

# betaMC: Methods

Ivan Jacob Agaloos Pesigan

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

```
set.seed(42)
out <- BetaMC(object)
str(out)

#> List of 9
#> $ call      : language BetaMC(object = object)
#> $ 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: 0x56477ade1410>
#> ..- 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 + 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: 0x56477ade1410>
#> ..- 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] 0.00655 -0.00359 -0.00202 -0.00359 0.00674 ...
#> ..- attr(*, "dimnames")=List of 2
#> ..$ : chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#> ..$ : chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#> $ thetahatstar: num [1:20000, 1:3] 0.399 0.575 0.313 0.439 0.653 ...
#> ..- attr(*, "dimnames")=List of 2
#> ..$ : NULL
#> ..$ : chr [1:3] "NARTIC" "PCTGRT" "PCTSUPP"
#> $ n : int 46
#> $ p : num 3

```

```

#> $ df          : int 42
#> - attr(*, "class")= chr [1:2] "betamc" "list"

set.seed(42)
BetaMC(object)

#> Call:
#> BetaMC(object = object)
#> HC3 sampling variance-covariance matrix:
#>      est      se      R  0.05%  0.5%  2.5% 97.5% 99.5% 99.95%
#> NARTIC  0.4951 0.0810 20000  0.1762 0.2710 0.3212 0.6362 0.6820 0.7242
#> PCTGRT  0.3915 0.0821 20000  0.0900 0.1644 0.2189 0.5426 0.5873 0.6291
#> PCTSUPP 0.2632 0.0856 20000 -0.0387 0.0246 0.0867 0.4232 0.4749 0.5311

```

## print

```

print(out)

#> Call:
#> BetaMC(object = object)
#> HC3 sampling variance-covariance matrix:
#>      est      se      R  0.05%  0.5%  2.5% 97.5% 99.5% 99.95%
#> NARTIC  0.4951 0.0810 20000  0.1762 0.2710 0.3212 0.6362 0.6820 0.7242
#> PCTGRT  0.3915 0.0821 20000  0.0900 0.1644 0.2189 0.5426 0.5873 0.6291
#> PCTSUPP 0.2632 0.0856 20000 -0.0387 0.0246 0.0867 0.4232 0.4749 0.5311

```

## coef

```

coef(out)

#>      NARTIC      PCTGRT      PCTSUPP
#> 0.4951451 0.3914887 0.2632477

```

## vcov

```

vcov(out)

#>      NARTIC      PCTGRT      PCTSUPP
#> NARTIC  0.006553462 -0.003590805 -0.002023911
#> PCTGRT -0.003590805  0.006742652 -0.002298333
#> PCTSUPP -0.002023911 -0.002298333  0.007330253

```

## confint

```
confint(out, level = 0.95)

#>           2.5%      97.5%
#> NARTIC  0.32119527 0.6362478
#> PCTGRT  0.21888204 0.5425632
#> PCTSUPP 0.08665619 0.4232492
```

## summary

```
summary(out)

#> Call:
#> BetaMC(object = object)
#>
#> Standardized regression slopes with HC3 standard errors:
#>      est      se      R  0.05%  0.5%  2.5%  97.5%  99.5% 99.95%
#> NARTIC  0.4951 0.0810 20000  0.1762 0.2710 0.3212 0.6362 0.6820 0.7242
#> PCTGRT  0.3915 0.0821 20000  0.0900 0.1644 0.2189 0.5426 0.5873 0.6291
#> PCTSUPP 0.2632 0.0856 20000 -0.0387 0.0246 0.0867 0.4232 0.4749 0.5311
```

## 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>
- Preacher, K. J., & Selig, J. P. (2012). Advantages of Monte Carlo confidence intervals for indirect effects. *Communication Methods and Measures*, 6(2), 77–98. <https://doi.org/10.1080/19312458.2012.679848>
- R Core Team. (2022). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. Vienna, Austria. <https://www.R-project.org/>