

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 0.0726 5 0.4287 0.4293 0.4320 0.5929 0.5942 0.5945
#> PCTGRT  0.3915 0.0572 5 0.2140 0.2144 0.2159 0.3411 0.3420 0.3422
#> PCTSUPP 0.2632 0.0832 5 0.1567 0.1594 0.1712 0.3570 0.3581 0.3584
#> 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 0.0726 5 0.4287 0.4293 0.4320 0.5929 0.5942 0.5945
#> PCTGRT  0.3915 0.0572 5 0.2140 0.2144 0.2159 0.3411 0.3420 0.3422
#> PCTSUPP 0.2632 0.0832 5 0.1567 0.1594 0.1712 0.3570 0.3581 0.3584
#> 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 0.0785 5 0.0760 0.0768 0.0802 0.2711 0.2778 0.2794
#> PCTGRT  0.1177 0.0244 5 0.0270 0.0272 0.0284 0.0853 0.0867 0.0870
#> PCTSUPP 0.0569 0.0351 5 0.0137 0.0144 0.0176 0.1073 0.1107 0.1114
#> 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 0.0785 5 0.0760 0.0768 0.0802 0.2711 0.2778 0.2794
#> PCTGRT  0.1177 0.0244 5 0.0270 0.0272 0.0284 0.0853 0.0867 0.0870
#> PCTSUPP 0.0569 0.0351 5 0.0137 0.0144 0.0176 0.1073 0.1107 0.1114
#> 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 0.1156 5  0.0866  0.0874  0.0909 0.3542 0.3599 0.3612
#> NARTIC-PCTSUPP 0.2319 0.1430 5  0.0880  0.0883  0.0896 0.4089 0.4194 0.4218
#> PCTGRT-PCTSUPP 0.1282 0.0980 5 -0.1308 -0.1297 -0.1249 0.1188 0.1293 0.1316
#> 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 0.1156 5  0.0866  0.0874  0.0909 0.3542 0.3599 0.3612
#> NARTIC-PCTSUPP 0.2319 0.1430 5  0.0880  0.0883  0.0896 0.4089 0.4194 0.4218
#> PCTGRT-PCTSUPP 0.1282 0.0980 5 -0.1308 -0.1297 -0.1249 0.1188 0.1293 0.1316
#> 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
#> 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 6.101989e-17 1.000000e+00
#>
#> $sigmacapx
#>      [,1]      [,2]
#> [1,] 1.000000e+00 6.101989e-17
#> [2,] 6.101989e-17 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 6.101989e-17
#> [3,] 0.5247135 6.101989e-17 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 6.101989e-17 1.000000e+00
#>
#> $sigmacapx
#>           [,1]           [,2]
#> [1,] 1.000000e+00 6.101989e-17
#> [2,] 6.101989e-17 1.000000e+00
#>
#> $sigmayx
#> [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 6.101989e-17
#> [3,] 0.5092979 6.101989e-17 1.000000e+00
#>
#> $pd
#> [1] TRUE

#> test-betaMC-mc-mi

#> MCMC(object = object, mi = mi, R = R, type = "mvn")
#> MCMC(object = object, mi = mi, R = R, type = "adf")
#> MCMC(object = object, mi = mi, R = R, type = "hc0")
#> MCMC(object = object, mi = mi, R = R, type = "hc1")
#> MCMC(object = object, mi = mi, R = R, type = "hc2")
#> MCMC(object = object, mi = mi, R = R, type = "hc3")
#> MCMC(object = object, mi = mi, R = R, type = "hc4")
#> MCMC(object = object, mi = mi, R = R, type = "hc4m")
#> MCMC(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 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
#> 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
#>
#> $sigmayxsq
#> [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 0.1162 5 0.3167 0.3169 0.3175 0.5808 0.5883 0.5900
#> PCTGRT  0.3757 0.0565 5 0.1414 0.1420 0.1446 0.2683 0.2690 0.2692
#> PCTSUPP 0.2254 0.1112 5 0.0644 0.0667 0.0770 0.3578 0.3660 0.3679
#> 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 0.1162 5 0.3167 0.3169 0.3175 0.5808 0.5883 0.5900
#> PCTGRT  0.3757 0.0565 5 0.1414 0.1420 0.1446 0.2683 0.2690 0.2692
#> PCTSUPP 0.2254 0.1112 5 0.0644 0.0667 0.0770 0.3578 0.3660 0.3679
#> 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 0.0307 5 0.7517 0.7526 0.7566 0.8335 0.8357 0.8362
#> adj 0.7906 0.0329 5 0.7340 0.7349 0.7393 0.8216 0.8240 0.8245
#> 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 0.0307 5 0.7517 0.7526 0.7566 0.8335 0.8357 0.8362
#> adj 0.7906 0.0329 5 0.7340 0.7349 0.7393 0.8216 0.8240 0.8245
#> 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 0.0964 5 0.2756 0.2769 0.2825 0.520 0.5270 0.5285
#> PCTGRT  0.3430 0.0525 5 0.1642 0.1650 0.1682 0.292 0.2943 0.2949
#> PCTSUPP 0.2385 0.0786 5 0.1168 0.1189 0.1281 0.327 0.3325 0.3337
#> 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 0.0964 5 0.2756 0.2769 0.2825 0.520 0.5270 0.5285
#> PCTGRT  0.3430 0.0525 5 0.1642 0.1650 0.1682 0.292 0.2943 0.2949
#> PCTSUPP 0.2385 0.0786 5 0.1168 0.1189 0.1281 0.327 0.3325 0.3337
#> 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 6.101989e-17 1.000000e+00
#>
#> [[1]][[8]]$value[[1]]$sigmacapx
#>           [,1]           [,2]
#> [1,] 1.000000e+00 6.101989e-17
#> [2,] 6.101989e-17 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 6.101989e-17
#> [3,] 0.5092979 6.101989e-17 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

```

```

#> 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"
```

Class

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

References

- Pesigan, I. J. A., & Cheung, S. F. (2023). Monte Carlo confidence intervals for the indirect effect with missing data. *Behavior Research Methods*, 56(3), 1678–1696. <https://doi.org/10.3758/s13428-023-02114-4>
- R Core Team. (2024). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. <https://www.R-project.org/>