betareg.r

denis

2021-07-16

```
#!/usr/bin/r

betareg <- expand.grid(A1 = 1.2, A2 = 2.3, A3 = 3.4,
            B1 = 2.1, B2 = 2.2, B3 = 2.3,
            C1 = 3.1, C2 = 3.2, C3 = 3.3,
            ID = "IP mixed add ID: ",
            Network = "Mask: sub rede",
            Add = "Address: ")

# interval value
intervals::as.matrix(betareg$A1)
```

```
##      [,1]
## [1,]  1.2
```

```
lap1 <- c(l1 = 1, l2 = 2, l3 = 3)
# lap 1
lap1 + betareg$A1
```

```
##  l1  l2  l3
## 2.2 3.2 4.2
```

```
lap1 + betareg$A2
```

```
##  l1  l2  l3
## 3.3 4.3 5.3
```

```
lap1 + betareg$A3
```

```
##  l1  l2  l3
## 4.4 5.4 6.4
```

```
# gamma value express
gamma(lap1 + betareg$A1)
```

```
##       l1       l2       l3
## 1.101802 2.423965 7.756690
```

```
gamma(lap1 + betareg$A2)
```

```
##        l1        l2        l3
##  2.683437  8.855343 38.077976
```

```
gamma(lap1 + betareg$A3)
```

```
##        l1        l2        l3
##  10.13610  44.59885 240.83378
```

```
# m sphere 1
l4 <- betareg$A1
mn <- t(l4)
mn
```

```
##      [,1]
## [1,]  1.2
```

```
l4
```

```
## [1] 1.2
```

```
# type of keys
g1 <- betareg$A1
g2 <- betareg$A2
# open window
window(g1)
```

```
## [1] 1.2
## attr(,"tsp")
## [1] 1 1 1
```

```
attr(1.2,"g1")
```

```
## NULL
```

```
window(g2)
```

```
## [1] 2.3
## attr(,"tsp")
## [1] 1 1 1
```

```
attr(2.3, "g2")
```

```
## NULL
```

```r
# fit top
drop(g1)
```

```
## [1] 1.2
```

```r
# narrative mean
mean(g1 + g2)
```

```
## [1] 3.5
```

```r
# range value checkup
range(g1, na.rm = FALSE)
```

```
## [1] 1.2 1.2
```

```r
# logit lap1

# checkup coffee
probit <- betareg$B1
# log algorithm
log(g1/(1 - g2))
```

```
## Warning in log(g1/(1 - g2)): NaNs produced
```

```
## [1] NaN
```

```r
# checkup git
ML <- path.expand(path = ".")

# score LM
LM <- languageEl(g1, ML)
# create the class
class(LM)
```

```
## [1] "NULL"
```

```r
# residual compile pop
# business
residuals(LM)
```

```
## NULL
```

```r
# loglik map poms
class(NULL)
```

```
## [1] "NULL"
```

```r
# summary scope
summary(LM)
```

```
## Length  Class   Mode
##      0   NULL   NULL
```

```r
# panoramas
c(betareg$C2)
```

```
## [1] 3.2
```

```r
# running skeleton

# analysis local
cars$speed + betareg$A1
```

```
##  [1]  5.2  5.2  8.2  8.2  9.2 10.2 11.2 11.2 11.2 12.2 12.2 13.2 13.2 13.2 13.2
## [16] 14.2 14.2 14.2 14.2 15.2 15.2 15.2 15.2 16.2 16.2 16.2 17.2 17.2 18.2 18.2
## [31] 18.2 19.2 19.2 19.2 19.2 20.2 20.2 20.2 21.2 21.2 21.2 21.2 21.2 23.2 24.2
## [46] 25.2 25.2 25.2 25.2 26.2
```

```r
# logical template
lrtest <- cars$speed + betareg$A1
lrtest
```

```
##  [1]  5.2  5.2  8.2  8.2  9.2 10.2 11.2 11.2 11.2 12.2 12.2 13.2 13.2 13.2 13.2
## [16] 14.2 14.2 14.2 14.2 15.2 15.2 15.2 15.2 16.2 16.2 16.2 17.2 17.2 18.2 18.2
## [31] 18.2 19.2 19.2 19.2 19.2 20.2 20.2 20.2 21.2 21.2 21.2 21.2 21.2 23.2 24.2
## [46] 25.2 25.2 25.2 25.2 26.2
```

```r
# formation academic compile speed running
waldo::compare(lrtest, betareg$A1)
```

```
## `old`: 5.2 5.2 8.2 8.2 9.2 10.2 11.2 11.2 11.2 12.2 and 40 more...
## `new`: 1.2                                                    ...
```

```r
# coef test
coeftest <- betareg$A1
coeftest
```

```
## [1] 1.2
```

```r
# linear hypothesis
betareg$ID
```

```
## [1] IP mixed add ID:
## Levels: IP mixed add ID:
```

```
betareg$Network
```

```
## [1] Mask: sub rede
## Levels: Mask: sub rede
```

```
betareg$Add
```

```
## [1] Address:
## Levels: Address:
```

```
# party test
partytest <- betareg$ID
partytest
```

```
## [1] IP mixed add ID:
## Levels: IP mixed add ID:
```

```
# beta mixed
betamix <- betareg$Network
betamix
```

```
## [1] Mask: sub rede
## Levels: Mask: sub rede
```

```
# flex mixed
flexmix <- betareg$Add
flexmix
```

```
## [1] Address:
## Levels: Address:
```

```
# coef value equation
p = 15
q = 16

# product of equation
S0 <- betareg$A1 * p + q
S0
```

```
## [1] 34
```

```
f = 1
At0 <- c(1 / 2 + (f - 1) + c(t = 1, p = 15, q = 16))
At0
```

```
##    t    p    q
##  1.5 15.5 16.5
```

```r
# (because the term on the left-hand side is 0 ?).
lp1sd = 0
if (lp1sd != 0){
   c(At0)
} else {
  c(lp1sd)
}
```

```
## [1] 0
```

```r
# The expansion (2.36) is a special case of a very useful expansion in an
# orthogonal basis set.
orth = 2.36 # very the value north
exp(orth)
```

```
## [1] 10.59095
```