paper simulation

```
# set.seed(4321)
set.seed(432)
n.sim <- 1e2 # sample size

p.all <- 1e4 # No. of dimensions of Y (signal + non-signal)
p.true <- 10 # No. of dimensions for signal
signal.true <- .6 # size of true signal
n.top.want <- 15
n.fold <- 10
library(MASS)
library(Matrix)
```

simulate data

```
# epsilon
epsilon <- matrix(rnorm(n = n.sim * p.all), nrow = n.sim, ncol = p.all)

# ------------------------------------------------------------------------------
# A
A.candidate <- list(rep(1, p.all), rep(0, p.all))
A.sample <- sample(A.candidate, size = n.sim, replace = TRUE)

A.sample <- do.call(rbind, A.sample)
A.sample.vec <- A.sample[,1]

# ------------------------------------------------------------------------------
# B1
b1.row <- c(rep(signal.true, p.true), rep(0, p.all - p.true))

rep.row<-function(x,n){
    matrix(rep(x,each=n),nrow=n)
}
b1 <- rep.row(b1.row, n = n.sim)
# ------------------------------------------------------------------------------
# B0
b0.row <- rnorm(n = p.all)
b0 <- rep.row(b0.row, n = n.sim)
# ------------------------------------------------------------------------------
# Y
temp1 <- b1 * A.sample
# ----------------------------------
rm(list = c('b1', 'A.sample'))
gc()
```

```
##           used (Mb) gc trigger (Mb) max used (Mb)
## Ncells 1065252 56.9    1770749 94.6  1442291 77.1
## Vcells 3888788 29.7    7427330 56.7  6921589 52.9
```

```
# ------------------------------------
Y <- b0 + temp1 + epsilon
# ------------------------------------
rm(list = c('b0', 'epsilon', 'temp1'))
gc()
```

```
##           used (Mb) gc trigger (Mb) max used (Mb)
## Ncells 1065599 57.0    1770749 94.6  1442291 77.1
## Vcells 1888995 14.5    5941864 45.4  6921589 52.9
```

fit DA.test

```
library(data.adapt.multi.test)
```

```
## data.adapt.multi.test: Data-adaptive test statistics for multiple testing in high-dimensional settin
```

```
## Version: 0.3.3
```

```
out_result <- data_adapt_multi_test(Y = Y, A = A.sample.vec, n.top = p.true + 5,
                                    n.fold = 4, parallel = FALSE) # BH on all Y
```

```
## [1] "Fold: 1"
## [1] 2
```

```
## Loading required package: nnls
```

```
## Loading required package: gam
```

```
## Loading required package: splines
```

```
## Loading required package: foreach
```

```
## Loaded gam 1.14
```

```
## Loading required package: earth
```

```
## Loading required package: plotmo
```

```
## Loading required package: plotrix
```

```
## Loading required package: TeachingDemos
```

```
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 10
## [1] 460
## [1] 487
## [1] 3024
## [1] 4001
## [1] 4305
## [1] 5550
## [1] 8231
## [1] 8716
## [1] "Fold: 2"
## [1] 2
```

```
## [1] 3
## [1] 6
## [1] 7
## [1] 8
## [1] 1406
## [1] 1578
## [1] 2299
## [1] 2512
## [1] 3409
## [1] 4658
## [1] 5181
## [1] 8716
## [1] 9709
## [1] 9790
## [1] "Fold: 3"
## [1] 2
## [1] 3
## [1] 4
## [1] 6
## [1] 7
## [1] 8
## [1] 758
## [1] 1452
## [1] 2118
## [1] 2946
## [1] 4061
## [1] 6488
## [1] 8231
## [1] 8302
## [1] 8718
## [1] "Fold: 4"
## [1] 2
## [1] 3
## [1] 4
## [1] 6
## [1] 7
## [1] 9
## [1] 505
## [1] 2008
## [1] 2118
## [1] 3409
## [1] 4001
## [1] 5181
## [1] 5251
## [1] 5550
## [1] 6974
```

```
print(out_result)
```

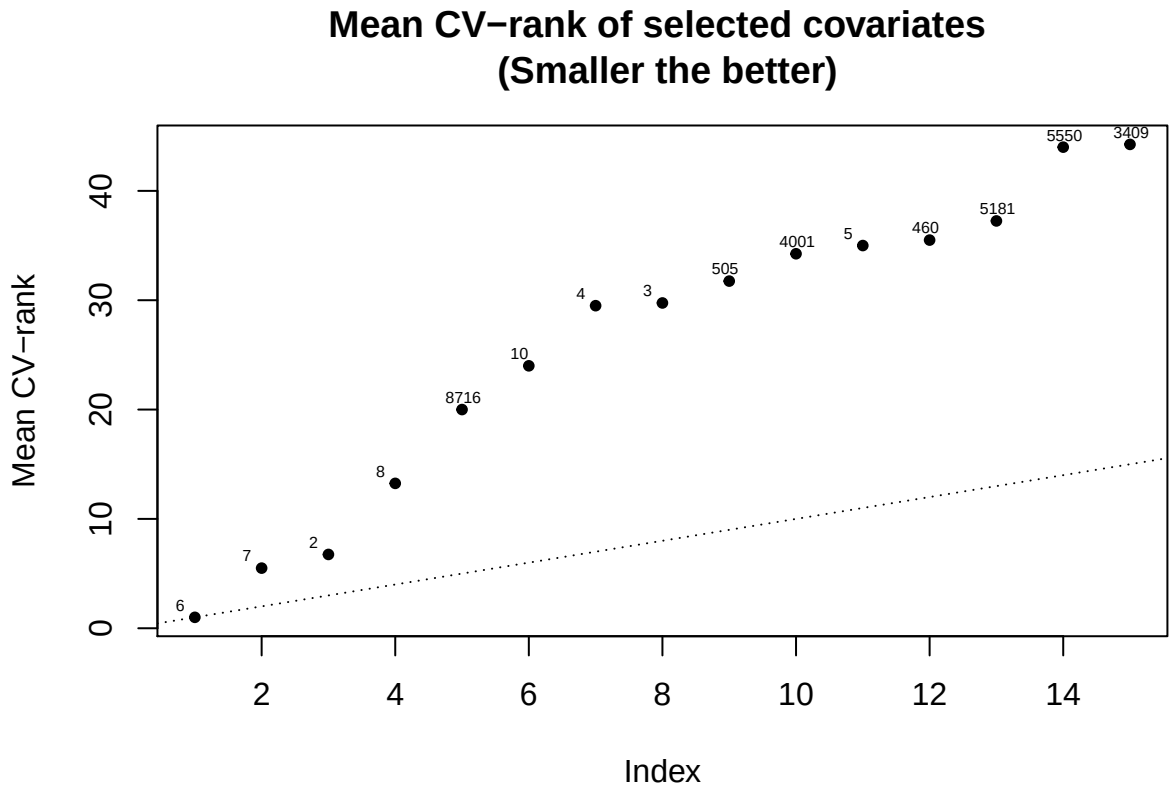
```
## [1] "The top covariates are"
## [[1]]
## x
## 2
## 1
##
```

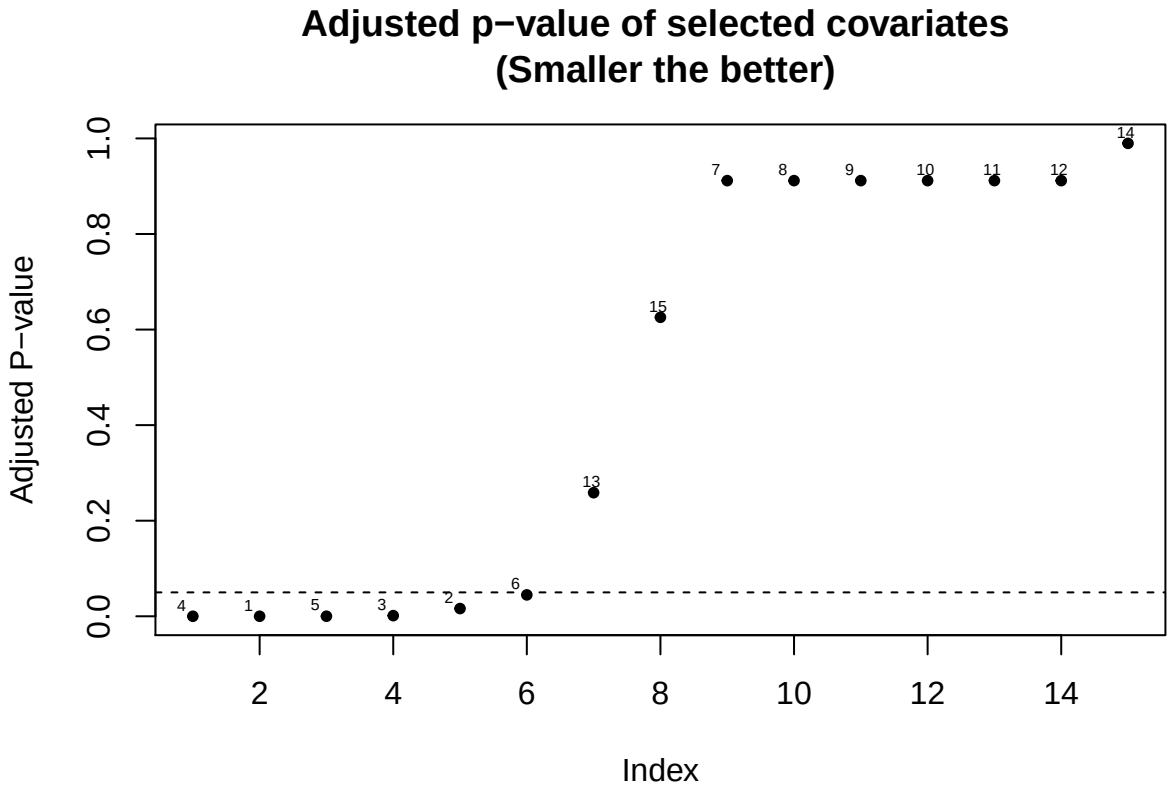
```
## [[2]]
## x
##    3    4
## 0.75 0.25
##
## [[3]]
## x
##    4    5    6
## 0.50 0.25 0.25
##
## [[4]]
## x
##    6    7
## 0.75 0.25
##
## [[5]]
## x
##    7    8
## 0.75 0.25
##
## [[6]]
## x
##    8    9 1406
## 0.50 0.25 0.25
##
## [[7]]
## x
##   10  505  758 1578
## 0.25 0.25 0.25 0.25
##
## [[8]]
## x
##  460 1452 2008 2299
## 0.25 0.25 0.25 0.25
##
## [[9]]
## x
##  487 2118 2512
## 0.25 0.50 0.25
##
## [[10]]
## x
## 2946 3024 3409
## 0.25 0.25 0.50
##
## [[11]]
## x
## 4001 4061 4658
## 0.50 0.25 0.25
##
## [[12]]
## x
## 4305 5181 6488
## 0.25 0.50 0.25
```

```
##
## [[13]]
## x
## 5251 5550 8231 8716
## 0.25 0.25 0.25 0.25
##
## [[14]]
## x
## 5550 8231 8302 9709
## 0.25 0.25 0.25 0.25
##
## [[15]]
## x
## 6974 8716 8718 9790
## 0.25 0.25 0.25 0.25
##
## [1] "The ATE estiamtes are"
##  [1]  0.824121612  0.512955440  0.672537103  0.917927449  0.877971744
##  [6]  0.432534900  0.065378076  0.042622574  0.083390119  0.034803681
## [11]  0.072996633  0.113768667  0.318284053 -0.002515117  0.186219075
## [1] "The raw p-values are"
##  [1] 5.572057e-06 5.364302e-03 3.499355e-04 6.231331e-07 2.305536e-05
##  [6] 1.789182e-02 7.594116e-01 8.234625e-01 6.239067e-01 8.509090e-01
## [11] 7.232268e-01 5.658912e-01 1.206756e-01 9.896647e-01 3.336796e-01
## [1] "The adjusted p-values are"
##  [1] 4.179043e-05 1.609291e-02 1.312258e-03 9.346997e-06 1.152768e-04
##  [6] 4.472955e-02 9.116882e-01 9.116882e-01 9.116882e-01 9.116882e-01
## [11] 9.116882e-01 9.116882e-01 2.585906e-01 9.896647e-01 6.256492e-01
## [1] "The top mean CV-rank are (the smaller the better)"
##  [1]  1.00  5.50  6.75 13.25 20.00 24.00 29.50 29.75 31.75 34.25 35.00
## [12] 35.50 37.25 44.00 44.25
## [1] "The percentage of appearing in top 15 are (the larger the better)"
##  [1] 100 100 100  75  50  25  75  75  25  50  25  25  50  50  50
## [1] "The covariates still significant are"
## [1] 1 2 3 4 5 6
## [1] "Their compositions are"
## [[1]]
## x
## 2
## 1
##
## [[2]]
## x
##    3    4
## 0.75 0.25
##
## [[3]]
## x
##    4    5    6
## 0.50 0.25 0.25
##
## [[4]]
## x
##    6    7
```

```
## 0.75 0.25
##
## [[5]]
## x
##    7    8
## 0.75 0.25
##
## [[6]]
## x
##    8    9 1406
## 0.50 0.25 0.25
```

```
plot(out_result)
```





fit naive BH

```
lm_out <- lm(Y ~ A.sample.vec)
B1_hat_lm <- lm_out$coefficients[2,]
names(B1_hat_lm) <- c(1:ncol(Y))

lm_summary <- summary(lm_out)
pval_lm <- sapply(lm_summary, function(x) x$coefficients[2,4])
qval_lm <- p.adjust(pval_lm, method = 'BH')
which(qval_lm < .05)
```

```
## Response Y6
##           6
```

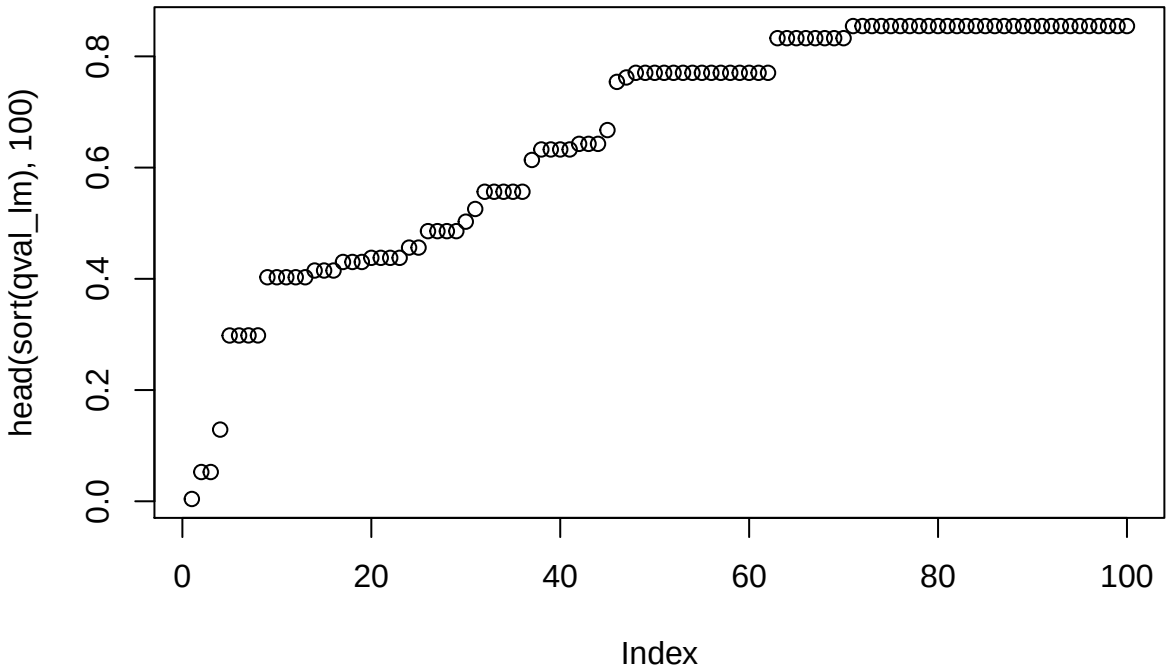
p value sorting

```
head(sort(qval_lm), 20)
```

```
##    Response Y6    Response Y2    Response Y3 Response Y7901    Response Y5
##    0.004147593    0.052627638    0.052627638    0.128874599    0.298087814
##    Response Y7    Response Y4    Response Y8 Response Y2648 Response Y4001
##    0.298087814    0.298232904    0.298232904    0.402835640    0.402835640
## Response Y7501 Response Y8231 Response Y8716   Response Y10 Response Y2804
##    0.402835640    0.402835640    0.402835640    0.414919993    0.414919993
## Response Y5181 Response Y2088 Response Y4321 Response Y8718  Response Y460
```

```
##    0.414919993    0.430310346    0.430310346    0.430310346    0.437727343
```

```
plot(head(sort(qval_lm), 100))
```

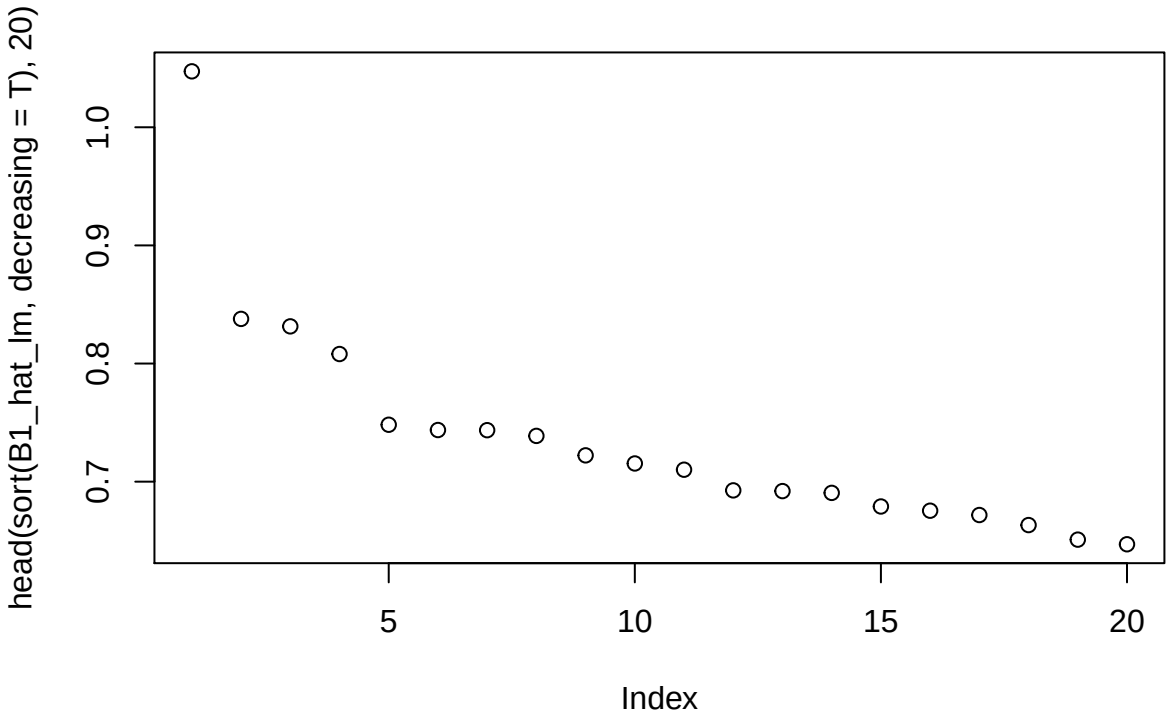



```
# head(B1_hat_lm[c(1:ncol(Y))[order(qval_lm)]], 20)

head(sort(B1_hat_lm, decreasing = T), 20)
```

```
##         6         2         7         3         8      4001      8716
## 1.0473395 0.8377845 0.8314589 0.8080560 0.7481866 0.7437065 0.7435837
##      5181         4      5550        10         5       460      8718
## 0.7387429 0.7222613 0.7154127 0.7101274 0.6926170 0.6920232 0.6905042
##      3409       505      5251         9      8302      2299
## 0.6789961 0.6754214 0.6717632 0.6632667 0.6509274 0.6470105
```

```
plot(head(sort(B1_hat_lm, decreasing = T), 20))
```


head(sort(B1_hat_lm, decreasing = T), 20)
0.7
0.8
0.9
1.0
5
10
15
20
Index