

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 0.0674 5 0.3876 0.3894 0.3977 0.5556 0.5563 0.5565
#> PCTGRT  0.3915 0.0408 5 0.3275 0.3281 0.3305 0.4283 0.4300 0.4304
#> PCTSUPP 0.2632 0.0646 5 0.1669 0.1678 0.1715 0.3316 0.3365 0.3376
#> 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 0.0718 5 0.5989 0.6003 0.6063 0.7623 0.7627 0.7628
#> 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 0.0905 5 0.3781 0.3797 0.3865 0.6138 0.6216 0.6234
#> PCTGRT  0.3915 0.0902 5 0.2673 0.2677 0.2694 0.4517 0.4526 0.4528
#> PCTSUPP 0.2632 0.0507 5 0.2125 0.2136 0.2184 0.3439 0.3463 0.3469
#> 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 0.0819 5 0.6942 0.6943 0.6947 0.8822 0.8909 0.8928
#> 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 0.0447 5 0.1259 0.1270 0.1321 0.2434 0.2456 0.2461
#> PCTGRT  0.1177 0.0333 5 0.0507 0.0516 0.0555 0.1356 0.1370 0.1373
#> PCTSUPP 0.0569 0.0291 5 0.0096 0.0097 0.0102 0.0757 0.0796 0.0805
#> 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 0.0308 5 0.1115 0.1117 0.1129 0.1882 0.1920 0.1929
#> PCTGRT  0.1177 0.0411 5 0.0824 0.0826 0.0836 0.1784 0.1847 0.1862
#> PCTSUPP 0.0569 0.0126 5 0.0363 0.0364 0.0373 0.0683 0.0693 0.0696
#> 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 0.1701 5  0.0076  0.0081  0.0101 0.4029 0.4211 0.4252
#> NARTIC-PCTSUPP 0.2319 0.1954 5 -0.0196 -0.0142  0.0100 0.4666 0.4684 0.4688
#> PCTGRT-PCTSUPP 0.1282 0.1298 5 -0.0276 -0.0264 -0.0207 0.2626 0.2639 0.2642
#> 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 0.1801 5 -0.1640 -0.1605 -0.1449 0.3023 0.3140 0.3166
#> NARTIC-PCTSUPP 0.2319 0.1363 5 -0.0278 -0.0245 -0.0096 0.3312 0.3388 0.3405
#> PCTGRT-PCTSUPP 0.1282 0.1718 5 -0.1568 -0.1539 -0.1410 0.2930 0.3069 0.3101
#> 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] 0.5201468 0.5059670
#>
#> $sigmasq
#> [1] 0.5152926
#>
#> $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] 1.041848
#>
#> $sigmayx
#> [1] 0.5201468 0.5059670
#>
#> $sigmacap
#>      [,1]      [,2]      [,3]
#> [1,] 1.0418480 5.201468e-01 5.059670e-01
#> [2,] 0.5201468 1.000000e+00 2.496804e-16
#> [3,] 0.5059670 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] 0.4644307 0.4897672
#>
#> $sigmasq
#> [1] 0.5331706
#>

```

```

#> $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
#>
#> $sigmayx
#> [1] 0.9887384
#>
#> $sigmayx
#> [1] 0.4644307 0.4897672
#>
#> $sigmacap
#>           [,1]           [,2]           [,3]
#> [1,] 0.9887384 4.644307e-01 4.897672e-01
#> [2,] 0.4644307 1.000000e+00 2.496804e-16
#> [3,] 0.4897672 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 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] 0.4790946 0.5000247
#>
#> $sigmasq
#> [1] 0.5478169
#>
#> $vechsigmacapx
#> [1] 1.03700884 0.01654671 0.98324369
#>
#> $sigmacapx
#>           [,1]      [,2]
#> [1,] 1.03700884 0.01654671
#> [2,] 0.01654671 0.98324369
#>
#> $sigmaysq
#> [1] 1.039606
#>
#> $sigmayx
#> [1] 0.5050991 0.4995736
#>
#> $sigmacap
#>           [,1]      [,2]      [,3]
#> [1,] 1.0396063 0.50509911 0.49957361

```

```

#> [2,] 0.5050991 1.03700884 0.01654671
#> [3,] 0.4995736 0.01654671 0.98324369
#>
#> $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] 0.4960487 0.4907616
#>
#> $sigmasq
#> [1] 0.5689688
#>
#> $vechsigmacapx
#> [1] 1.068483753 -0.002145986 0.939094918
#>
#> $sigmacapx
#>      [,1]      [,2]
#> [1,] 1.068483753 -0.002145986
#> [2,] -0.002145986 0.939094918
#>
#> $sigmayx
#> [1] 1.057018
#>
#> $sigmayx
#> [1] 0.5289668 0.4598072
#>
#> $sigmacap
#>      [,1]      [,2]      [,3]
#> [1,] 1.0570178 0.528966807 0.459807198
#> [2,] 0.5289668 1.068483753 -0.002145986
#> [3,] 0.4598072 -0.002145986 0.939094918
#>
#> $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 0.0845 5 0.4426 0.4427 0.4429 0.6263 0.6398 0.6428
#> PCTGRT  0.3757 0.1257 5 0.2725 0.2727 0.2736 0.5604 0.5694 0.5715
#> PCTSUPP 0.2254 0.1208 5 0.0615 0.0640 0.0750 0.3830 0.3970 0.4001
#> 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 0.0723 5 0.3237 0.3239 0.3245 0.4885 0.4930 0.4940
#> PCTGRT  0.3757 0.0861 5 0.1896 0.1915 0.2001 0.4149 0.4196 0.4207
#> PCTSUPP 0.2254 0.1304 5 0.0493 0.0493 0.0494 0.3179 0.3222 0.3231
#> 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 0.0626 5 0.7039 0.7043 0.7062 0.8559 0.8625 0.8639
#> adj 0.7906 0.0671 5 0.6828 0.6832 0.6853 0.8456 0.8526 0.8542
#> 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 0.1591 5 0.2914 0.2969 0.3213 0.6768 0.6782 0.6785
#> adj 0.5714 0.1627 5 0.2753 0.2809 0.3059 0.6695 0.6709 0.6712
#> 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 0.0692 5 0.6969 0.6976 0.7006 0.8688 0.8744 0.8757
#> adj 0.7906 0.0742 5 0.6753 0.6760 0.6792 0.8594 0.8655 0.8668
#> 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 0.1007 5 0.4314 0.4318 0.4337 0.6580 0.6617 0.6625
#> adj 0.5714 0.1030 5 0.4185 0.4189 0.4209 0.6503 0.6540 0.6549
#> 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 0.0627 5 0.4069 0.4085 0.4152 0.5662 0.5678 0.5682
#> PCTGRT  0.3430 0.0730 5 0.2776 0.2776 0.2777 0.4433 0.4506 0.4522
#> PCTSUPP 0.2385 0.0710 5 0.1407 0.1416 0.1459 0.3235 0.3295 0.3309
#> 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 0.1410 5 0.2721 0.2723 0.2731 0.5913 0.6009 0.6030
#> PCTGRT  0.3430 0.1018 5 0.2341 0.2355 0.2415 0.4950 0.5079 0.5108
#> PCTSUPP 0.2385 0.0919 5 0.1437 0.1441 0.1455 0.3626 0.3719 0.3740
#> 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 0.6063032 0.7623279
#>
#>
#> [[1]][[1]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[2]]
#> [[1]][[2]]$value
#> [[1]][[2]]$value[[1]]
#>          2.5 %    97.5 %
#> NARTIC 0.6947237 0.8821525
#>
#>
#> [[1]][[2]]$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]]$coef
#> [1] 0.4644307 0.4897672
#>
#> [[1]][[8]]$value[[1]]$sigmasq

```

```

#> [1] 0.5331706
#>
#> [[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] 0.9887384
#>
#> [[1]][[8]]$value[[1]]$sigmayx
#> [1] 0.4644307 0.4897672
#>
#> [[1]][[8]]$value[[1]]$sigmacap
#>           [,1]           [,2]           [,3]
#> [1,] 0.9887384 4.644307e-01 4.897672e-01
#> [2,] 0.4644307 1.000000e+00 2.496804e-16
#> [3,] 0.4897672 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

```

```

#> 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/>