Package **CompSign**

Lena Morrill

October 2017

```
## knitr preferences
## no chache
```

```
## install latest version
library(devtools)
devtools::install_github("lm687/CompSign")

## Downloading GitHub repo lm687/CompSign@master
## from URL https://api.github.com/repos/lm687/CompSign/zipball/master
## Installing CompSign
## '/Library/Frameworks/R.framework/Resources/bin/R' --no-site-file
\
##  --no-environ --no-save --no-restore --quiet CMD INSTALL  \
##  '/private/var/folders/22/nzk7280n61jd5qrjhqm5cwph0000gn/T/RtmptScku8/devtools66f18b07ad
\
##  --library='/Library/Frameworks/R.framework/Versions/3.4/Resources/library'
\
##  --install-tests
##

library(CompSign)
```

```
############################
####### Dummy data #######
############################

### Example of matrix transformed into sign object
input_dummy <- matrix(runif(100), 4)
colnames(input_dummy) <- paste0('s', 1:25); rownames(input_dummy) <- paste0('sam', 1:4)
sign_dummy <- to_sign(input_dummy)
```

1 Summarise the signature matrix

```
add_together_matrix(sign_dummy)
```

```
## An object of class "sign"
## Slot "id":
## [1] "input_dummy"
##
## Slot "id_samples":
## [1] "sam1" "sam2" "sam3" "sam4"
##
## Slot "id_signatures":
##  [1] "s1"  "s2"  "s3"  "s4"  "s5"  "s6"  "s7"  "s8"  "s9"  "s10" "s11"
## [12] "s12" "s13" "s14" "s15" "s16" "s17" "s18" "s19" "s20" "s21" "s22"
## [23] "s23" "s24" "s25"
##
## Slot "count_matrix":
##             s1        s2        s3        s4        s5        s6        s7
## sam1 0.6907831 0.8228524 0.2524015 0.2744543 0.3000512 0.1322686 0.2194736
## sam2 0.8818848 0.9148024 0.4953640 0.3326483 0.8978098 0.9684040 0.5022404
## sam3 0.6368819 0.1914090 0.8556098 0.7969298 0.1121239 0.7307226 0.7358153
## sam4 0.2607221 0.4356568 0.2446198 0.6252434 0.5345280 0.7105411 0.4965340
##              s8          s9       s10       s11        s12       s13
## sam1 0.01923001 0.007304247 0.8551043 0.8121970 0.12980839 0.2961542
## sam2 0.26354396 0.649942506 0.2710597 0.2352673 0.09139877 0.6805692
## sam3 0.20337556 0.786678446 0.8520911 0.6531096 0.11114697 0.4819755
## sam4 0.98687697 0.839449452 0.4043220 0.1414012 0.82397333 0.3746479
##            s14       s15       s16        s17         s18       s19
## sam1 0.9559235 0.8283332 0.4244773 0.62294107 0.543067955 0.9656360
## sam2 0.7063705 0.5644384 0.2842755 0.51062850 0.711746154 0.1414930
## sam3 0.6775065 0.9901806 0.6629162 0.43285737 0.529813803 0.4695726
## sam4 0.5806252 0.3017961 0.6773294 0.02001661 0.001374035 0.6979142
##            s20        s21        s22        s23        s24       s25
## sam1 0.9658056 0.92127553 0.38571641 0.19556602 0.06867065 0.7349321
## sam2 0.7008028 0.05619648 0.87290842 0.19533150 0.03583676 0.4261994
## sam3 0.9243638 0.43182421 0.24641396 0.01843368 0.20574887 0.3837493
## sam4 0.4448937 0.57538916 0.04109801 0.72449780 0.70333235 0.5251124
##
## Slot "modified":
## [1] TRUE
```

```
results_sumarise <- summarise(add_together_matrix(sign_dummy))
```

```
## Error in switch(robust, pearson = {:  EXPR must be a length 1 vector
```

```
results_sumarise$General
```

```
## Error in eval(expr, envir, enclos):  object 'results_sumarise' not
found
```

2 Linear model for numerical predictors

```
tmp_merged_compositional <- new("merged_compositional",
                                id='adas',
                                id_samples=paste0("sam", 1:30),
                                id_signatures= c('s1', 's2', 's3', 's4'), ## signature names
                                count_matrix=MCMCpack::rdirichlet(30, c(1,1,1,1)),
                                df=data.frame(a=sample(1:1e4, 30), b=rep(10, 30)))
comp_lm(tmp_merged_compositional)

## [[1]]
## Response Y1 :
##
## Call:
## lm(formula = Y1 ~ as.matrix((x@df)[, indices_predictor]))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4639 -0.6986 -0.1645  0.6090  2.4207
##
## Coefficients: (1 not defined because of singularities)
##                                          Estimate Std. Error t value
## (Intercept)                             -1.0977949  0.5412697  -2.028
## as.matrix((x@df)[, indices_predictor])a  0.0001310  0.0000846   1.549
## as.matrix((x@df)[, indices_predictor])b         NA         NA      NA
##                                         Pr(>|t|)
## (Intercept)                               0.0522 .
## as.matrix((x@df)[, indices_predictor])a   0.1327
## as.matrix((x@df)[, indices_predictor])b       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.236 on 28 degrees of freedom
## Multiple R-squared:  0.07889,Adjusted R-squared:  0.04599
## F-statistic: 2.398 on 1 and 28 DF,  p-value: 0.1327
##
##
## Response Y2 :
##
## Call:
```

```
## lm(formula = Y2 ~ as.matrix((x@df)[, indices_predictor]))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3945 -0.6119 -0.2015  0.7405  2.2804
##
## Coefficients: (1 not defined because of singularities)
##                                          Estimate Std. Error t value
## (Intercept)                              4.103e-01  5.494e-01   0.747
## as.matrix((x@df)[, indices_predictor])a -9.292e-05  8.587e-05  -1.082
## as.matrix((x@df)[, indices_predictor])b         NA         NA      NA
##                                          Pr(>|t|)
## (Intercept)                                 0.461
## as.matrix((x@df)[, indices_predictor])a     0.288
## as.matrix((x@df)[, indices_predictor])b        NA
##
## Residual standard error: 1.255 on 28 degrees of freedom
## Multiple R-squared:  0.04014,Adjusted R-squared:  0.005857
## F-statistic: 1.171 on 1 and 28 DF,  p-value: 0.2885
##
##
## Response Y3 :
##
## Call:
## lm(formula = Y3 ~ as.matrix((x@df)[, indices_predictor]))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.1368 -0.3773 -0.2044  0.5526  2.7553
##
## Coefficients: (1 not defined because of singularities)
##                                          Estimate Std. Error t value
## (Intercept)                             -1.590e-01  6.170e-01  -0.258
## as.matrix((x@df)[, indices_predictor])a  4.871e-06  9.644e-05   0.051
## as.matrix((x@df)[, indices_predictor])b         NA         NA      NA
##                                          Pr(>|t|)
## (Intercept)                                 0.798
## as.matrix((x@df)[, indices_predictor])a     0.960
## as.matrix((x@df)[, indices_predictor])b        NA
##
## Residual standard error: 1.409 on 28 degrees of freedom
## Multiple R-squared:  9.109e-05,Adjusted R-squared:  -0.03562
## F-statistic: 0.002551 on 1 and 28 DF,  p-value: 0.9601
```

3 Importing data

```
#biplot(princomp(acomp(MCMCpack::rdirichlet(30, rep(1, 4)))))
```