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 0.0628 5 0.4295 0.4301 0.4330 0.5718 0.5732 0.5735
#> PCTGRT  0.3915 0.0542 5 0.2174 0.2183 0.2224 0.3452 0.3457 0.3459
#> PCTSUPP 0.2632 0.0715 5 0.2678 0.2687 0.2727 0.4496 0.4589 0.4610
#> 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 0.0402 5 0.7083 0.7085 0.7093 0.8037 0.8092 0.8105
#> 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 0.0755 5 0.3197 0.3216 0.3301 0.5142 0.5181 0.5189
#> PCTGRT  0.3915 0.0731 5 0.3419 0.3433 0.3497 0.5229 0.5243 0.5246
#> PCTSUPP 0.2632 0.0506 5 0.2026 0.2028 0.2033 0.3170 0.3195 0.3201
#> 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 0.0402 5 0.7083 0.7085 0.7093 0.8037 0.8092 0.8105
#> 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 0.0642 5 0.1160 0.1163 0.1175 0.2613 0.2647 0.2655
#> PCTGRT  0.1177 0.0222 5 0.0312 0.0315 0.0331 0.0848 0.0851 0.0852
#> PCTSUPP 0.0569 0.0461 5 0.0501 0.0504 0.0516 0.1608 0.1674 0.1689
#> 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 0.0505 5 0.0643 0.0654 0.0703 0.1884 0.1890 0.1891
#> PCTGRT  0.1177 0.0624 5 0.0787 0.0789 0.0798 0.2226 0.2262 0.2270
#> PCTSUPP 0.0569 0.0220 5 0.0287 0.0288 0.0293 0.0818 0.0840 0.0845
#> 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 0.0751 5  0.1232  0.1232  0.1234 0.2851 0.2876 0.2881
#> NARTIC-PCTSUPP 0.2319 0.1264 5 -0.0315 -0.0288 -0.0167 0.2978 0.3042 0.3057
#> PCTGRT-PCTSUPP 0.1282 0.1179 5 -0.2436 -0.2404 -0.2258 0.0469 0.0486 0.0490
#> 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 0.1322 5 -0.2048 -0.2018 -0.1884 0.1259 0.1285 0.1291
#> NARTIC-PCTSUPP 0.2319 0.1046 5  0.0531  0.0547  0.0614 0.3108 0.3153 0.3163
#> PCTGRT-PCTSUPP 0.1282 0.1075 5  0.0218  0.0244  0.0356 0.2937 0.2968 0.2975
#> Call:
#> DiffBetaMC(object = mc)
```

```
#>
#> 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
#> Test passed
```

```
#> 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] 0.4811796 0.5247135
#>
#> $sigmasq
#> [1] 0.5564911
#>
#> $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] 1.063349
#>
#> $sigmayx
#> [1] 0.4811796 0.5247135
#>
#> $sigmacap
#>           [,1]         [,2]         [,3]
#> [1,] 1.0633492 4.811796e-01 5.247135e-01
#> [2,] 0.4811796 1.000000e+00 2.496804e-16
#> [3,] 0.5247135 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] 0.4971297 0.5092979
#>
#> $sigmasq
#> [1] 0.5110473
#>
```

```
#> $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] 1.01757
#>
#> $sigmayx
#> [1] 0.4971297 0.5092979
#>
#> $sigmacap
#>           [,1]         [,2]         [,3]
#> [1,] 1.0175696 4.971297e-01 5.092979e-01
#> [2,] 0.4971297 1.000000e+00 2.496804e-16
#> [3,] 0.5092979 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] 0.4948543 0.5180275
#>
#> $sigmasq
#> [1] 0.5517818
#>
#> $vechsigmacapx
#> [1]  0.94325321 -0.01001239  1.00981987
#>
#> $sigmacapx
#>             [,1]        [,2]
#> [1,]  0.94325321 -0.01001239
#> [2,] -0.01001239  1.00981987
#>
#> $sigmaysq
#> [1] 1.048621
#>
#> $sigmayx
#> [1] 0.4615862 0.5181598
#>
#> $sigmacap
#>           [,1]        [,2]        [,3]
#> [1,] 1.0486207  0.46158623  0.51815979
```

```
#> [2,] 0.4615862  0.94325321 -0.01001239
#> [3,] 0.5181598 -0.01001239  1.00981987
#>
#> $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] 0.4873036 0.5138718
#>
#> $sigmasq
#> [1] 0.519847
#>
#> $vechsigmacapx
#> [1] 0.92654372 0.01659553 1.01126328
#>
#> $sigmacapx
#>            [,1]       [,2]
#> [1,] 0.92654372 0.01659553
#> [2,] 0.01659553 1.01126328
#>
#> $sigmaysq
#> [1] 1.015218
#>
#> $sigmayx
#> [1] 0.4600361 0.5277467
#>
#> $sigmacap
#>           [,1]       [,2]       [,3]
#> [1,] 1.0152184 0.46003610 0.52774671
#> [2,] 0.4600361 0.92654372 0.01659553
#> [3,] 0.5277467 0.01659553 1.01126328
#>
#> $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 0.0928 5 0.3581 0.3590 0.3629 0.5889 0.5960 0.5976
#> PCTGRT  0.3757 0.0786 5 0.1304 0.1317 0.1372 0.3249 0.3275 0.3281
#> PCTSUPP 0.2254 0.0827 5 0.2506 0.2506 0.2508 0.4339 0.4455 0.4481
#> 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 0.1108 5 0.2569 0.2596 0.2717 0.5552 0.5655 0.5679
#> PCTGRT  0.3757 0.0854 5 0.2907 0.2926 0.3007 0.4931 0.4939 0.4941
#> PCTSUPP 0.2254 0.0675 5 0.1258 0.1266 0.1303 0.2982 0.3041 0.3054
#> 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 0.0397 5 0.7445 0.7454 0.7492 0.8477 0.8497 0.8501
#> adj 0.7906 0.0425 5 0.7263 0.7272 0.7313 0.8368 0.8389 0.8394
#> 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 0.0613 5 0.5017 0.5020 0.5031 0.6463 0.6549 0.6569
#> adj 0.5714 0.0627 5 0.4904 0.4907 0.4918 0.6383 0.6471 0.6491
#> 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 0.0375 5 0.7582 0.7591 0.7631 0.8581 0.8607 0.8613
#> adj 0.7906 0.0402 5 0.7409 0.7419 0.7462 0.8480 0.8508 0.8514
#> 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 0.0613 5 0.5017 0.5020 0.5031 0.6463 0.6549 0.6569
#> adj 0.5714 0.0627 5 0.4904 0.4907 0.4918 0.6383 0.6471 0.6491
#> 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 0.0748 5 0.3406 0.3410 0.3427 0.5110 0.5145 0.5153
#> PCTGRT  0.3430 0.0471 5 0.1766 0.1774 0.1813 0.2912 0.2918 0.2919
#> PCTSUPP 0.2385 0.0710 5 0.2239 0.2245 0.2270 0.3994 0.4088 0.4109
#> 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 0.0730 5 0.2536 0.2554 0.2634 0.4341 0.4347 0.4349
#> PCTGRT  0.3430 0.0833 5 0.2806 0.2810 0.2824 0.4715 0.4756 0.4765
#> PCTSUPP 0.2385 0.0476 5 0.1694 0.1697 0.1711 0.2855 0.2898 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 0.7093103 0.8036672
#>
#>
#> [[1]][[1]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[2]]
#> [[1]][[2]]$value
#> [[1]][[2]]$value[[1]]
#>            2.5 %    97.5 %
#> NARTIC 0.7093103 0.8036672
#>
#>
#> [[1]][[2]]$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]]$coef
#> [1] 0.4971297 0.5092979
#>
#> [[1]][[8]]$value[[1]]$sigmasq
```

```
#> [1] 0.5110473
#>
#> [[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] 1.01757
#>
#> [[1]][[8]]$value[[1]]$sigmayx
#> [1] 0.4971297 0.5092979
#>
#> [[1]][[8]]$value[[1]]$sigmacap
#>           [,1]         [,2]         [,3]
#> [1,] 1.0175696 4.971297e-01 5.092979e-01
#> [2,] 0.4971297 1.000000e+00 2.496804e-16
#> [3,] 0.5092979 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
```

```
#>
#>
#> [[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

```
ls()

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

Class

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

References

Pesigan, I. J. A., & Cheung, S. F. (2023). Monte Carlo confidence intervals for the indirect effect with missing data. *Behavior Research Methods*. https://doi.org/10.3758/s13428-023-02114-4

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