Cross-platform integration and Differential Expression analysis of T2DM and HTN transcriptomic profiles

Loranda_Calderon

2025-04-01

```
library(GEOquery)
```

```
## Loading required package: Biobase
```

```
## Loading required package: BiocGenerics
```

```
##
## Attaching package: 'BiocGenerics'
```

```
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
```

```
## The following objects are masked from 'package:base':
##
##     anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##     colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##     get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
##     match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##     Position, rank, rbind, Reduce, rownames, sapply, setdiff, sort,
##     table, tapply, union, unique, unsplit, which.max, which.min
```

```
## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
## Setting options('download.file.method.GEOquery'='auto')
```

```
## Setting options('GEOquery.inmemory.gpl'=FALSE)
```

```
library(affy)
library(limma)
```

```
##
## Attaching package: 'limma'
```

```
## The following object is masked from 'package:BiocGenerics':
##
##     plotMA
```

```
library(sva)
```

```
## Loading required package: mgcv
```

```
## Loading required package: nlme
```

```
## This is mgcv 1.9-1. For overview type 'help("mgcv-package")'.
```

```
## Loading required package: genefilter
```

```
## Loading required package: BiocParallel
```

```
library(WGCNA)
```

```
## Loading required package: dynamicTreeCut
```

```
## Loading required package: fastcluster
```

```
##
## Attaching package: 'fastcluster'
```

```
## The following object is masked from 'package:stats':
##
##     hclust
```

```
##
```

```
##
## Attaching package: 'WGCNA'
```

```
## The following object is masked from 'package:stats':
##
##     cor
```

```
library(ensembldb)
```

```
## Loading required package: GenomicRanges
```

```
## Loading required package: stats4
```

```
## Loading required package: S4Vectors
```

```
##
## Attaching package: 'S4Vectors'
```

```
## The following object is masked from 'package:utils':
##
##     findMatches

## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname

## Loading required package: IRanges

##
## Attaching package: 'IRanges'

## The following object is masked from 'package:nlme':
##
##     collapse

## Loading required package: GenomeInfoDb

## Loading required package: GenomicFeatures

## Loading required package: AnnotationDbi

## Loading required package: AnnotationFilter

##
## Attaching package: 'ensembldb'

## The following object is masked from 'package:stats':
##
##     filter
```

```
library(biomaRt)
```

```
# Download GSE25724 data (T2DM)
gse_T2DM <- getGEO("GSE25724", GSEMatrix =TRUE,getGPL=FALSE)
```

```
## Found 1 file(s)

## GSE25724_series_matrix.txt.gz
```

```
datMeta_T2DM <- pData(gse_T2DM[[1]])
rownames(datMeta_T2DM) <- datMeta_T2DM$geo_accession
```

```
# Read GSE25724 data
setwd("/Users/lorandacalderonzamora/GSE25724/")
data.affy_T2DM <- ReadAffy(celfile.path = "./")
datExpr_T2DM <- exprs(data.affy_T2DM)
```

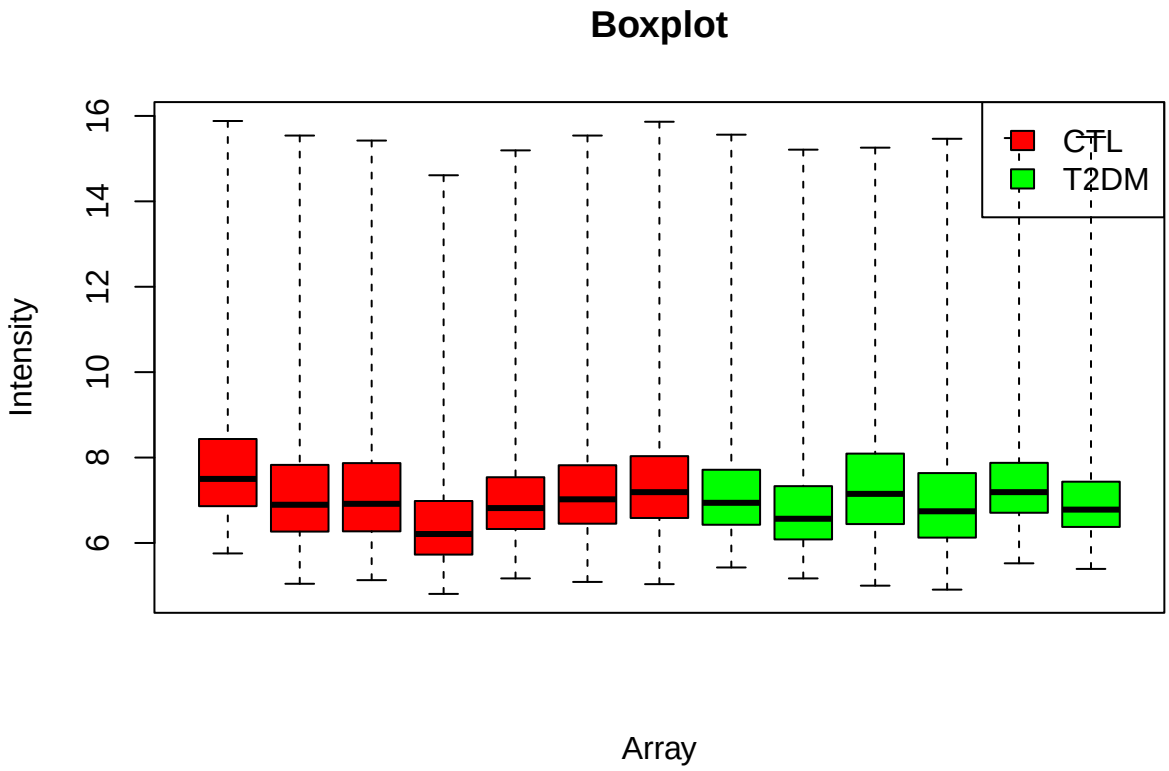
```
# Align datMeta_T2DM and datExpr_T2DM by sample identifiers
GSM_T2DM <- rownames(pData(data.affy_T2DM))
GSM_T2DM <- substr(GSM_T2DM,1,9)
idx <- match(GSM_T2DM, datMeta_T2DM$geo_accession)
datMeta_T2DM <- datMeta_T2DM[idx,]
colnames(datExpr_T2DM)=rownames(datMeta_T2DM)
```

```
# Cleaning and formatting of GSE25724 metadata
datMeta_T2DM <- datMeta_T2DM[,-c(3:7,14:36)]
colnames(datMeta_T2DM)[2] <- c("Dx")
datMeta_T2DM$Dx[rownames(datMeta_T2DM) %in% c("GSM631755", "GSM631756", "GSM631757", "GSM631758", "GSM63
datMeta_T2DM$Dx[rownames(datMeta_T2DM) %in% c("GSM631762", "GSM631763", "GSM631764", "GSM631765", "GSM63
datMeta_T2DM$Dx <- as.factor(datMeta_T2DM$Dx)
```

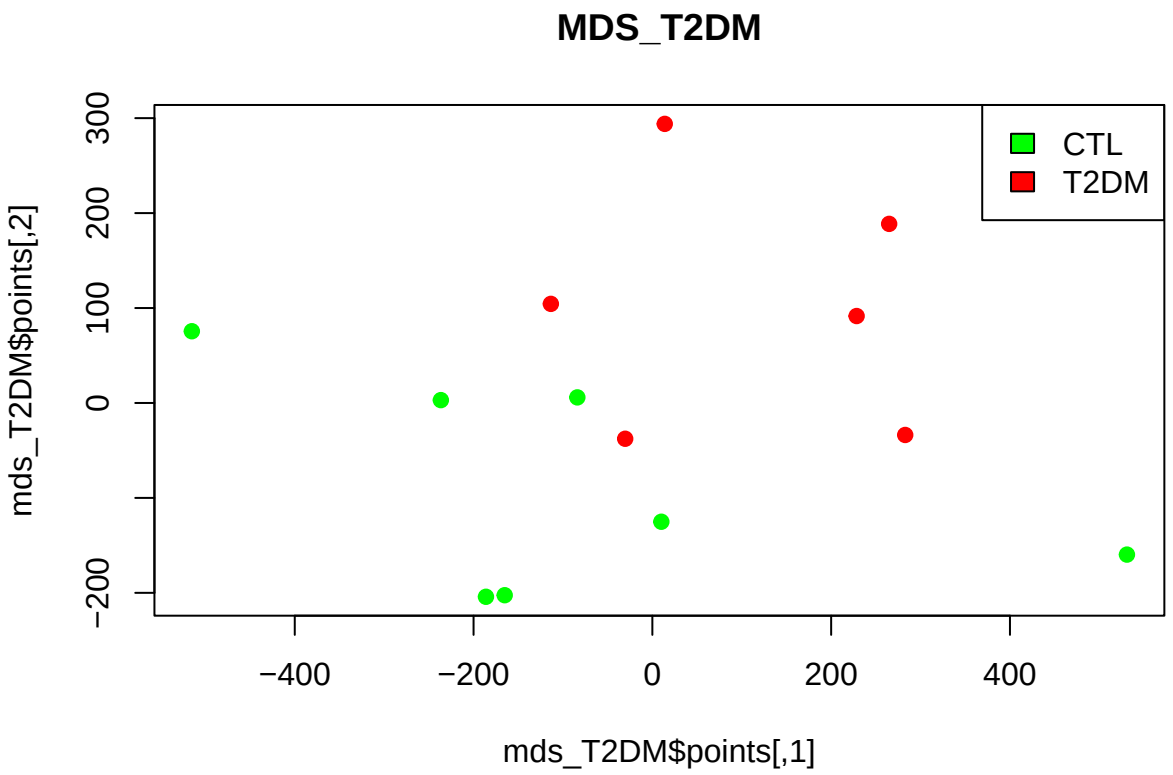
```
# Preprocessing and quality assessment of GSE25724 raw expression data
datExpr_T2DM <- log2(datExpr_T2DM)
dim(datExpr_T2DM)
```

```
## [1] 506944     13
```

```
# Exploratory visualization of GSE25724 raw data
boxplot(datExpr_T2DM,range=0, col=c('red', 'green')[as.numeric(datMeta_T2DM$Dx)], xaxt='n', xlab = "Arra
legend("topright",legend = levels(datMeta_T2DM$Dx),fill = c('red', 'green')[as.numeric(as.factor(levels
```



```
mds_T2DM = cmdscale(dist(t(datExpr_T2DM)),eig=TRUE)
plot(mds_T2DM$points,col=c('green', 'red')[as.numeric(datMeta_T2DM$Dx)],pch=19,main="MDS_T2DM")
legend("topright",legend = levels(datMeta_T2DM$Dx),fill =c('green', 'red')[as.numeric(as.factor(levels(
```




```
# Normalization using RMA
datExpr_T2DM <- rma(data.affy_T2DM, background=T, normalize=T, verbose=T)
```

```
## Warning: replacing previous import 'AnnotationDbi::tail' by 'utils::tail' when
## loading 'hgu133acdf'
```

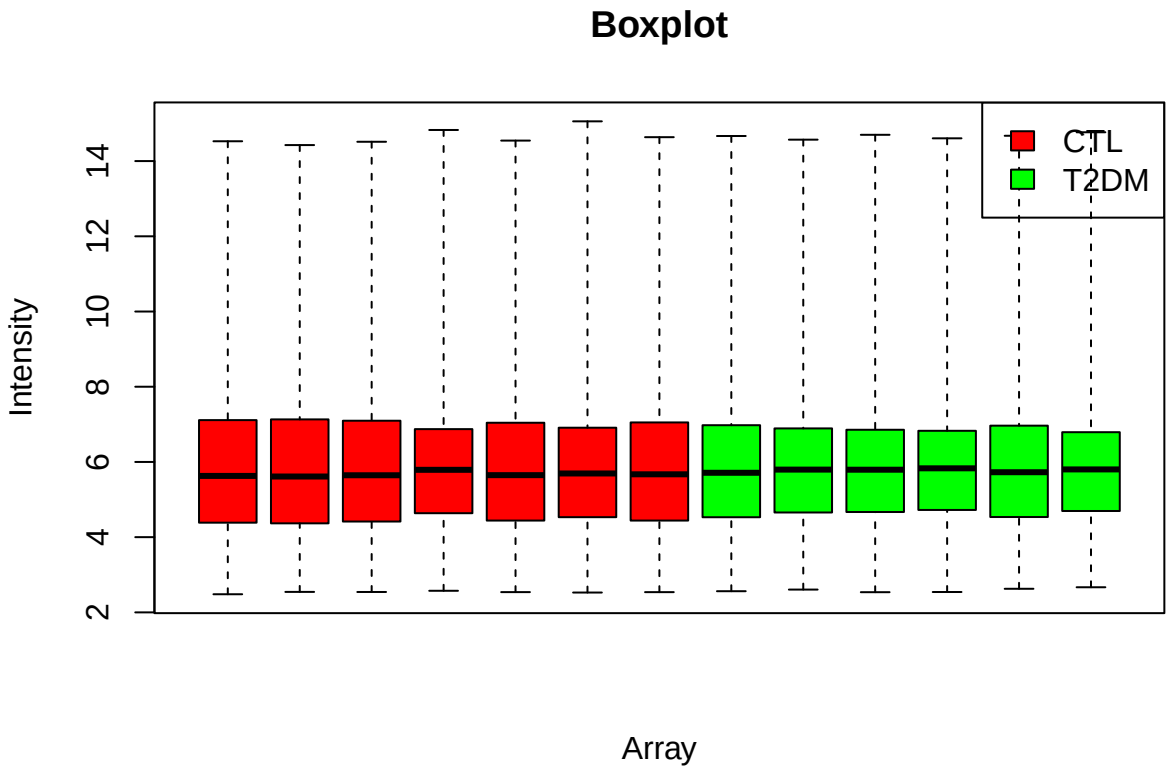
```
## Warning: replacing previous import 'AnnotationDbi::head' by 'utils::head' when
## loading 'hgu133acdf'
```

```
##
```

```
## Background correcting
## Normalizing
## Calculating Expression
```

```
datExpr_T2DM <- exprs(datExpr_T2DM)
```

```r
# Exploratory visualization of GSE25724 normalized data
boxplot(datExpr_T2DM,range=0, col=c('red', 'green')[as.numeric(datMeta_T2DM$Dx)], xaxt='n', xlab = "Arra
legend("topright",legend = levels(datMeta_T2DM$Dx),fill = c('red', 'green')[as.numeric(as.factor(levels
```

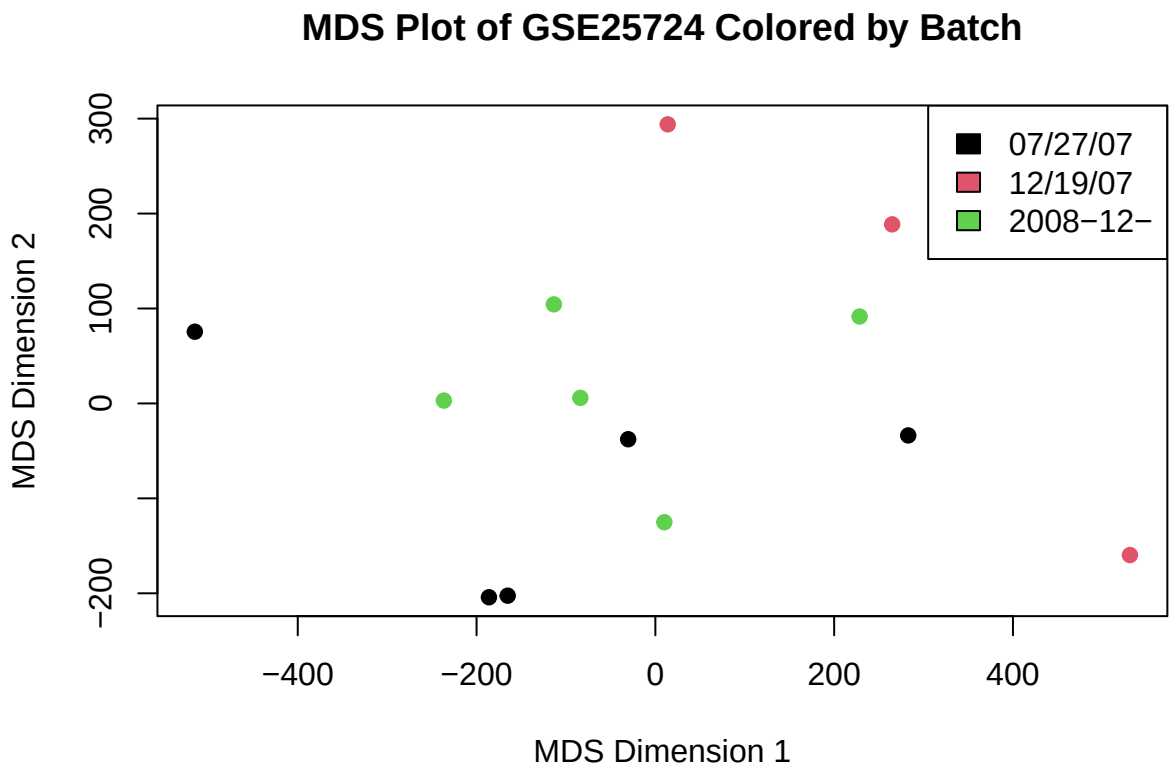



```r
# Extract ScanDate from GSE25724 for batch effect correction
batch_T2DM <- protocolData(data.affy_T2DM)$ScanDate
batch_T2DM <- substr(batch_T2DM,1,8)
batch_T2DM <- as.factor(batch_T2DM)
table(batch_T2DM)
```

```
## batch_T2DM
## 07/27/07 12/19/07 2008-12-
##        5        3        5
```

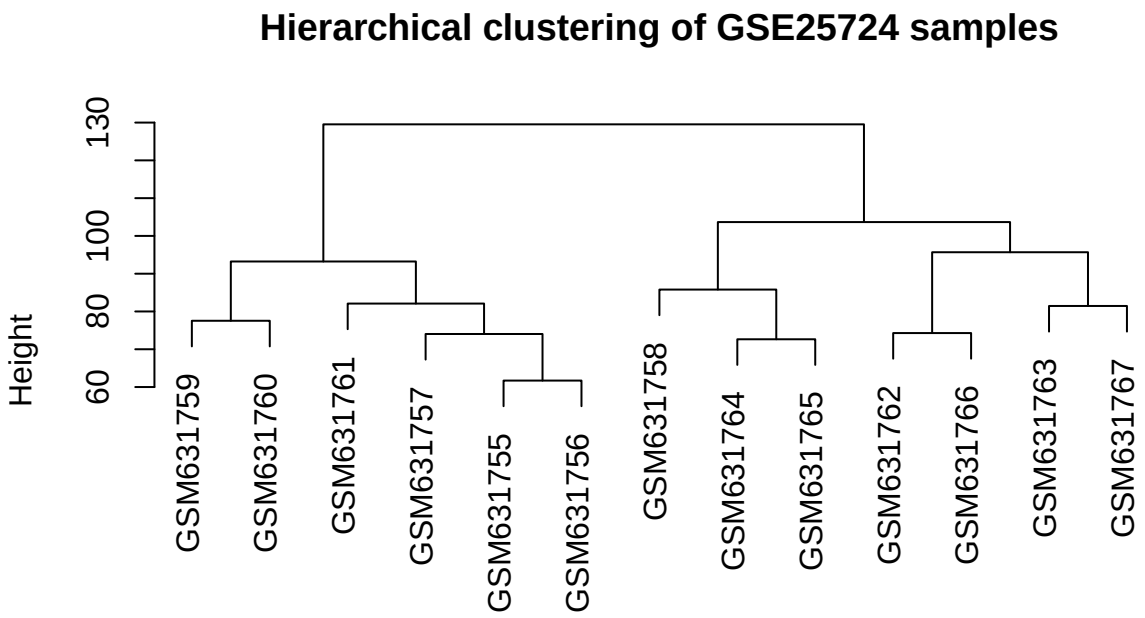
```r
datMeta_T2DM$Batch <- batch_T2DM
```

```r
# Visualization of ScanDate metadata from GSE25724 to identify potential batch effects
plot(mds_T2DM$points,col = as.numeric(datMeta_T2DM$Batch),pch=19,main="MDS Plot of GSE25724 Colored by
legend("topright",legend = levels(datMeta_T2DM$Batch),fill = as.numeric(as.factor(levels(datMeta_T2DM$Ba
```

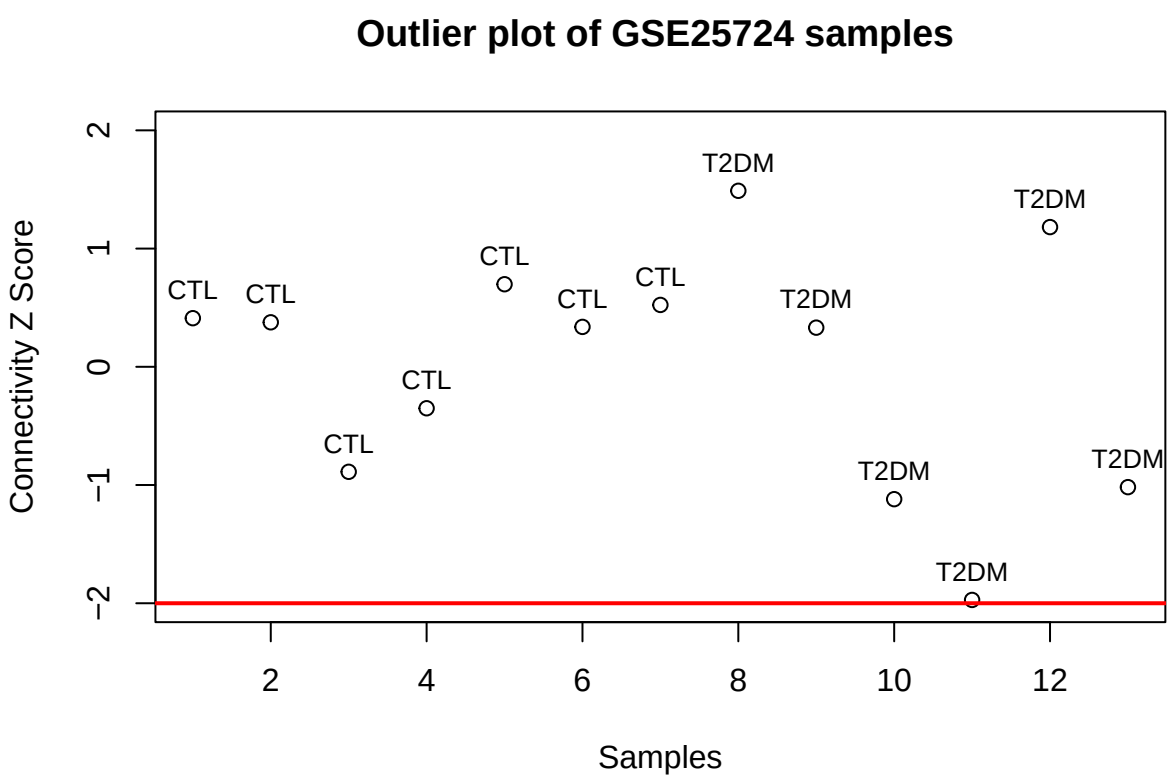



```
# Create ExpressionSet object after Batch effect assessment
datMeta_T2DM$Batch <- batch_T2DM
datMeta_proc_T2DM <- new("AnnotatedDataFrame", data = datMeta_T2DM)
colnames(datExpr_T2DM) <- rownames(datMeta_T2DM)
datAll_T2DM <- new("ExpressionSet", exprs = datExpr_T2DM, phenoData = datMeta_proc_T2DM)
# No singular batch was detected in the GSE25724 dataset.
# Therefore, batch correction with ComBat is technically feasible.
# However, as no evident batch effect was observed in exploratory analyses (MDS),
# ComBat was not applied, and no batch removal was necessary.
```

```
# Sample Clustering and outlier detection
tree_T2DM <- hclust(dist(t(exprs(datAll_T2DM))), method = "average")
plot(tree_T2DM, main = "Hierarchical clustering of GSE25724 samples", xlab = "", sub = "")
```




```
normadj_T2DM <- (0.5 + 0.5*bicor(exprs(datAll_T2DM)))^2
netsummary_T2DM <- fundamentalNetworkConcepts(normadj_T2DM)
C_T2DM <- netsummary_T2DM$Connectivity
Z.C_T2DM <- (C_T2DM - mean(C_T2DM)) / sqrt(var(C_T2DM))

datLabel_T2DM <- pData(datAll_T2DM)$Dx
plot(1:length(Z.C_T2DM),Z.C_T2DM,main="Outlier plot of GSE25724 samples ",xlab = "Samples",ylab="Connec
text(1:length(Z.C_T2DM),Z.C_T2DM,label=datLabel_T2DM,pos=3,cex=0.8)
abline(h= -2, col="red", lwd = 2)
```




```
# Identify and remove potential outlier from GSE25724 samples based on connectivity Z-score
# No samples exceeded the threshold (Z < -2), so none were removed
to_keep_T2DM <- abs(Z.C_T2DM) < 2
table(to_keep_T2DM)
```

```
## to_keep_T2DM
## TRUE
##   13
```

```
colnames(exprs(datAll_T2DM))[!to_keep_T2DM]
```

```
## character(0)
```

```
datAll_T2DM <- datAll_T2DM[, to_keep_T2DM]
```

```
# Annotating Probes using Ensembl
ensembl <- useEnsembl(biomart = "ensembl", dataset = "hsapiens_gene_ensembl")
```

```
# Annotating Probes for GSE25724 dataset
identifier <- "affy_hg_u133a_2"
getinfo <- c("affy_hg_u133a_2", "ensembl_gene_id", "entrezgene_id", "external_gene_name")
geneDat_T2DM <- getBM(attributes = getinfo,
                      filters = identifier,
```

```
                      values = rownames(exprs(datAll_T2DM)),
                      mart = ensembl)
idx_T2DM <- match(rownames(exprs(datAll_T2DM)), geneDat_T2DM$affy_hg_u133a_2)
geneDat_T2DM <- geneDat_T2DM[idx_T2DM, ]
table(is.na(geneDat_T2DM$ensembl_gene_id))
```

```
##
## FALSE  TRUE
## 20259  2024
```

```
to_keep_T2DM <- !is.na(geneDat_T2DM$ensembl_gene_id)
geneDat_T2DM <- geneDat_T2DM[to_keep_T2DM, ]
datAll_T2DM <- datAll_T2DM[to_keep_T2DM, ]
```

```
# Collapse Rows for GSE25724 by Ensembl Gene ID
table(duplicated(geneDat_T2DM$affy_hg_u133a_2))
```

```
##
## FALSE
## 20259
```

```
table(duplicated(geneDat_T2DM$ensembl_gene_id))
```

```
##
## FALSE  TRUE
## 13366  6893
```

```
CR_T2DM <- collapseRows(exprs(datAll_T2DM),
                        rowGroup = geneDat_T2DM$ensembl_gene_id,
                        rowID = geneDat_T2DM$affy_hg_u133a_2)
CRdata_T2DM <- CR_T2DM$datETcollapsed
idx_T2DM <- match(CR_T2DM$group2row[,"selectedRowID"], geneDat_T2DM$affy_hg_u133a_2)
geneDat_T2DM <- geneDat_T2DM[idx_T2DM, ]
rownames(geneDat_T2DM) <- geneDat_T2DM$ensembl_gene_id
```

```
# Differential Expression Analysis from GSE25724
mod_T2DM <- model.matrix(~pData(datAll_T2DM)$Dx)
fit_T2DM <- lmFit(CR_T2DM$datETcollapsed,mod_T2DM)
fit_T2DM <- eBayes(fit_T2DM)
tt_T2DM <- topTable(fit_T2DM,coef = 2,n = Inf,genelist = geneDat_T2DM)
head(tt_T2DM)
```

```
##                 affy_hg_u133a_2 ensembl_gene_id entrezgene_id
## ENSG00000147642       218692_at ENSG00000147642         55638
## ENSG00000171109     207098_s_at ENSG00000171109         55669
## ENSG00000156413     211465_x_at ENSG00000156413          2528
## ENSG00000143575       201145_at ENSG00000143575         10456
## ENSG00000187735     216241_s_at ENSG00000187735          6917
## ENSG00000086619       220012_at ENSG00000086619         56605
##                 external_gene_name      logFC  AveExpr         t      P.Value
```

```
## ENSG00000147642               SYBU -1.7852973 7.456118 -8.898821 3.975337e-07
## ENSG00000171109               MFN1 -2.0197137 5.854147 -8.708803 5.146010e-07
## ENSG00000156413               FUT6  0.9909830 8.111291  7.684775 2.221989e-06
## ENSG00000143575               HAX1 -0.9649762 8.280549 -7.464098 3.096722e-06
## ENSG00000187735              TCEA1 -1.9923078 8.559339 -7.444819 3.188778e-06
## ENSG00000086619              ERO1B -2.4723116 7.598684 -7.394610 3.442341e-06
##                    adj.P.Val        B
## ENSG00000147642 0.003439078 6.555426
## ENSG00000171109 0.003439078 6.335236
## ENSG00000156413 0.007096584 5.061263
## ENSG00000143575 0.007096584 4.766488
## ENSG00000187735 0.007096584 4.740382
## ENSG00000086619 0.007096584 4.672123
```

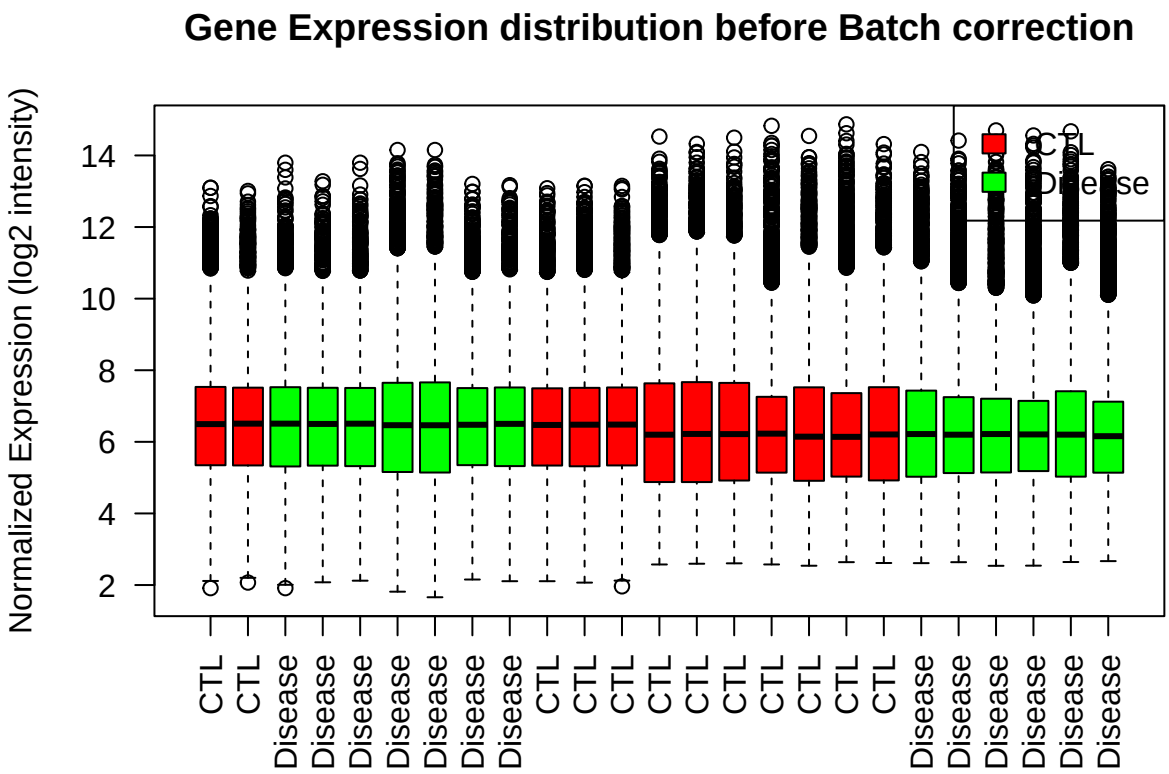
```
# Load a Normalized_expression_matrix_from_HTN csv file
data_NEM_HTN <- read.csv("/Users/lorandacalderonzamora/Downloads/combat_expr", row.names = 1)
```

```
# Preprocessing sample identifiers and group assignment
colnames(data_NEM_HTN) <- gsub("\\..*", "", colnames(data_NEM_HTN))
group <- factor(colnames(data_NEM_HTN))
```

```
# Merging T2DM and HTN Expression matrices by common genes
common_genes <- intersect(rownames(data_NEM_HTN), rownames(CRdata_T2DM))
data_NEM_HTN_common <- data_NEM_HTN[common_genes, ]
CRdata_T2DM_common <- CRdata_T2DM[common_genes, ]

unificated_expr_matrix <- cbind(data_NEM_HTN_common, CRdata_T2DM_common)
unificated_expr_matrix <- as.data.frame(unificated_expr_matrix)
```

```
# Relabeling sample identifiers with group labels for Differential Expression analysis
sample_ids <- colnames(unificated_expr_matrix)

group_labels <- sample_ids
group_labels[group_labels %in% c("GSM631755", "GSM631756", "GSM631757", "GSM631758", "GSM631759", "GSM63
                                 "GSM609530")] <- "CTL"

group_labels[group_labels %in% c("GSM631762", "GSM631763", "GSM631764", "GSM631765", "GSM631766", "GSM63
                                 "GSM609526", "GSM609527")] <- "Disease"

colnames(unificated_expr_matrix) <- group_labels
```

```
# Boxplot of the merged Expression matrix from T2DM and HTN prior to Batch correction
boxplot(unificated_expr_matrix, main = "Gene Expression distribution before Batch correction", col = c(
legend("topright", legend = levels(factor(group_labels)), fill = c("red", "green"))
```

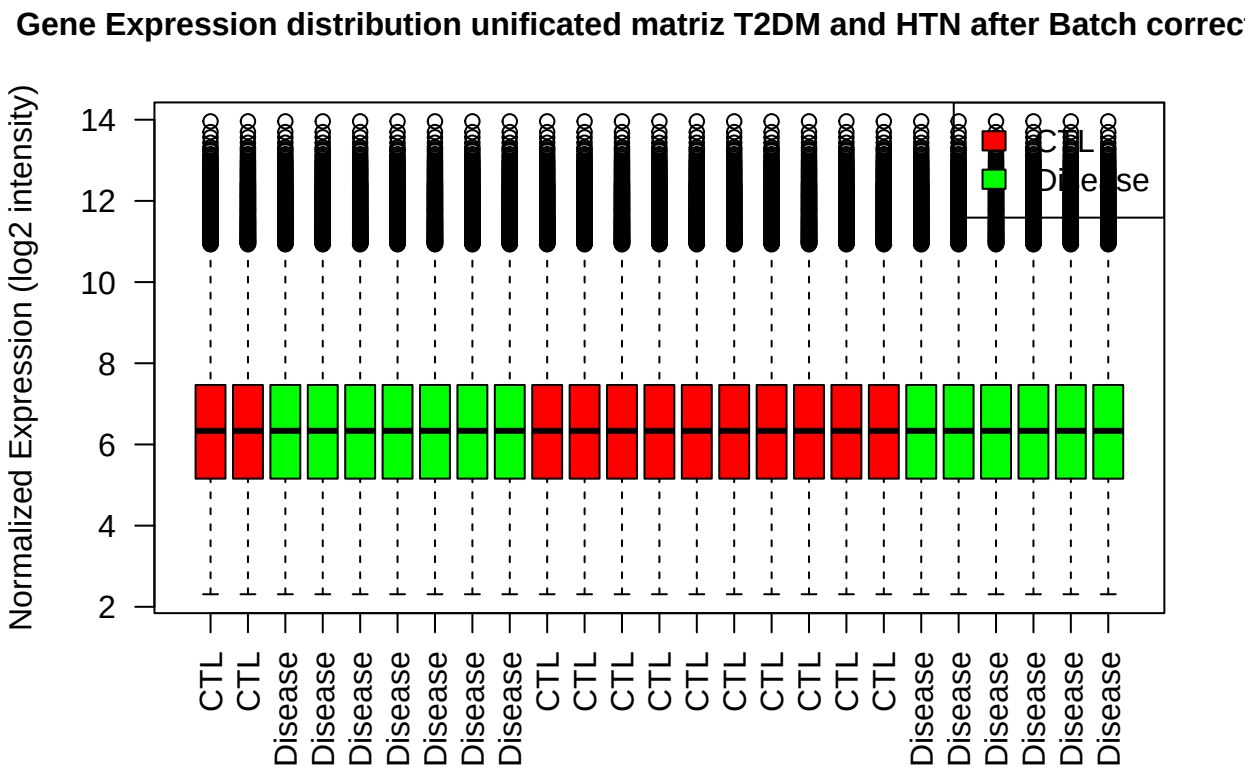



```r
# Quantile normalization of merged Expression matrix
library(preprocessCore)
expr_matrix_qn <- normalize.quantiles(as.matrix(unificated_expr_matrix))
rownames(expr_matrix_qn) <- rownames(unificated_expr_matrix)
colnames(expr_matrix_qn) <- colnames(unificated_expr_matrix)
```

```r
# Batch effect correction between T2DM and HTN Datasets using ComBat
n_HTN <- ncol(data_NEM_HTN_common)
n_T2DM <- ncol(CRdata_T2DM_common)
batch <- c(rep("HTN", n_HTN), rep("T2DM", n_T2DM))

combat_expr <- ComBat(dat = expr_matrix_qn, batch = batch, par.prior = TRUE, prior.plots = FALSE)
```

```
## Found2batches

## Adjusting for0covariate(s) or covariate level(s)

## Standardizing Data across genes

## Fitting L/S model and finding priors

## Finding parametric adjustments

## Adjusting the Data
```

```r
# Boxplot of the merged Expression matrix from T2DM and HTN after to Batch correction
boxplot(expr_matrix_qn, main = "Gene Expression distribution unificated matriz T2DM and HTN after Batch
legend("topright", legend = levels(factor(group_labels)), fill = c("red", "green"))
```




```r
# Define group factor for Differential Expression analysis
sample_ids <- colnames(combat_expr)
group <- as.factor(colnames(combat_expr))
```

‘

```r
# Differential Expression Analysis between Control and Disease samples from integrated T2DM and HTN Dat
design <- model.matrix(~ group)
fit_merge <- lmFit(combat_expr, design)
fit_merge <- eBayes(fit_merge)
tt_merge <- topTable(fit_merge, coef = 2, number = Inf)
head(tt_merge)
```

```
##                      logFC  AveExpr         t      P.Value  adj.P.Val        B
## ENSG00000133226 -0.9124525 7.509201 -5.620966 6.025814e-06 0.04549518 3.822115
## ENSG00000118495 -1.0503591 5.976003 -5.123157 2.263210e-05 0.04549518 2.658900
## ENSG00000167633  0.4355697 4.519460  5.080068 2.539053e-05 0.04549518 2.557451
## ENSG00000134884 -0.8300671 7.907104 -5.022713 2.959263e-05 0.04549518 2.422281
## ENSG00000178078  0.7516481 7.001916  4.990679 3.223628e-05 0.04549518 2.346723
## ENSG00000126804 -1.0117875 6.690346 -4.987182 3.253889e-05 0.04549518 2.338471
```

```r
# Gene annotation of Differential Expression using Ensembl gene IDs
gene_ids <- rownames(tt_merge)
annot_attributes <- c("ensembl_gene_id", "external_gene_name", "entrezgene_id")
geneDat <- getBM(attributes = annot_attributes,
                 filters = "ensembl_gene_id",
                 values = gene_ids,
                 mart = ensembl)

tt_merge$ensembl_gene_id <- rownames(tt_merge)
tt_annotated <- merge(tt_merge, geneDat, by = "ensembl_gene_id")

tt_annotated <- tt_annotated[, c("ensembl_gene_id", "external_gene_name", "entrezgene_id",
                                 "logFC", "AveExpr", "t", "P.Value", "adj.P.Val", "B")]

head(tt_annotated)
```

```
##   ensembl_gene_id external_gene_name entrezgene_id       logFC  AveExpr
## 1 ENSG00000000003             TSPAN6          7105  0.06385849 6.545041
## 2 ENSG00000000005               TNMD         64102  0.18128788 3.845240
## 3 ENSG00000000419               DPM1          8813 -0.69726956 8.056324
## 4 ENSG00000000457              SCYL3         57147 -0.16583302 6.040446
## 5 ENSG00000000460              FIRRM         55732  0.21194351 3.730130
## 6 ENSG00000000938                FGR          2268  0.00536728 6.778804
##             t     P.Value  adj.P.Val         B
## 1  0.17129185 0.865287307 0.93522714 -5.946200
## 2  1.96263019 0.060199360 0.23944823 -4.223632
## 3 -3.09296738 0.004608104 0.08138296 -2.039369
## 4 -1.24559146 0.223746720 0.47454200 -5.230485
## 5  2.20787740 0.036043337 0.18504539 -3.801831
## 6  0.01988833 0.984280506 0.99344443 -5.960229
```