betaSandwich: Internal Tests

Ivan Jacob Agaloos Pesigan

Tests

```
#> test-betaSandwich-beta-sandwich-adf
#> Test passed
#> Test passed

#> test-betaSandwich-beta-sandwich-hc

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

#> test-betaSandwich-beta-sandwich-methods

#> Call:
#> BetaHC(object = object)
#>
#> Standardized regression slopes with HC3 standard errors:
#>            est     se      t df      p   0.05%   0.5%   2.5%  97.5%  99.5%
#> NARTIC  0.4951 0.0786 6.3025 42 0.0000  0.2172 0.2832 0.3366 0.6537 0.7071
#> PCTGRT  0.3915 0.0818 4.7831 42 0.0000  0.1019 0.1707 0.2263 0.5567 0.6123
#> PCTSUPP 0.2632 0.0855 3.0786 42 0.0037 -0.0393 0.0325 0.0907 0.4358 0.4940
#>         99.95%
#> NARTIC  0.7731
#> PCTGRT  0.6810
#> PCTSUPP 0.5658
#> Call:
#> BetaHC(object = object)
#>
#> Standardized regression slopes with HC3 standard errors:
#>            est     se      t df      p   0.05%   0.5%   2.5%  97.5%  99.5%
#> NARTIC  0.4951 0.0786 6.3025 42 0.0000  0.2172 0.2832 0.3366 0.6537 0.7071
```

```
#> PCTGRT  0.3915 0.0818 4.7831 42 0.0000  0.1019 0.1707 0.2263 0.5567 0.6123
#> PCTSUPP 0.2632 0.0855 3.0786 42 0.0037 -0.0393 0.0325 0.0907 0.4358 0.4940
#>         99.95%
#> NARTIC  0.7731
#> PCTGRT  0.6810
#> PCTSUPP 0.5658
#> Call:
#> BetaN(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>            est     se      t df     p   0.05%   0.5%   2.5%  97.5%  99.5%
#> NARTIC  0.4951 0.0759 6.5272 42 0.000  0.2268 0.2905 0.3421 0.6482 0.6998
#> PCTGRT  0.3915 0.0770 5.0824 42 0.000  0.1190 0.1837 0.2360 0.5469 0.5993
#> PCTSUPP 0.2632 0.0747 3.5224 42 0.001 -0.0011 0.0616 0.1124 0.4141 0.4649
#>         99.95%
#> NARTIC  0.7635
#> PCTGRT  0.6640
#> PCTSUPP 0.5276
#> Call:
#> BetaN(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>            est     se      t df     p   0.05%   0.5%   2.5%  97.5%  99.5%
#> NARTIC  0.4951 0.0759 6.5272 42 0.000  0.2268 0.2905 0.3421 0.6482 0.6998
#> PCTGRT  0.3915 0.0770 5.0824 42 0.000  0.1190 0.1837 0.2360 0.5469 0.5993
#> PCTSUPP 0.2632 0.0747 3.5224 42 0.001 -0.0011 0.0616 0.1124 0.4141 0.4649
#>         99.95%
#> NARTIC  0.7635
#> PCTGRT  0.6640
#> PCTSUPP 0.5276
#> Call:
#> BetaADF(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>            est     se      t df      p   0.05%   0.5%   2.5%  97.5%  99.5%
#> NARTIC  0.4951 0.0674 7.3490 42 0.0000  0.2568 0.3134 0.3592 0.6311 0.6769
#> PCTGRT  0.3915 0.0710 5.5164 42 0.0000  0.1404 0.2000 0.2483 0.5347 0.5830
#> PCTSUPP 0.2632 0.0769 3.4231 42 0.0014 -0.0088 0.0558 0.1081 0.4184 0.4707
#>         99.95%
#> NARTIC  0.7335
#> PCTGRT  0.6426
#> PCTSUPP 0.5353
#> Call:
#> BetaADF(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
```

```
#>            est     se      t df      p   0.05%   0.5%   2.5%  97.5%  99.5%
#> NARTIC  0.4951 0.0674 7.3490 42 0.0000  0.2568 0.3134 0.3592 0.6311 0.6769
#> PCTGRT  0.3915 0.0710 5.5164 42 0.0000  0.1404 0.2000 0.2483 0.5347 0.5830
#> PCTSUPP 0.2632 0.0769 3.4231 42 0.0014 -0.0088 0.0558 0.1081 0.4184 0.4707
#>         99.95%
#> NARTIC  0.7335
#> PCTGRT  0.6426
#> PCTSUPP 0.5353
#> Call:
#> BetaHC(object = object)
#>
#> Standardized regression slopes with HC3 standard errors:
#>           est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC 0.7622 0.0645 11.8222 44 0 0.5349 0.5886 0.6322 0.8921 0.9357 0.9895
#> Call:
#> BetaHC(object = object)
#>
#> Standardized regression slopes with HC3 standard errors:
#>           est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC 0.7622 0.0645 11.8222 44 0 0.5349 0.5886 0.6322 0.8921 0.9357 0.9895
#> Call:
#> BetaN(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>           est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC 0.7622 0.0618 12.3341 44 0 0.5443 0.5958 0.6376 0.8867 0.9285   0.98
#> Call:
#> BetaN(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>           est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC 0.7622 0.0618 12.3341 44 0 0.5443 0.5958 0.6376 0.8867 0.9285   0.98
#> Call:
#> BetaADF(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>           est     se      t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC 0.7622 0.0604 12.625 44 0 0.5493 0.5996 0.6405 0.8838 0.9247  0.975
#> Call:
#> BetaADF(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>           est     se      t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC 0.7622 0.0604 12.625 44 0 0.5493 0.5996 0.6405 0.8838 0.9247  0.975
#> Test passed
```

```
#> test-betaSandwich-beta-sandwich-mvn

#> Test passed
#> Test passed

#> test-betaSandwich-diff-beta-sandwich-methods

#> Call:
#> DiffBetaSandwich(object = BetaN(object))
#>
#> Difference between standardized regression coefficients with MVN standard errors:
#>                    est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1357 0.7640 0.4449 -0.3428 -0.2458 -0.1623 0.3696
#> NARTIC-PCTSUPP 0.2319 0.1252 1.8524 0.0640 -0.1800 -0.0906 -0.0135 0.4773
#> PCTGRT-PCTSUPP 0.1282 0.1227 1.0451 0.2960 -0.2755 -0.1878 -0.1123 0.3688
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4531 0.5501
#> NARTIC-PCTSUPP 0.5544 0.6438
#> PCTGRT-PCTSUPP 0.4443 0.5320
#> Call:
#> DiffBetaSandwich(object = BetaN(object))
#>
#> Difference between standardized regression coefficients with MVN standard errors:
#>                    est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1357 0.7640 0.4449 -0.3428 -0.2458 -0.1623 0.3696
#> NARTIC-PCTSUPP 0.2319 0.1252 1.8524 0.0640 -0.1800 -0.0906 -0.0135 0.4773
#> PCTGRT-PCTSUPP 0.1282 0.1227 1.0451 0.2960 -0.2755 -0.1878 -0.1123 0.3688
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4531 0.5501
#> NARTIC-PCTSUPP 0.5544 0.6438
#> PCTGRT-PCTSUPP 0.4443 0.5320
#> Call:
#> DiffBetaSandwich(object = BetaADF(object))
#>
#> Difference between standardized regression coefficients with MVN standard errors:
#>                    est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1212 0.8555 0.3923 -0.2950 -0.2084 -0.1338 0.3411
#> NARTIC-PCTSUPP 0.2319 0.1181 1.9642 0.0495 -0.1566 -0.0722  0.0005 0.4633
#> PCTGRT-PCTSUPP 0.1282 0.1215 1.0555 0.2912 -0.2716 -0.1847 -0.1099 0.3664
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4158 0.5024
#> NARTIC-PCTSUPP 0.5360 0.6204
#> PCTGRT-PCTSUPP 0.4412 0.5281
#> Call:
#> DiffBetaSandwich(object = BetaADF(object))
#>
#> Difference between standardized regression coefficients with MVN standard errors:
```

```
#>                    est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1212 0.8555 0.3923 -0.2950 -0.2084 -0.1338 0.3411
#> NARTIC-PCTSUPP 0.2319 0.1181 1.9642 0.0495 -0.1566 -0.0722  0.0005 0.4633
#> PCTGRT-PCTSUPP 0.1282 0.1215 1.0555 0.2912 -0.2716 -0.1847 -0.1099 0.3664
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4158 0.5024
#> NARTIC-PCTSUPP 0.5360 0.6204
#> PCTGRT-PCTSUPP 0.4412 0.5281
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc0"))
#>
#> Difference between standardized regression coefficients with HC0 standard errors:
#>                    est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1201 0.8629 0.3882 -0.2916 -0.2058 -0.1318 0.3391
#> NARTIC-PCTSUPP 0.2319 0.1169 1.9840 0.0473 -0.1527 -0.0692  0.0028 0.4610
#> PCTGRT-PCTSUPP 0.1282 0.1201 1.0674 0.2858 -0.2671 -0.1812 -0.1072 0.3637
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4131 0.4989
#> NARTIC-PCTSUPP 0.5330 0.6165
#> PCTGRT-PCTSUPP 0.4377 0.5236
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc0"))
#>
#> Difference between standardized regression coefficients with HC0 standard errors:
#>                    est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1201 0.8629 0.3882 -0.2916 -0.2058 -0.1318 0.3391
#> NARTIC-PCTSUPP 0.2319 0.1169 1.9840 0.0473 -0.1527 -0.0692  0.0028 0.4610
#> PCTGRT-PCTSUPP 0.1282 0.1201 1.0674 0.2858 -0.2671 -0.1812 -0.1072 0.3637
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4131 0.4989
#> NARTIC-PCTSUPP 0.5330 0.6165
#> PCTGRT-PCTSUPP 0.4377 0.5236
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc1"))
#>
#> Difference between standardized regression coefficients with HC1 standard errors:
#>                    est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1257 0.8245 0.4097 -0.3100 -0.2202 -0.1427 0.3501
#> NARTIC-PCTSUPP 0.2319 0.1223 1.8958 0.0580 -0.1706 -0.0832 -0.0078 0.4716
#> PCTGRT-PCTSUPP 0.1282 0.1257 1.0199 0.3078 -0.2855 -0.1956 -0.1182 0.3747
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4275 0.5173
#> NARTIC-PCTSUPP 0.5470 0.6344
#> PCTGRT-PCTSUPP 0.4521 0.5420
#> Call:
```

```
#> DiffBetaSandwich(object = BetaHC(object, type = "hc1"))
#>
#> Difference between standardized regression coefficients with HC1 standard errors:
#>                     est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1257 0.8245 0.4097 -0.3100 -0.2202 -0.1427 0.3501
#> NARTIC-PCTSUPP 0.2319 0.1223 1.8958 0.0580 -0.1706 -0.0832 -0.0078 0.4716
#> PCTGRT-PCTSUPP 0.1282 0.1257 1.0199 0.3078 -0.2855 -0.1956 -0.1182 0.3747
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4275 0.5173
#> NARTIC-PCTSUPP 0.5470 0.6344
#> PCTGRT-PCTSUPP 0.4521 0.5420
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc2"))
#>
#> Difference between standardized regression coefficients with HC2 standard errors:
#>                     est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1302 0.7960 0.4260 -0.3248 -0.2318 -0.1516 0.3589
#> NARTIC-PCTSUPP 0.2319 0.1240 1.8704 0.0614 -0.1761 -0.0875 -0.0111 0.4749
#> PCTGRT-PCTSUPP 0.1282 0.1284 0.9990 0.3178 -0.2942 -0.2024 -0.1234 0.3798
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4391 0.5321
#> NARTIC-PCTSUPP 0.5513 0.6399
#> PCTGRT-PCTSUPP 0.4589 0.5506
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc2"))
#>
#> Difference between standardized regression coefficients with HC2 standard errors:
#>                     est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1302 0.7960 0.4260 -0.3248 -0.2318 -0.1516 0.3589
#> NARTIC-PCTSUPP 0.2319 0.1240 1.8704 0.0614 -0.1761 -0.0875 -0.0111 0.4749
#> PCTGRT-PCTSUPP 0.1282 0.1284 0.9990 0.3178 -0.2942 -0.2024 -0.1234 0.3798
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4391 0.5321
#> NARTIC-PCTSUPP 0.5513 0.6399
#> PCTGRT-PCTSUPP 0.4589 0.5506
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc3"))
#>
#> Difference between standardized regression coefficients with HC3 standard errors:
#>                     est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1417 0.7316 0.4644 -0.3626 -0.2613 -0.1741 0.3814
#> NARTIC-PCTSUPP 0.2319 0.1318 1.7595 0.0785 -0.2018 -0.1076 -0.0264 0.4902
#> PCTGRT-PCTSUPP 0.1282 0.1375 0.9329 0.3509 -0.3241 -0.2259 -0.1412 0.3977
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4686 0.5699
```

```
#> NARTIC-PCTSUPP 0.5714 0.6656
#> PCTGRT-PCTSUPP 0.4823 0.5806
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc3"))
#>
#> Difference between standardized regression coefficients with HC3 standard errors:
#>                   est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1417 0.7316 0.4644 -0.3626 -0.2613 -0.1741 0.3814
#> NARTIC-PCTSUPP 0.2319 0.1318 1.7595 0.0785 -0.2018 -0.1076 -0.0264 0.4902
#> PCTGRT-PCTSUPP 0.1282 0.1375 0.9329 0.3509 -0.3241 -0.2259 -0.1412 0.3977
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4686 0.5699
#> NARTIC-PCTSUPP 0.5714 0.6656
#> PCTGRT-PCTSUPP 0.4823 0.5806
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc4"))
#>
#> Difference between standardized regression coefficients with HC4 standard errors:
#>                   est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1452 0.7138 0.4753 -0.3742 -0.2704 -0.1809 0.3883
#> NARTIC-PCTSUPP 0.2319 0.1296 1.7892 0.0736 -0.1946 -0.1020 -0.0221 0.4859
#> PCTGRT-PCTSUPP 0.1282 0.1361 0.9420 0.3462 -0.3197 -0.2224 -0.1386 0.3951
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4777 0.5815
#> NARTIC-PCTSUPP 0.5658 0.6584
#> PCTGRT-PCTSUPP 0.4789 0.5762
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc4"))
#>
#> Difference between standardized regression coefficients with HC4 standard errors:
#>                   est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1452 0.7138 0.4753 -0.3742 -0.2704 -0.1809 0.3883
#> NARTIC-PCTSUPP 0.2319 0.1296 1.7892 0.0736 -0.1946 -0.1020 -0.0221 0.4859
#> PCTGRT-PCTSUPP 0.1282 0.1361 0.9420 0.3462 -0.3197 -0.2224 -0.1386 0.3951
#>                 99.5% 99.95%
#> NARTIC-PCTGRT  0.4777 0.5815
#> NARTIC-PCTSUPP 0.5658 0.6584
#> PCTGRT-PCTSUPP 0.4789 0.5762
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc4m"))
#>
#> Difference between standardized regression coefficients with HC4M standard errors:
#>                   est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1465 0.7077 0.4791 -0.3783 -0.2736 -0.1834 0.3907
#> NARTIC-PCTSUPP 0.2319 0.1338 1.7331 0.0831 -0.2084 -0.1128 -0.0304 0.4941
#> PCTGRT-PCTSUPP 0.1282 0.1406 0.9123 0.3616 -0.3343 -0.2338 -0.1473 0.4037
```

```
#>                  99.5% 99.95%
#> NARTIC-PCTGRT  0.4809 0.5856
#> NARTIC-PCTSUPP 0.5766 0.6722
#> PCTGRT-PCTSUPP 0.4903 0.5908
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc4m"))
#>
#> Difference between standardized regression coefficients with HC4M standard errors:
#>                   est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1465 0.7077 0.4791 -0.3783 -0.2736 -0.1834 0.3907
#> NARTIC-PCTSUPP 0.2319 0.1338 1.7331 0.0831 -0.2084 -0.1128 -0.0304 0.4941
#> PCTGRT-PCTSUPP 0.1282 0.1406 0.9123 0.3616 -0.3343 -0.2338 -0.1473 0.4037
#>                  99.5% 99.95%
#> NARTIC-PCTGRT  0.4809 0.5856
#> NARTIC-PCTSUPP 0.5766 0.6722
#> PCTGRT-PCTSUPP 0.4903 0.5908
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc5"))
#>
#> Difference between standardized regression coefficients with HC5 standard errors:
#>                   est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1312 0.7899 0.4296 -0.3282 -0.2344 -0.1536 0.3609
#> NARTIC-PCTSUPP 0.2319 0.1227 1.8906 0.0587 -0.1717 -0.0841 -0.0085 0.4723
#> PCTGRT-PCTSUPP 0.1282 0.1274 1.0067 0.3141 -0.2909 -0.1999 -0.1214 0.3779
#>                  99.5% 99.95%
#> NARTIC-PCTGRT  0.4417 0.5355
#> NARTIC-PCTSUPP 0.5478 0.6355
#> PCTGRT-PCTSUPP 0.4564 0.5474
#> Call:
#> DiffBetaSandwich(object = BetaHC(object, type = "hc5"))
#>
#> Difference between standardized regression coefficients with HC5 standard errors:
#>                   est     se      z      p   0.05%    0.5%    2.5%  97.5%
#> NARTIC-PCTGRT  0.1037 0.1312 0.7899 0.4296 -0.3282 -0.2344 -0.1536 0.3609
#> NARTIC-PCTSUPP 0.2319 0.1227 1.8906 0.0587 -0.1717 -0.0841 -0.0085 0.4723
#> PCTGRT-PCTSUPP 0.1282 0.1274 1.0067 0.3141 -0.2909 -0.1999 -0.1214 0.3779
#>                  99.5% 99.95%
#> NARTIC-PCTGRT  0.4417 0.5355
#> NARTIC-PCTSUPP 0.5478 0.6355
#> PCTGRT-PCTSUPP 0.4564 0.5474
#> Test passed
```

#> test-betaSandwich-diff-beta-sandwich

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

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

#> test-betaSandwich-r-sq-beta-sandwich-methods

```
#> Call:
#> RSqBetaSandwich(object = BetaN(object))
#>
#> Multiple correlation with MVN standard errors:
#>        est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.8045 0.0328 24.5344 42 0 0.6885 0.7161 0.7383 0.8707 0.8930 0.9205
#> adj 0.7906 0.0351 22.5014 42 0 0.6663 0.6958 0.7197 0.8615 0.8854 0.9149
#> Call:
#> RSqBetaSandwich(object = BetaN(object))
#>
#> Multiple correlation with MVN standard errors:
#>        est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.8045 0.0328 24.5344 42 0 0.6885 0.7161 0.7383 0.8707 0.8930 0.9205
#> adj 0.7906 0.0351 22.5014 42 0 0.6663 0.6958 0.7197 0.8615 0.8854 0.9149
#> Call:
#> RSqBetaSandwich(object = BetaADF(object))
#>
#> Multiple correlation with MVN standard errors:
#>        est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.8045 0.0287 28.0431 42 0 0.7030 0.7271 0.7466 0.8624 0.8819 0.9060
#> adj 0.7906 0.0307 25.7193 42 0 0.6818 0.7076 0.7285 0.8526 0.8735 0.8993
#> Call:
#> RSqBetaSandwich(object = BetaADF(object))
#>
#> Multiple correlation with MVN standard errors:
#>        est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.8045 0.0287 28.0431 42 0 0.7030 0.7271 0.7466 0.8624 0.8819 0.9060
#> adj 0.7906 0.0307 25.7193 42 0 0.6818 0.7076 0.7285 0.8526 0.8735 0.8993
#> Call:
```

```
#> RSqBetaSandwich(object = BetaHC(object))
#>
#> Multiple correlation with HC3 standard errors:
#>        est     se       t df p  0.05% 0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.8045 0.0313 25.6916 42 0 0.6937 0.72 0.7413 0.8677 0.8890 0.9153
#> adj 0.7906 0.0336 23.5627 42 0 0.6719 0.70 0.7229 0.8583 0.8811 0.9093
#> Call:
#> RSqBetaSandwich(object = BetaHC(object))
#>
#> Multiple correlation with HC3 standard errors:
#>        est     se       t df p  0.05% 0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.8045 0.0313 25.6916 42 0 0.6937 0.72 0.7413 0.8677 0.8890 0.9153
#> adj 0.7906 0.0336 23.5627 42 0 0.6719 0.70 0.7229 0.8583 0.8811 0.9093
#> Call:
#> RSqBetaSandwich(object = BetaN(object))
#>
#> Multiple correlation with MVN standard errors:
#>        est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.5809 0.0508 11.4431 44 0 0.4019 0.4442 0.4786 0.6832 0.7176 0.7599
#> adj 0.5714 0.0519 11.0053 44 0 0.3883 0.4316 0.4667 0.6760 0.7111 0.7544
#> Call:
#> RSqBetaSandwich(object = BetaN(object))
#>
#> Multiple correlation with MVN standard errors:
#>        est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.5809 0.0508 11.4431 44 0 0.4019 0.4442 0.4786 0.6832 0.7176 0.7599
#> adj 0.5714 0.0519 11.0053 44 0 0.3883 0.4316 0.4667 0.6760 0.7111 0.7544
#> Call:
#> RSqBetaSandwich(object = BetaADF(object))
#>
#> Multiple correlation with MVN standard errors:
#>        est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.5809 0.0491 11.8379 44 0 0.4079 0.4488 0.4820 0.6798 0.7130 0.7539
#> adj 0.5714 0.0502 11.3850 44 0 0.3944 0.4363 0.4702 0.6725 0.7065 0.7483
#> Call:
#> RSqBetaSandwich(object = BetaADF(object))
#>
#> Multiple correlation with MVN standard errors:
#>        est     se       t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.5809 0.0491 11.8379 44 0 0.4079 0.4488 0.4820 0.6798 0.7130 0.7539
#> adj 0.5714 0.0502 11.3850 44 0 0.3944 0.4363 0.4702 0.6725 0.7065 0.7483
#> Call:
#> RSqBetaSandwich(object = BetaHC(object))
#>
#> Multiple correlation with HC3 standard errors:
```

```
#>        est     se      t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.5809 0.0511 11.371 44 0 0.4008 0.4434 0.4779 0.6839 0.7184 0.7610
#> adj 0.5714 0.0522 10.936 44 0 0.3872 0.4307 0.4661 0.6767 0.7120 0.7556
#> Call:
#> RSqBetaSandwich(object = BetaHC(object))
#>
#> Multiple correlation with HC3 standard errors:
#>        est     se      t df p  0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> rsq 0.5809 0.0511 11.371 44 0 0.4008 0.4434 0.4779 0.6839 0.7184 0.7610
#> adj 0.5714 0.0522 10.936 44 0 0.3872 0.4307 0.4661 0.6767 0.7120 0.7556
#> Test passed

#> test-betaSandwich-r-sq-beta-sandwich

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

#> test-betaSandwich

#> 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
#> Test passed
```

```
#> [[1]]
#> [[1]][[1]]
#> [[1]][[1]]$value
#> [[1]][[1]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[1]]$visible
#> [1] TRUE
#>
#>
#> [[1]][[2]]
#> [[1]][[2]]$value
#> [[1]][[2]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[2]]$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]]$value[[1]]
#> [1] TRUE
#>
#>
#> [[1]][[8]]$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]]
#> [1] TRUE
#>
#>
#> [[1]][[10]]$visible
#> [1] TRUE
```

Environment

```r
ls()

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

Class

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

References

Pesigan, I. J. A., Sun, R. W., & Cheung, S. F. (2023). betaDelta and betaSandwich: Confidence intervals for standardized regression coefficients in R. *Multivariate Behavioral Research*, *58*(6), 1183–1186. https://doi.org/10.1080/00273171.2023.2201277

R Core Team. (2025). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/