```
#####################################################################################################
#####################################################################################################
##                                                                                                 ##
## Supplementary material to Comments on "Outlier-sums for differential gene expression analysis" ##
##                                                                                                 ##
## Sarah C. Emerson, Department of Statistics, Oregon State University, Corvallis, OR 97331,       ##
##     emersosa@stat.oregonstate.edu                                                               ##
## Scott S. Emerson, Department of Biostatistics, University of Washington, Seattle, WA 98195      ##
##                                                                                                 ##
#####################################################################################################
#####################################################################################################


###############
##
## The code in this file was used to compute and/or simulate the distribution of "outliers" for
## mixtures of normals.
##
###############

###############
##
## Mixture of normals
##
## rD = nD / nH
##  Healthy population: X ~ a N(0,tau^2) + (1-a) N(1,tau^2)
##  Diseased population: W ~ B(1,p); Y ~ X + W
##
###############

# compute the CDF for a mixture of two normals

ataupCDF <- function (x, a, tau, p, rD, target=0.5) {
    (rD + (1-p)) / (rD + 1) * (a * pnorm(x,0,tau) + (1-a) * pnorm(x,1,tau)) +
        p / (rD + 1) * (a * pnorm(x,1,tau) + (1-a) * pnorm(x,2,tau)) - target
}

# compute the mad for a mixture of two normals

ataupMAD <- function (x, m, a, tau, p, rD, target=0.5) {
    ataupCDF (x+m, a, tau, p, rD) + 0.5 - ataupCDF (m - x, a, tau, p, rD) - 0.5 - target
}

# compute the integral between -\infty and x for uf(u)du, where f(u) is the density for N(mu,sigma^2)
# this also simulates the value of the integral as validation of the code

ptlMean <- function (x, mu, sigma) {
    y <- rnorm(1000000,mu,sigma)
    c(mu * pnorm((x-mu) / sigma) - sigma / sqrt(2 * pi) * exp(- (x - mu)^2 / 2 / sigma^2),
    mean(ifelse(y < x, y, 0)))
}

# a function to compute the median and mad of healthy only or combined healthy and diseased
# where the healthy are a mixture of normals, and the diseased is a shift hypothesis in a subset
# the function finds the true values in a search, as well as simulating the results as validation

medMAD <- function (a, tau, p, rD=1, nH=1000000) {
    X <- rnorm(nH,0,tau) + rbinom(nH,1,1-a)
    Y <- rnorm(rD * nH,0,tau) + rbinom(rD * nH,1,1-a) + rbinom(rD * nH,1,p)
    mX <- uniroot (ataupCDF, lower= -0.5, upper= 2.5, a=a, tau=tau, p=0, rD=0)$root
    mComb <- uniroot (ataupCDF, lower= -0.5, upper= 2.5, a=a, tau=tau, p=p, rD=rD)$root
    tmadX <- uniroot (ataupMAD, lower= -0.5, upper= 5, a=a, m=mX, tau=tau, p=0, rD=0)$root
    tmadComb <- uniroot (ataupMAD, lower= 0, upper= 5, a=a, m=mComb, tau=tau, p=p, rD=rD)$root
    cbind(sdX = sqrt(var(X)), sdComb = sqrt(var(c(X,Y))),
       tmdnX= mX, tmdnComb= mComb, tmadX= tmadX, tmadComb= tmadComb,
       mdnX=quantile(X,prob=0.5),mdnComb=quantile(c(X,Y),prob=0.5),madX=mad(X,constant=1),madComb=mad(c
```

```
 (X,Y),constant=1))
}

# a function to compute the propensity to outliers for a distribution that is a mixture of
# normals: X ~ a N(0, tau^2) + (1 - a) N(1, tau^2), where outliers are observations that
# exceed the 75th percentile by more than one inter-quartile range.

outlierProb2 <- function (a, tau, nH=1000000) {
    q25 <- uniroot (ataupCDF, lower= -1.5, upper= 7.5, a=a, tau=tau, p=0, rD=0, target=0.25)$root
    q75 <- uniroot (ataupCDF, lower= -1.5, upper= 7.5, a=a, tau=tau, p=0, rD=0, target=0.75)$root
    thresh <- 2 * q75 - q25
    probOutlier <- 1 - ataupCDF (thresh, a=a, tau=tau, p=0, rD=0, target=0)
    mX <- uniroot (ataupCDF, lower= -0.5, upper= 2.5, a=a, tau=tau, p=0, rD=0)$root
    tmadX <- uniroot (ataupMAD, lower= -0.5, upper= 5, a=a, m=mX, tau=tau, p=0, rD=0)$root
    X <- rnorm(nH,0,tau) + rbinom(nH,1,1-a)
    X <- ifelse(X > thresh, (X-mX)/tmadX, 0)
    meanOutlier <- mean(X)
    sdOutlier <- sqrt(var(X))
    cbind(a=a, tau=tau, probOutlier=probOutlier, meanOutlier=meanOutlier, sdOutlier=sdOutlier)
}

# analogous functions for a mixture of 3 normals

ataup3CDF <- function (x, a, tau, p, rD, target=0.5) {
    (rD + (1-p)) / (rD + 1) * (a * pnorm(x,0,tau) + (1-a)/2 * pnorm(x,-1,tau) + (1-a)/2 * pnorm(x,1,tau)) +
        p / (rD + 1) * (a * pnorm(x,1,tau) + (1-a)/2 * pnorm(x,0,tau) + (1-a)/2 * pnorm(x,2,tau)) - target
}

ataup3MAD <- function (x, m, a, tau, p, rD, target=0.5) {
    ataup3CDF (x+m, a, tau, p, rD) + 0.5 - ataup3CDF (m - x, a, tau, p, rD) - 0.5 - target
}

outlierProb3 <- function (a, tau, nH=1000000) {
    q25 <- uniroot (ataup3CDF, lower= -12.5, upper= 17.5, a=a, tau=tau, p=0, rD=0, target=0.25)$root
    q75 <- uniroot (ataup3CDF, lower= -12.5, upper= 17.5, a=a, tau=tau, p=0, rD=0, target=0.75)$root
    thresh <- 2 * q75 - q25
    probOutlier <- 1 - ataup3CDF (thresh, a=a, tau=tau, p=0, rD=0, target=0)
    mX <- uniroot (ataup3CDF, lower= -12.5, upper= 12.5, a=a, tau=tau, p=0, rD=0)$root
    tmadX <- uniroot (ataup3MAD, lower= -12.5, upper= 15, a=a, m=mX, tau=tau, p=0, rD=0)$root
    X <- rnorm(nH,0,tau) + rbinom(nH,1,1-a) * c(-1,1)[rbinom(nH,1,0.5)+1]
    X <- ifelse(X > thresh, (X-mX)/tmadX, 0)
    meanOutlier <- mean(X)
    sdOutlier <- sqrt(var(X))
    cbind(a=a, tau=tau, probOutlier=probOutlier, meanOutlier=meanOutlier, sdOutlier=sdOutlier)
}

####################################################################################################
####################################################################################################

###############
##
## Median and MAD of the healthy group (X) and the combined sample (X & Y)
## for a variety of mixture distributions and subgroup sizes.  Results from
## numerical integration (tmdnX, tmdnComb, tmadX, tmadComb) as well as from
## simulations (mdnX, mdnComb, madX, madComb).  Also presented are the
## ratios sdRat = sdComb/sdX, tmadRat=tmadComb/tmadX, and madRat=madComb/madX
##
###############

rslt <- NULL
for (a in c(0.2, 0.3, 0.4, 0.5)) {
    for (tau in c(0.1,0.5,1)) {
        for (p in c(0.25,0.5,0.75,1)) {
            rslt <- rbind(rslt,cbind(a=a,tau=tau,p=p,medMAD(a,tau,p)))
        }
    }
```

```
}
options(width=200)
cbind(rslt,sdRat=rslt[,"sdComb"]/rslt[,"sdX"],tmadRat=rslt[,"tmadComb"]/rslt[,"tmadX"],madRat=rslt
[,"madComb"]/rslt[,"madX"])

      a tau    p       sdX    sdComb     tmdnX  tmdnComb       tmadX  tmadComb      mdnX   mdnComb
 madX   madComb    sdRat   tmadRat    madRat
50% 0.2 0.1 0.25 0.4117533 0.5281149 0.9681068 0.9869837 0.09327568 0.1023037 0.9680113 0.9869605
 0.09320930 0.1020811 1.282600 1.0967884 1.0951814
50% 0.2 0.1 0.50 0.4120468 0.5977458 0.9681068 1.0096585 0.09327568 0.1203855 0.9681879 1.0096848
 0.09320997 0.1201805 1.450675 1.2906424 1.2893524
50% 0.2 0.1 0.75 0.4119183 0.6354388 0.9681068 1.0391261 0.09327568 0.1625570 0.9681310 1.0390077
 0.09310750 0.1620515 1.542633 1.7427582 1.7404777
50% 0.2 0.1 1.00 0.4122482 0.6480590 0.9681068 1.0841430 0.09327568 0.5025800 0.9680152 1.0840796
 0.09324146 0.5568710 1.572012 5.3881145 5.9723544
50% 0.2 0.5 0.25 0.6404638 0.7208651 0.8550051 0.9411276 0.41896836 0.4558069 0.8550664 0.9401454
 0.41959072 0.4558151 1.125536 1.0879268 1.0863326
50% 0.2 0.5 0.50 0.6398418 0.7733663 0.8550051 1.0429134 0.41896836 0.5012704 0.8551081 1.0434611
 0.41859505 0.5013406 1.208684 1.1964396 1.1976744
50% 0.2 0.5 0.75 0.6401320 0.8033391 0.8550051 1.1653033 0.41896836 0.5469828 0.8547142 1.1647563
 0.41795010 0.5472513 1.254958 1.3055467 1.3093701
50% 0.2 0.5 1.00 0.6402498 0.8123727 0.8550051 1.3116186 0.41896836 0.5747335 0.8563823 1.3120638
 0.41902869 0.5746644 1.268837 1.3717826 1.3714201
50% 0.2 1.0 0.25 1.0758640 1.1259005 0.8155714 0.9281232 0.72552773 0.7562448 0.8186210 0.9296415
 0.72406502 0.7554443 1.046508 1.0423375 1.0433377
50% 0.2 1.0 0.50 1.0769455 1.1606980 0.8155714 1.0495857 0.72552773 0.7822739 0.8160654 1.0486187
 0.72535152 0.7820657 1.077768 1.0782137 1.0781886
50% 0.2 1.0 0.75 1.0779035 1.1804479 0.8155714 1.1784252 0.72552773 0.8001640 0.8188032 1.1803438
 0.72715761 0.8005813 1.095133 1.1028717 1.1009735
50% 0.2 1.0 1.00 1.0763788 1.1882382 0.8155714 1.3118447 0.72552773 0.8069998 0.8141393 1.3117228
 0.72527769 0.8080175 1.103922 1.1122935 1.1140802
50% 0.3 0.1 0.25 0.4689005 0.5737467 0.9433896 0.9656180 0.12400785 0.1268914 0.9434367 0.9654961
 0.12405107 0.1270211 1.223600 1.0232528 1.0239418
50% 0.3 0.1 0.50 0.4691625 0.6381273 0.9433896 0.9895333 0.12400785 0.1390830 0.9434241 0.9894856
 0.12407306 0.1389776 1.360141 1.1215659 1.1201270
50% 0.3 0.1 0.75 0.4690305 0.6741016 0.9433896 1.0171465 0.12400785 0.1715314 0.9435784 1.0172040
 0.12391168 0.1715857 1.437224 1.3832298 1.3847419
50% 0.3 0.1 1.00 0.4687216 0.6854159 0.9433896 1.0524475 0.12400785 0.5038325 0.9434474 1.0524260
 0.12369439 0.4064404 1.462309 4.0629082 3.2858431
50% 0.3 0.5 0.25 0.6792066 0.7548473 0.7571311 0.8498249 0.46837520 0.4989797 0.7570010 0.8501179
 0.46874301 0.4986061 1.111366 1.0653419 1.0637088
50% 0.3 0.5 0.50 0.6786934 0.8050252 0.7571311 0.9542573 0.46837520 0.5355140 0.7575832 0.9543279
 0.46817947 0.5355082 1.186140 1.1433441 1.1438097
50% 0.3 0.5 0.75 0.6782696 0.8335123 0.7571311 1.0734124 0.46837520 0.5717382 0.7558282 1.0724545
 0.46855339 0.5722860 1.228881 1.2206842 1.2213891
50% 0.3 0.5 1.00 0.6788890 0.8431776 0.7571311 1.2095935 0.46837520 0.5948274 0.7561208 1.2105117
 0.46861476 0.5944860 1.241996 1.2699805 1.2686027
50% 0.3 1.0 0.25 1.1006747 1.1487473 0.7142088 0.8275347 0.74507025 0.7748211 0.7148830 0.8274923
 0.74503253 0.7749522 1.043676 1.0399303 1.0401589
50% 0.3 1.0 0.50 1.0999653 1.1816475 0.7142088 0.9492589 0.74507025 0.7998162 0.7146810 0.9493547
 0.74386725 0.7992137 1.074259 1.0734775 1.0744036
50% 0.3 1.0 0.75 1.0995633 1.2009819 0.7142088 1.0778628 0.74507025 0.8168729 0.7170855 1.0779454
 0.74411616 0.8159216 1.092235 1.0963703 1.0964976
50% 0.3 1.0 1.00 1.1009777 1.2092409 0.7142088 1.2106538 0.74507025 0.8233568 0.7131674 1.2104810
 0.74570814 0.8231050 1.098334 1.1050728 1.1037897
50% 0.4 0.1 0.25 0.5000520 0.5997529 0.9032624 0.9359287 0.19420976 0.1800284 0.9030457 0.9359216
 0.19462205 0.1805215 1.199381 0.9269792 0.9275492
50% 0.4 0.1 0.50 0.4999548 0.6615282 0.9032624 0.9651409 0.19420976 0.1790375 0.9029204 0.9650053
 0.19485645 0.1798219 1.323176 0.9218770 0.9228429
50% 0.4 0.1 0.75 0.5002087 0.6960915 0.9032624 0.9940281 0.19420976 0.1984283 0.9030838 0.9939591
 0.19457247 0.1984245 1.391602 1.0217218 1.0197975
50% 0.4 0.1 1.00 0.5001245 0.7072702 0.9032624 1.0253341 0.19420976 0.5053829 0.9032035 1.0252991
 0.19418767 0.6127076 1.414188 2.6022526 3.1552343
50% 0.4 0.5 0.25 0.6996078 0.7735949 0.6373650 0.7409752 0.50921923 0.5369736 0.6362857 0.7405691
 0.50967703 0.5366355 1.105755 1.0545039 1.0528933
50% 0.4 0.5 0.50 0.6999871 0.8230587 0.6373650 0.8533276 0.50921923 0.5664185 0.6376096 0.8543372
```

```
 0.50982788 0.5663945 1.175820 1.1123275 1.1109523
50% 0.4 0.5 0.75 0.7003724 0.8514862 0.6373650 0.9746230 0.50921923 0.5931958 0.6358395 0.9747729
 0.50993899 0.5939943 1.215762 1.1649125 1.1648341
50% 0.4 0.5 1.00 0.6998622 0.8600035 0.6373650 1.1052417 0.50921923 0.6088960 0.6374090 1.1051063
 0.50949478 0.6087560 1.228818 1.1957443 1.1948228
50% 0.4 1.0 0.25 1.1151842 1.1627207 0.6083124 0.7226520 0.75779365 0.7871038 0.6089532 0.7224391
 0.75875779 0.7870840 1.042627 1.0386783 1.0373323
50% 0.4 1.0 0.50 1.1127445 1.1940786 0.6083124 0.8449596 0.75779365 0.8114225 0.6060196 0.8438033
 0.75741083 0.8106496 1.073093 1.0707697 1.0702905
50% 0.4 1.0 0.75 1.1139617 1.2141720 0.6083124 0.9736971 0.75779365 0.8277892 0.6087947 0.9736365
 0.75811473 0.8284026 1.089958 1.0923676 1.0927140
50% 0.4 1.0 1.00 1.1139519 1.2209279 0.6083124 1.1062136 0.75779365 0.8338204 0.6091077 1.1066503
 0.75865911 0.8337155 1.096033 1.1003265 1.0989330
50% 0.5 0.1 0.25 0.5099093 0.6074444 0.5000003 0.8849885 0.49999999 0.5016464 0.6996724 0.8851703
 0.49842347 0.4260590 1.191279 1.0032927 0.8548133
50% 0.5 0.1 0.50 0.5099326 0.6693486 0.5000003 0.9325702 0.49999999 0.5031691 0.2994299 0.9325437
 0.49874531 0.6345454 1.312622 1.0063381 1.2722835
50% 0.5 0.1 0.75 0.5099329 0.7034127 0.5000003 0.9681411 0.49999999 0.5049450 0.3067321 0.9679852
 0.49911876 0.6499286 1.379422 1.0098900 1.3021523
50% 0.5 0.1 1.00 0.5096747 0.7143686 0.5000003 1.0000000 0.49999999 0.5067056 0.3120215 1.0001309
 0.49882467 0.6954025 1.401617 1.0134112 1.3940820
50% 0.5 0.5 0.25 0.7073783 0.7812626 0.5000000 0.6146536 0.52526682 0.5572141 0.4996662 0.6140589
 0.52555278 0.5576414 1.104448 1.0608210 1.0610569
50% 0.5 0.5 0.50 0.7071318 0.8289445 0.5000000 0.7389836 0.52526682 0.5853664 0.5001113 0.7391221
 0.52557566 0.5854582 1.172263 1.1144173 1.1139371
50% 0.5 0.5 0.75 0.7070331 0.8566561 0.5000000 0.8685433 0.52526682 0.6061243 0.4990654 0.8686321
 0.52470544 0.6058205 1.211621 1.1539360 1.1545915
50% 0.5 0.5 1.00 0.7066780 0.8657732 0.5000000 1.0000000 0.52526682 0.6140134 0.4990902 0.9992996
 0.52481916 0.6134365 1.225131 1.1689552 1.1688531
50% 0.5 1.0 0.25 1.1190848 1.1663593 0.5000000 0.6151779 0.76222033 0.7916830 0.4992032 0.6145497
 0.76225853 0.7917131 1.042244 1.0386537 1.0386412
50% 0.5 1.0 0.50 1.1181587 1.1986682 0.5000000 0.7382242 0.76222033 0.8158353 0.4989353 0.7374989
 0.76200653 0.8157015 1.072002 1.0703405 1.0704652
50% 0.5 1.0 0.75 1.1187093 1.2184075 0.5000000 0.8674446 0.76222033 0.8318190 0.4994467 0.8680251
 0.76351947 0.8321190 1.089119 1.0913105 1.0898465
50% 0.5 1.0 1.00 1.1182184 1.2254737 0.5000000 1.0000000 0.76222033 0.8374248 0.4992306 0.9996552
 0.76099632 0.8374131 1.095916 1.0986649 1.1004168


###############
##
## Outlier probability for a variety of mixture distributions
##
###############


rslt2 <- NULL
for (a in c(0.1,0.5,0.9)) {
    for (tau in c(0.1,0.25,0.5,1,2)) {
        rslt2 <- rbind(rslt2,outlierProb2(a,tau))
    }
}
options(width=200)
round(rslt2,4)

        a  tau probOutlier meanOutlier sdOutlier
 [1,] 0.1 0.10      0.0143      0.0486    0.4085
 [2,] 0.1 0.25      0.0143      0.0489    0.4095
 [3,] 0.1 0.50      0.0165      0.0566    0.4425
 [4,] 0.1 1.00      0.0201      0.0717    0.5018
 [5,] 0.1 2.00      0.0213      0.0754    0.5170
 [6,] 0.5 0.10      0.0000      0.0000    0.0000
 [7,] 0.5 0.25      0.0000      0.0001    0.0125
 [8,] 0.5 0.50      0.0079      0.0265    0.2975
 [9,] 0.5 1.00      0.0198      0.0691    0.4920
[10,] 0.5 2.00      0.0214      0.0757    0.5177
```

```
[11,] 0.9 0.10      0.1052      1.2910    3.8481
[12,] 0.9 0.25      0.0982      0.5009    1.5608
[13,] 0.9 0.50      0.0414      0.1598    0.7815
[14,] 0.9 1.00      0.0241      0.0872    0.5603
[15,] 0.9 2.00      0.0218      0.0779    0.5258

###############
##
## Outlier probability for a variety of mixture distributions, this time
## for mixtures of 3 normal distributions:
## X ~ (1-a)/2 N(-1, tau^2) + a N(0, tau^2) + (1-a)/2 N(1, tau^2)
##
###############


rslt3 <- NULL
for (a in c(0.45,0.49,0.5,0.51,0.55)) {
    for (tau in c(0.1,0.25,0.5,1,2)) {
        rslt3 <- rbind(rslt3,outlierProb3(a,tau))
    }
}
options(width=200)
round(rslt3,4)

         a  tau probOutlier meanOutlier sdOutlier
 [1,] 0.45 0.10      0.0000      0.0000    0.0000
 [2,] 0.45 0.25      0.0000      0.0000    0.0115
 [3,] 0.45 0.50      0.0072      0.0240    0.2807
 [4,] 0.45 1.00      0.0183      0.0637    0.4705
 [5,] 0.45 2.00      0.0212      0.0742    0.5127
 [6,] 0.49 0.10      0.0000      0.0000    0.0000
 [7,] 0.49 0.25      0.0006      0.0020    0.0791
 [8,] 0.49 0.50      0.0106      0.0356    0.3436
 [9,] 0.49 1.00      0.0192      0.0671    0.4839
[10,] 0.49 2.00      0.0213      0.0752    0.5155
[11,] 0.50 0.10      0.0000      0.0000    0.0000
[12,] 0.50 0.25      0.0018      0.0057    0.1345
[13,] 0.50 0.50      0.0116      0.0380    0.3554
[14,] 0.50 1.00      0.0194      0.0679    0.4866
[15,] 0.50 2.00      0.0213      0.0757    0.5168
[16,] 0.51 0.10      0.2447      1.0488    1.8550
[17,] 0.51 0.25      0.0046      0.0148    0.2168
[18,] 0.51 0.50      0.0126      0.0418    0.3730
[19,] 0.51 1.00      0.0196      0.0685    0.4890
[20,] 0.51 2.00      0.0213      0.0751    0.5160
[21,] 0.55 0.10      0.2250      1.3304    2.4854
[22,] 0.55 0.25      0.0382      0.1273    0.6402
[23,] 0.55 0.50      0.0170      0.0570    0.4382
[24,] 0.55 1.00      0.0203      0.0710    0.4989
[25,] 0.55 2.00      0.0214      0.0751    0.5152
```