betaSandwich: Methods

Ivan Jacob Agaloos Pesigan

1 HC

```
object <- lm(QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
```

```
out <- BetaHC(object, type = "hc3")
str(out)

#> List of 8
#>  $ call: language BetaHC(object = object, type = "hc3")
#>  $ lm  :List of 12
#>   ..$ coefficients : Named num [1:4] 10.3592 0.0842 0.216 0.1126
#>   .. ..- attr(*, "names")= chr [1:4] "(Intercept)" "NARTIC" "PCTGRT" "PCTSUPP"
#>   ..$ residuals    : Named num [1:46] -3.068 -0.688 2.675 2.052 4.039 ...
#>   .. ..- attr(*, "names")= chr [1:46] "1" "2" "3" "4" ...
#>   ..$ effects      : Named num [1:46] -187.4 51.49 27.59 -16.11 4.51 ...
#>   .. ..- attr(*, "names")= chr [1:46] "(Intercept)" "NARTIC" "PCTGRT" "PCTSUPP" ...
#>   ..$ rank         : int 4
#>   ..$ fitted.values: Named num [1:46] 15.1 23.7 26.3 33.9 40 ...
#>   .. ..- attr(*, "names")= chr [1:46] "1" "2" "3" "4" ...
#>   ..$ assign       : int [1:4] 0 1 2 3
#>   ..$ qr           :List of 5
#>   .. ..$ qr   : num [1:46, 1:4] -6.782 0.147 0.147 0.147 0.147 ...
#>   .. .. ..- attr(*, "dimnames")=List of 2
#>   .. .. .. ..$ : chr [1:46] "1" "2" "3" "4" ...
#>   .. .. .. ..$ : chr [1:4] "(Intercept)" "NARTIC" "PCTGRT" "PCTSUPP"
#>   .. .. ..- attr(*, "assign")= int [1:4] 0 1 2 3
#>   .. ..$ qraux: num [1:4] 1.15 1.04 1.06 1.09
#>   .. ..$ pivot: int [1:4] 1 2 3 4
#>   .. ..$ tol  : num 1e-07
#>   .. ..$ rank : int 4
#>   .. ..- attr(*, "class")= chr "qr"
#>   ..$ df.residual  : int 42
#>   ..$ xlevels      : Named list()
#>   ..$ call         : language lm(formula = QUALITY ~ NARTIC + PCTGRT + PCTSUPP, data = nas1982)
#>   ..$ terms        :Classes 'terms', 'formula'  language QUALITY ~ NARTIC + PCTGRT + PCTSUPP
```

```
#>   .. .. ..- attr(*, "variables")= language list(QUALITY, NARTIC, PCTGRT, PCTSUPP)
#>   .. .. ..- attr(*, "factors")= int [1:4, 1:3] 0 1 0 0 0 0 1 0 0 0 ...
#>   .. .. .. ..- attr(*, "dimnames")=List of 2
#>   .. .. .. .. ..$ : chr [1:4] "QUALITY" "NARTIC" "PCTGRT" "PCTSUPP"
#>   .. .. .. .. ..$ : chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#>   .. .. ..- attr(*, "term.labels")= chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#>   .. .. ..- attr(*, "order")= int [1:3] 1 1 1
#>   .. .. ..- attr(*, "intercept")= int 1
#>   .. .. ..- attr(*, "response")= int 1
#>   .. .. ..- attr(*, ".Environment")=<environment: 0x563250123108>
#>   .. .. ..- attr(*, "predvars")= language list(QUALITY, NARTIC, PCTGRT, PCTSUPP)
#>   .. .. ..- attr(*, "dataClasses")= Named chr [1:4] "numeric" "numeric" "numeric" "numeric"
#>   .. .. .. ..- attr(*, "names")= chr [1:4] "QUALITY" "NARTIC" "PCTGRT" "PCTSUPP"
#>   ..$ model        :'data.frame': 46 obs. of  4 variables:
#>   .. ..$ QUALITY: int [1:46] 12 23 29 36 44 21 40 42 24 30 ...
#>   .. ..$ NARTIC : int [1:46] 14 61 68 49 130 65 79 187 32 50 ...
#>   .. ..$ PCTGRT : int [1:46] 8 3 13 63 53 29 35 40 19 8 ...
#>   .. ..$ PCTSUPP: int [1:46] 16 67 66 52 64 59 81 65 87 43 ...
#>   .. ..- attr(*, "terms")=Classes 'terms', 'formula'  language QUALITY ~ NARTIC + PCTGRT + PCTSUP
#>   .. .. .. ..- attr(*, "variables")= language list(QUALITY, NARTIC, PCTGRT, PCTSUPP)
#>   .. .. .. ..- attr(*, "factors")= int [1:4, 1:3] 0 1 0 0 0 0 1 0 0 0 ...
#>   .. .. .. .. ..- attr(*, "dimnames")=List of 2
#>   .. .. .. .. .. ..$ : chr [1:4] "QUALITY" "NARTIC" "PCTGRT" "PCTSUPP"
#>   .. .. .. .. .. ..$ : chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#>   .. .. .. ..- attr(*, "term.labels")= chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#>   .. .. .. ..- attr(*, "order")= int [1:3] 1 1 1
#>   .. .. .. ..- attr(*, "intercept")= int 1
#>   .. .. .. ..- attr(*, "response")= int 1
#>   .. .. .. ..- attr(*, ".Environment")=<environment: 0x563250123108>
#>   .. .. .. ..- attr(*, "predvars")= language list(QUALITY, NARTIC, PCTGRT, PCTSUPP)
#>   .. .. .. ..- attr(*, "dataClasses")= Named chr [1:4] "numeric" "numeric" "numeric" "numeric"
#>   .. .. .. .. ..- attr(*, "names")= chr [1:4] "QUALITY" "NARTIC" "PCTGRT" "PCTSUPP"
#>   ..- attr(*, "class")= chr "lm"
#>  $ type: chr "hc3"
#>  $ beta: Named num [1:3] 0.495 0.391 0.263
#>   ..- attr(*, "names")= chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#>  $ vcov: num [1:3, 1:3] 0.00617 -0.0036 -0.00194 -0.0036 0.0067 ...
#>   ..- attr(*, "dimnames")=List of 2
#>   .. ..$ : chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#>   .. ..$ : chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#>  $ n   : int 46
#>  $ p   : num 3
#>  $ df  : int 42
#>  - attr(*, "class")= chr [1:2] "betasandwich" "list"

BetaHC(object, type = "hc3")
```

```
#> Call:
#> BetaHC(object = object, type = "hc3")
#>
#> Standardized regression slopes with HC3 standard errors:
#>            est     se      t      p   0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC  0.4951 0.0786 6.3025 0.0000  0.2172 0.2832 0.3366 0.6537 0.7071 0.7731
#> PCTGRT  0.3915 0.0818 4.7831 0.0000  0.1019 0.1707 0.2263 0.5567 0.6123 0.6810
#> PCTSUPP 0.2632 0.0855 3.0786 0.0037 -0.0393 0.0325 0.0907 0.4358 0.4940 0.5658
```

print

```
print(out)

#> Call:
#> BetaHC(object = object, type = "hc3")
#>
#> Standardized regression slopes with HC3 standard errors:
#>            est     se      t      p   0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC  0.4951 0.0786 6.3025 0.0000  0.2172 0.2832 0.3366 0.6537 0.7071 0.7731
#> PCTGRT  0.3915 0.0818 4.7831 0.0000  0.1019 0.1707 0.2263 0.5567 0.6123 0.6810
#> PCTSUPP 0.2632 0.0855 3.0786 0.0037 -0.0393 0.0325 0.0907 0.4358 0.4940 0.5658
```

coef

```
coef(out)

#>    NARTIC    PCTGRT   PCTSUPP
#> 0.4951451 0.3914887 0.2632477
```

vcov

```
vcov(out)

#>                NARTIC       PCTGRT      PCTSUPP
#> NARTIC   0.006172168 -0.003602529 -0.001943469
#> PCTGRT  -0.003602529  0.006699155 -0.002443584
#> PCTSUPP -0.001943469 -0.002443584  0.007311625
```

confint

```
confint(out, level = 0.95)

#>               2.5%     97.5%
#> NARTIC  0.33659828 0.6536920
#> PCTGRT  0.22631203 0.5566654
#> PCTSUPP 0.09068548 0.4358099
```

summary

```
summary(out)

#> Call:
#> BetaHC(object = object, type = "hc3")
#>
#> Standardized regression slopes with HC3 standard errors:
#>            est     se      t      p   0.05%   0.5%   2.5%  97.5%  99.5% 99.95%
#> NARTIC  0.4951 0.0786 6.3025 0.0000  0.2172 0.2832 0.3366 0.6537 0.7071 0.7731
#> PCTGRT  0.3915 0.0818 4.7831 0.0000  0.1019 0.1707 0.2263 0.5567 0.6123 0.6810
#> PCTSUPP 0.2632 0.0855 3.0786 0.0037 -0.0393 0.0325 0.0907 0.4358 0.4940 0.5658
```

2 Multivariate Normal

```
object <- lm(rating ~ ., data = attitude)
```

```
out <- BetaN(object)
str(out)

#> List of 8
#>  $ call: language BetaN(object = object)
#>  $ lm  :List of 12
#>   ..$ coefficients : Named num [1:7] 10.7871 0.6132 -0.0731 0.3203 0.0817 ...
#>   .. ..- attr(*, "names")= chr [1:7] "(Intercept)" "complaints" "privileges" "learning" ...
#>   ..$ residuals    : Named num [1:30] -8.11 1.647 1.061 -0.227 6.546 ...
#>   .. ..- attr(*, "names")= chr [1:30] "1" "2" "3" "4" ...
#>   ..$ effects      : Named num [1:30] -354.011 54.107 2.742 11.715 -0.971 ...
#>   .. ..- attr(*, "names")= chr [1:30] "(Intercept)" "complaints" "privileges" "learning" ...
#>   ..$ rank         : int 7
#>   ..$ fitted.values: Named num [1:30] 51.1 61.4 69.9 61.2 74.5 ...
#>   .. ..- attr(*, "names")= chr [1:30] "1" "2" "3" "4" ...
#>   ..$ assign       : int [1:7] 0 1 2 3 4 5 6
#>   ..$ qr           :List of 5
```

```
#>   .. ..$ qr   : num [1:30, 1:7] -5.477 0.183 0.183 0.183 0.183 ...
#>   .. .. ..- attr(*, "dimnames")=List of 2
#>   .. .. .. ..$ : chr [1:30] "1" "2" "3" "4" ...
#>   .. .. .. ..$ : chr [1:7] "(Intercept)" "complaints" "privileges" "learning" ...
#>   .. .. ..- attr(*, "assign")= int [1:7] 0 1 2 3 4 5 6
#>   .. ..$ qraux: num [1:7] 1.18 1 1.29 1.1 1.07 ...
#>   .. ..$ pivot: int [1:7] 1 2 3 4 5 6 7
#>   .. ..$ tol  : num 1e-07
#>   .. ..$ rank : int 7
#>   .. ..- attr(*, "class")= chr "qr"
#>   ..$ df.residual  : int 23
#>   ..$ xlevels      : Named list()
#>   ..$ call         : language lm(formula = rating ~ ., data = attitude)
#>   ..$ terms        :Classes 'terms', 'formula'  language rating ~ complaints + privileges + learn
#>   .. .. ..- attr(*, "variables")= language list(rating, complaints, privileges, learning, raises,
#>   .. .. ..- attr(*, "factors")= int [1:7, 1:6] 0 1 0 0 0 0 0 0 0 1 ...
#>   .. .. .. ..- attr(*, "dimnames")=List of 2
#>   .. .. .. .. ..$ : chr [1:7] "rating" "complaints" "privileges" "learning" ...
#>   .. .. .. .. ..$ : chr [1:6] "complaints" "privileges" "learning" "raises" ...
#>   .. .. ..- attr(*, "term.labels")= chr [1:6] "complaints" "privileges" "learning" "raises" ...
#>   .. .. ..- attr(*, "order")= int [1:6] 1 1 1 1 1 1
#>   .. .. ..- attr(*, "intercept")= int 1
#>   .. .. ..- attr(*, "response")= int 1
#>   .. .. ..- attr(*, ".Environment")=<environment: 0x563250123108>
#>   .. .. ..- attr(*, "predvars")= language list(rating, complaints, privileges, learning, raises,
#>   .. .. ..- attr(*, "dataClasses")= Named chr [1:7] "numeric" "numeric" "numeric" "numeric" ...
#>   .. .. .. ..- attr(*, "names")= chr [1:7] "rating" "complaints" "privileges" "learning" ...
#>   ..$ model        :'data.frame': 30 obs. of  7 variables:
#>   .. ..$ rating    : num [1:30] 43 63 71 61 81 43 58 71 72 67 ...
#>   .. ..$ complaints: num [1:30] 51 64 70 63 78 55 67 75 82 61 ...
#>   .. ..$ privileges: num [1:30] 30 51 68 45 56 49 42 50 72 45 ...
#>   .. ..$ learning  : num [1:30] 39 54 69 47 66 44 56 55 67 47 ...
#>   .. ..$ raises    : num [1:30] 61 63 76 54 71 54 66 70 71 62 ...
#>   .. ..$ critical  : num [1:30] 92 73 86 84 83 49 68 66 83 80 ...
#>   .. ..$ advance   : num [1:30] 45 47 48 35 47 34 35 41 31 41 ...
#>   .. ..- attr(*, "terms")=Classes 'terms', 'formula'  language rating ~ complaints + privileges +
#>   .. .. .. ..- attr(*, "variables")= language list(rating, complaints, privileges, learning, rais
#>   .. .. .. ..- attr(*, "factors")= int [1:7, 1:6] 0 1 0 0 0 0 0 0 0 1 ...
#>   .. .. .. .. ..- attr(*, "dimnames")=List of 2
#>   .. .. .. .. .. ..$ : chr [1:7] "rating" "complaints" "privileges" "learning" ...
#>   .. .. .. .. .. ..$ : chr [1:6] "complaints" "privileges" "learning" "raises" ...
#>   .. .. .. ..- attr(*, "term.labels")= chr [1:6] "complaints" "privileges" "learning" "raises" ..
#>   .. .. .. ..- attr(*, "order")= int [1:6] 1 1 1 1 1 1
#>   .. .. .. ..- attr(*, "intercept")= int 1
#>   .. .. .. ..- attr(*, "response")= int 1
```

```
#>   .. .. .. ..- attr(*, ".Environment")=<environment: 0x563250123108>
#>   .. .. .. ..- attr(*, "predvars")= language list(rating, complaints, privileges, learning, raise
#>   .. .. .. ..- attr(*, "dataClasses")= Named chr [1:7] "numeric" "numeric" "numeric" "numeric" ..
#>   .. .. .. .. ..- attr(*, "names")= chr [1:7] "rating" "complaints" "privileges" "learning" ...
#>   ..- attr(*, "class")= chr "lm"
#>  $ type: chr "mvn"
#>  $ beta: Named num [1:6] 0.6707 -0.0734 0.3089 0.0698 0.0312 ...
#>   ..- attr(*, "names")= chr [1:6] "complaints" "privileges" "learning" "raises" ...
#>  $ vcov: num [1:6, 1:6] 0.020531 -0.006381 -0.009324 -0.013812 -0.000242 ...
#>   ..- attr(*, "dimnames")=List of 2
#>   .. ..$ : chr [1:6] "complaints" "privileges" "learning" "raises" ...
#>   .. ..$ : chr [1:6] "complaints" "privileges" "learning" "raises" ...
#>  $ n   : int 30
#>  $ p   : num 6
#>  $ df  : int 23
#>  - attr(*, "class")= chr [1:2] "betasandwich" "list"
```

```
BetaN(object)
```

```
#> Call:
#> BetaN(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>                est     se       t      p   0.05%    0.5%    2.5%  97.5%  99.5%
#> complaints  0.6707 0.1433  4.6810 0.0001  0.1309  0.2685  0.3743 0.9671 1.0730
#> privileges -0.0734 0.1197 -0.6136 0.5455 -0.5243 -0.4094 -0.3210 0.1741 0.2625
#> learning    0.3089 0.1431  2.1580 0.0416 -0.2304 -0.0929  0.0128 0.6049 0.7107
#> raises      0.0698 0.1657  0.4213 0.6774 -0.5545 -0.3954 -0.2730 0.4126 0.5350
#> critical    0.0312 0.1047  0.2980 0.7684 -0.3632 -0.2627 -0.1854 0.2478 0.3251
#> advance    -0.1835 0.1338 -1.3717 0.1834 -0.6874 -0.5590 -0.4602 0.0932 0.1920
#>            99.95%
#> complaints 1.2106
#> privileges 0.3774
#> learning   0.8481
#> raises     0.6941
#> critical   0.4256
#> advance    0.3205
```

print

```
print(out)
```

```
#> Call:
#> BetaN(object = object)
#>
```

```
#> Standardized regression slopes with MVN standard errors:
#>                est     se       t      p   0.05%    0.5%    2.5%  97.5%  99.5%
#> complaints  0.6707 0.1433  4.6810 0.0001  0.1309  0.2685  0.3743 0.9671 1.0730
#> privileges -0.0734 0.1197 -0.6136 0.5455 -0.5243 -0.4094 -0.3210 0.1741 0.2625
#> learning    0.3089 0.1431  2.1580 0.0416 -0.2304 -0.0929  0.0128 0.6049 0.7107
#> raises      0.0698 0.1657  0.4213 0.6774 -0.5545 -0.3954 -0.2730 0.4126 0.5350
#> critical    0.0312 0.1047  0.2980 0.7684 -0.3632 -0.2627 -0.1854 0.2478 0.3251
#> advance    -0.1835 0.1338 -1.3717 0.1834 -0.6874 -0.5590 -0.4602 0.0932 0.1920
#>            99.95%
#> complaints 1.2106
#> privileges 0.3774
#> learning   0.8481
#> raises     0.6941
#> critical   0.4256
#> advance    0.3205
```

coef

```
coef(out)

#>  complaints  privileges    learning      raises    critical     advance
#>  0.67072520 -0.07342743  0.30887024  0.06981172  0.03119975 -0.18346445
```

vcov

```
vcov(out)

#>               complaints    privileges     learning       raises      critical
#> complaints  0.0205314876 -0.0063811296 -0.009324286 -0.013811718 -0.0002422133
#> privileges -0.0063811296  0.0143201460 -0.002170471  0.001552377 -0.0002768442
#> learning   -0.0093242861 -0.0021704714  0.020484826 -0.004998152  0.0028372586
#> raises     -0.0138117179  0.0015523774 -0.004998152  0.027456049 -0.0048713593
#> critical   -0.0002422133 -0.0002768442  0.002837259 -0.004871359  0.0109607636
#> advance     0.0096650976 -0.0029432354 -0.006326814 -0.009305030 -0.0017608366
#>                 advance
#> complaints  0.009665098
#> privileges -0.002943235
#> learning   -0.006326814
#> raises     -0.009305030
#> critical   -0.001760837
#> advance     0.017890011
```

confint

```
confint(out, level = 0.95)

#>                   2.5%      97.5%
#> complaints  0.37431113 0.96713928
#> privileges -0.32097709 0.17412223
#> learning    0.01279319 0.60494730
#> raises     -0.27296210 0.41258553
#> critical   -0.18537560 0.24777510
#> advance    -0.46015474 0.09322584
```

summary

```
summary(out)

#> Call:
#> BetaN(object = object)
#>
#> Standardized regression slopes with MVN standard errors:
#>                est     se       t      p   0.05%    0.5%    2.5%  97.5%  99.5%
#> complaints  0.6707 0.1433  4.6810 0.0001  0.1309  0.2685  0.3743 0.9671 1.0730
#> privileges -0.0734 0.1197 -0.6136 0.5455 -0.5243 -0.4094 -0.3210 0.1741 0.2625
#> learning    0.3089 0.1431  2.1580 0.0416 -0.2304 -0.0929  0.0128 0.6049 0.7107
#> raises      0.0698 0.1657  0.4213 0.6774 -0.5545 -0.3954 -0.2730 0.4126 0.5350
#> critical    0.0312 0.1047  0.2980 0.7684 -0.3632 -0.2627 -0.1854 0.2478 0.3251
#> advance    -0.1835 0.1338 -1.3717 0.1834 -0.6874 -0.5590 -0.4602 0.0932 0.1920
#>            99.95%
#> complaints 1.2106
#> privileges 0.3774
#> learning   0.8481
#> raises     0.6941
#> critical   0.4256
#> advance    0.3205
```

References

Dudgeon, P. (2017). Some improvements in confidence intervals for standardized regression coefficients. *Psychometrika*, *82*(4), 928–951. https://doi.org/10.1007/s11336-017-9563-z

R Core Team. (2022). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org/