library(Biobase)

library(GEOquery)

# load series and platform data from GEO

gset <- getGEO("GSE42589", GSEMatrix =TRUE, getGPL=FALSE)

if (length(gset) > 1) idx <- grep("GPL6244", attr(gset, "names")) else idx <- 1

gset <- gset[[idx]]

# set parameters and draw the plot

dev.new(width=4+dim(gset)[[2]]/5, height=6)

par(mar=c(2+round(max(nchar(sampleNames(gset)))/2),4,2,1))

title <- paste ("GSE42589", '/', annotation(gset), " selected samples", sep ='')

boxplot(exprs(gset), boxwex=0.7, notch=T, main=title, outline=FALSE, las=2)

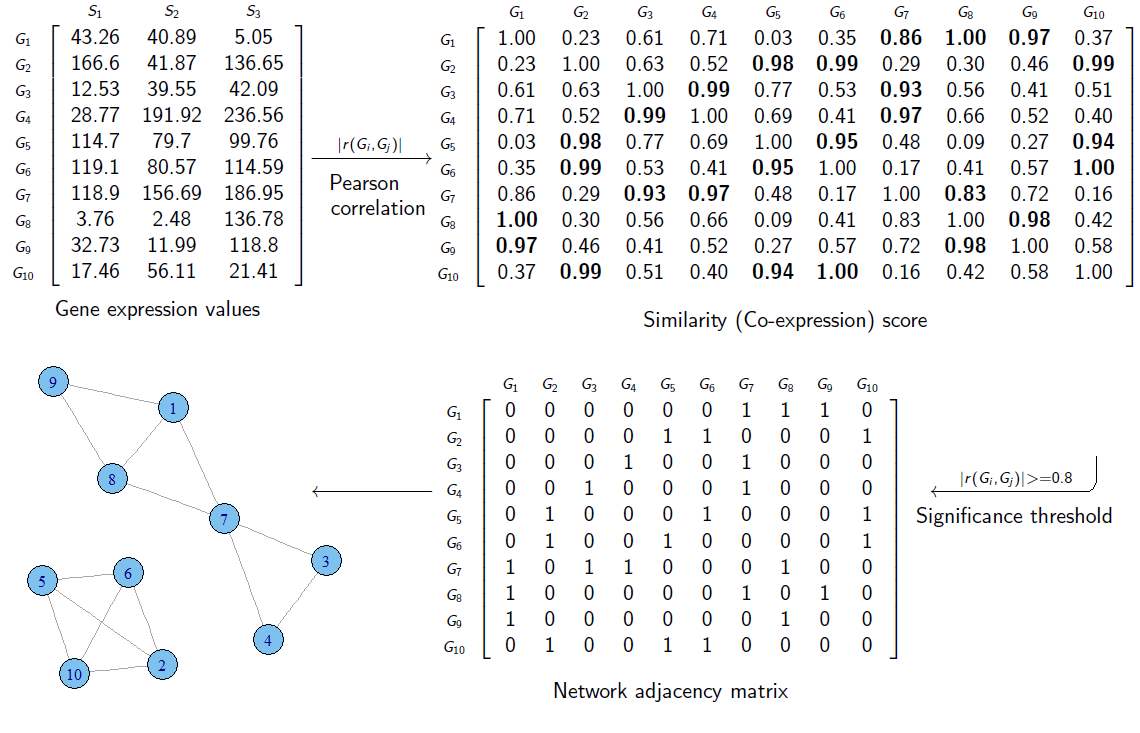
library(preprocessCore)

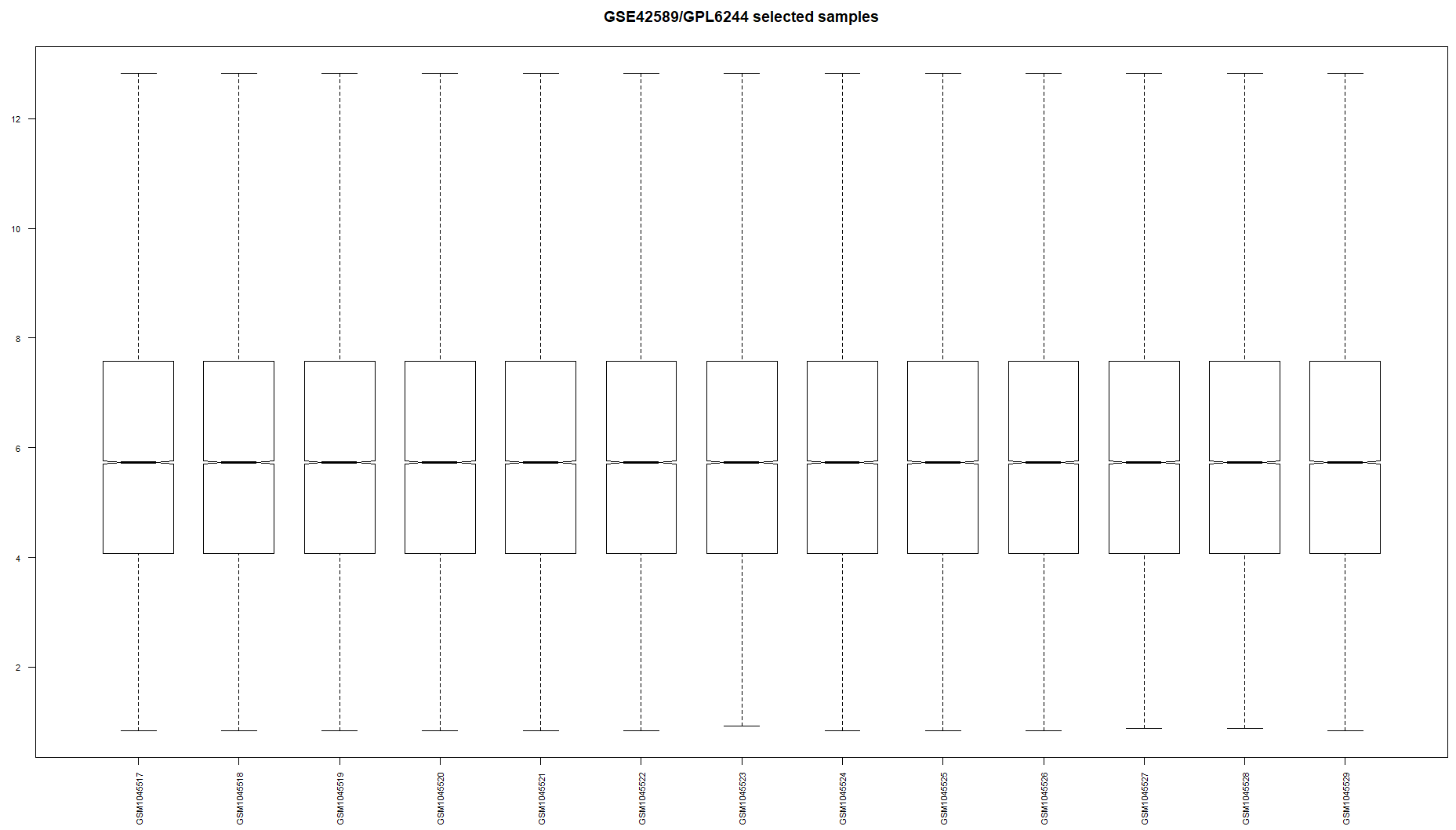
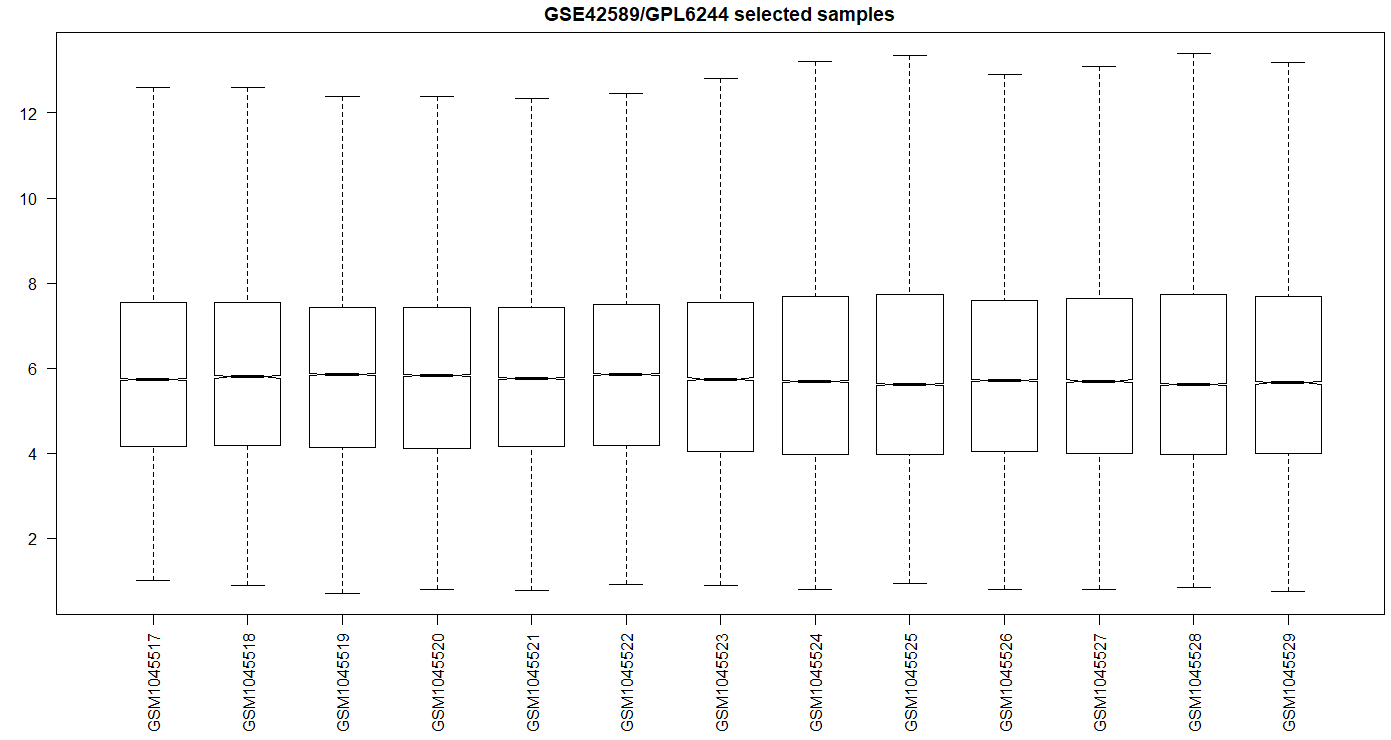
exprs\_gset\_Q <- normalize.quantiles(exprs(gset))

dimnames(exprs\_gset\_Q) <- dimnames(exprs(gset))

par(cex.axis=0.7)

boxplot(exprs\_gset\_Q, boxwex=0.7, notch=T, main=title, outline=FALSE, las=2)





> idx <- match(rownames(exprs\_gset\_Q),anno[,1])

> test <- anno[idx,]

> exprs\_new <- data.frame(test[,3],exprs\_gset\_Q)

> gpl <- getGEO("GPL6244", AnnotGPL = TRUE)

File stored at:

C:\Users\fyan\AppData\Local\Temp\Rtmp4oVQTG/GPL6244.annot.gz

> anno <- Table(gpl)

> names(anno)

[1] "ID" "Gene title" "Gene symbol" "Gene ID" "UniGene title" "UniGene symbol" "UniGene ID"

[8] "Nucleotide Title" "GI" "GenBank Accession" "Platform\_CLONEID" "Platform\_ORF" "Platform\_SPOTID" "Chromosome location"

[15] "Chromosome annotation" "GO:Function" "GO:Process" "GO:Component" "GO:Function ID" "GO:Process ID" "GO:Component ID"

> anno\_new <- anno[,c(1,3)]

> names(anno\_new)

[1] "ID" "Gene symbol"

> names(anno\_new) <- c("ID","Symbol")

> names(anno\_new)

[1] "ID" "Symbol"

> geneProbes <- which(!is.na(anno\_new$Symbol))

> probeids <- as.character(anno\_new$ID[geneProbes])

> probes <- intersect(probeids, rownames(exprs\_gset\_Q))

> length(probes)

[1] 33297

> geneMatrix <- exprs\_gset\_Q[probes, ]

> inds <- which(anno\_new$ID %in% probes)

> head(probes)

[1] "7896736" "7896738" "7896740" "7896742" "7896744" "7896746"

> head(as.character(anno\_new$ID[inds]))

[1] "7896736" "7896738" "7896740" "7896742" "7896744" "7896746"

> geneMatTable <- cbind(geneMatrix, anno\_new[inds,])

> head(geneMatTable)

GSM1045517 GSM1045518 GSM1045519 GSM1045520 GSM1045521 GSM1045522 GSM1045523 GSM1045524 GSM1045525

7896736 3.709795 4.339049 4.670770 5.607805 5.461106 4.247761 4.524943 3.241511 4.794915

7896738 2.732016 4.405623 3.213253 2.587224 2.846853 2.797087 2.572694 2.856354 2.622934

7896740 2.766976 3.624355 3.051015 3.331938 3.185500 2.912254 3.171253 2.928478 2.857676

7896742 4.673250 5.393126 4.392853 5.323297 4.975064 5.727892 4.973749 5.181275 6.182672

7896744 4.335009 6.906489 4.387489 3.987254 4.635438 4.822016 4.496222 7.278641 5.488412

7896746 7.086680 9.757809 9.148946 7.837590 6.788087 9.061966 7.558813 7.744745 6.601148

GSM1045526 GSM1045527 GSM1045528 GSM1045529 ID Symbol

7896736 4.279825 4.628924 4.486740 5.181107 7896736

7896738 2.327342 2.297365 2.226336 2.562228 7896738

7896740 3.034703 2.889066 2.942933 2.841911 7896740 OR4F17///OR4F5///OR4F4

7896742 4.811886 5.767975 6.176741 6.335267 7896742 LOC100134822///SEPT14///LINC00266-1///PCMTD2

7896744 4.113332 5.539998 3.972791 5.564460 7896744 OR4F29///OR4F21///OR4F16///OR4F3

7896746 6.677303 6.708996 6.381895 6.575412 7896746

> dim(geneMatTable)

[1] 33297 15

> geneMatTable\_new <- aggregate(geneMatTable[, -c(14,15)],by=list(Symbol = geneMatTable$Symbol),FUN=mean,na.rm=TRUE)

> dim(geneMatTable\_new)

[1] 20359 14

> head(geneMatTable\_new,1)

Symbol GSM1045517 GSM1045518 GSM1045519 GSM1045520 GSM1045521 GSM1045522 GSM1045523 GSM1045524 GSM1045525 GSM1045526 GSM1045527 GSM1045528 GSM1045529

1 4.95651 4.973512 4.884099 4.850769 4.858558 5.002483 4.865615 4.927613 4.936914 4.864022 4.868596 4.912908 4.887578

> geneMatTable\_new <- geneMatTable\_new [-1,]

> case\_matrix\_o <- geneMatTable\_new[,1:8]

> control\_matrix\_o <- geneMatTable\_new[, c(1,9:14)]

> case\_matrix <- t(case\_matrix\_o [,-1])

> colnames(case\_matrix) <- case\_matrix\_o [,1]

> dim(case\_matrix)

[1] 7 20358

> control\_matrix <- t(control\_matrix\_o [,-1])

> colnames(control\_matrix) <- control\_matrix\_o [,1]

> dim(control\_matrix)

[1] 6 20358

> case\_corr <- cor(case\_matrix)

> dim(case\_corr)

[1] 20358 20358

> control\_corr <- cor(control\_matrix)

> dim(control\_corr)

[1] 20358 20358

> network <- read.delim("C:/Users/fyan/Desktop/18cleft/HPRD\_Release9\_041310/network.txt",header=TRUE,as.is=T)

> corr\_df\_case <- data.frame(Gene1=character(), Gene2=character(), cor=numeric(),stringsAsFactors = FALSE)

> for (i in 1:nrow(network)) {

r = case\_corr[match(network[i,1], rownames(case\_corr)), match(network[i,2], rownames(case\_corr)) ]

corr\_df\_case[nrow(corr\_df\_case)+1,] = list(network[i,1], network[i,2], r) }

> head(corr\_df\_case)

Gene1 Gene2 cor

1 ALDH1A1 ALDH1A1 1.00000000

2 ITGA7 CHRNA1 -0.01227838

3 PPP1R9A ACTG1 0.55299936

4 SRGN CD44 -0.03183222

5 GRB7 ERBB2 0.18755670

6 PAK1 ERBB2 -0.46268400

> corr\_df\_control <- data.frame(Gene1=character(), Gene2=character(), cor=numeric(),stringsAsFactors = FALSE)

> for (i in 1:nrow(network)) {

r = control\_corr[match(network[i,1], rownames(control\_corr)), match(network[i,2], rownames(control\_corr)) ]

corr\_df\_control[nrow(corr\_df\_control)+1,] <- list(network[i,1], network[i,2], r) }

> head(corr\_df\_control)

Gene1 Gene2 cor

1 ALDH1A1 ALDH1A1 1

2 ITGA7 CHRNA1 -0.778712323755542

3 PPP1R9A ACTG1 -0.529183872202047

4 SRGN CD44 0.653574531267934

5 GRB7 ERBB2 0.764267188695985

6 PAK1 ERBB2 -0.239254642255144

> write.table(corr\_df\_case,"corr\_case.txt",sep="\t",row.names=FALSE)

> write.table(corr\_df\_control,"corr\_control.txt",sep="\t",row.names=FALSE)

> df <- data.frame(corr\_df\_case,corr\_df\_control$cor)

> head(df)

Gene1 Gene2 cor corr\_df\_control.cor

1 ALDH1A1 ALDH1A1 1.00000000 1.0000000

2 ITGA7 CHRNA1 -0.01227838 -0.7787123

3 PPP1R9A ACTG1 0.55299936 -0.5291839

4 SRGN CD44 -0.03183222 0.6535745

5 GRB7 ERBB2 0.18755670 0.7642672

6 PAK1 ERBB2 -0.46268400 -0.2392546

> dim(df)

[1] 39240 4

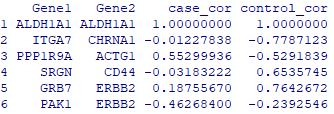
> df <- na.omit(df)

> dim(df)

[1] 33829 4

> colnames(df) <- c("Gene1","Gene2","case\_cor","control\_cor")

> head(df)



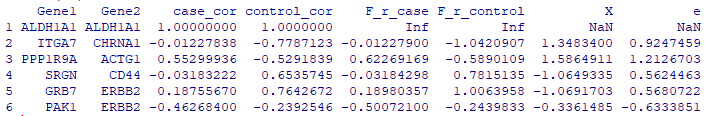
> df$F\_r\_case <- log((1+ df$case\_cor)/(1-df$case\_cor))/2

> df$F\_r\_control <- log((1+ df$control\_cor)/(1-df$control\_cor))/2

> df$X <- (df$F\_r\_case - df$F\_r\_control)/ sqrt(1/4+1/3)

> df$e <- qnorm(1-2\*(1-pnorm(abs(df$X))))

> head(df)

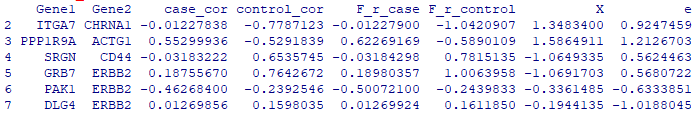


> dim(df)

[1] 33829 8

> df <- na.omit(df)

> head(df)

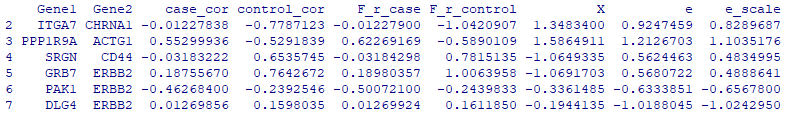


> dim(df)

[1] 31855 8

> scale(df$e) -> df$e\_scale

> head(df)

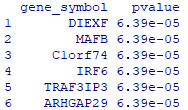


> write.table(df,"edge\_weight.txt",sep="\t",row.names=FALSE)

> write.table(df[,c(1,2,9)],"edge\_weight\_scale.txt",sep="\t",row.names=FALSE)

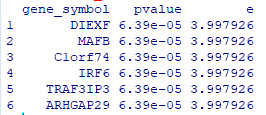
> gwas\_asian <- read.delim("C:/Users/fyan/Desktop/18cleft/dmGWAS\_R/PASCAL\_original\_10\_26/adjusted/GWAS\_Asian\_adjusted.txt",as.is=T)

> head(gwas\_asian)



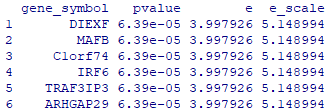
> gwas\_asian$e <- qnorm(1- gwas\_asian$pvalue/2)

> head(gwas\_asian)



> gwas\_asian$e\_scale <- scale(gwas\_asian$e)

> head(gwas\_asian)



> write.table(gwas\_asian,"node\_weight.txt",sep="\t",row.names=FALSE)

> write.table(gwas\_asian[,c(1,4)],"node\_weight\_scale.txt",sep="\t",row.names=FALSE)

## Java EWdmGWAS

c:

cd C:\Users\fyan\EW\_dmGWAS

java EW\_dmGWAS\_v3\_1 model=EW\_dmGWAS node\_weight\_file=Asian\_node\_weight\_scale.txt edge\_weight\_file=edge\_weight\_scale.txt lambda=1 r=0.2 output=Asian.lambda\_1\_permutated.txt permutation=true

java EW\_dmGWAS\_v3\_1 model=EW\_dmGWAS node\_weight\_file=Euro\_node\_weight\_scale.txt edge\_weight\_file=edge\_weight\_scale.txt lambda=0.5 r=0.2 output=Euro.lambda\_0.5.txt permutation=true

l =list()

for (i in 1:40) { length = length(subnetwork$modules[[i]])

l= c (l, length)}

test <- read.delim("test.lambda\_1.txt", as.is=T)

genes = strsplit(test[1,1], split=" ")[[1]]

genes

test = test[order(test[,3], decreasing=T),]

genes = strsplit(test[1,1], split=" ")[[1]]

genes

induced.subgraph(G, genes) -> subG

vcount(subG)

E(subG)

tkplot(subG)

tkplot(subG, vertex.color=V(subG)$weight)

tkplot(subG, vertex.color=0.5\* V(subG)$weight)

tkplot(subG, vertex.color="grey", vertex.size=0.5\*V(subG)$weight)

tkplot.off()

tkplot(subG, vertex.color="grey", vertex.size=V(subG)$weight)

tkplot(subG, vertex.color="grey", vertex.size=3\*V(subG)$weight)

> V(subG)$weight

> E(subG)

+ 3/3 edges from 4037847 (vertex names):

[1] ATF7IP--GMEB1 ATF7IP--ZNF350 ATF7IP--MACF1

> case\_corr["ATF7IP", "GMEB1"]

[1] 0.8917036

> case\_corr["ATF7IP", "ZNF350"]

[1] 0.8086786

> case\_corr["ATF7IP", "MACF1"]

[1] -0.3473027

> get.edgelist(subG)

[,1] [,2]

[1,] "ATF7IP" "GMEB1"

[2,] "ATF7IP" "ZNF350"

[3,] "ATF7IP" "MACF1"

> ee <- get.edgelist(subG)

> control\_corr[ee[,1], ee[,2]]

GMEB1 ZNF350 MACF1

ATF7IP 0.116498 0.7590003 0.8509796

ATF7IP 0.116498 0.7590003 0.8509796

ATF7IP 0.116498 0.7590003 0.8509796

> dim(nodeweight)

NULL

> head(nodeweight)

ITGA7 PPP1R9A SRGN GRB7 PAK1 DLG4

2.037499620 0.007519956 0.830953321 -0.072756358 -1.727934322 0.677639965

> V(subG)$weight

[1] 2.068426 3.143045 2.626768 2.800649

> head(sort(nodeweight, decreasing=T))

MAFB ARHGAP29 TRAF3IP3 ABCA4 PITX2 SYT14

3.830660 3.830660 3.830660 3.751471 3.709097 3.690288

> degree(G) -> deg

> names(deg) = V(G)$name

> head(deg)

ITGA7 PPP1R9A SRGN GRB7 PAK1 DLG4

4 7 7 14 60 91

> deg["VIM"]

VIM

101

> deg["UBE3A"]

UBE3A

17

> head(sort(deg, decreasing=T))

TP53 YWHAG EP300 SRC CREBBP GRB2

256 222 202 194 189 186

test <- read.delim("test.lambda\_1.txt", as.is=T)

Asian\_gene\_list = strsplit(test[1,1], split=" ")[[1]]

for (i in 2:nrow(test)) {

genes = strsplit(test[i,1], split=" ")[[1]]

Asian\_gene\_list = union (Asian\_gene\_list, genes)}

lapply(Asian\_gene\_list, write, " Asian\_lambda\_1\_gene\_list.txt", append=TRUE)

euro <- read.delim("Euro.lambda\_0.5.txt", as.is=T)

Euro\_gene\_list = strsplit(euro[1,1], split=" ")[[1]]

for (i in 2:nrow(euro)) {

genes = strsplit(euro[i,1], split=" ")[[1]]

Euro\_gene\_list = union (Euro\_gene\_list, genes)}

lapply(Euro\_gene\_list, write, "Euro\_lambda\_0.5\_gene\_list.txt", append=TRUE)

**Network**

> edge.weight = read.table("C:/Users/fyan/EW\_dmGWAS/1\_29/edge\_weight\_scale.txt", header=F, as.is=T)

> dim(edge.weight)

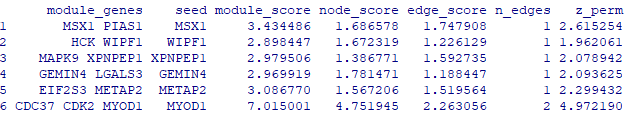
[1] 31855 3

> G = graph.data.frame(edge.weight[,1:2], directed=F)

> E(G)$weight = edge.weight[, 3]

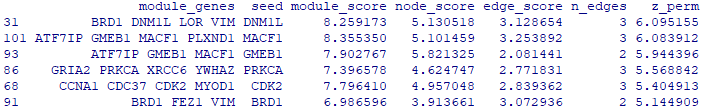
> res.mat = read.delim("C:/Users/fyan/EW\_dmGWAS/1\_29/Asian.lambda\_1\_permutated.txt", as.is=T)

> head(res.mat)



> res.mat = res.mat[order(res.mat[, "z\_perm"], decreasing=T), ]

> head(res.mat)



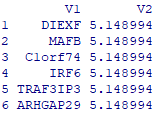
> genes = strsplit(res.mat[1,1], split=" ")[[1]]

> genes

[1] "BRD1" "DNM1L" "LOR" "VIM"

> node.weight = read.table("C:/Users/fyan/EW\_dmGWAS/1\_29/Asian\_node\_weight\_scale.txt", as.is=T)

> head(node.weight)



> z\_node = node.weight[,2]

> names(z\_node) = node.weight[,1]

> V(G)$weight = z\_node[V(G)$name]

> vcount(G)

[1] 8421

> ecount(G)

[1] 31855

> summary(E(G)$weight)

Min. 1st Qu. Median Mean 3rd Qu. Max.

-4.01812 -0.67666 -0.02482 0.00000 0.66361 4.80456

> summary(V(G)$weight)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

-1.3552 -0.7956 -0.2104 -0.0080 0.5672 5.1490 518

> dim(node.weight)

[1] 21301 2

> length(intersect(node.weight[,1], V(G)$name))

[1] 7903

> genes

[1] "BRD1" "DNM1L" "LOR" "VIM"

> subG <- induced.subgraph(G, genes)

> ecount(subG)

[1] 3

> get.edgelist(subG)

[,1] [,2]

[1,] "VIM" "DNM1L"

[2,] "VIM" "LOR"

[3,] "VIM" "BRD1"

> cbind(get.edgelist(subG), E(subG)$weight)

[,1] [,2] [,3]

[1,] "VIM" "DNM1L" "1.664421964"

[2,] "VIM" "LOR" "0.891062835"

[3,] "VIM" "BRD1" "2.863502913"

> cbind(V(subG)$name, V(subG)$weight)

[,1] [,2]

[1,] "VIM" "1.704778164"

[2,] "DNM1L" "2.413573591"

[3,] "LOR" "3.185303923"

[4,] "BRD1" "2.957381302"

> sig.res.mat = res.mat[res.mat$z\_perm > 1.96, ]

> module.genes <- sapply(sig.res.mat[,1], function(u)strsplit(u, split=" ")[[1]] )

> length(module.genes)

[1] 119

> dim(res.mat)

[1] 132 7

> dim(sig.res.mat)

[1] 119 7

> module.genes = unique(unlist(module.genes))

> length(module.genes)

[1] 245

> sort(module.genes)

[1] "ABL1" "ACTR3B" "ADCY5" "AKAP9" "APC" "APLF" "APP" "ARF5" "ARRB2" "ATF7IP" "ATP2B4" "BAK1" "BCL3" "BNIP2"

[15] "BRD1" "BRF2" "BUB1" "CABLES1" "CAPZA1" "CASP2" "CASP3" "CASP6" "CASP9" "CCDC85B" "CCNA1" "CD2" "CD2AP" "CD96"

[29] "CDC37" "CDC73" "CDK1" "CDK2" "CHRNA1" "CLDN8" "CLTC" "CNTN1" "COL1A1" "COL8A1" "CORO1A" "CP" "CPSF1" "CREB3"

[43] "CSNK1A1" "CSNK2A2" "CTBP1" "CTNNB1" "DBF4" "DCP2" "DCTN1" "DDX24" "DGKA" "DNM1L" "DNMT3B" "DPT" "DRAP1" "DRD3"

[57] "DSP" "DTX2" "EED" "EFEMP2" "EHMT2" "EIF2S3" "EIF3E" "ELN" "EP300" "EPHX2" "EVPL" "EYA3" "FANCA" "FANCD2"

[71] "FANCE" "FCGR2B" "FEZ1" "FLNC" "FSCN1" "FYN" "FZD4" "GADD45G" "GCC1" "GEMIN4" "GHR" "GKAP1" "GLRX2" "GMEB1"

[85] "GNAI1" "GNGT1" "GOSR2" "GRB2" "GRIA2" "GRIN2B" "GTF2A2" "GTF2E1" "HCK" "HDAC5" "HGS" "HIST1H4F" "HNRNPK" "HPCAL4"

[99] "HSBP1" "HSF1" "HSPB1" "IKZF1" "IL18" "INPPL1" "IRX5" "ITGA7" "ITGAV" "JAK2" "KIRREL" "KPNB1" "LGALS3" "LIMS1"

[113] "LOR" "LOX" "LRP2BP" "LTBR" "LTF" "MACF1" "MAG" "MAPK9" "MAPRE1" "MCM7" "MED31" "MEGF10" "METAP2" "MITF"

[127] "MMP12" "MSX1" "MVP" "MYCN" "MYOD1" "NAGA" "NAGK" "NDEL1" "NFATC3" "NGFR" "NIPSNAP3A" "NLRP2" "NOS1" "NOTCH1"

[141] "NTRK2" "OPTN" "OSTM1" "OTX2" "PARP4" "PAX6" "PDE4D" "PDE4DIP" "PECAM1" "PELP1" "PHLDB2" "PIAS1" "PIAS3" "PIAS4"

[155] "PICK1" "PIM1" "PLAUR" "PLXNB1" "PLXND1" "PPP1R16A" "PPP4C" "PPP4R1" "PRC1" "PRKACB" "PRKCA" "PRPF40A" "PRRT2" "PSMA4"

[169] "PSMA5" "PSMB1" "PTPN11" "PVR" "PYDC2" "RAB11FIP3" "RAB3IP" "RAB5B" "RAB8A" "RASSF1" "RBBP6" "RBM8A" "RBPJ" "RFC1"

[183] "RFC3" "RGMA" "RGS17" "RNF8" "ROBO2" "RPL35A" "RPS6KB1" "RSRC1" "SAFB" "SAFB2" "SEC16A" "SEC23A" "SEMA4D" "SH3GL3"

[197] "SHMT1" "SIX5" "SLC18A2" "SLC9A1" "SLIT3" "SNAP23" "SNAP25" "SPP1" "SRC" "SREBF2" "SRP68" "SRP72" "STAM" "STX12"

[211] "SURF2" "TADA3" "TAF11" "TAF12" "TAOK2" "TBL1XR1" "TBP" "TESC" "TGIF1" "TGM1" "THBS1" "TJP1" "TJP2" "TMPRSS6"

[225] "TNC" "TNFRSF1A" "TP53" "TRAF5" "TRDMT1" "TRIP10" "TXNRD1" "UBQLN1" "UBQLN2" "USO1" "VAMP8" "VIM" "VPS52" "VPS53"

[239] "WIPF1" "XPNPEP1" "XRCC6" "YTHDC1" "YWHAE" "YWHAZ" "ZBTB38"

> module.genes\_top10 <- sapply(sig.res.mat[1:10,1], function(u)strsplit(u, split=" ")[[1]] )

> module.genes\_top10 = unique(unlist(module.genes\_top10))

> length(module.genes\_top10)

[1] 27

> subG\_top10 <- induced.subgraph(G, module.genes\_top10)

> ecount(subG\_top10)

[1] 27

> vcount(subG\_top10)

[1] 27

> get.edgelist(subG\_top10)

> edge\_top10 <- cbind(get.edgelist(subG\_top10), E(subG\_top10)$weight)

[,1] [,2] [,3]

[1,] "VIM" "DNM1L" "1.664421964"

[2,] "VIM" "LOR" "0.891062835"

[3,] "VIM" "BRD1" "2.863502913"

> node\_top10 <- cbind(V(subG\_top10)$name, V(subG\_top10)$weight)

> write.table(edge\_top10,file="edge\_top10.txt", row.names=FALSE, sep="\t")

> write.table(node\_top10,file="node\_top10.txt", row.names=FALSE, sep="\t")

> subG\_all <- induced.subgraph(G, module.genes)

> ecount(subG\_all)

[1] 354

> vcount(subG\_all)

[1] 245

> get.edgelist(subG\_all)

> edge\_all <- cbind(get.edgelist(subG\_all), E(subG\_all)$weight)

[,1] [,2] [,3]

[1,] "VIM" "DNM1L" "1.664421964"

[2,] "VIM" "LOR" "0.891062835"

[3,] "VIM" "BRD1" "2.863502913"

> node\_all <- cbind(V(subG\_all)$name, V(subG\_all)$weight)

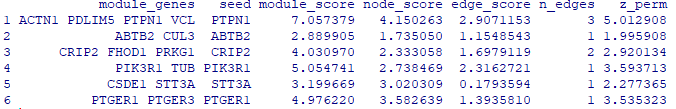
> write.table(edge\_all,file="edge\_all.txt", row.names=FALSE, sep="\t")

> write.table(node\_all,file="node\_all.txt", row.names=FALSE, sep="\t")

**European**

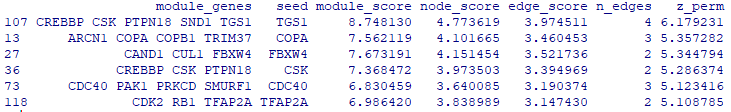
> res.mat\_euro = read.delim("C:/Users/fyan/EW\_dmGWAS/1\_29/Euro.lambda\_1\_permutated.txt", as.is=T)

> head(res.mat\_euro)



> res.mat\_euro = res.mat\_euro[order(res.mat\_euro[, "z\_perm"], decreasing=T), ]

> head(res.mat)



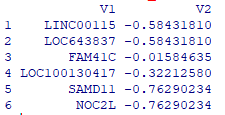
> genes\_euro = strsplit(res.mat\_euro[1,1], split=" ")[[1]]

> genes\_euro

[1] "CREBBP" "CSK" "PTPN18" "SND1" "TGS1"

> node.weight\_euro = read.table("C:/Users/fyan/EW\_dmGWAS/1\_29/Euro\_node\_weight\_scale.txt", as.is=T)

> head(node.weight\_euro)



> z\_node\_euro = node.weight\_euro[,2]

> names(z\_node\_euro) = node.weight\_euro[,1]

> G = graph.data.frame(edge.weight[,1:2], directed=F)

> E(G)$weight = edge.weight[, 3]

> V(G)$weight = z\_node\_euro[V(G)$name]

> vcount(G)

[1] 8421

> ecount(G)

[1] 31855

> summary(E(G)$weight)

Min. 1st Qu. Median Mean 3rd Qu. Max.

-4.01812 -0.67666 -0.02482 0.00000 0.66361 4.80456

> summary(V(G)$weight)

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

-1.3506 -0.8021 -0.2045 -0.0044 0.5825 4.6679 473

> dim(node.weight)

[1] 21301 2

> length(intersect(node.weight[,1], V(G)$name))

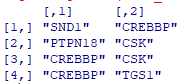
[1] 7903

> subG\_euro\_top1 <- induced.subgraph(G, genes\_euro)

> ecount(subG\_euro\_top1)

[1] 4

> get.edgelist(subG\_euro\_top1)



> edge\_top1 <- cbind(get.edgelist(subG\_euro\_top1), E(subG\_euro\_top1)$weight)

> node\_top1 <- cbind(V(subG\_euro\_top1)$name, V(subG\_euro\_top1)$weight)

> sig.res.mat\_euro = res.mat\_euro[res.mat\_euro$z\_perm > 1.96, ]

> module.genes\_euro <- sapply(sig.res.mat\_euro[,1], function(u)strsplit(u, split=" ")[[1]] )

> module.genes\_euro\_all = unique(unlist(module.genes\_euro))

> length(module.genes\_euro\_all)

[1] 253

> module.genes\_euro\_top10 <- sapply(sig.res.mat\_euro[1:10,1], function(u)strsplit(u, split=" ")[[1]] )

> module.genes\_euro\_top10 = unique(unlist(module.genes\_euro\_top10))

> length(module.genes\_euro\_top10)

[1] 31

> subG\_top10 <- induced.subgraph(G, module.genes\_euro\_top10)

> ecount(subG\_top10)

[1] 29

> vcount(subG\_top10)

[1] 31

> edge\_top10 <- cbind(get.edgelist(subG\_top10), E(subG\_top10)$weight)

[,1] [,2] [,3]

[1,] "VIM" "DNM1L" "1.664421964"

[2,] "VIM" "LOR" "0.891062835"

[3,] "VIM" "BRD1" "2.863502913"

> node\_top10 <- cbind(V(subG\_top10)$name, V(subG\_top10)$weight)

> write.table(edge\_top10,file="edge\_top10.txt", row.names=FALSE, sep="\t")

> write.table(node\_top10,file="node\_top10.txt", row.names=FALSE, sep="\t")

> subG\_all <- induced.subgraph(G, module.genes\_euro\_all)

> ecount(subG\_all)

[1] 302

> vcount(subG\_all)

[1] 253

> edge\_all <- cbind(get.edgelist(subG\_all), E(subG\_all)$weight)

[,1] [,2] [,3]

[1,] "VIM" "DNM1L" "1.664421964"

[2,] "VIM" "LOR" "0.891062835"

[3,] "VIM" "BRD1" "2.863502913"

> node\_all <- cbind(V(subG\_all)$name, V(subG\_all)$weight)

> write.table(edge\_all,file="edge\_all.txt", row.names=FALSE, sep="\t")

> write.table(node\_all,file="node\_all.txt", row.names=FALSE, sep="\t")

setwd("C:/Users/fyan/Documents")

library(dmGWAS)

selected\_top1 = simpleChoose(res.list, top=1, plot=T)

selected\_top10 = simpleChoose(res.list, top=10, plot=T)

selected\_top50 = simpleChoose(res.list, top=50, plot=T)

edge\_top1 <- get.edgelist(selected\_top1$subnetwork)

node\_top1 <- cbind (V(selected\_top1$subnetwork)$name, V(selected\_top1$subnetwork)$weight)

write.table(edge\_top1,file="edge\_top1.txt", row.names=FALSE, sep="\t")

write.table(node\_top1,file="node\_top1.txt", row.names=FALSE, sep="\t")

edge\_top10 <- get.edgelist(selected\_top10$subnetwork)

node\_top10 <- cbind (V(selected\_top10$subnetwork)$name, V(selected\_top10$subnetwork)$weight)

write.table(edge\_top10,file="edge\_top10.txt", row.names=FALSE, sep="\t")

write.table(node\_top10,file="node\_top10.txt", row.names=FALSE, sep="\t")

edge\_top50 <- get.edgelist(selected\_top50$subnetwork)

node\_top50 <- cbind (V(selected\_top50$subnetwork)$name, V(selected\_top50$subnetwork)$weight)

write.table(edge\_top50,file="edge\_top50.txt", row.names=FALSE, sep="\t")

write.table(node\_top50,file="node\_top50.txt", row.names=FALSE, sep="\t")

edge <- get.edgelist(a)

node <- cbind (V(a)$name, V(a)$weight)

write.table(edge,file="edge.txt", row.names=FALSE, sep="\t")

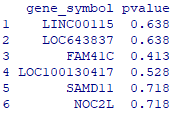
write.table(node,file="node.txt", row.names=FALSE, sep="\t")

> euro.gene.pascal = read.delim("C:/Users/fyan/Desktop/18cleft/dmGWAS\_R/PASCAL\_original\_10\_26/adjusted/GWAS\_Euro.txt", as.is=T)

> dim(euro.gene.pascal)

[1] 21457 2

> head(euro.gene.pascal)



> euro.gene.p = euro.gene.pascal[,2]

> names(euro.gene.p) = euro.gene.pascal[,1]

> summary(euro.gene.p[module.genes])

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.000566 0.011750 0.029400 0.065914 0.084250 0.478000

> table(euro.gene.p[module.genes] < 0.05)

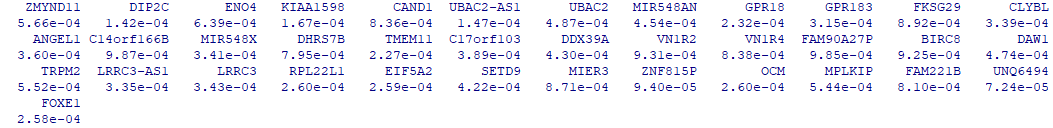
FALSE TRUE

39 72

> sum(euro.gene.p < 0.001)

[1] 37

> euro.gene.p[euro.gene.p < 0.001]



> G = graph.data.frame(network, directed=F)

> intersect(names(which(euro.gene.p < 0.001)), V(G)$name)

[1] "ZMYND11" "KIAA1598" "CAND1" "UBAC2" "GPR183" "DHRS7B" "TMEM11" "BIRC8" "TRPM2" "EIF5A2" "FOXE1"

> intersect(names(which(euro.gene.p < 0.001)), V(G)$name) -> x

> length(x)

[1] 11

> dim(res.mat)

[1] 717 5

> grep("ZMYND11", res.mat[,1])

[1] 15 65 318 319 341

> grep("UBAC2", res.mat[,1])

integer(0)

> sapply(x, function(u)grep(x, res.mat[,1]))

