
#> [1] TRUE
#>
#> Call:
#> MC(object = object, R = 5L, decomposition = "svd")
#> The first set of simulated parameter estimates
#> and model-implied covariance matrix.
#>
#> $coef
#> [1] 0.5306815 0.4324981
#>
#> $sigmasq
#> [1] 0.4837773
#>
#> $vechsigmacapx
#> [1] 1.041535669 0.000887221 1.017708053
#>
#> $sigmacapx
#>              [,1]        [,2]
#> [1,] 1.041535669 0.000887221
#> [2,] 0.000887221 1.017708053
#>
#> $sigmaysq
#> [1] 0.9678719
#>
#> $sigmayx
#> [1] 0.5531074 0.4406277
#>
#> $sigmacap
#>           [,1]        [,2]        [,3]
#> [1,] 0.9678719 0.553107427 0.440627675
#> [2,] 0.5531074 1.041535669 0.000887221
#> [3,] 0.4406277 0.000887221 1.017708053
#>
#> $pd
#> [1] TRUE
#> test-betaMC-p-cor-mc-est-mi

#> Call:
#> PCorMC(object = mc)
#>
#> Squared partial correlations
#> type = "mvn"
```

```
#>             est     se R  0.05%    0.5%    2.5%  97.5%  99.5% 99.95%
#> NARTIC   0.4874 0.0892 5 0.3826 0.3834 0.3866 0.6036 0.6122 0.6141
#> PCTGRT   0.3757 0.1882 5 0.1696 0.1701 0.1725 0.5765 0.5856 0.5876
#> PCTSUPP  0.2254 0.0908 5 0.0520 0.0533 0.0594 0.2665 0.2688 0.2693
#> Call:
#> PCorMC(object = mc)
#>
#> Squared partial correlations
#> type = "mvn"
#> Test passed
#> Test passed

#> test-betaMC-p-cor-mc-est

#> Call:
#> PCorMC(object = mc)
#>
#> Squared partial correlations
#> type = "mvn"
#>             est     se R  0.05%    0.5%    2.5%  97.5%  99.5% 99.95%
#> NARTIC   0.4874 0.1552 5 0.2215 0.2232 0.2311 0.5788 0.5805 0.5809
#> PCTGRT   0.3757 0.0903 5 0.2430 0.2434 0.2452 0.4480 0.4625 0.4657
#> PCTSUPP  0.2254 0.0714 5 0.1311 0.1312 0.1319 0.2868 0.2980 0.3005
#> Call:
#> PCorMC(object = mc)
#>
#> Squared partial correlations
#> type = "mvn"
#> Test passed
#> Test passed

#> test-betaMC-r-sq-mc-est-mi

#> Call:
#> RSqMC(object = mc)
#>
#> R-squared and adjusted R-squared
#> type = "mvn"
#>         est     se R  0.05%    0.5%    2.5%  97.5%  99.5% 99.95%
#> rsq 0.8045 0.0766 5 0.6717 0.6720 0.6730 0.8409 0.8434 0.8439
#> adj 0.7906 0.0821 5 0.6483 0.6485 0.6496 0.8296 0.8322 0.8328
#> Call:
#> RSqMC(object = mc)
#>
#> R-squared and adjusted R-squared
#> type = "mvn"
#> Test passed
```

```
#> Call:
#> RSqMC(object = mc)
#>
#> R-squared and adjusted R-squared
#> type = "mvn"
#>        est      se R 0.05%    0.5%    2.5%  97.5%  99.5% 99.95%
#> rsq 0.5809 0.0806 5 0.473 0.4744 0.4807 0.6843 0.6938 0.6959
#> adj 0.5714 0.0824 5 0.461 0.4624 0.4689 0.6771 0.6868 0.6890
#> Call:
#> RSqMC(object = mc)
#>
#> R-squared and adjusted R-squared
#> type = "mvn"
#> Test passed

#> test-betaMC-r-sq-mc-est

#> Call:
#> RSqMC(object = mc)
#>
#> R-squared and adjusted R-squared
#> type = "mvn"
#>        est      se R  0.05%    0.5%    2.5%  97.5%  99.5% 99.95%
#> rsq 0.8045 0.0897 5 0.6820 0.6826 0.6851 0.8878 0.8912 0.8919
#> adj 0.7906 0.0961 5 0.6593 0.6599 0.6626 0.8797 0.8834 0.8842
#> Call:
#> RSqMC(object = mc)
#>
#> R-squared and adjusted R-squared
#> type = "mvn"
#> Test passed
#> Call:
#> RSqMC(object = mc)
#>
#> R-squared and adjusted R-squared
#> type = "mvn"
#>        est      se R  0.05%    0.5%    2.5%  97.5%  99.5% 99.95%
#> rsq 0.5809 0.1313 5 0.3802 0.3808 0.3834 0.6645 0.6684 0.6693
#> adj 0.5714 0.1343 5 0.3661 0.3667 0.3694 0.6569 0.6609 0.6618
#> Call:
#> RSqMC(object = mc)
#>
#> R-squared and adjusted R-squared
#> type = "mvn"
#> Test passed

#> test-betaMC-s-cor-mc-est-mi
```

```
#> Call:
#> SCorMC(object = mc)
#>
#> Semipartial correlations
#> type = "mvn"
#>           est     se R  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC  0.4312 0.0475 5 0.3725 0.3737 0.3792 0.4917 0.4931 0.4935
#> PCTGRT  0.3430 0.1081 5 0.1980 0.1989 0.2031 0.4610 0.4699 0.4719
#> PCTSUPP 0.2385 0.0504 5 0.1627 0.1638 0.1688 0.2905 0.2926 0.2931
#> Call:
#> SCorMC(object = mc)
#>
#> Semipartial correlations
#> type = "mvn"
#> Test passed
#> Test passed

#> test-betaMC-s-cor-mc-est

#> Call:
#> SCorMC(object = mc)
#>
#> Semipartial correlations
#> type = "mvn"
#>           est     se R  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC  0.4312 0.0311 5 0.3892 0.3892 0.3896 0.4595 0.4616 0.4620
#> PCTGRT  0.3430 0.0414 5 0.3097 0.3100 0.3113 0.4086 0.4109 0.4114
#> PCTSUPP 0.2385 0.0573 5 0.1608 0.1613 0.1635 0.2891 0.2904 0.2907
#> Call:
#> SCorMC(object = mc)
#>
#> Semipartial correlations
#> type = "mvn"
#> Test passed
#> Test passed

#> test-zzz-coverage

#>              beta1     beta2     beta3 sigmasq   sigmax1x1   sigmax2x1   sigmax3x1
#> sigmaysq   909.1981  257.2976  276.0367       1 0.007091036 0.03637752 0.01896371
#> sigmayx1  3507.1691  471.2058  510.5430       0 0.084208291 0.21599726 0.11260003
#> sigmayx2   471.2058  333.2295  150.9121       0 0.000000000 0.08420829 0.00000000
#> sigmayx3   510.5430  150.9121  554.4386       0 0.000000000 0.00000000 0.08420829
#> sigmax1x1    0.0000    0.0000    0.0000       0 1.000000000 0.00000000 0.00000000
#> sigmax2x1    0.0000    0.0000    0.0000       0 0.000000000 1.00000000 0.00000000
#> sigmax3x1    0.0000    0.0000    0.0000       0 0.000000000 0.00000000 1.00000000
#> sigmax2x2    0.0000    0.0000    0.0000       0 0.000000000 0.00000000 0.00000000
```

```
#> sigmax3x2    0.0000    0.0000    0.0000         0 0.000000000 0.00000000 0.00000000
#> sigmax3x3    0.0000    0.0000    0.0000         0 0.000000000 0.00000000 0.00000000
#>             sigmax2x2 sigmax3x2  sigmax3x3
#> sigmaysq  0.04665482 0.0486426 0.01267877
#> sigmayx1  0.00000000 0.0000000 0.00000000
#> sigmayx2  0.21599726 0.1126000 0.00000000
#> sigmayx3  0.00000000 0.2159973 0.11260003
#> sigmax1x1 0.00000000 0.0000000 0.00000000
#> sigmax2x1 0.00000000 0.0000000 0.00000000
#> sigmax3x1 0.00000000 0.0000000 0.00000000
#> sigmax2x2 1.00000000 0.0000000 0.00000000
#> sigmax3x2 0.00000000 1.0000000 0.00000000
#> sigmax3x3 0.00000000 0.0000000 1.00000000
#>                beta1    beta2    beta3 sigmasq
#> sigmaysq   909.1981 257.2976 276.0367       1
#> sigmayx1  3507.1691 471.2058 510.5430       0
#> sigmayx2   471.2058 333.2295 150.9121       0
#> sigmayx3   510.5430 150.9121 554.4386       0
#> sigmax1x1    0.0000    0.0000    0.0000       0
#> sigmax2x1    0.0000    0.0000    0.0000       0
#> sigmax3x1    0.0000    0.0000    0.0000       0
#> sigmax2x2    0.0000    0.0000    0.0000       0
#> sigmax3x2    0.0000    0.0000    0.0000       0
#> sigmax3x3    0.0000    0.0000    0.0000       0
#>                beta1    beta2    beta3       rsq   sigmax1x1  sigmax2x1
#> sigmaysq   909.1981 257.2976 276.0367 -126.0843 0.007091036 0.03637752
#> sigmayx1  3507.1691 471.2058 510.5430    0.0000 0.084208291 0.21599726
#> sigmayx2   471.2058 333.2295 150.9121    0.0000 0.000000000 0.08420829
#> sigmayx3   510.5430 150.9121 554.4386    0.0000 0.000000000 0.00000000
#> sigmax1x1    0.0000    0.0000    0.0000    0.0000 1.000000000 0.00000000
#> sigmax2x1    0.0000    0.0000    0.0000    0.0000 0.000000000 1.00000000
#> sigmax3x1    0.0000    0.0000    0.0000    0.0000 0.000000000 0.00000000
#> sigmax2x2    0.0000    0.0000    0.0000    0.0000 0.000000000 0.00000000
#> sigmax3x2    0.0000    0.0000    0.0000    0.0000 0.000000000 0.00000000
#> sigmax3x3    0.0000    0.0000    0.0000    0.0000 0.000000000 0.00000000
#>             sigmax3x1  sigmax2x2 sigmax3x2  sigmax3x3
#> sigmaysq  0.01896371 0.04665482 0.0486426 0.01267877
#> sigmayx1  0.11260003 0.00000000 0.0000000 0.00000000
#> sigmayx2  0.00000000 0.21599726 0.1126000 0.00000000
#> sigmayx3  0.08420829 0.00000000 0.2159973 0.11260003
#> sigmax1x1 0.00000000 0.00000000 0.0000000 0.00000000
#> sigmax2x1 0.00000000 0.00000000 0.0000000 0.00000000
#> sigmax3x1 1.00000000 0.00000000 0.0000000 0.00000000
#> sigmax2x2 0.00000000 1.00000000 0.0000000 0.00000000
#> sigmax3x2 0.00000000 0.00000000 1.0000000 0.00000000
#> sigmax3x3 0.00000000 0.00000000 0.0000000 1.00000000
```

```
#>                beta1    beta2    beta3      rsq
#> sigmaysq    909.1981 257.2976 276.0367 -126.0843
#> sigmayx1   3507.1691 471.2058 510.5430    0.0000
#> sigmayx2    471.2058 333.2295 150.9121    0.0000
#> sigmayx3    510.5430 150.9121 554.4386    0.0000
#> sigmax1x1     0.0000   0.0000   0.0000    0.0000
#> sigmax2x1     0.0000   0.0000   0.0000    0.0000
#> sigmax3x1     0.0000   0.0000   0.0000    0.0000
#> sigmax2x2     0.0000   0.0000   0.0000    0.0000
#> sigmax3x2     0.0000   0.0000   0.0000    0.0000
#> sigmax3x3     0.0000   0.0000   0.0000    0.0000
#> [[1]]
#> [[1]][[1]]
#> [[1]][[1]]$value
#> [[1]][[1]]$value[[1]]
#>           2.5 %    97.5 %
#> NARTIC 0.7302652 0.8461801
#>
#>
#> [[1]][[1]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[2]]
#> [[1]][[2]]$value
#> [[1]][[2]]$value[[1]]
#>          2.5 %    97.5 %
#> NARTIC 0.640227 0.7582161
#>
#>
#> [[1]][[2]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[3]]
#> [[1]][[3]]$value
#> [[1]][[3]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[3]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[4]]
#> [[1]][[4]]$value
```

```
#> [[1]][[4]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[4]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[5]]
#> [[1]][[5]]$value
#> [[1]][[5]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[5]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[6]]
#> [[1]][[6]]$value
#> [[1]][[6]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[6]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[7]]
#> [[1]][[7]]$value
#> [[1]][[7]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[7]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[8]]
#> [[1]][[8]]$value
#> [[1]][[8]]$value[[1]]
#> [[1]][[8]]$value[[1]]$coef
#> [1] 0.4719623 0.4979797
#>
#> [[1]][[8]]$value[[1]]$sigmasq
```

```
#> [1] 0.5270782
#>
#> [[1]][[8]]$value[[1]]$vechsigmacapx
#> [1] 1.000000e+00 2.496804e-16 1.000000e+00
#>
#> [[1]][[8]]$value[[1]]$sigmacapx
#>               [,1]          [,2]
#> [1,] 1.000000e+00 2.496804e-16
#> [2,] 2.496804e-16 1.000000e+00
#>
#> [[1]][[8]]$value[[1]]$sigmaysq
#> [1] 0.9978104
#>
#> [[1]][[8]]$value[[1]]$sigmayx
#> [1] 0.4719623 0.4979797
#>
#> [[1]][[8]]$value[[1]]$sigmacap
#>             [,1]          [,2]          [,3]
#> [1,] 0.9978104 4.719623e-01 4.979797e-01
#> [2,] 0.4719623 1.000000e+00 2.496804e-16
#> [3,] 0.4979797 2.496804e-16 1.000000e+00
#>
#> [[1]][[8]]$value[[1]]$pd
#> [1] TRUE
#>
#>
#>
#> [[1]][[8]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[9]]
#> [[1]][[9]]$value
#> [[1]][[9]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[9]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[10]]
#> [[1]][[10]]$value
#> [[1]][[10]]$value[[1]]
#>           beta1 beta2 rsq sigmax1x1 sigmax2x1 sigmax2x2
```

```
#> sigmaysq      1      1   -2     0.25      0.5      0.25
#> sigmayx1      1      0    0     0.50      0.5      0.00
#> sigmayx2      0      1    0     0.00      0.5      0.50
#> sigmax1x1     0      0    0     1.00      0.0      0.00
#> sigmax2x1     0      0    0     0.00      1.0      0.00
#> sigmax2x2     0      0    0     0.00      0.0      1.00
#>
#>
#> [[1]][[10]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[11]]
#> [[1]][[11]]$value
#> [[1]][[11]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[11]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[12]]
#> [[1]][[12]]$value
#> [[1]][[12]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[12]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[13]]
#> [[1]][[13]]$value
#> [[1]][[13]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[13]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[14]]
#> [[1]][[14]]$value
#> [[1]][[14]]$value[[1]]
#> [1] TRUE
```

16

```
#> 
#> 
#> [[1]][[14]]$visible
#> [1] TRUE
#> 
#> 
#> [[1]][[15]]
#> [[1]][[15]]$value
#> [[1]][[15]]$value[[1]]
#> [1] TRUE
#> 
#> 
#> [[1]][[15]]$visible
#> [1] TRUE
#> 
#> 
#> [[1]][[16]]
#> [[1]][[16]]$value
#> [[1]][[16]]$value[[1]]
#> [1] TRUE
#> 
#> 
#> [[1]][[16]]$visible
#> [1] TRUE
#> 
#> 
#> [[1]][[17]]
#> [[1]][[17]]$value
#> [[1]][[17]]$value[[1]]
#> [1] FALSE
#> 
#> 
#> [[1]][[17]]$visible
#> [1] TRUE
```

Environment

```r
ls()

#> [1] "nas1982"  "root"     "tex_file"
```

Class

```
#> [[1]]
#> [1] "data.frame"
#>
#> [[2]]
#> [1] "root_criterion"
#>
#> [[3]]
#> [1] "character"
```

References

R Core Team. (2023). *R: A language and environment for statistical computing.* R Foundation for
      Statistical Computing. Vienna, Austria. https://www.R-project.org/