

# Lehmann's alternative on Credit Card dataset

$k=2$ , proportion of outliers from 0 to 1

2023-05-24

We consider Lehmann's alternative with  $k = 2$  and we show that the theoretical result that closed testing procedure with Wilcoxon-Mann-Whitney local test is the Locally Most Powerful Invariant test for the global null is validated also by numerical simulations.

```
library(doSNOW)
library(foreach)
library(nout)
library(tictoc)
library(isotree)
library(readr)
library(R.matlab)

compact_results = function(res){
  resT=as.data.frame(t(res))
  discoveries = as.data.frame(cbind("d_BH"=unlist(resT$d_BH),
                                     "d_StoBH"=unlist(resT$d_StoBH),
                                     "d_Sim"=unlist(resT$d_Sim),
                                     "d_StoSimes"=unlist(resT$d_StoSimes),
                                     "d_WMW"=unlist(resT$d_WMW)))
  mean.discoveries = apply(discoveries, MARGIN = 2, FUN = mean)
  power.GlobalNull = as.data.frame(discoveries>0)
  mean.powerGlobalNull = apply(power.GlobalNull, MARGIN = 2, FUN = mean)
  return(list("discoveries" = discoveries,
             "mean.discoveries" = mean.discoveries,
             "power.GlobalNull" = power.GlobalNull,
             "mean.powerGlobalNull" = mean.powerGlobalNull,
             "pi.not" = unlist(resT$pi.not),
             "uniques"=unlist(resT$uniques),
             "n1"=unlist(resT$n1),
             "alpha"=unlist(resT$alpha)))
}
```

## Credit Card dataset

```
dataset = read_csv("~/nout/trials/RealData/Datasets/Dataset creditcard/creditcard.csv")

in_ind = which(dataset$Class==0)
out_ind = which(dataset$Class==1)

# Initializing parameters
```

```

set.seed(321)

B = 4000

l = 1683
m = 249
n = 50
k = 2 # exponent of Lehmann's alternative
myalpha = n/(m+1)

tr_ind = sample(in_ind, size = 1)
in_ind2 = setdiff(in_ind, tr_ind)
tr = dataset[tr_ind,]
n_cpus = parallel::detectCores()
iso.fo = isotree::isolation.forest(tr, ndim = ncol(dataset), ntrees = 150, sample_size = 256,
                                   nthreads = n_cpus, scoring_metric = "depth",
                                   output_score = TRUE)

isofo.model = iso.fo$model
mycrit = nout::critWMW(m = n, n = m, alpha = myalpha)

prop.out = seq(0, 1, by=0.02)
n1_vec = round(prop.out*n)

cl <- makeCluster(parallel::detectCores())
clusterEvalQ(cl, {library(isotree)})

## [[1]]
## [1] "isotree"      "snow"        "stats"       "graphics"    "grDevices"  "utils"
## [7] "datasets"    "methods"     "base"
##
## [[2]]
## [1] "isotree"      "snow"        "stats"       "graphics"    "grDevices"  "utils"
## [7] "datasets"    "methods"     "base"
##
## [[3]]
## [1] "isotree"      "snow"        "stats"       "graphics"    "grDevices"  "utils"
## [7] "datasets"    "methods"     "base"
##
## [[4]]
## [1] "isotree"      "snow"        "stats"       "graphics"    "grDevices"  "utils"
## [7] "datasets"    "methods"     "base"

registerDoSNOW(cl)

res = list()

for(n1 in n1_vec){
  i = which(n1_vec==n1)
  res[[i]] = foreach(b = 1:B, .combine=cbind) %dopar% {
    n0 = n - n1
    N = n0 + m + k*n1
    in_index3 = sample(in_ind2, size = N)
    cal_ind = in_index3[1:m]

```

```

te_ind.augmented = in_index3[(m+1):N]
cal = dataset[cal_ind,]
te = dataset[te_ind.augmented,]
S_cal = predict.isolation_forest(isofo.model, cal, type = "score")
augmented.S_te = predict.isolation_forest(isofo.model, te, type = "score")
if(n1==0)
  S_te = augmented.S_te
if(n1==n)
  S_te = sapply(0:(n1-1), FUN=function(i) max(augmented.S_te[1+k*i], augmented.S_te[k+k*i]))
if(0<n1&n1<n)
  S_te = c(augmented.S_te[1:n0],
            sapply(0:(n1-1), FUN=function(i) max(augmented.S_te[1+k*i], augmented.S_te[k+k*i]))

d_WMW = nout::d_mannwhitney(S_Y = S_te, S_X = S_cal, crit = mycrit)
d_Sim = nout::d_Simes(S_X = S_cal, S_Y = S_te, alpha = myalpha)
StoSimes = nout::d_StoreySimes(S_X = S_cal, S_Y = S_te, alpha = myalpha)
d_StoSimes = StoSimes$d
pi.not = StoSimes$pi.not
d_BH = nout::d_benjhoch(S_X = S_cal, S_Y = S_te, alpha = myalpha)
d_StoBH = nout::d_StoreyBH(S_X = S_cal, S_Y = S_te, alpha = myalpha)
uniques = length(unique(c(S_cal, S_te)))
return(list("d_BH" = d_BH,
            "d_StoBH" = d_StoBH,
            "d_Sim" = d_Sim,
            "d_StoSimes" = d_StoSimes,
            "d_WMW" = d_WMW,
            "uniques" = uniques,
            "n1" = n1,
            "pi.not" = pi.not,
            "alpha" = myalpha))

}
}

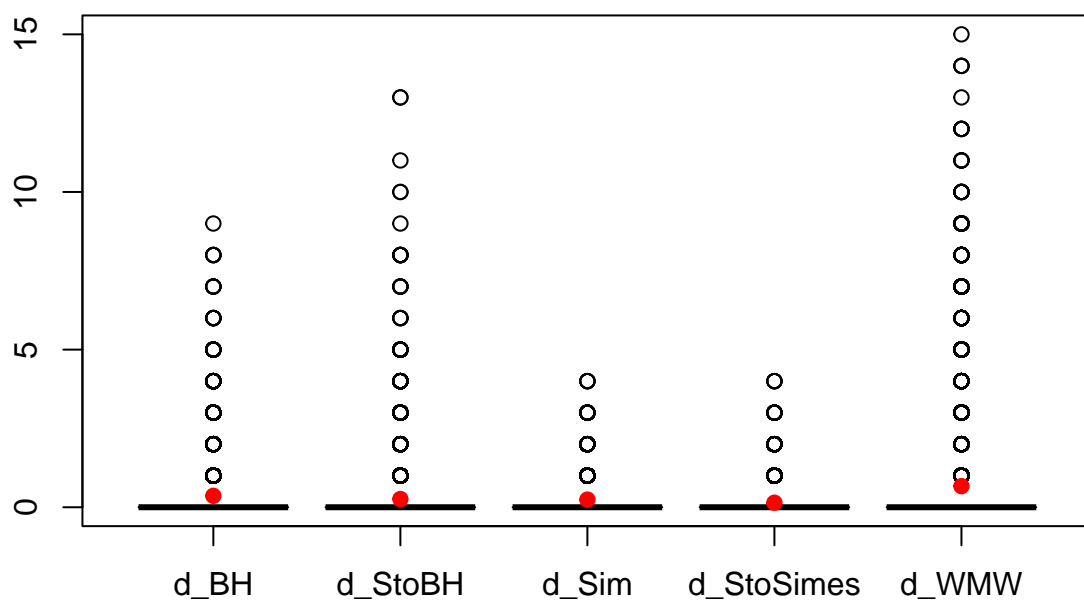
stopCluster(cl)

results = lapply(res, compact_results)

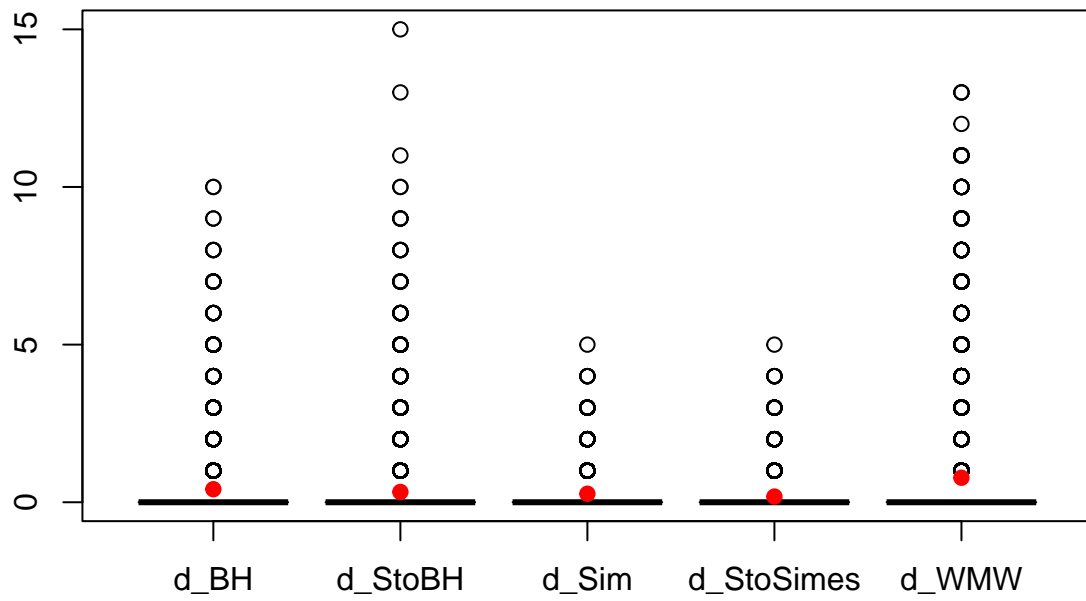
for(i in 1:length(n1_vec)){
  boxplot(results[[i]]$discoveries, main=paste0("Digits | Number of discoveries with ", n1_vec[[i]], "
  points(x=1:5, y=results[[i]]$mean.discoveries, pch=19, col="red")
  results[[i]]$mean.discoveries
  results[[i]]$mean.powerGlobalNull
}

```

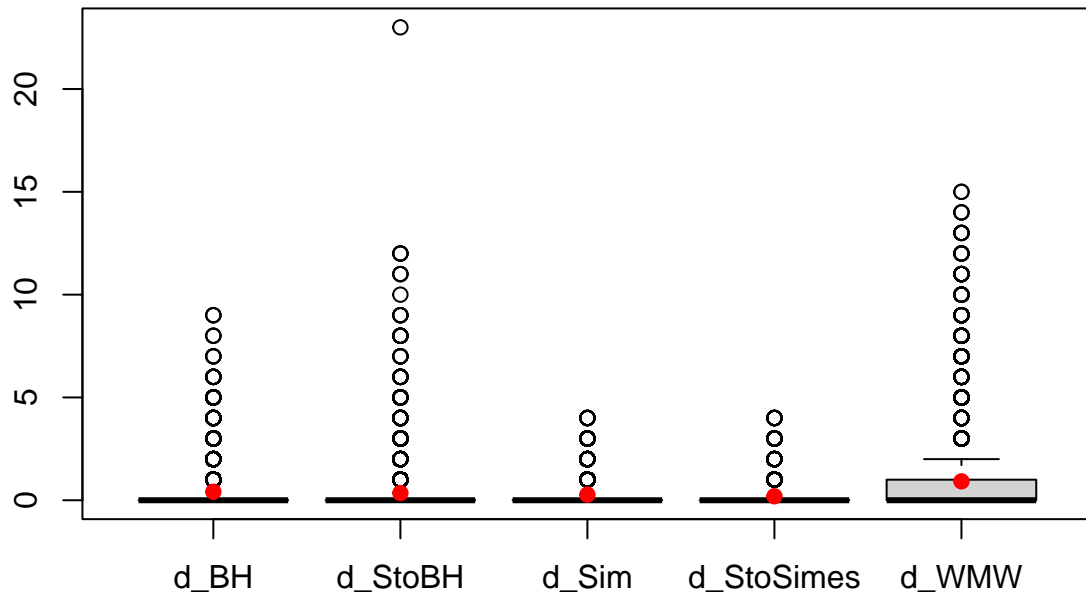
### Digits | Number of discoveries with 0 outliers



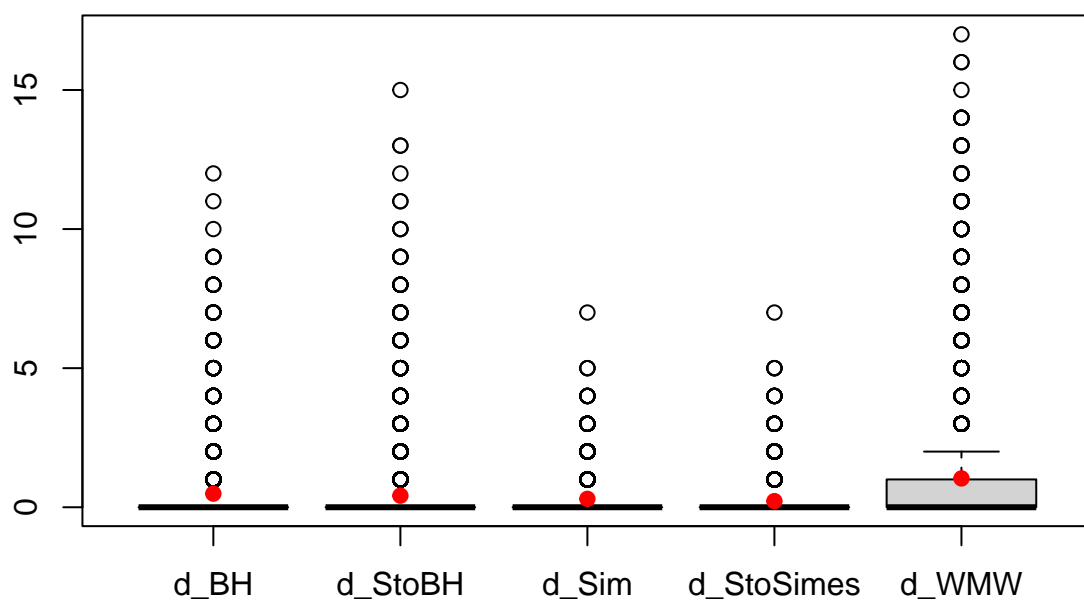
### Digits | Number of discoveries with 1 outliers



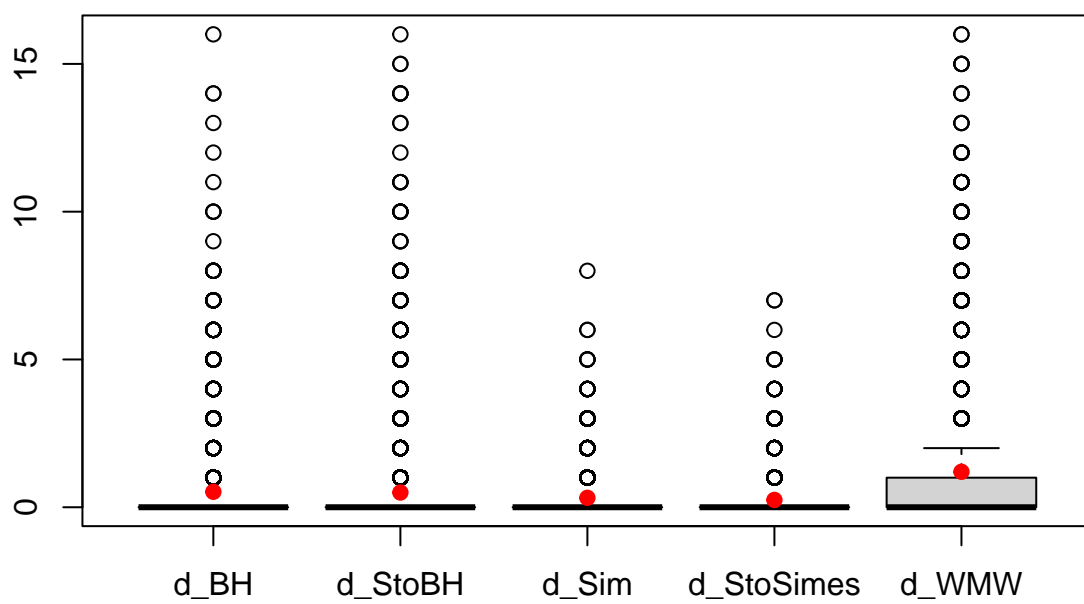
## Digits | Number of discoveries with 2 outliers



## Digits | Number of discoveries with 3 outliers

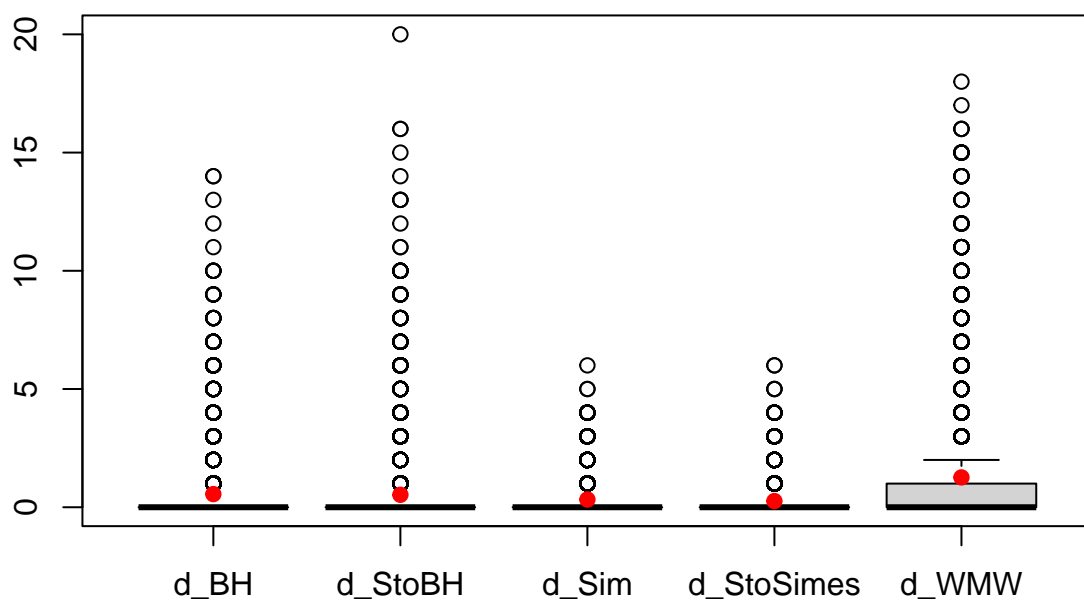


### Digits | Number of discoveries with 4 outliers

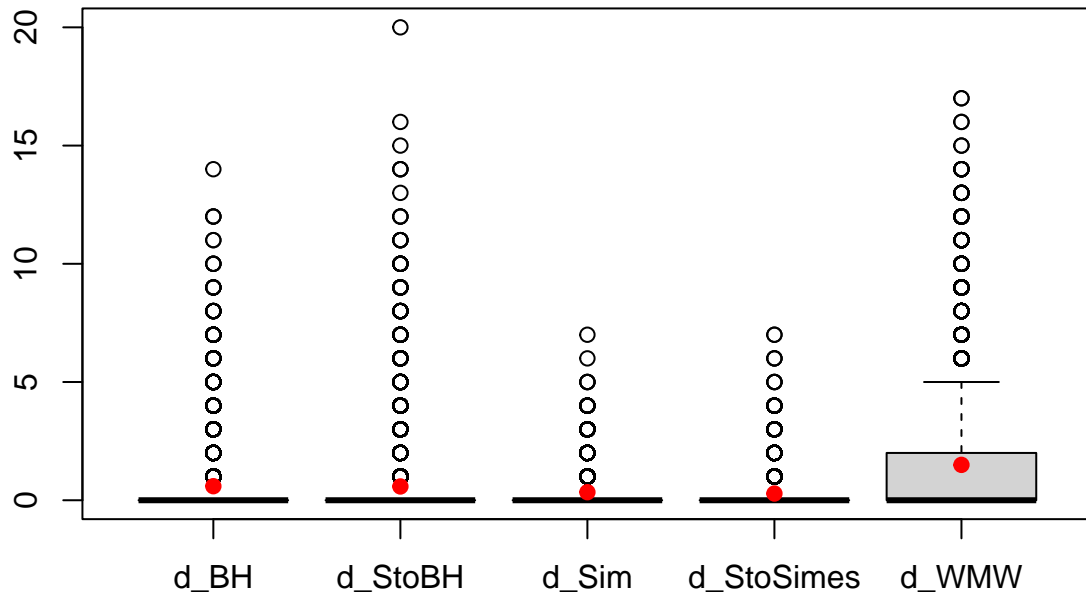




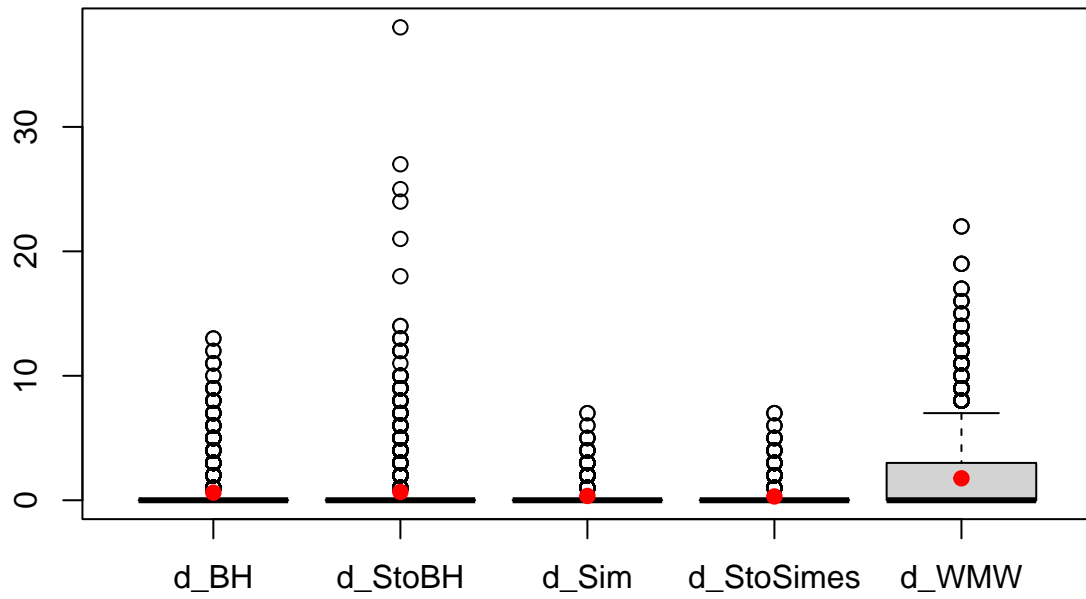
## Digits | Number of discoveries with 5 outliers



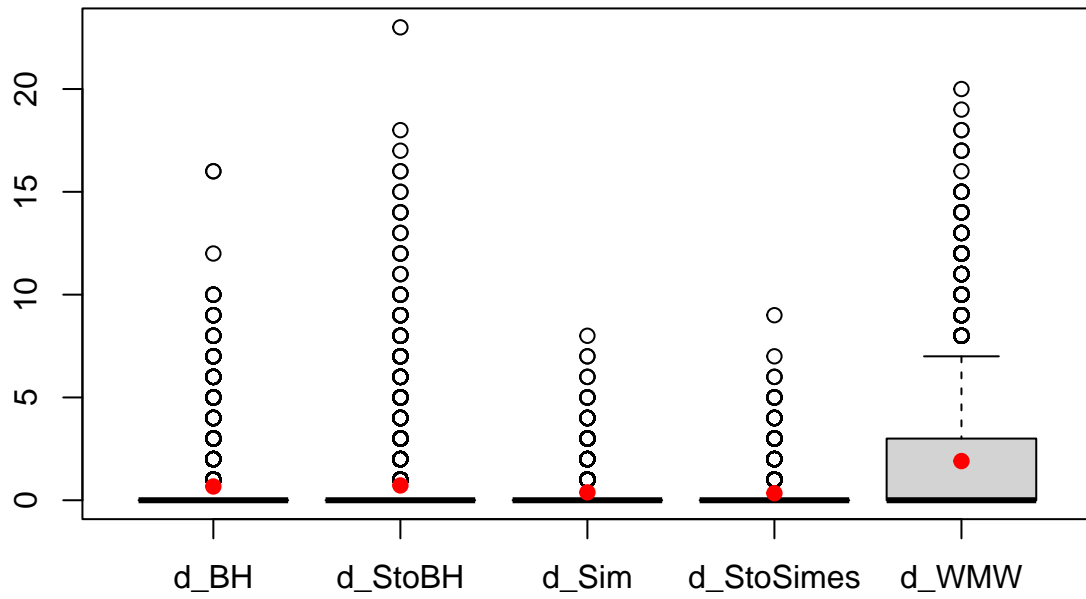
## Digits | Number of discoveries with 6 outliers



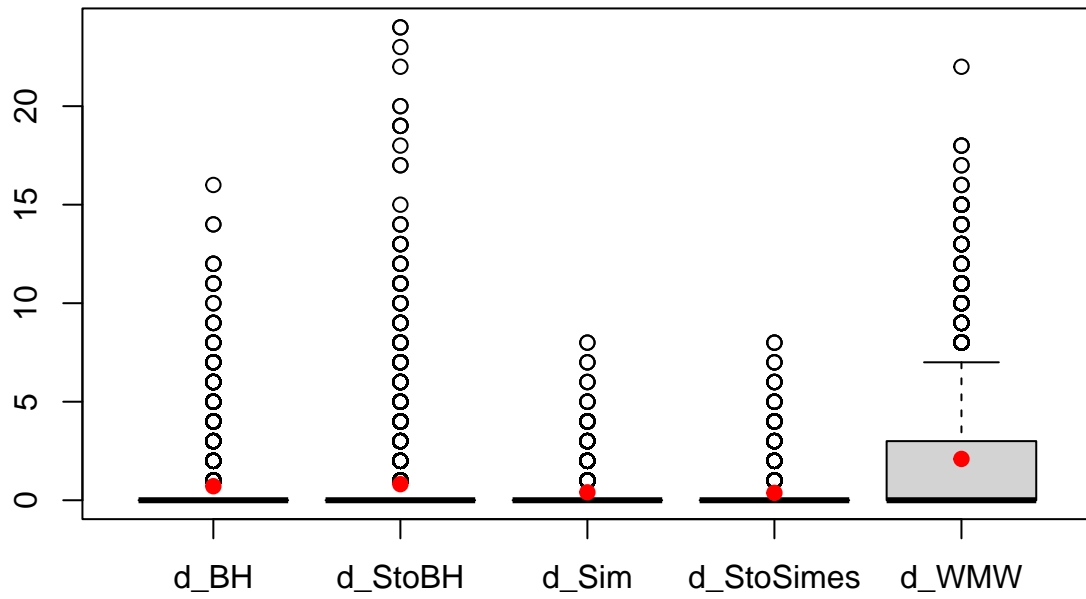
## Digits | Number of discoveries with 7 outliers



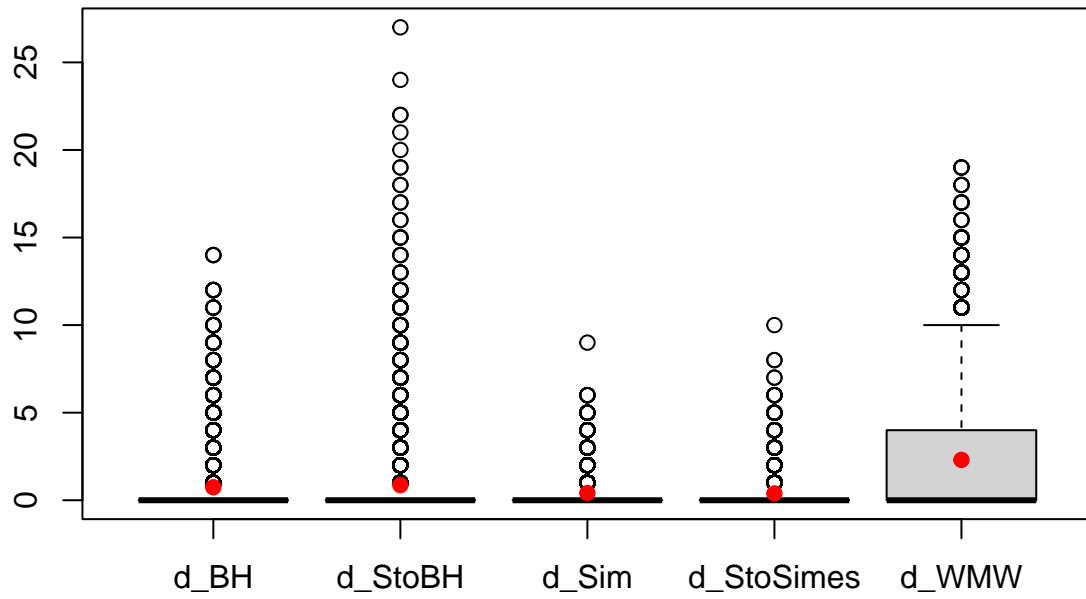
## Digits | Number of discoveries with 8 outliers



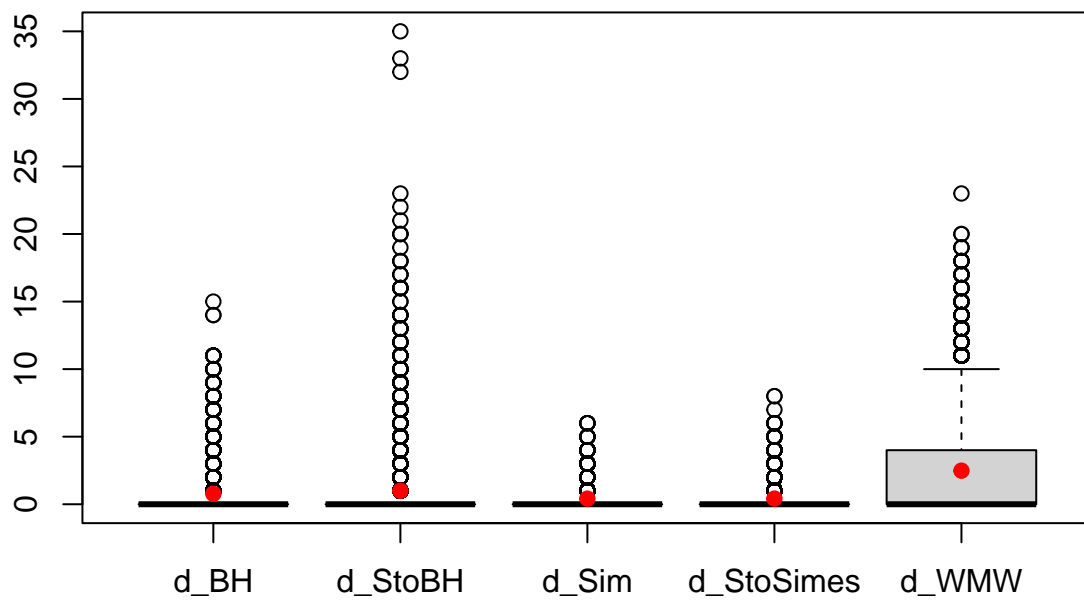
## Digits | Number of discoveries with 9 outliers



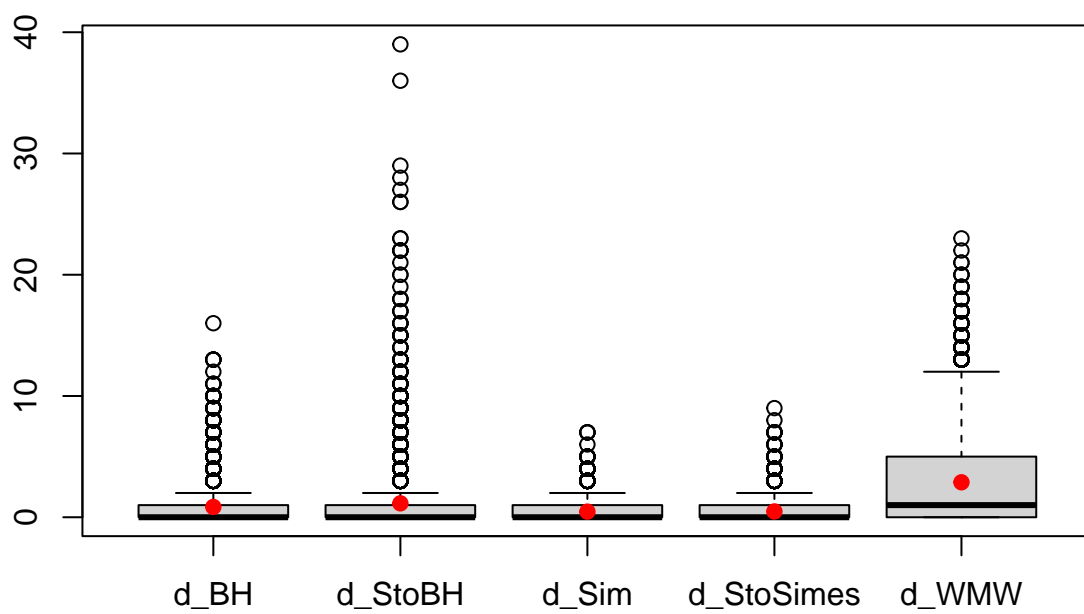
## Digits | Number of discoveries with 10 outliers



## Digits | Number of discoveries with 11 outliers

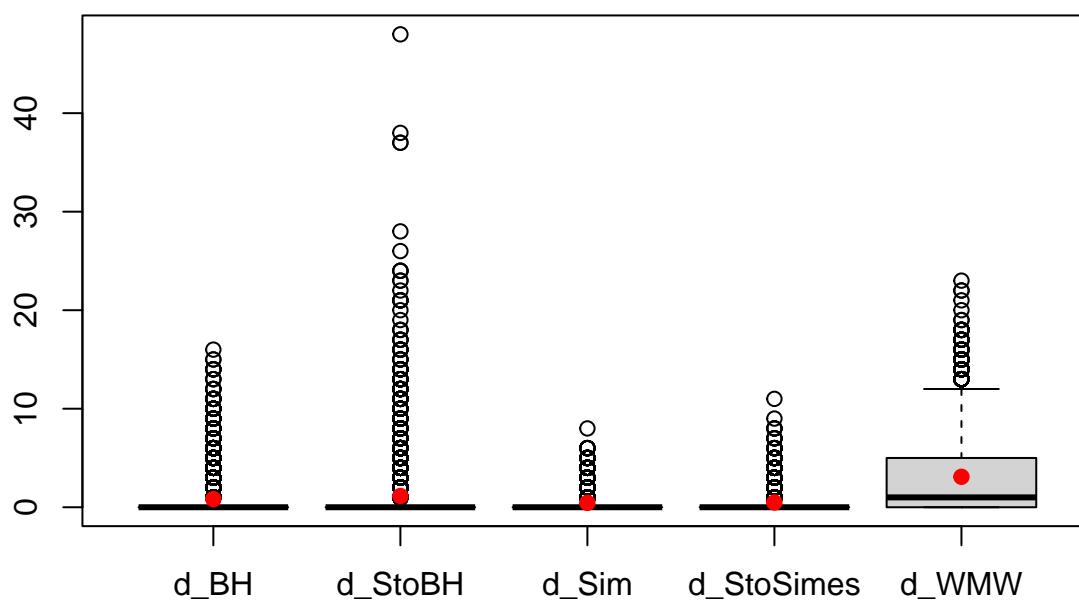


## Digits | Number of discoveries with 12 outliers

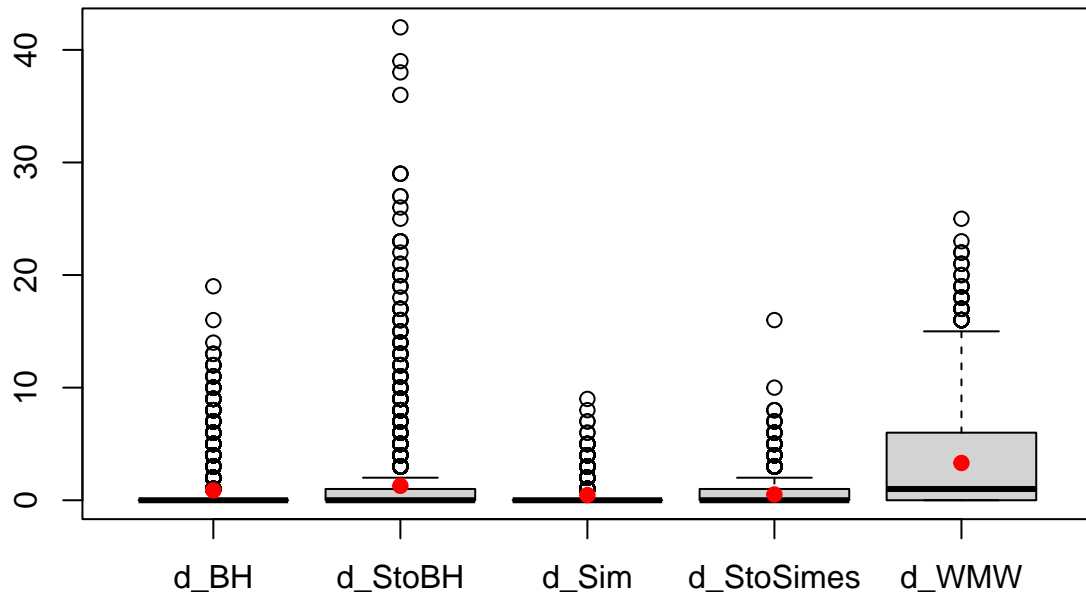




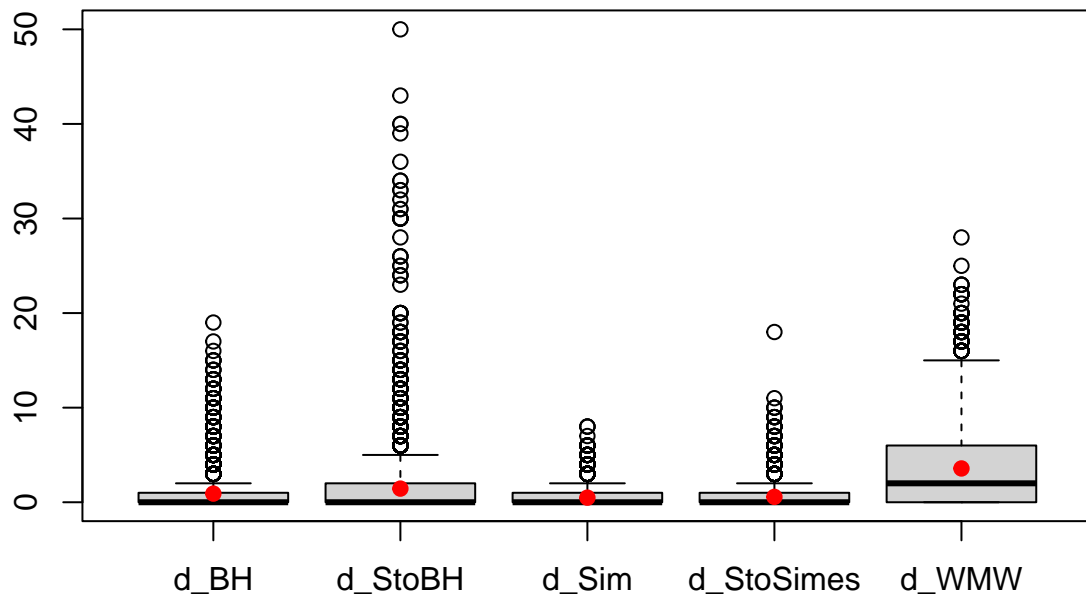
## Digits | Number of discoveries with 13 outliers



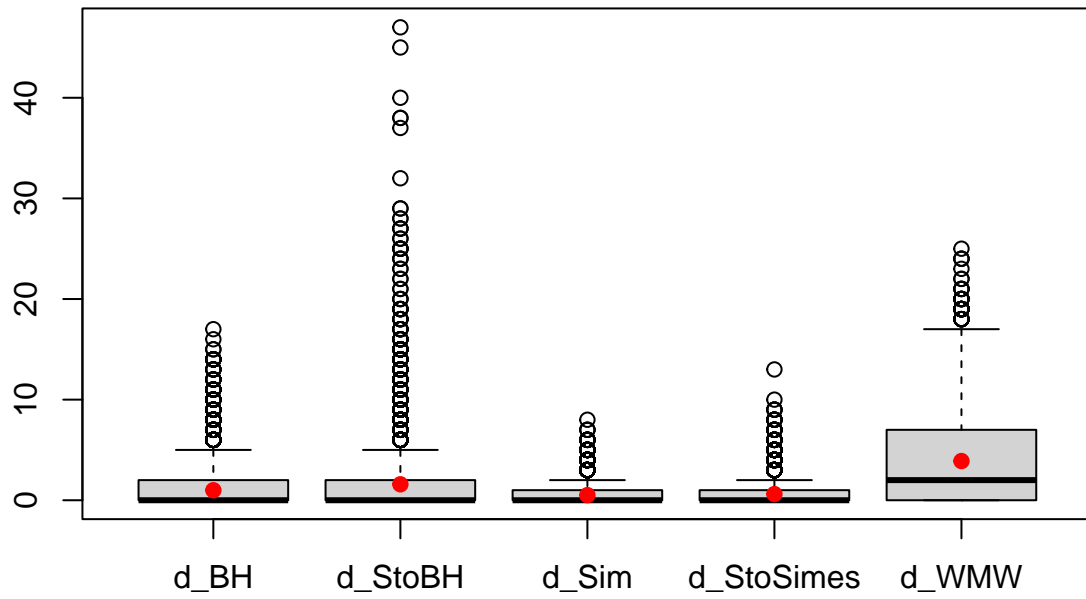
## Digits | Number of discoveries with 14 outliers



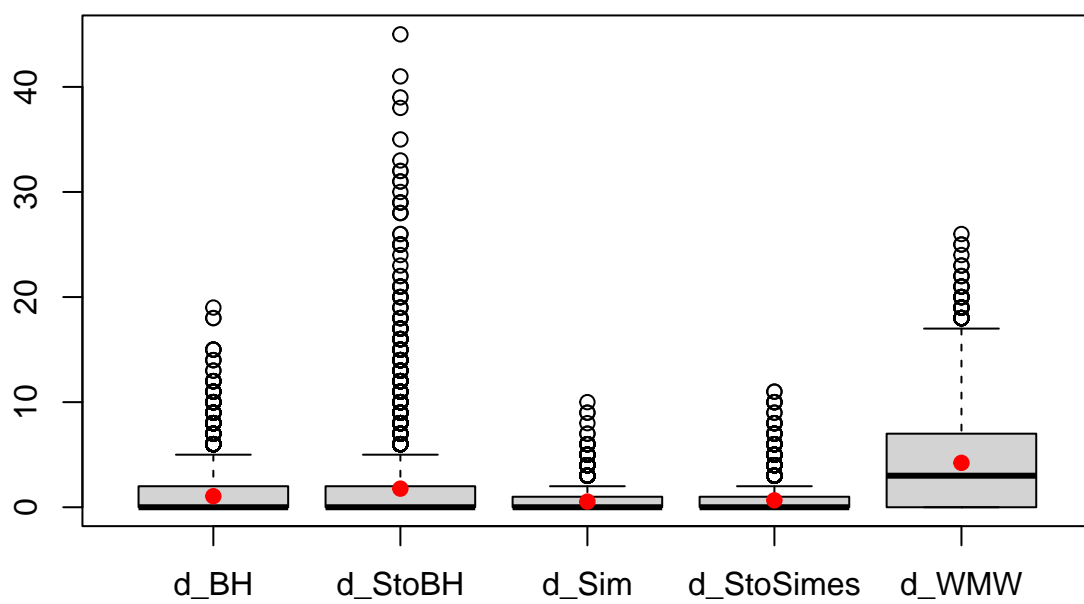
## Digits | Number of discoveries with 15 outliers



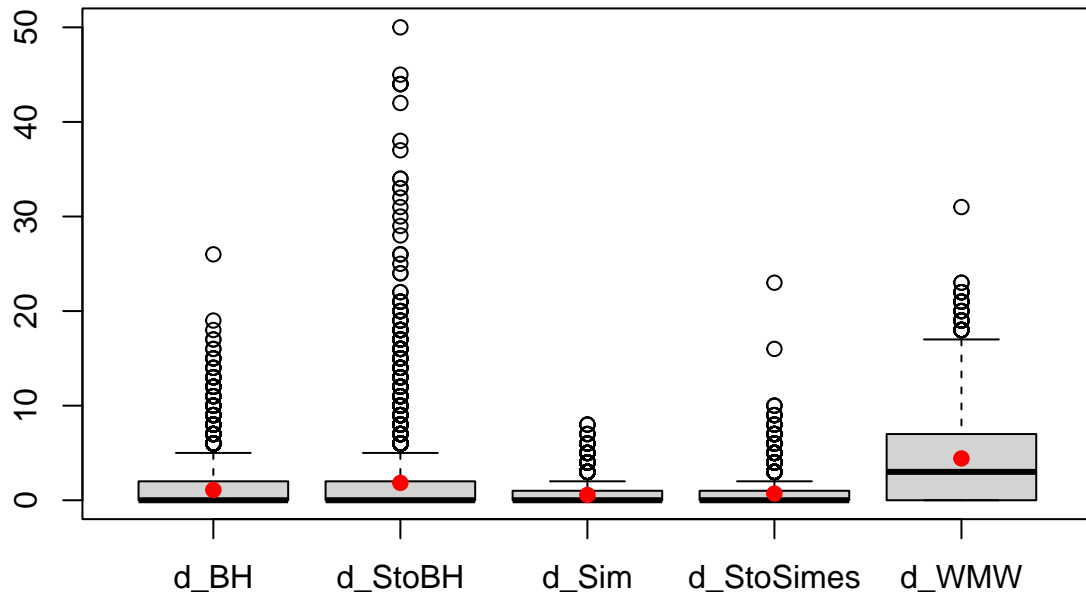
## Digits | Number of discoveries with 16 outliers



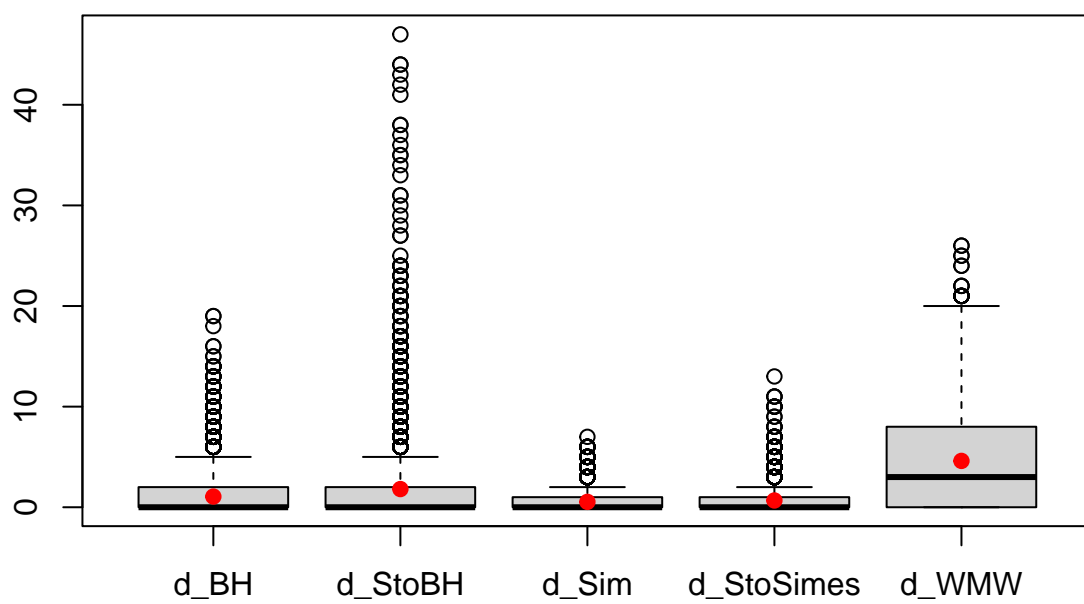
## Digits | Number of discoveries with 17 outliers



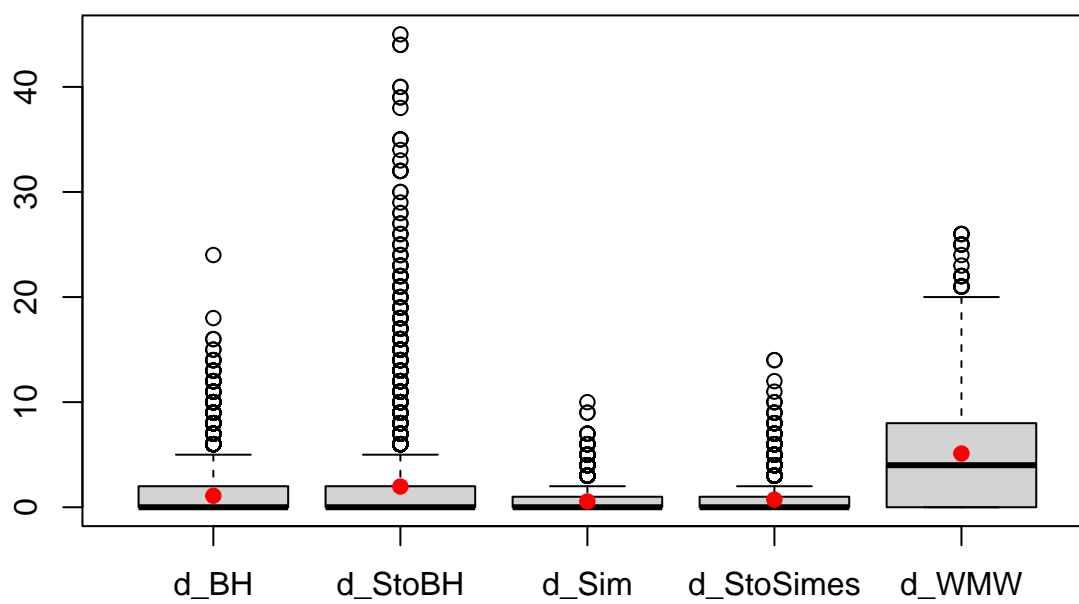
## Digits | Number of discoveries with 18 outliers



## Digits | Number of discoveries with 19 outliers

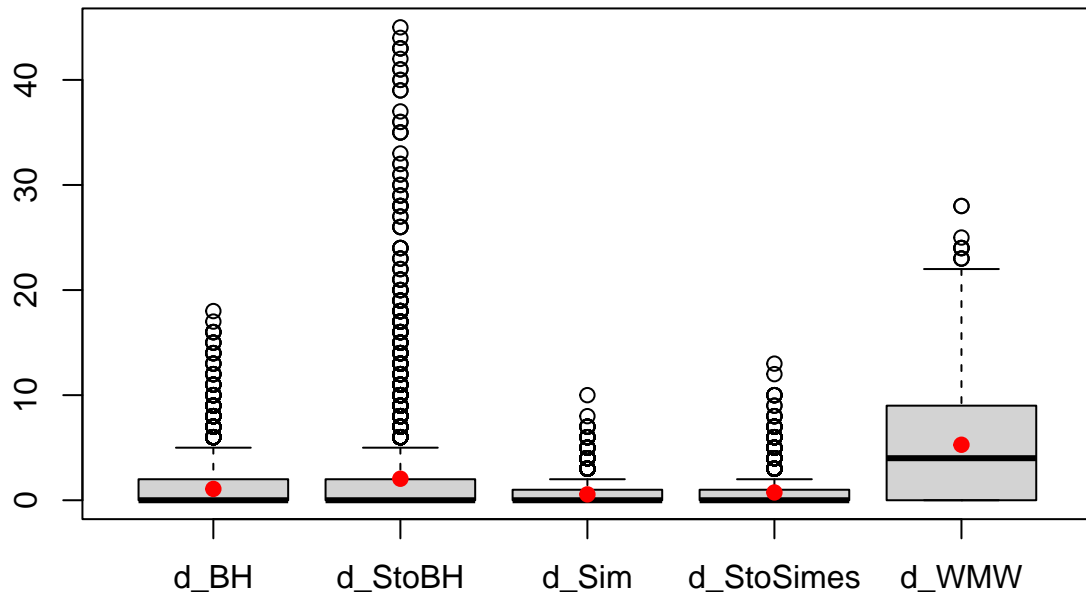


## Digits | Number of discoveries with 20 outliers

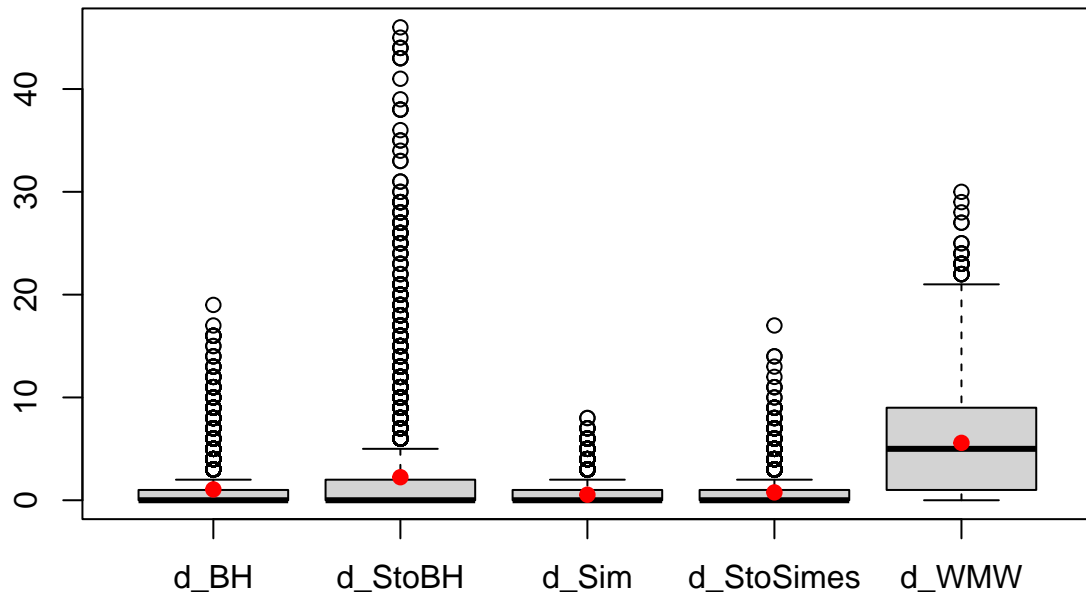




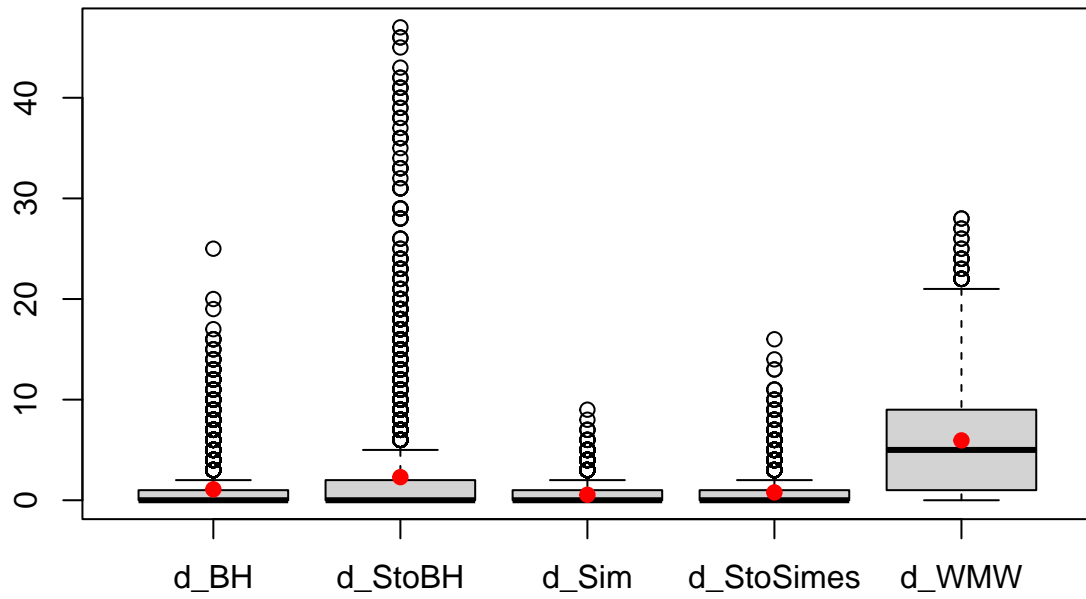
## Digits | Number of discoveries with 21 outliers



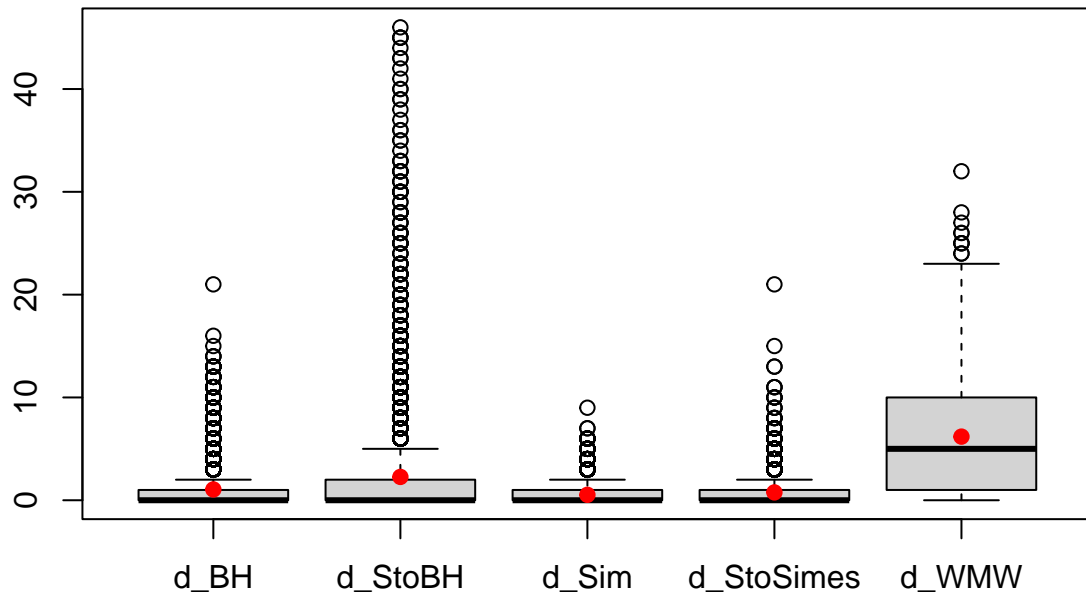
## Digits | Number of discoveries with 22 outliers



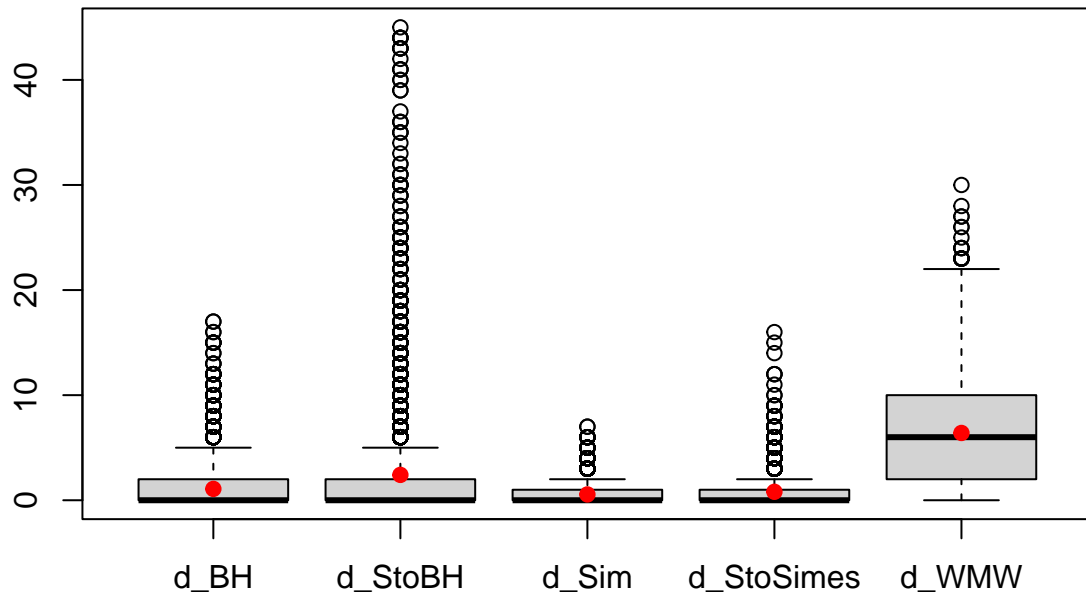
## Digits | Number of discoveries with 23 outliers



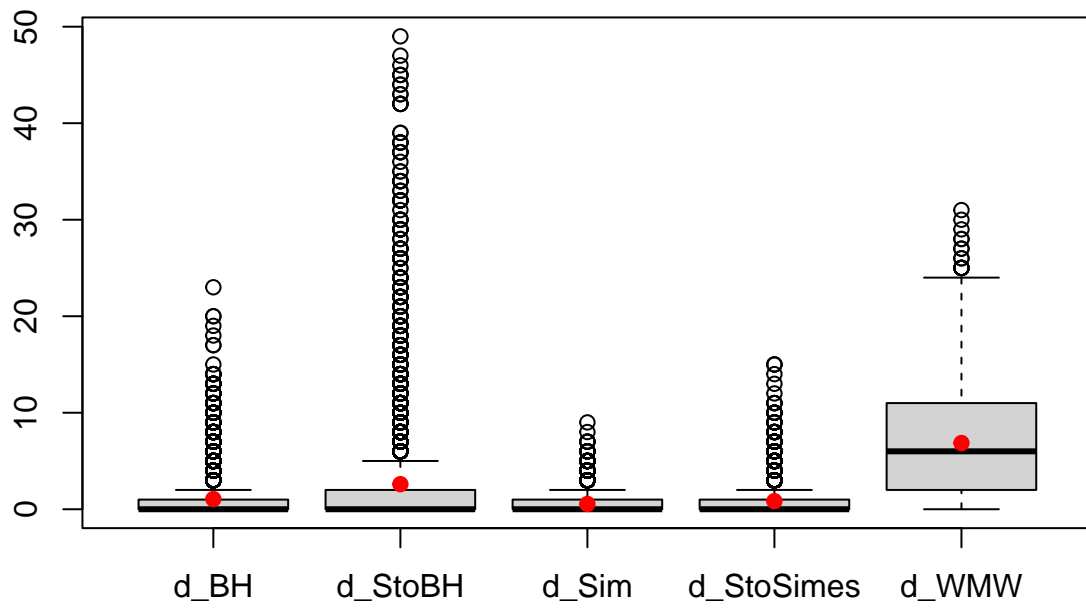
## Digits | Number of discoveries with 24 outliers



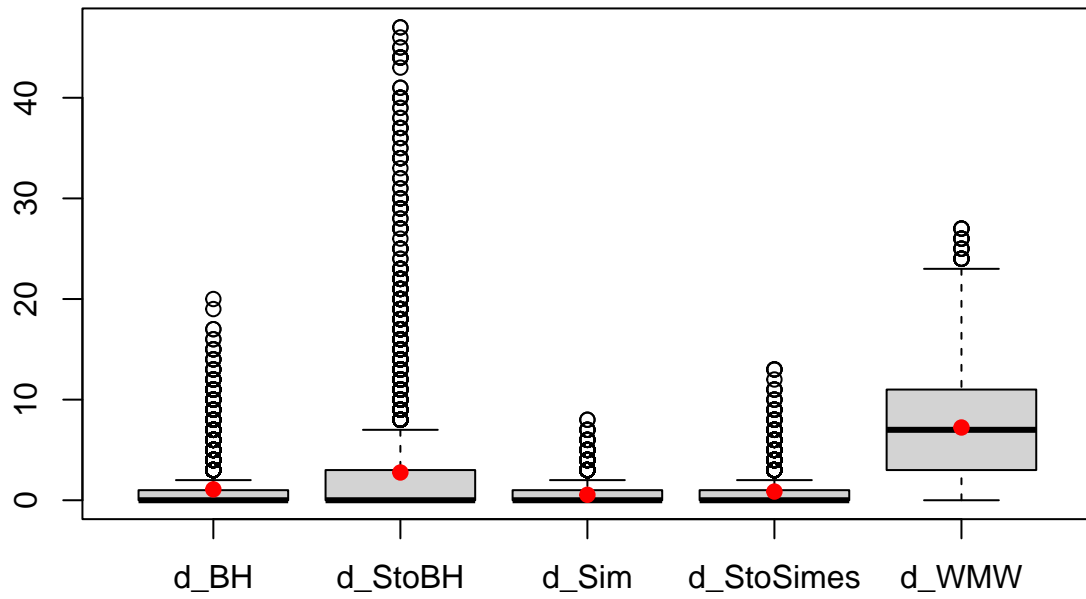
## Digits | Number of discoveries with 25 outliers



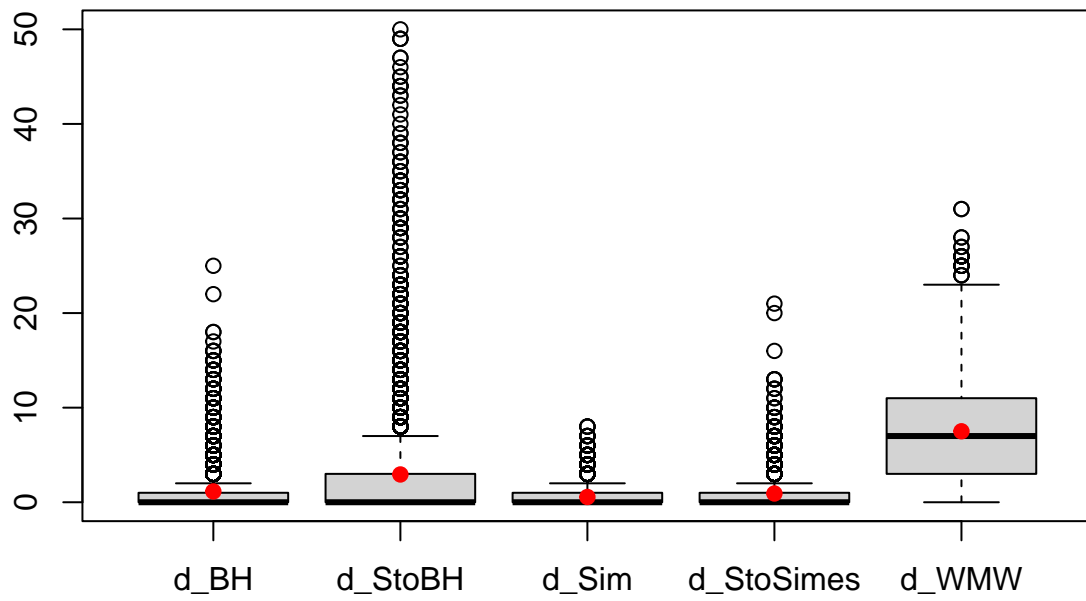
## Digits | Number of discoveries with 26 outliers



## Digits | Number of discoveries with 27 outliers

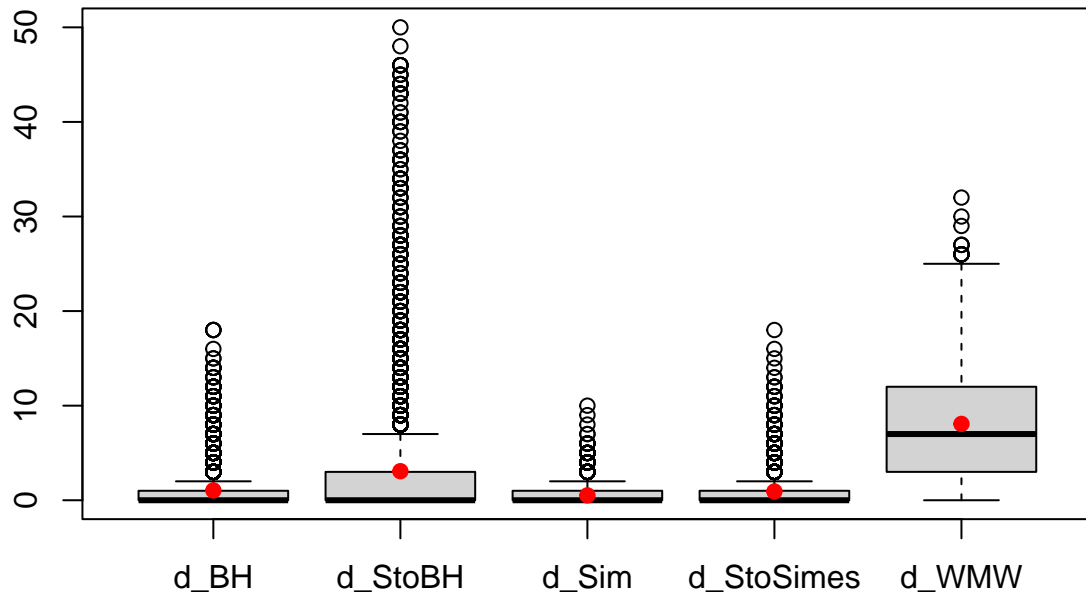


## Digits | Number of discoveries with 28 outliers

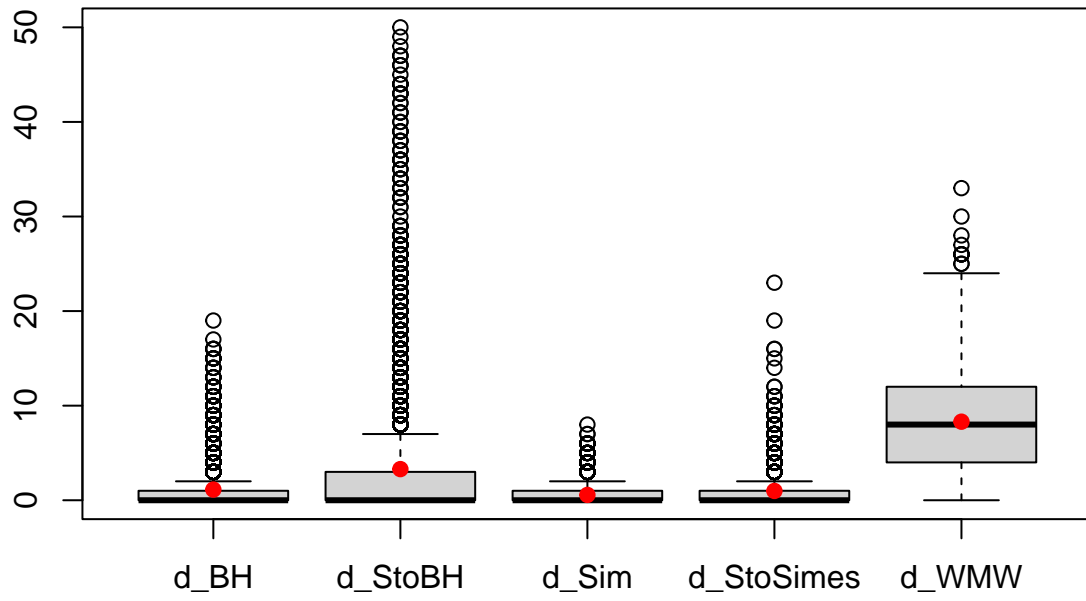




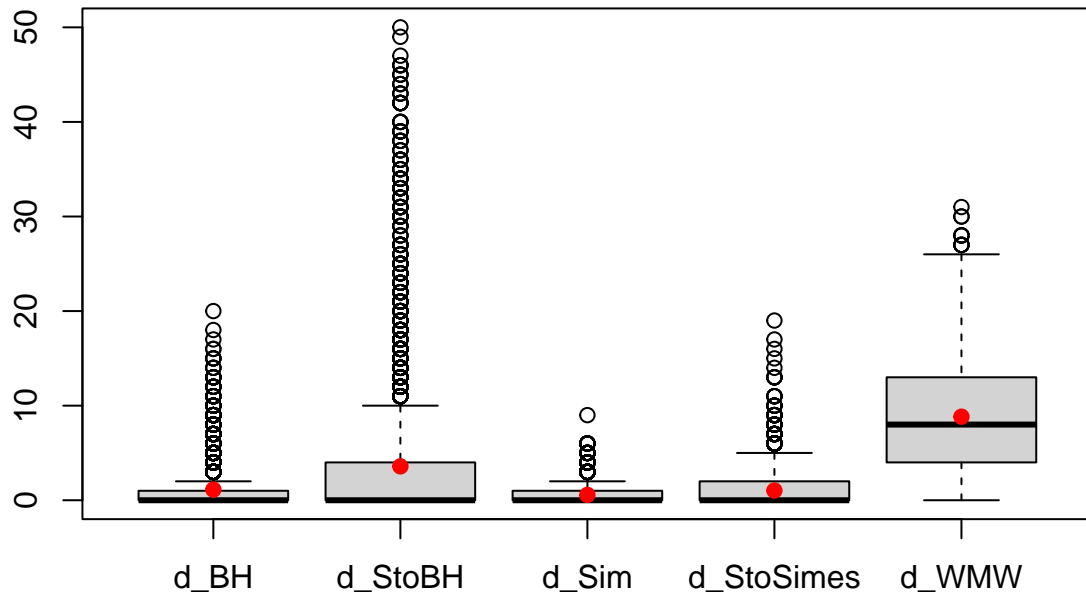
## Digits | Number of discoveries with 29 outliers



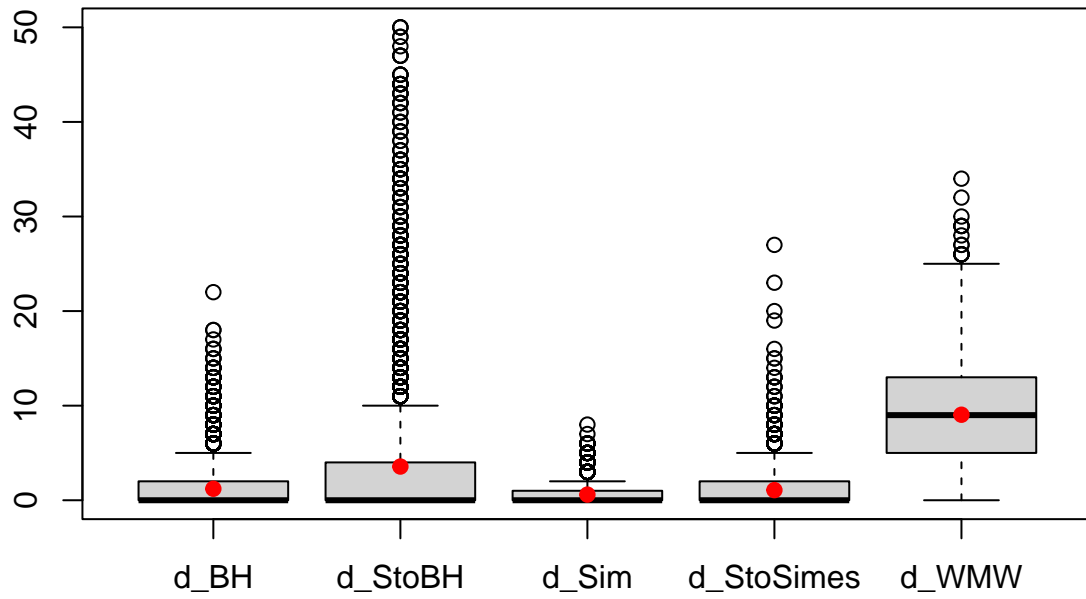
## Digits | Number of discoveries with 30 outliers



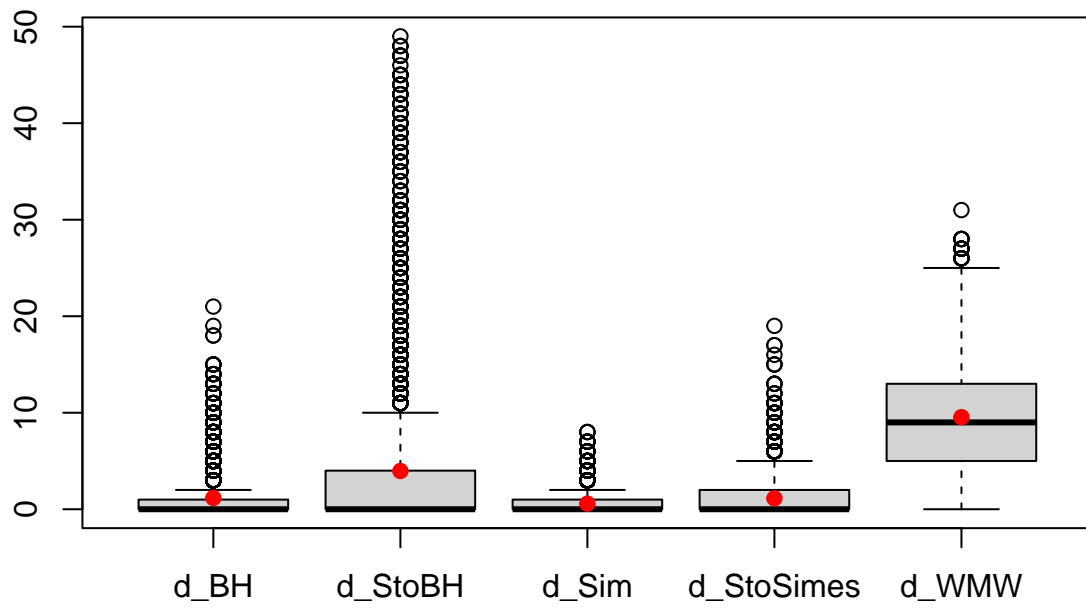
## Digits | Number of discoveries with 31 outliers



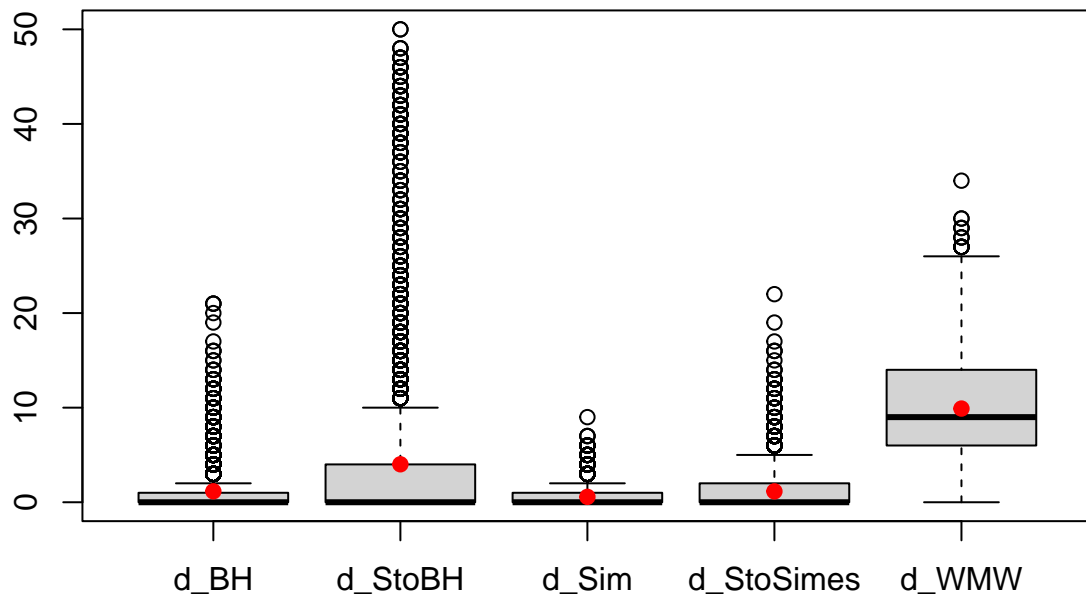
## Digits | Number of discoveries with 32 outliers



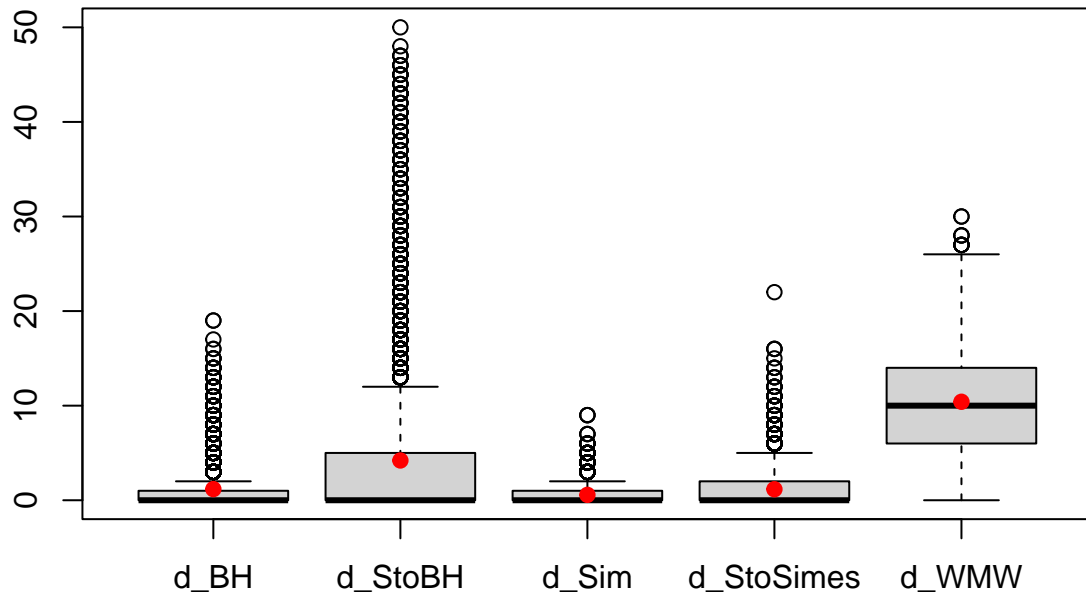
## Digits | Number of discoveries with 33 outliers



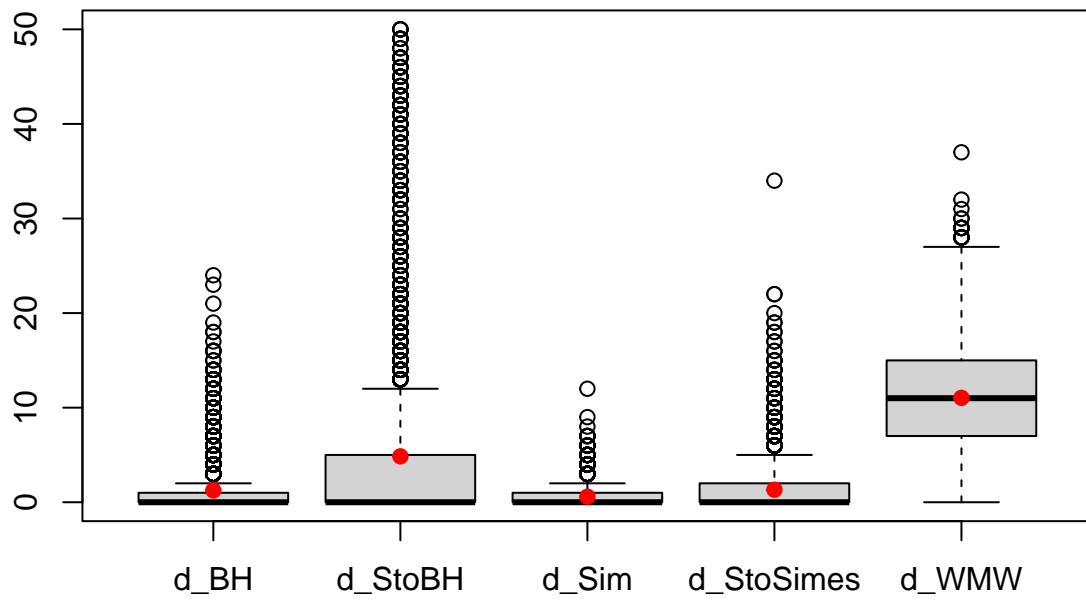
## Digits | Number of discoveries with 34 outliers



## Digits | Number of discoveries with 35 outliers

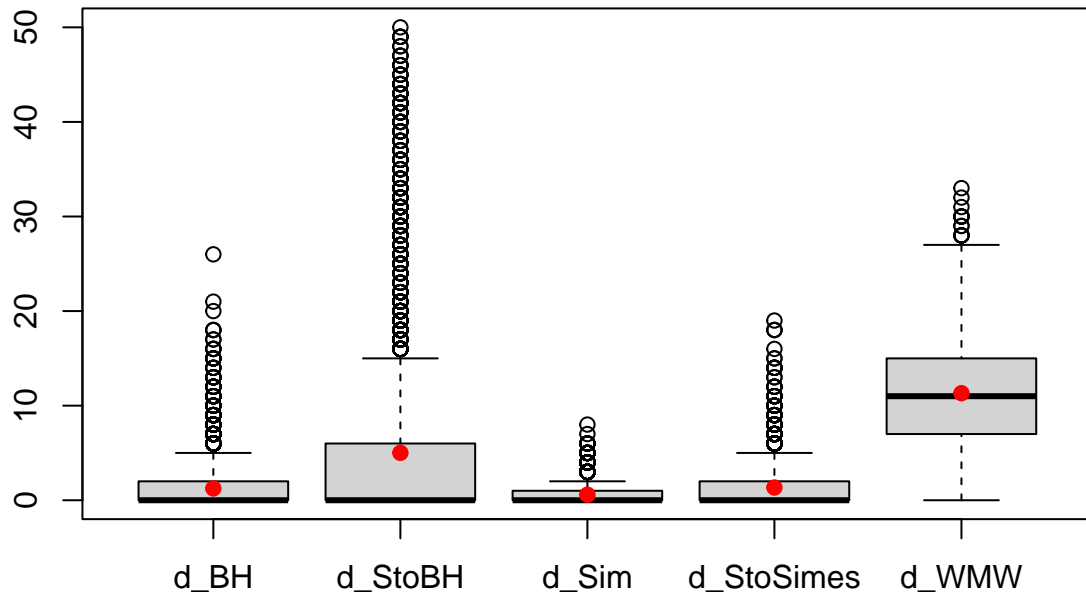


## Digits | Number of discoveries with 36 outliers

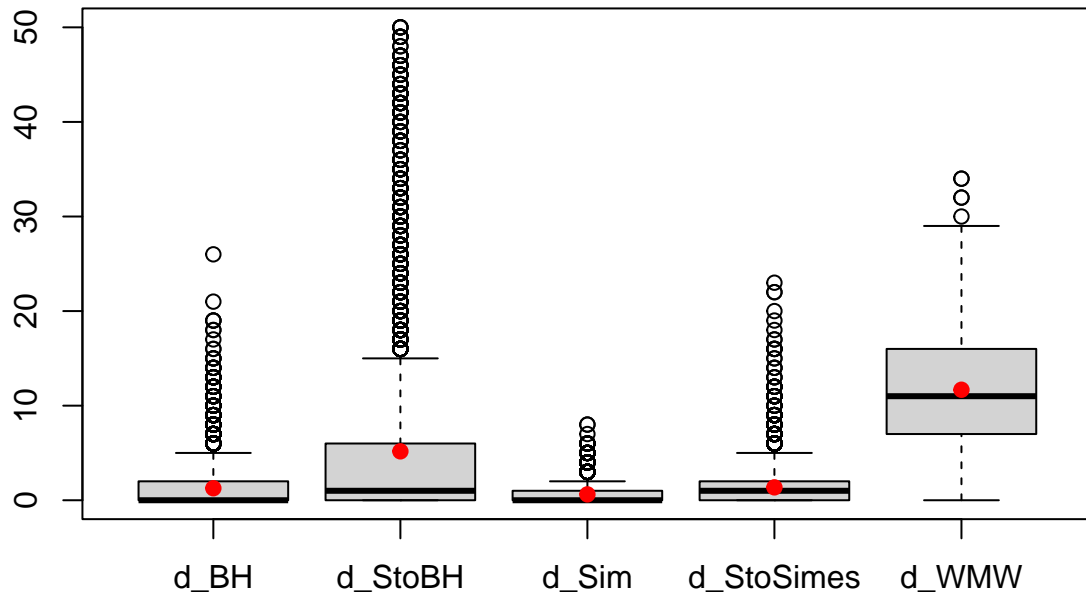




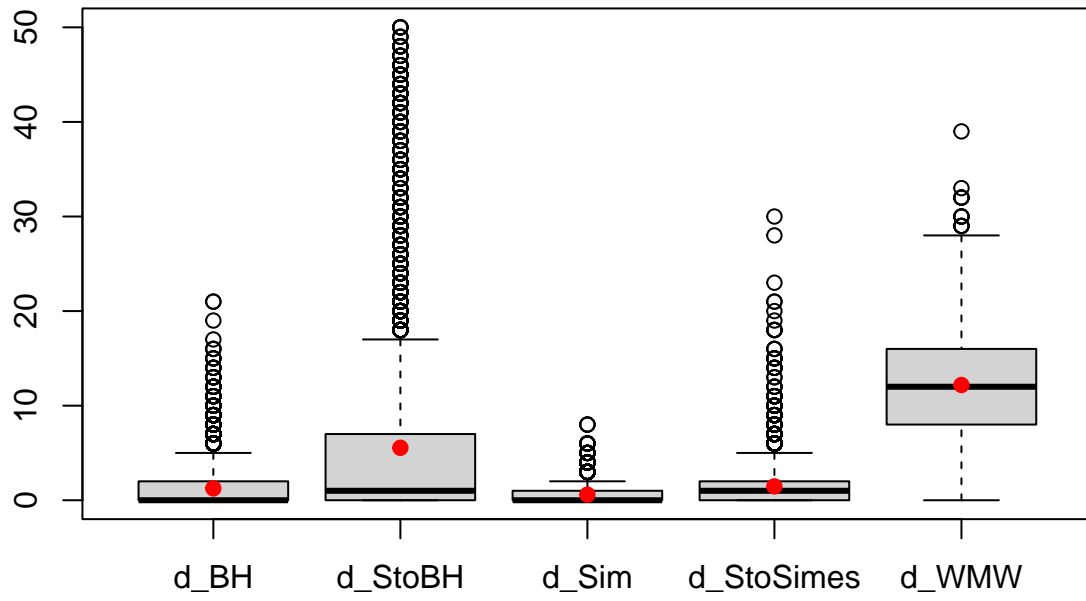
## Digits | Number of discoveries with 37 outliers



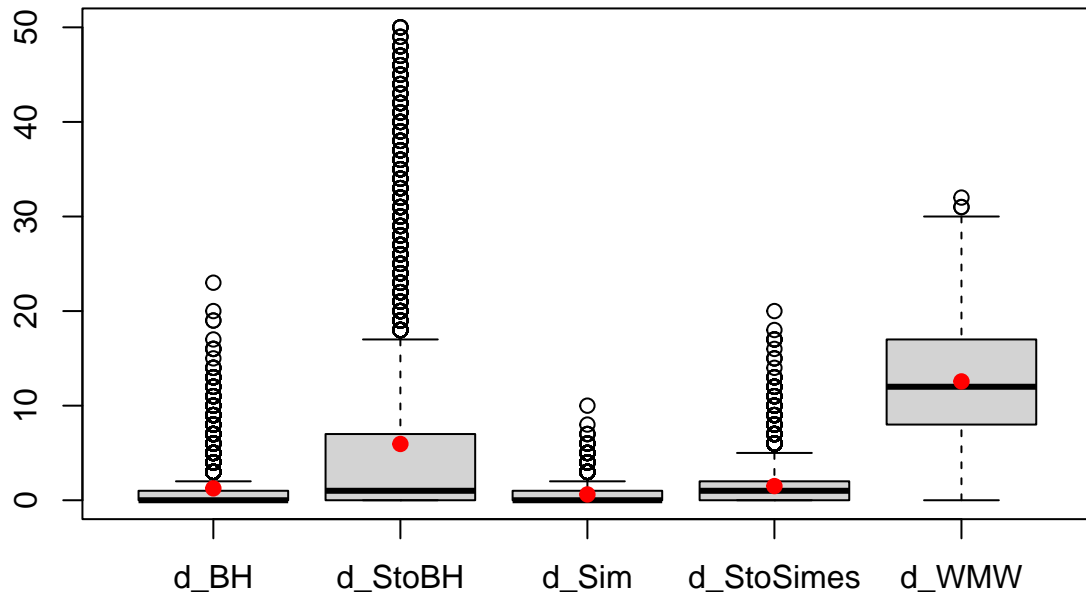
## Digits | Number of discoveries with 38 outliers



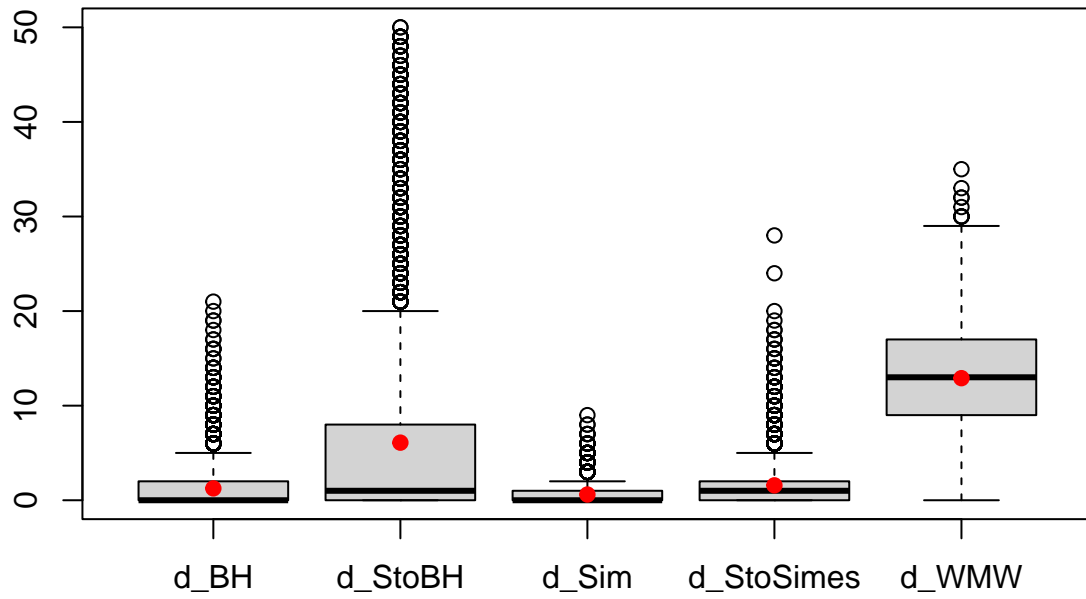
## Digits | Number of discoveries with 39 outliers



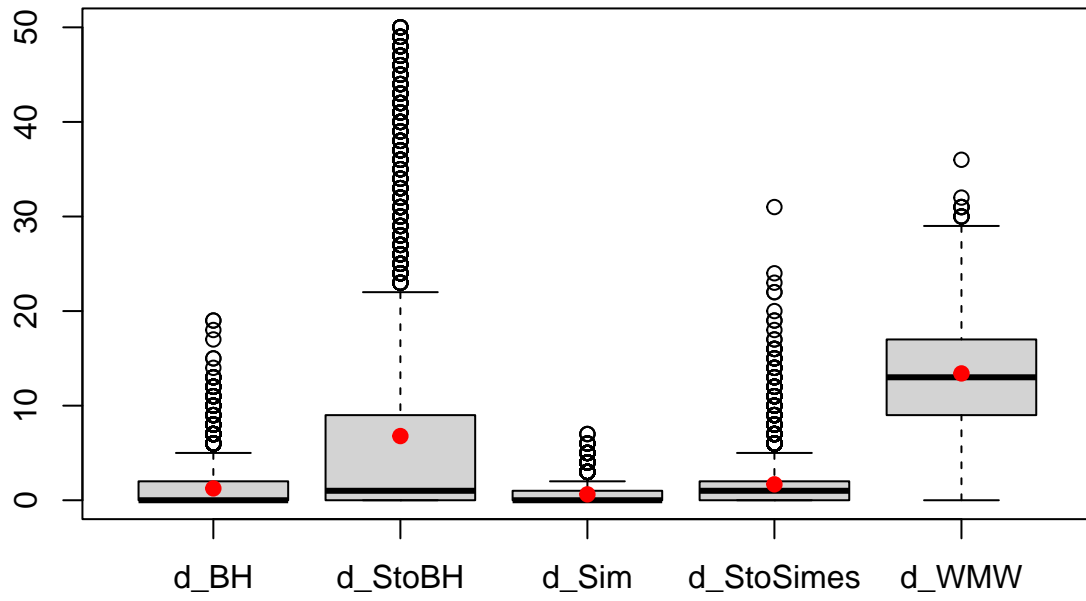
## Digits | Number of discoveries with 40 outliers



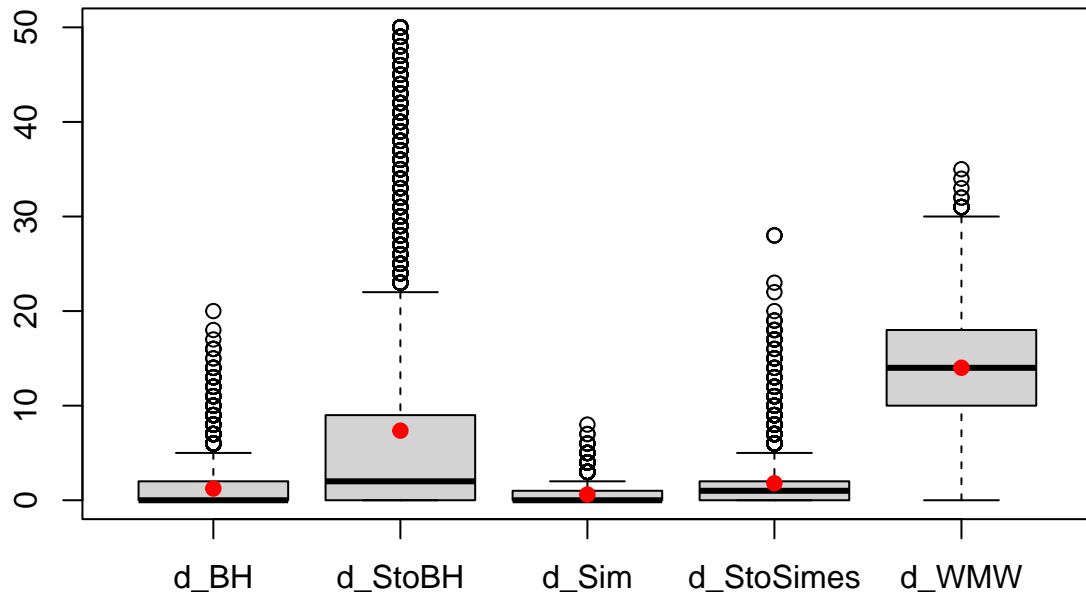
## Digits | Number of discoveries with 41 outliers



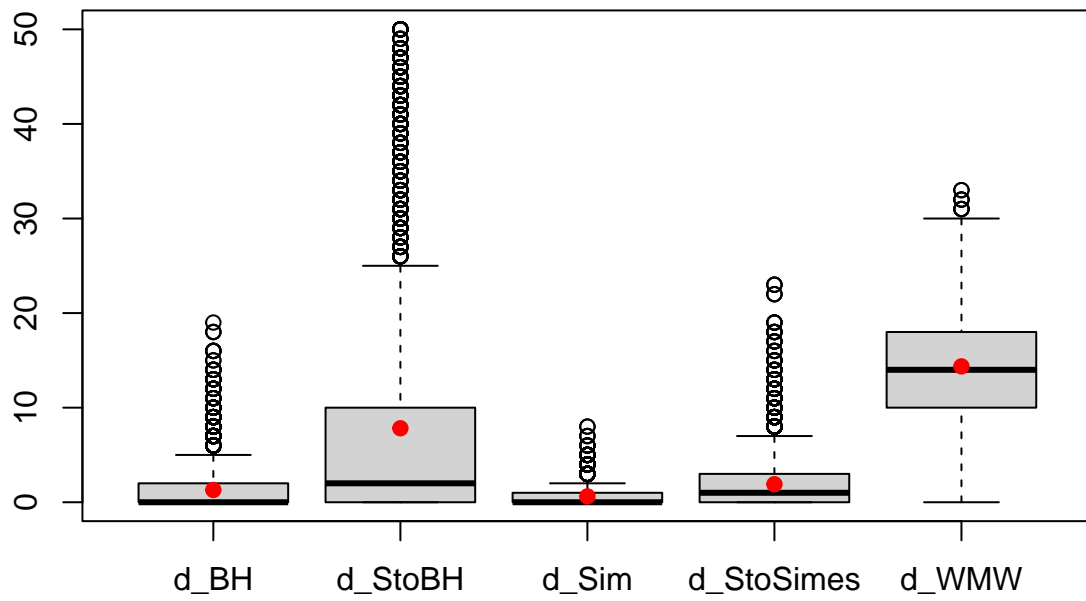
## Digits | Number of discoveries with 42 outliers



## Digits | Number of discoveries with 43 outliers

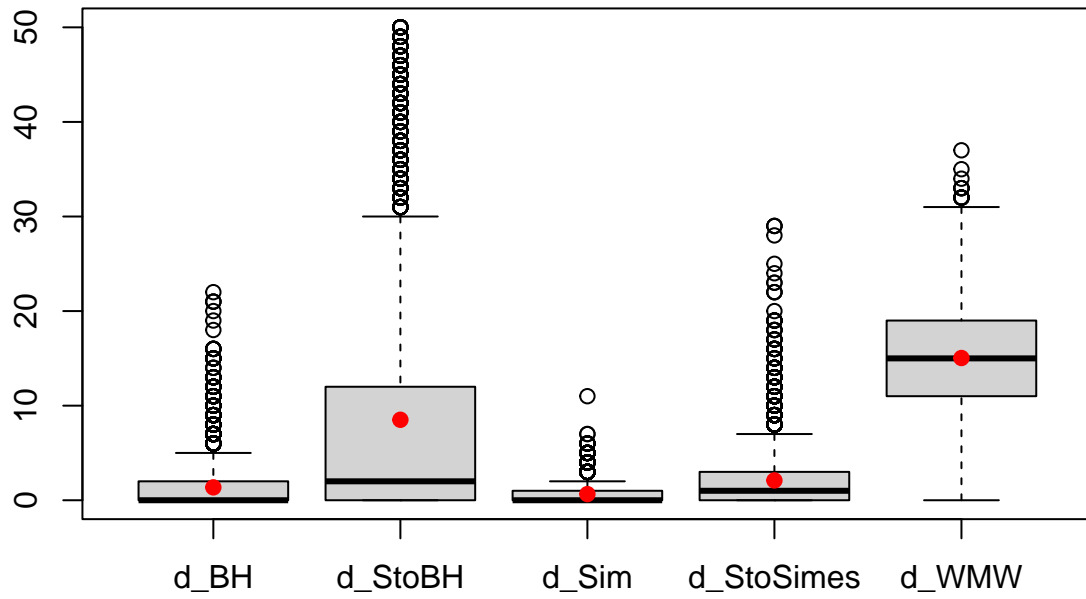


## Digits | Number of discoveries with 44 outliers

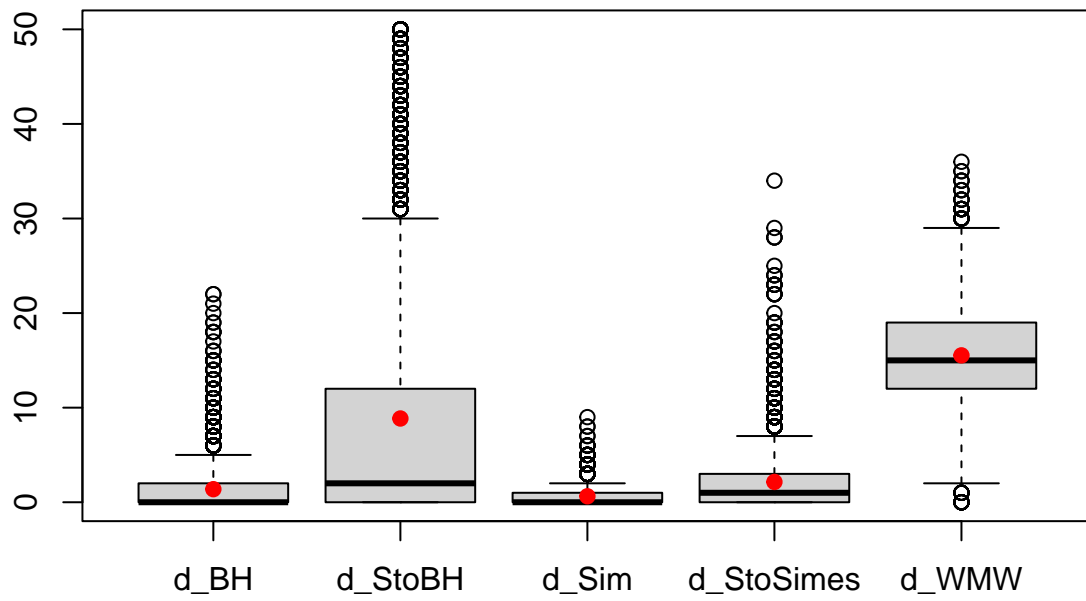




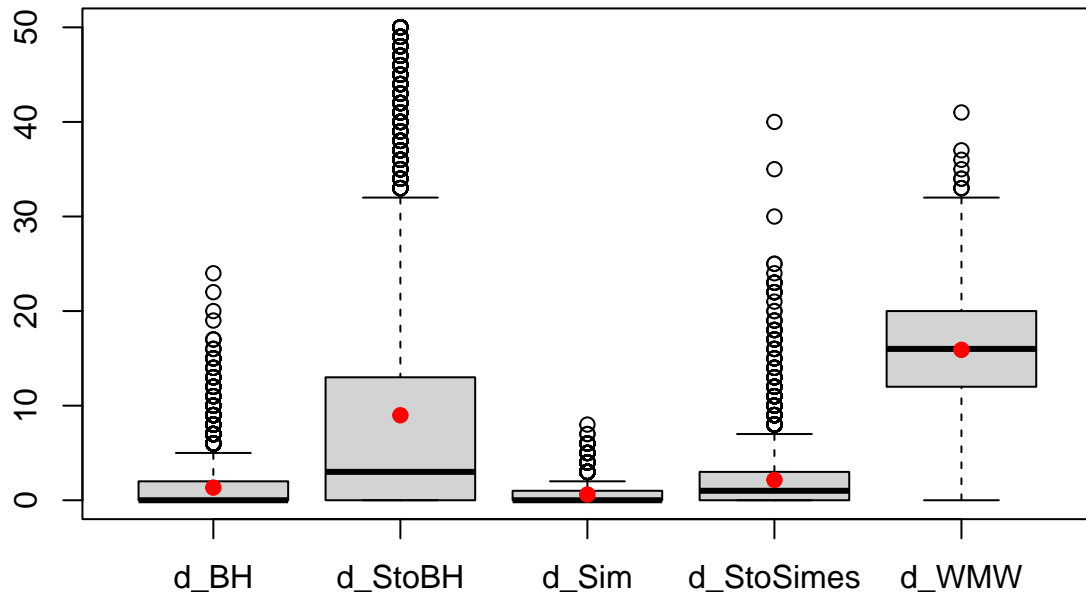
## Digits | Number of discoveries with 45 outliers



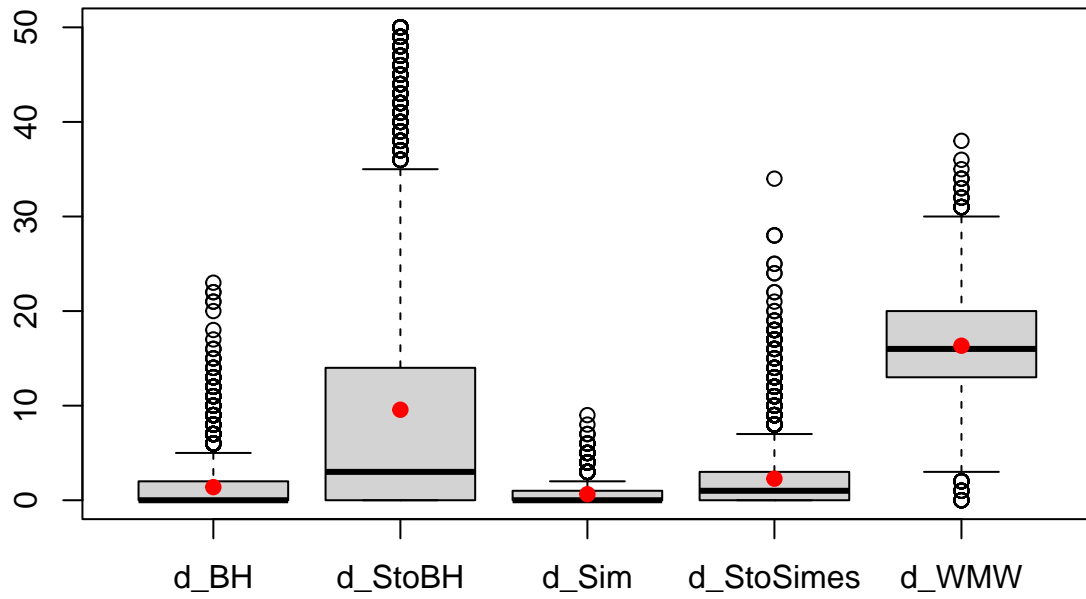
## Digits | Number of discoveries with 46 outliers



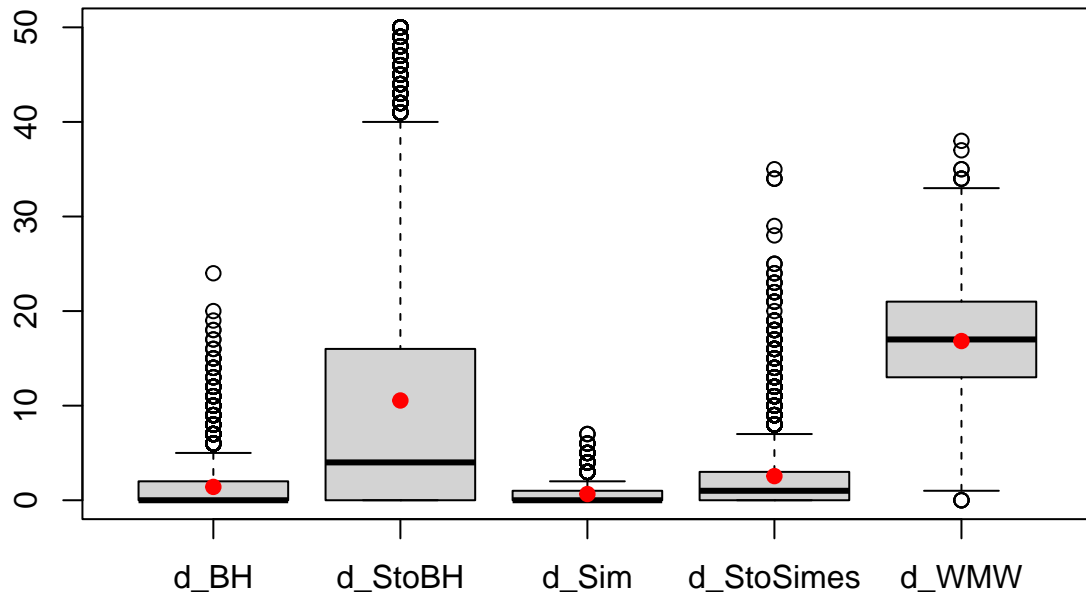
## Digits | Number of discoveries with 47 outliers



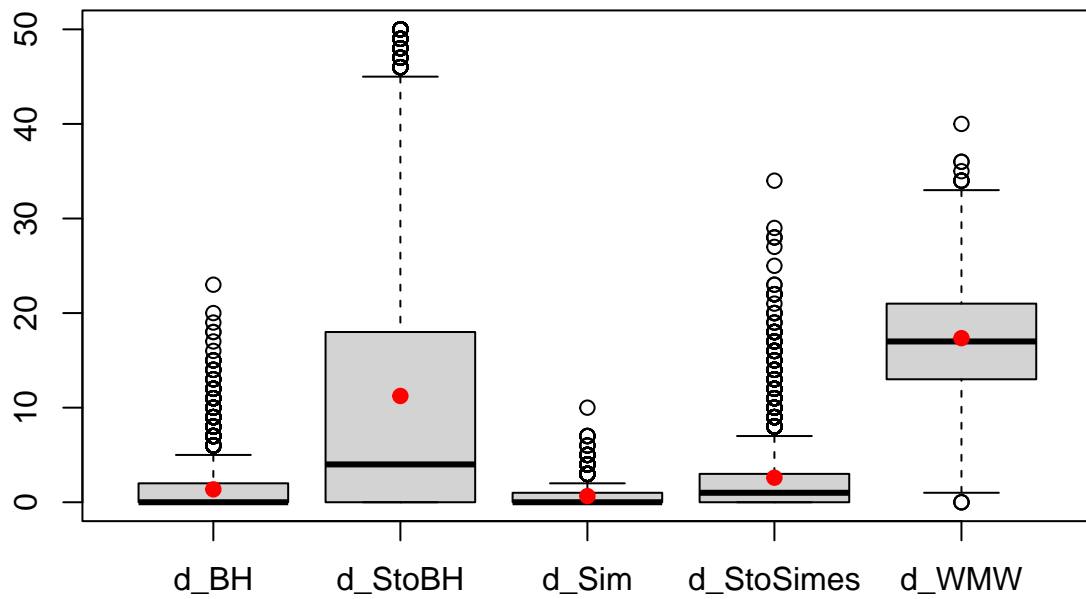
## Digits | Number of discoveries with 48 outliers



## Digits | Number of discoveries with 49 outliers



## Digits | Number of discoveries with 50 outliers



```
resDigits_alpha02_k2_alln1s = results
save(resDigits_alpha02_k2_alln1s,
      file="~/nout/trials/RealData/PowerStudy/New!/alpha0.2/CreditCardOnly0.2/Lehmann k=2/resCreditCard_
```