Alternative Splicing and Regulary Events in Browning

Code

Hamza Umut Karakurt

24/09/2023

1 Introduction

1.1 Aim

Mainly, 2 types of adipocyte cell can be found in mammals; white and brown adipocytes. They originate from different stem cells and basically white adipocytes stores fat while brown adipocytes burns fat with their high mitochondrial activity. Normally, brown adipocytes cannot be found in adult humans but can be found in newborns, rodents and hibernating animals. On the other hand, decrease in temperature (and some certain chemicals) gives white adipocytes brownish characters and beige, white adipocytes with increased fat burning potential, adipocytes forms. This process is called *browning* and/or *thermogenesis*. Uncoupling Protein-1 (UCP-1) is a marker for thermogenesis and with increase in its expression level, thermogenesis process is activated. In this study, we aimed to use RNA-Seq data of Cold Beige, Warm Beige and Warm White adipocytes to identify alternative splicing and regulatory events that associated with thermogenesis.

1.2 Data Source and Experiments

In this study, we used an RNA-Seq data which involves Cold Beige, Warm Beige, Warm White, Warm Brown and Cold Brown adipocytes. The article also uses ChIP-Seq data for integrative analysis of these 5 adipocyte types. Researchers used NuTRAP mouse, a mouse that enables researcher to analyze epigenome and transcriptome at the same time. Experimental technique uses mRNAs via seperation/purification of ribosomes.

1.3 Notebook Content

This notebook includes analyses of RNA-Seq data of Cold Beige, Warm Beige and Warm White Adipocytes from the data set “GSE108077”.

2 Pre-process of Data

2.1 Alignment of Raw Fastq Reads

Analyses includes Alignment of raw RNA-Seq reads (in FASTQ format) using STAR (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3530905/) aligner to Mouse Genome and annotation Version 39 from Ensembl (https://www.ensembl.org/index.html), differential expression using DESeq2 (https://genomebiology.biomedcentral.com/articles/10.1186/s13059-014-0550-8) , alternative splicing analysis using ASPli (https://academic.oup.com/bioinformatics/article-abstract/37/17/2609/6156815?redirectedFrom=fulltext) and Reporter Regulatory Elements Analysis (RREA) (results are exported from Matlab 2021a).

Hide

```
hisat2 -p 12 -x "hisat2_genome_index" -1 "FASTQ_File_1" -2 "FASTQ_File_1" -S "output.sam"

samtools view -S -b "output.sam" > "output.bam"

samtools sort -@ 12 -o "output__sorted.bam" "output.bam"

samtools index -@ 12 "output_sorted.bam"
```

2.2 Processing of Genomic Properties for ASPli

Hide

```
library(ASpli)
library(GenomicFeatures)
library(biomaRt)

genomeTxDb <- makeTxDbFromGFF("GRCm39.105.gtf" , format = "gtf" , organism = "Mus musculus" , taxonom
yId = 10090)
genes <- genes(genomeTxDb)
gene_ids <- genes$gene_id

ensembl <- useMart("ensembl", dataset="mmusculus_gene_ensembl")

gs_heatdata <- getBM(attributes = c('external_gene_name','ensembl_gene_id'),filters = 'ensembl_gene_i
d',values = gene_ids,mart = ensembl)

symbols <- as.data.frame(gs_heatdata$external_gene_name)
rownames(symbols) <- gs_heatdata$ensembl_gene_id
colnames(symbols) <- c("symbol")
features <- binGenome(genomeTxDb , cores = 10)

save(features , genomeTxDb , file = "MM39_features_ensembl.RData")
```

2.3 Required packages For Notebook

Hide

```
library(ASpli)
```

```
Zorunlu paket yükleniyor: parallel
Zorunlu paket yükleniyor: edgeR
Zorunlu paket yükleniyor: limma
Zorunlu paket yükleniyor: AnnotationDbi
Zorunlu paket yükleniyor: stats4
Zorunlu paket yükleniyor: BiocGenerics

Attaching package: ‘BiocGenerics’

The following object is masked from ‘package:limma’:

    plotMA

The following objects are masked from ‘package:parallel’:

    clusterApply, clusterApplyLB, clusterCall, clusterEvalQ, clusterExport, clusterMap, parApply,
    parCapply, parLapply, parLapplyLB, parRapply, parSapply, parSapplyLB

The following objects are masked from ‘package:stats’:

    IQR, mad, sd, var, xtabs

The following objects are masked from ‘package:base’:

    anyDuplicated, 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

Zorunlu paket yükleniyor: Biobase
Welcome to Bioconductor

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

Zorunlu paket yükleniyor: IRanges
Zorunlu paket yükleniyor: S4Vectors

Attaching package: ‘S4Vectors’

The following objects are masked from ‘package:base’:

    expand.grid, I, unname


Attaching package: ‘IRanges’

The following object is masked from ‘package:grDevices’:

    windows

Registered S3 methods overwritten by 'dbplyr':
  method         from
  print.tbl_lazy
  print.tbl_sql
Registered S3 method overwritten by 'data.table':
  method           from
  print.data.table
Registered S3 method overwritten by 'htmlwidgets':
```

```
  method           from
  print.htmlwidget tools:rstudio
```

Hide

```
library(GenomicFeatures)
```

```
Zorunlu paket yükleniyor: GenomeInfoDb
Zorunlu paket yükleniyor: GenomicRanges
```

Hide

```
library(biomaRt)
library(clusterProfiler)
```

```
Registered S3 methods overwritten by 'treeio':
  method              from
  MRCA.phylo          tidytree
  MRCA.treedata       tidytree
  Nnode.treedata      tidytree
  Ntip.treedata       tidytree
  ancestor.phylo      tidytree
  ancestor.treedata   tidytree
  child.phylo         tidytree
  child.treedata      tidytree
  full_join.phylo     tidytree
  full_join.treedata  tidytree
  groupClade.phylo    tidytree
  groupClade.treedata tidytree
  groupOTU.phylo      tidytree
  groupOTU.treedata   tidytree
  is.rooted.treedata  tidytree
  nodeid.phylo        tidytree
  nodeid.treedata     tidytree
  nodelab.phylo       tidytree
  nodelab.treedata    tidytree
  offspring.phylo     tidytree
  offspring.treedata  tidytree
  parent.phylo        tidytree
  parent.treedata     tidytree
  root.treedata       tidytree
  rootnode.phylo      tidytree
  sibling.phylo       tidytree
Registered S3 method overwritten by 'ggtree':
  method      from
  identify.gg ggfun
clusterProfiler v4.0.5  For help: https://yulab-smu.top/biomedical-knowledge-mining-book/


Attaching package: 'clusterProfiler'

The following object is masked from 'package:biomaRt':

    select

The following object is masked from 'package:AnnotationDbi':

    select

The following object is masked from 'package:IRanges':

    slice

The following object is masked from 'package:S4Vectors':

    rename

The following object is masked from 'package:stats':

    filter
```

Hide

```
library(ggvenn)
```

```
Zorunlu paket yükleniyor: dplyr

Attaching package: 'dplyr'

The following object is masked from 'package:biomaRt':

    select

The following objects are masked from 'package:GenomicRanges':

    intersect, setdiff, union

The following object is masked from 'package:GenomeInfoDb':

    intersect

The following object is masked from 'package:AnnotationDbi':

    select

The following objects are masked from 'package:IRanges':

    collapse, desc, intersect, setdiff, slice, union

The following objects are masked from 'package:S4Vectors':

    first, intersect, rename, setdiff, setequal, union

The following object is masked from 'package:Biobase':

    combine

The following objects are masked from 'package:BiocGenerics':

    combine, intersect, setdiff, union

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union

Zorunlu paket yükleniyor: grid
Zorunlu paket yükleniyor: ggplot2
```

Hide

```
library(DESeq2)
```

```
Zorunlu paket yükleniyor: SummarizedExperiment
Zorunlu paket yükleniyor: MatrixGenerics
Zorunlu paket yükleniyor: matrixStats

Attaching package: 'matrixStats'

The following object is masked from 'package:dplyr':

    count

The following objects are masked from 'package:Biobase':

    anyMissing, rowMedians


Attaching package: 'MatrixGenerics'

The following objects are masked from 'package:matrixStats':

    colAlls, colAnyNAs, colAnys, colAvgsPerRowSet, colCollapse, colCounts, colCummaxs, colCummins,
    colCumprods, colCumsums, colDiffs, colIQRDiffs, colIQRs, colLogSumExps, colMadDiffs, colMads,
    colMaxs, colMeans2, colMedians, colMins, colOrderStats, colProds, colQuantiles, colRanges,
    colRanks, colSdDiffs, colSds, colSums2, colTabulates, colVarDiffs, colVars, colWeightedMads,
    colWeightedMeans, colWeightedMedians, colWeightedSds, colWeightedVars, rowAlls, rowAnyNAs,
    rowAnys, rowAvgsPerColSet, rowCollapse, rowCounts, rowCummaxs, rowCummins, rowCumprods,
    rowCumsums, rowDiffs, rowIQRDiffs, rowIQRs, rowLogSumExps, rowMadDiffs, rowMads, rowMaxs,
    rowMeans2, rowMedians, rowMins, rowOrderStats, rowProds, rowQuantiles, rowRanges, rowRanks,
    rowSdDiffs, rowSds, rowSums2, rowTabulates, rowVarDiffs, rowVars, rowWeightedMads,
    rowWeightedMeans, rowWeightedMedians, rowWeightedSds, rowWeightedVars

The following object is masked from 'package:Biobase':

    rowMedians
```

Hide

```
library(stringr)
library(ggsci)
library(RColorBrewer)
library(expss)
```

```
Zorunlu paket yükleniyor: maditr

To modify variables or add new variables:
             let(mtcars, new_var = 42, new_var2 = new_var*hp) %>% head()


Attaching package: 'maditr'

The following object is masked from 'package:SummarizedExperiment':

    shift

The following objects are masked from 'package:dplyr':

    between, coalesce, first, last

The following object is masked from 'package:biomaRt':

    columns

The following object is masked from 'package:GenomicRanges':

    shift

The following object is masked from 'package:AnnotationDbi':

    columns

The following object is masked from 'package:IRanges':

    shift

The following object is masked from 'package:S4Vectors':

    first

Registered S3 methods overwritten by 'expss':
  method                 from
  [.labelled             Hmisc
  as.data.frame.labelled base
  print.labelled         Hmisc

Use 'expss_output_rnotebook()' to display tables inside R Notebooks.
 To return to the console output, use 'expss_output_default()'.


Attaching package: 'expss'

The following objects are masked from 'package:stringr':

    fixed, regex

The following object is masked from 'package:ggplot2':

    vars

The following objects are masked from 'package:dplyr':

    compute, contains, na_if, recode, vars, where
```

```
The following objects are masked from 'package:IRanges':

    from, to

The following objects are masked from 'package:S4Vectors':

    from, to
```

Hide

```
library(org.Mm.eg.db)
```

Hide

```
library(reshape)
```

```

Attaching package: 'reshape'

The following object is masked from 'package:maditr':

    melt

The following object is masked from 'package:dplyr':

    rename

The following object is masked from 'package:clusterProfiler':

    rename

The following objects are masked from 'package:S4Vectors':

    expand, rename
```

Hide

```
library(gridExtra)
```

```

Attaching package: 'gridExtra'

The following object is masked from 'package:dplyr':

    combine

The following object is masked from 'package:Biobase':

    combine

The following object is masked from 'package:BiocGenerics':

    combine
```

Hide

```
library(knitr)
library(mygene)
```

```
Attaching package: 'mygene'

The following object is masked from 'package:maditr':

    query

The following object is masked from 'package:biomaRt':

    getGene
```

3 Data Analysis

3.1 Features (Genes, Exons, Introns, Junctions) Count Matrices

This section shows the preparation of count matrices using ASPli. Since ASPli also created gene counts matrix, an extra step using FeatureCounts did not included here.

12 samples (4 Cold Beige, 3 Warm Beige and 5 Warm White) are used. BAM files are stored in a folder and for each condition merged bam files were generated using Samtools (https://pubmed.ncbi.nlm.nih.gov/19505943/) in order to use in ASPli.

This section has "eval=FALSE" parameter and it will not run. The section takes about 2 hours (1 hour for feature generation and 1 hour for ASPli counts).

Hide

```
load("~/Documents/R/ADmanu_mm/MM39_features_ensembl.RData")

BAMFiles <- list.files(path="/home/huk/Documents/R/ADmanu_mm/BAM_Files/STAR" , pattern = ".bam" , ful
l.names = TRUE)

targets <- data.frame(row.names = paste0('Sample',c(1:12)),
                      bam = BAMFiles[seq(3,25,by=2)],
                      temperature = c(rep("Cold",4),rep("Warm",8)),
                      cell_type = c(rep("Beige",7) , rep("White",5)),
                      stringsAsFactors = FALSE)

mBAMs <- data.frame(bam = BAMFiles[c(1,27,29)],
                    condition = c("Cold_Beige","Warm_Beige","Warm_White"))


gbcounts <- gbCounts(features=features, targets=targets,
                     minReadLength = 50, maxISize = 50000 , libType = "PE")


asd <- jCounts(counts=gbcounts, features=features, minReadLength=50 , libType = "PE")
```

Design matrix includes temperature and cell type parameters. Formula uses only temperature for comparison. Biologically, temperature decreases and certain chemicals gives white adipocytes a brownish character and generates beige adipocytes. Due to white adipocytes can transform to beige adipocytes, another analysis for cell types did not applied.

In this section bins (exons, introns and junctions) were analyzed.

*This section has "eval=FALSE" parameter and it will not run. The section takes about 2 hours

Hide

```
######################### ASPLI ANALYSES ############################

form <- formula(~temperature)
model.matrix(form,targets)

gb <- gbDUreport(gbcounts, formula = form)

jdur <- jDUreport(asd, formula = form)

sr <- splicingReport(gb, jdur, counts=gbcounts)

is <- integrateSignals(sr,asd , bin.fdr = 0.01)

exportSplicingReports(sr , output.dir = "cbg_wbg_wwh_splice" , maxBinFDR = 0.01 , maxJunctionFDR = 0.
01)

exportIntegratedSignals(is,sr=sr,
                        output.dir = "cbg_wbg_wwh_splice_integrated",
                        counts=gbcounts,features=features,asd=asd,
                        mergedBams = mBAMs , bforce = TRUE)
```

GeneSymbols of Ensembl IDs in order to use in conversion of Ensembl IDs to GeneSymbols.

Hide

```
genomeTxDb <- makeTxDbFromGFF("GRCm39.105.gtf" , format = "gtf" , organism = "Mus musculus" , taxonom
yId = 10090)
genes <- genes(genomeTxDb)
gene_ids <- genes$gene_id
ensembl <- useMart("ensembl", dataset="mmusculus_gene_ensembl")
gs_heatdata <- getBM(attributes = c('external_gene_name','ensembl_gene_id'),filters = 'ensembl_gene_i
d',values = gene_ids,mart = ensembl)
```

Data From Previous Analyses are saved in a file:

Hide

```
load("cbg_wbg_wwh_ensembl.RData")
```

3.2 Differential Expression Analysis

DESeq2 package is used for differential expression analysis. Gene Count Matrix is extracted from ASPli object. Genes with total expression in all samples lower than 36 are removed from the data. Results are restored in a variable.

```
using pre-existing size factors
estimating dispersions
gene-wise dispersion estimates
mean-dispersion relationship
final dispersion estimates
fitting model and testing
-- replacing outliers and refitting for 18 genes
-- DESeq argument 'minReplicatesForReplace' = 7
-- original counts are preserved in counts(dds)
estimating dispersions
fitting model and testing
```

Genes and Adjusted P-Values are stored in in “gene_pval_rep” variable for using in Reporter Regulatory Elements Analysis (RREA) in Matlab2020a via in-house scripts. Ensembl Gene IDs are converted to GeneSymbols using *vlookup* function from expss (https://cran.r-project.org/web/packages/expss/index.html) package.

Hide

```
gene_pval_rep <- as.data.frame(cbind(as.character(resdata_dds$Row.names) , as.numeric(resdata_dds$pad
j)))
gene_pval_rep[,1] <- as.character(vlookup(gene_pval_rep[,1] , dict = gs_heatdata , result_column = 1
, lookup_column = 2))
colnames(gene_pval_rep) <- c("Symbol" , "P_Val")
gene_pval_rep <- gene_pval_rep %>% group_by(Symbol) %>% summarise(P=min(P_Val))
gene_pval_rep <- as.data.frame(gene_pval_rep)
tf_gene_network <- read.table("TF-Gene.txt" , header = F , sep = "\t")
non_tfs <- unique(tf_gene_network[,2])
rownames(gene_pval_rep) <- gene_pval_rep$Symbol
gene_pval_rep <- na.omit(gene_pval_rep)
write.table(gene_pval_rep , file = "symbol_padjs.txt" , sep = " \t" , row.names = F , col.names = F ,
quote = F)
```

Genes with adjusted p-values lower than 0.01 and absolute Log2Fold Change Value higher than 1 are considered as *significantly changed genes*. Differentially expressed genes are stored in “gene_de” variable.

3.3 Enrichment Analysis of Differentially Expressed Genes

Enrichment analysis using GeneOntology Biological Process (BP), Molecular Function (MF) and Cellular Component (CC) is applied to significantly changed genes using clusterProfiler (https://www.cell.com/the-innovation/fulltext/S2666-6758(21)00066-7?_returnURL=https%3A%2F%2Flinkinghub.elsevier.com%2Fretrieve%2Fpii%2FS2666675821000667%3Fshowall%3Dtrue) package.

Hide

```
gene_de <- as.data.frame(cbind(as.numeric(resdata_dds$log2FoldChange) , as.numeric(resdata_dds$pad
j)))
colnames(gene_de) <- c("log2FC" , "p_adj")
rownames(gene_de) <- resdata_dds$Row.names
gene_de <- gene_de[gene_de$p_adj < 0.01,]
gene_de <- gene_de[(abs(gene_de$log2FC) > 1.5),]

gene_enrich_up <- enrichGO(rownames(gene_de)[gene_de$log2FC > 0] , OrgDb = "org.Mm.eg.db", keyType =
"ENSEMBL" , ont = "all")
gene_enrich_down <- enrichGO(rownames(gene_de)[gene_de$log2FC < 0] , OrgDb = "org.Mm.eg.db", keyType
= "ENSEMBL" , ont = "all")
```

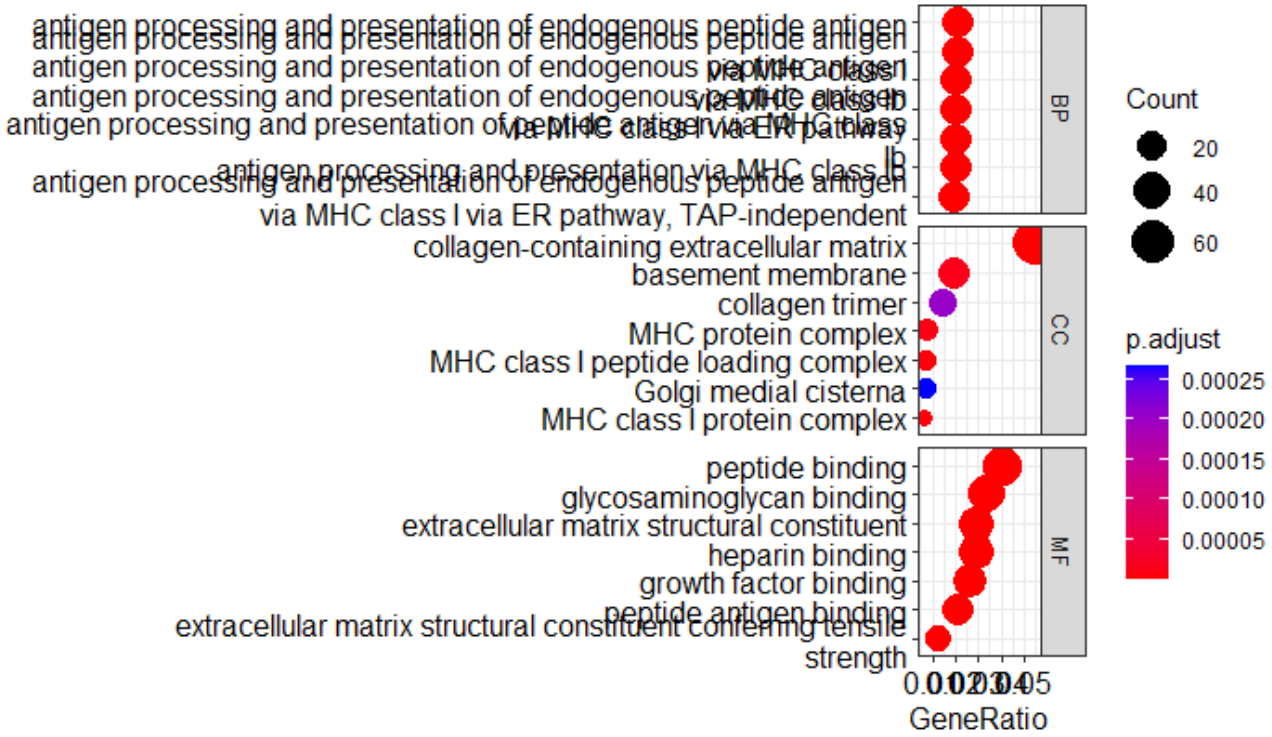
3.3.1 Dotplots of DE Genes.

Plots correspond to BP, CC and MF terms of GeneOntology respectively (If there is any term enriched)

3.3.1.1 Upregulated | 3.3.1.2 Downregulated

Hide

```
dotplot(gene_enrich_up, split="ONTOLOGY" , showCategory = 7 , label_format = 70) + facet_grid(ONTOLOG
Y~., scale="free")
```



3.4 Differential AS Analysis

ASPli package has edgeR package for significancy analyses. Using built-in functions of libraries, differential alternative splicing (AS) events are identified significantly. Since there are different types of alternative splicing events and ASPli can identify numerous of events (including Novel AS Events and Unidentified AS Events), here we focused 4 main classes of alternative splicing events. Exon Skipping (ES), Intron Retention (IR), Alternative 3' Splicing Site (Alt3ss) and Alternative 5' Splicing Site (Alt5ss)

Differential Signals from all detected AS events. IR, ES, Alt3ss and Alt5ss are chosen.

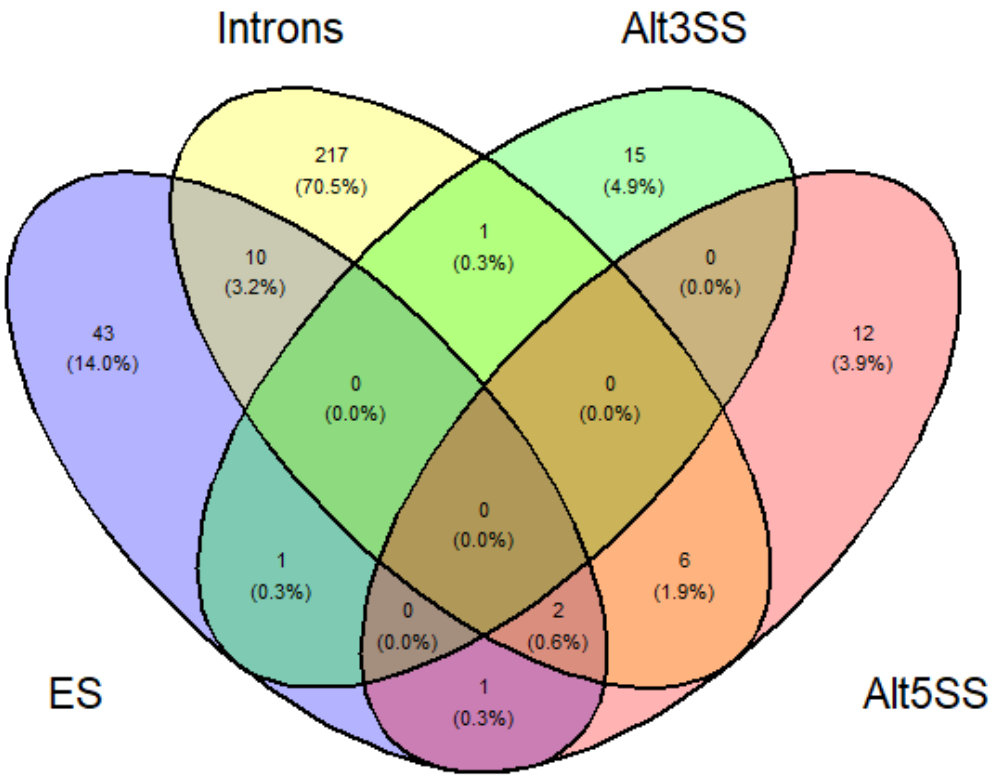
Hide

```
signals_all <- as.data.frame(is@signals)
signal_types <- as.data.frame(cbind(signals_all$locus , signals_all$bin.event))
signal_types[,1] <- as.character(vlookup(signal_types[,1] , dict = gs_heatdata , result_column = 1 ,
lookup_column = 2))
write.table(signal_types , file = "as_types.txt" , row.names = F , col.names = F , quote = F , sep =
"\t")

signals <- signals_all[signals_all$bin.event == "IR" | signals_all$bin.event == "ES" | signals_all$bi
n.event == "Alt5ss" | signals_all$bin.event == "Alt3ss",]
es_signals <- signals[signals$bin.event == "ES",]
ir_signals <- signals[signals$bin.event == "IR",]
ir_signals <- ir_signals[ir_signals$feature == "I",]
alt3_signals <- signals[signals$bin.event == "Alt3ss",]
alt5_signals <- signals[signals$bin.event == "Alt5ss",]


write.table(signals , file = "AS_Events.txt" , row.names = T , col.names = T , quote = F , sep = "\t"
, dec = ",")
```

1294 significant AS events are identified via ASPli analyses (326 Unique Genes). Venn Scheme to identify if a gene have multiple AS events. Number of AS Events in Each Class

Hide

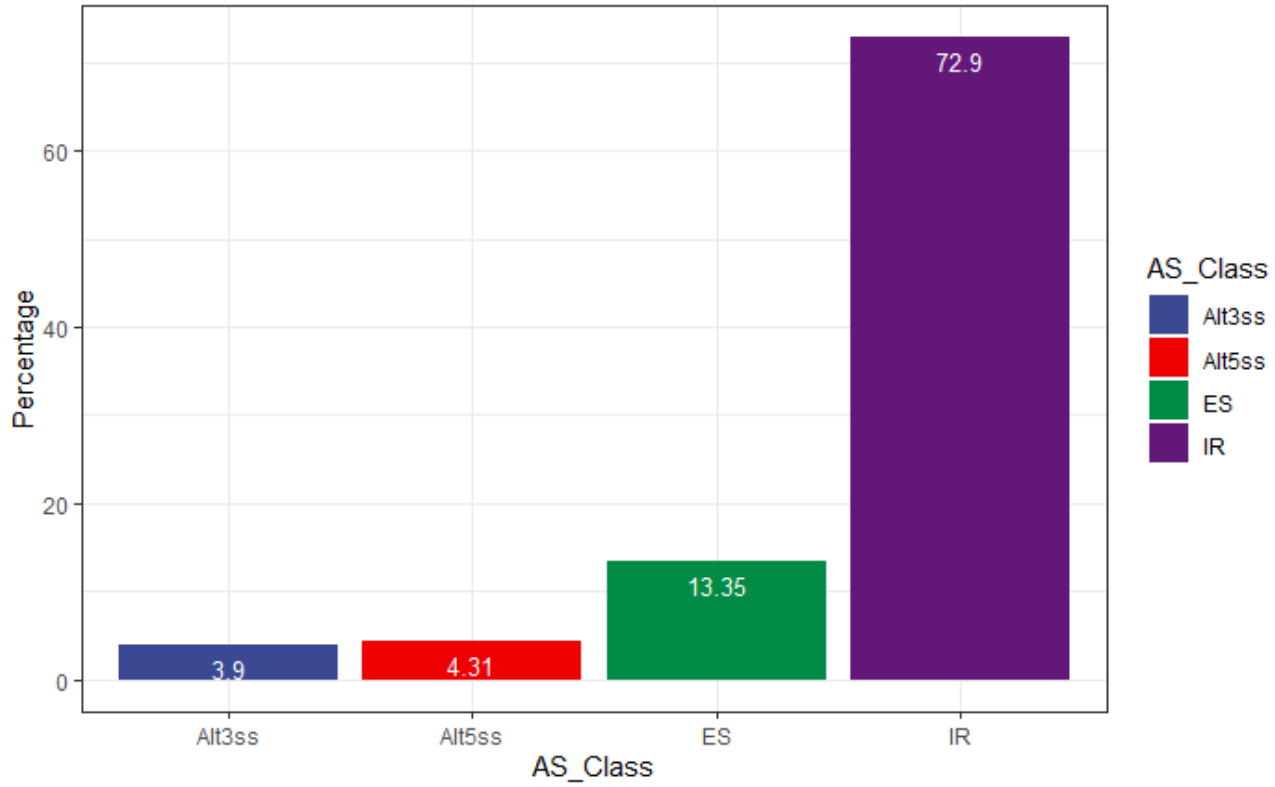
```
ggvenn(list(ES = es_signals$locus , Introns = ir_signals$locus , Alt3SS = alt3_signals$locus , Alt5SS = alt5_signals$locus) , fill_alpha = 0.3 , text_size = 3)
```



Percentage of 4 Selected AS Events in all AS Events. For the selected 4 classes, the number of AS events are 460.

Hide

```
number_of_as_in_each_class = data.frame(AS_Class = c("ES" , "IR" ,"Alt3ss" , "Alt5ss") ,
                                        Percentage = c((nrow(es_signals) / nrow(signals) * 100) , (nrow(ir_signals) / nrow(signals) * 100) , (nrow(alt3_signals) / nrow(signals) * 100) , (nrow(alt5_signals) / nrow(signals) * 100)))

ggplot(data=number_of_as_in_each_class, aes(x=AS_Class, y=Percentage , fill = AS_Class)) + geom_bar(stat="identity") + scale_fill_aaas() + theme_bw() + geom_text(aes(label=round(Percentage,digits = 2)), vjust=1.6, color="white",position = position_dodge(0.9), size=3.5)
```



3.4.1 Enrichment Analysis of Genes with Significant AS Events.

For the class of AS Events (ES, IR, Alt3ss and Alt5ss) Ensembl Gene IDs directly used for enrichment analysis and results are plotted as dotplots (if there is significantly enriched terms.)

Hide

```
es_enrich_up <- enrichGO(es_signals$locus[es_signals$b.logfc > 0] , OrgDb = "org.Mm.eg.db", keyType =
"ENSEMBL" , ont = "all")
es_enrich_down <- enrichGO(es_signals$locus[es_signals$b.logfc < 0] , OrgDb = "org.Mm.eg.db", keyType
= "ENSEMBL" , ont = "all")

ir_enrich_up <- enrichGO(ir_signals$locus[ir_signals$b.logfc > 0] , OrgDb = "org.Mm.eg.db", keyType =
"ENSEMBL" , ont = "all")
ir_enrich_down <- enrichGO(ir_signals$locus[ir_signals$b.logfc < 0] , OrgDb = "org.Mm.eg.db", keyType
= "ENSEMBL" , ont = "all")
```

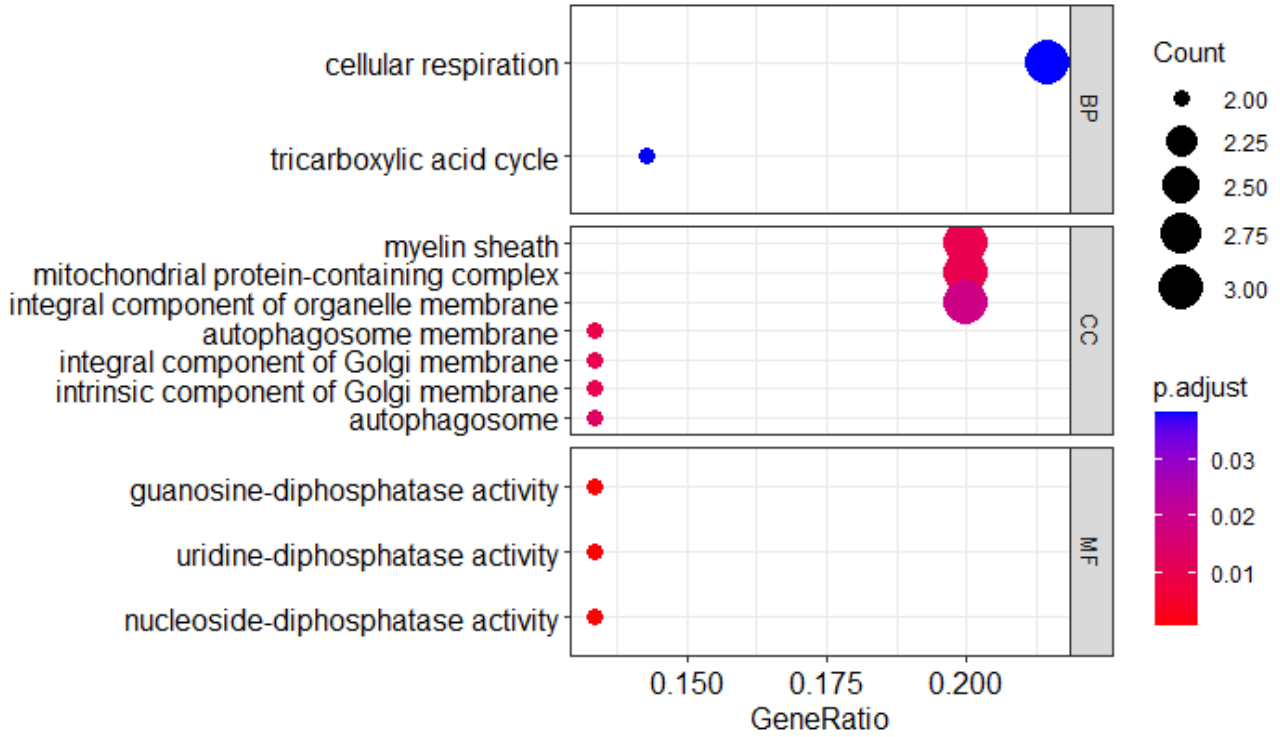
3.4.2 Dotplots of Upregulated and Downregulated with differential AS events.

Plots correspond to BP, CC and MF terms of GeneOntology respectively (If there is any term enriched)

3.4.2.1 Upregulated | 3.4.2.2 Downregulated

Hide

```
dotplot(es_enrich_up, split="ONTOLOGY" , showCategory = 7 , label_format = 70) + facet_grid(ONTOLOGY
~., scale="free")
```



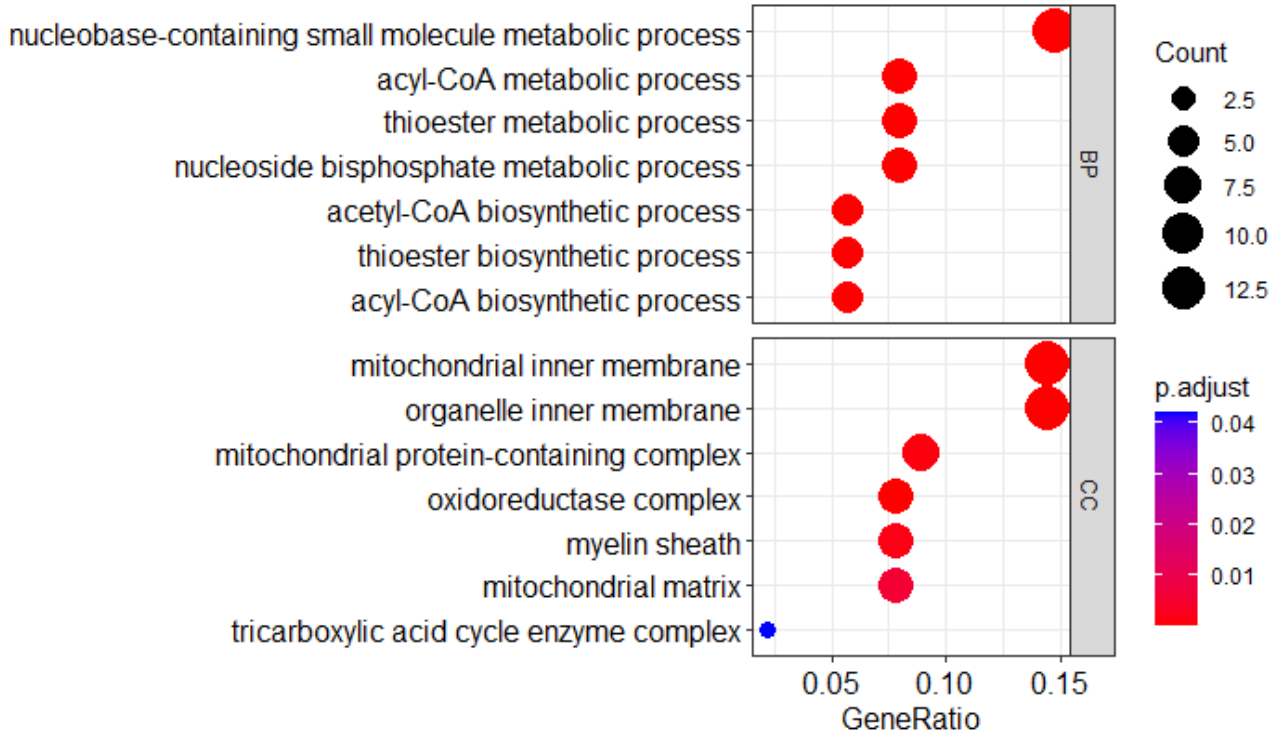
3.4.3 Dotplots of genes with differential IR events.

Plots correspond to BP, CC and MF terms of GeneOntology respectively (If there is any term enriched)

3.4.3.1 Upregulated | 3.4.3.2 Downregulated

Hide

```
dotplot(ir_enrich_up, split="ONTOLOGY" , showCategory = 7 , label_format = 70) + facet_grid(ONTOLOGY ~., scale="free")
```



3.5 Heatmap of Significant ES and IR Events

Count matrices of Exons and Introns are extracted from ASPli object. For both of the matrices, features with total count lower than 36 across samples are removed. Matrices are stored (only stored) in DESeq Objects for easier manipulation

Hide

```
##### EXON COUNTS #####
BinCounts <- countsb(gbcounts)
exons <- BinCounts[BinCounts$feature == "E",]
exon_matrix <- as.matrix(exons[,c(10:21)])
exon_matrix <- exon_matrix[rowSums(exon_matrix) > 36,]
sub <- apply(exon_matrix, 1, function(row) all(row !=0 ))
exon_matrix <- exon_matrix[sub,]
exon_matrix <- na.omit(exon_matrix)
colnames(exon_matrix) <- c(paste("CBG" , c(1:4) , sep = "") , paste("WBG" , c(1:3) , sep = "") , pas
te("WWH" , c(1:5) , sep = ""))

##### INTRON COUNTS #####
BinCounts <- countsb(gbcounts)
introns <- BinCounts[BinCounts$feature == "I",]
intron_matrix <- as.matrix(introns[,c(10:21)])
intron_matrix <- intron_matrix[rowSums(intron_matrix) > 36,]
sub <- apply(intron_matrix, 1, function(row) all(row !=0 ))
intron_matrix <- intron_matrix[sub,]
intron_matrix <- na.omit(intron_matrix)
colnames(intron_matrix) <- c(paste("CBG" , c(1:4) , sep = "") , paste("WBG" , c(1:3) , sep = "") , p
aste("WWH" , c(1:5) , sep = ""))

condition <- c(paste("CBG" , c(1:4) , sep = "") , paste("WBG" , c(1:3) , sep = "") , paste("WWH" , c
(1:5) , sep = ""))
```

3.5.1 Percentages of features.

Hide

```
a <- nrow(intron_matrix) / (nrow(intron_matrix) + nrow(exon_matrix)) * 100
b <- nrow(exon_matrix) / (nrow(intron_matrix) + nrow(exon_matrix)) * 100
c <- nrow(intron_matrix) / sum(features@bins$feature == "I") * 100
d <- nrow(exon_matrix) / sum(features@bins$feature == "E") * 100
e <- sum(features@bins$feature == "I") / sum(features@bins$feature == "E") * 100
print(paste("The Ratio of Introns: " , round(a , digits = 2)))
```

```
[1] "The Ratio of Introns:  24.36"
```

Hide

```
print(paste("The Ratio of Exons: " , round(b , digits = 2)))
```

```
[1] "The Ratio of Exons:  75.64"
```

Hide

```
print(paste("The Ratio of Expressed Introns to All Introns: " , round(c , digits = 2)))
```

```
[1] "The Ratio of Expressed Introns to All Introns:  21.76"
```

Hide

```
print(paste("The Ratio of Expressed Exons to All Exons: " , round(d , digits = 2)))
```

```
[1] "The Ratio of Expressed Exons to All Exons:  33.27"
```

Hide

```
print(paste("The Ratio of All Introns to Exons in Total Features: " , round(e , digits = 2)))
```

```
[1] "The Ratio of All Introns to Exons in Total Features:  49.24"
```

Hide

```
rm(a,b,c,d,e)
```

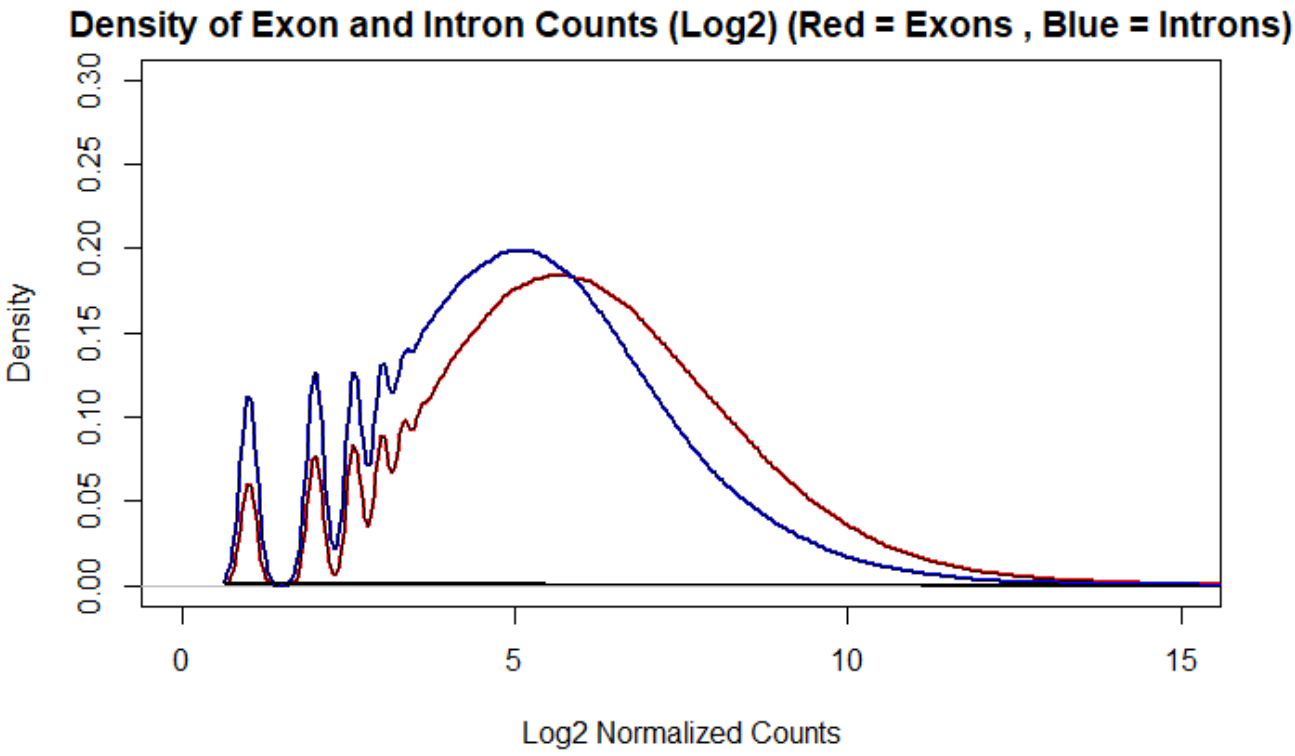
3.5.2 Density plot of exon and intron counts.

Hide

```
d_ex <- density(log2(exon_matrix) + 1)
d_in <- density(log2(intron_matrix) + 1)

plot(d_ex, lwd = 2, col = "red",main = "Density of Exon and Intron Counts (Log2) (Red = Exons , Blue
= Introns)",
     xlab = "Log2 Normalized Counts",xlim = c(0 , 15), ylim = c(0 , 0.3))
polygon(d_ex, col = rgb(1, 0, 0, alpha = 0.5))
```

Hide

```


lines(d_in, col = "blue", lwd = 2)
polygon(d_in, col = rgb(0, 0, 1, alpha = 0.5))
```



3.5.3 Heatmaps of Alternatively Spliced Introns and Exons.

Bin Counts of Alternatively Spliced Introns and Exons extracted from BinCounts object. All exon counts with their gene symbols are plotted while the most variant introns (equal to number of exons) are plotted due to have better visualization. Counts are shown in Supplementary File.

Associated Exon and Intron counts of genes

Hide

```
genes_with_es <- unique(es_signals$bin)
genes_with_ir <- unique(ir_signals$bin)

#match_ex <- grep(paste(genes_with_es,collapse = "|") , rownames(BinCounts))
#match_ir <- grep(paste(genes_with_ir,collapse = "|") , rownames(BinCounts))

selected_ex <- BinCounts[genes_with_es,]
selected_ex <- selected_ex[,c(10:21)]
selected_ex <- na.omit(selected_ex)
selected_ex <- as.matrix(selected_ex[rowSums(selected_ex) > 24,])
selected_ir <- BinCounts[genes_with_ir,]
selected_ir <- selected_ir[,c(10:21)]
selected_ir <- na.omit(selected_ir)
selected_ir <- as.matrix(selected_ir[rowSums(selected_ir) > 24,])

colnames(selected_ex) <- c(paste("CBG" , c(1:4) , sep = "") , paste("WBG" , c(1:3) , sep = "") , pas
te("WWH" , c(1:5) , sep = ""))
colnames(selected_ir) <- c(paste("CBG" , c(1:4) , sep = "") , paste("WBG" , c(1:3) , sep = "") , pas
te("WWH" , c(1:5) , sep = ""))
```

Hide

```
selected_ex <- log2(selected_ex + 1)
selected_ir <- log2(selected_ir + 1)

df <- melt(selected_ex)
```

```
Warning in type.convert.default(X[[i]], ...) :
  'as.is' should be specified by the caller; using TRUE
Warning in type.convert.default(X[[i]], ...) :
  'as.is' should be specified by the caller; using TRUE
```

Hide

```
colnames(df) <- c("Exon", "Sample", "value")
selected_ex_genes <- str_split_fixed(df$Exon , ":" , 2)
selected_ex_genes[,1] <- vlookup(selected_ex_genes[,1] ,dict = gs_heatdata , result_column = 1 , look
up_column = 2)
df$Exon <- paste(selected_ex_genes[,1] , selected_ex_genes[,2] , sep = ":")


var_intron <- apply(selected_ir, 1, var)
selected_intron <- names(sort(var_intron, decreasing=TRUE))[1:nrow(selected_ex)]

df2 <- melt(selected_ir[selected_intron,])
```

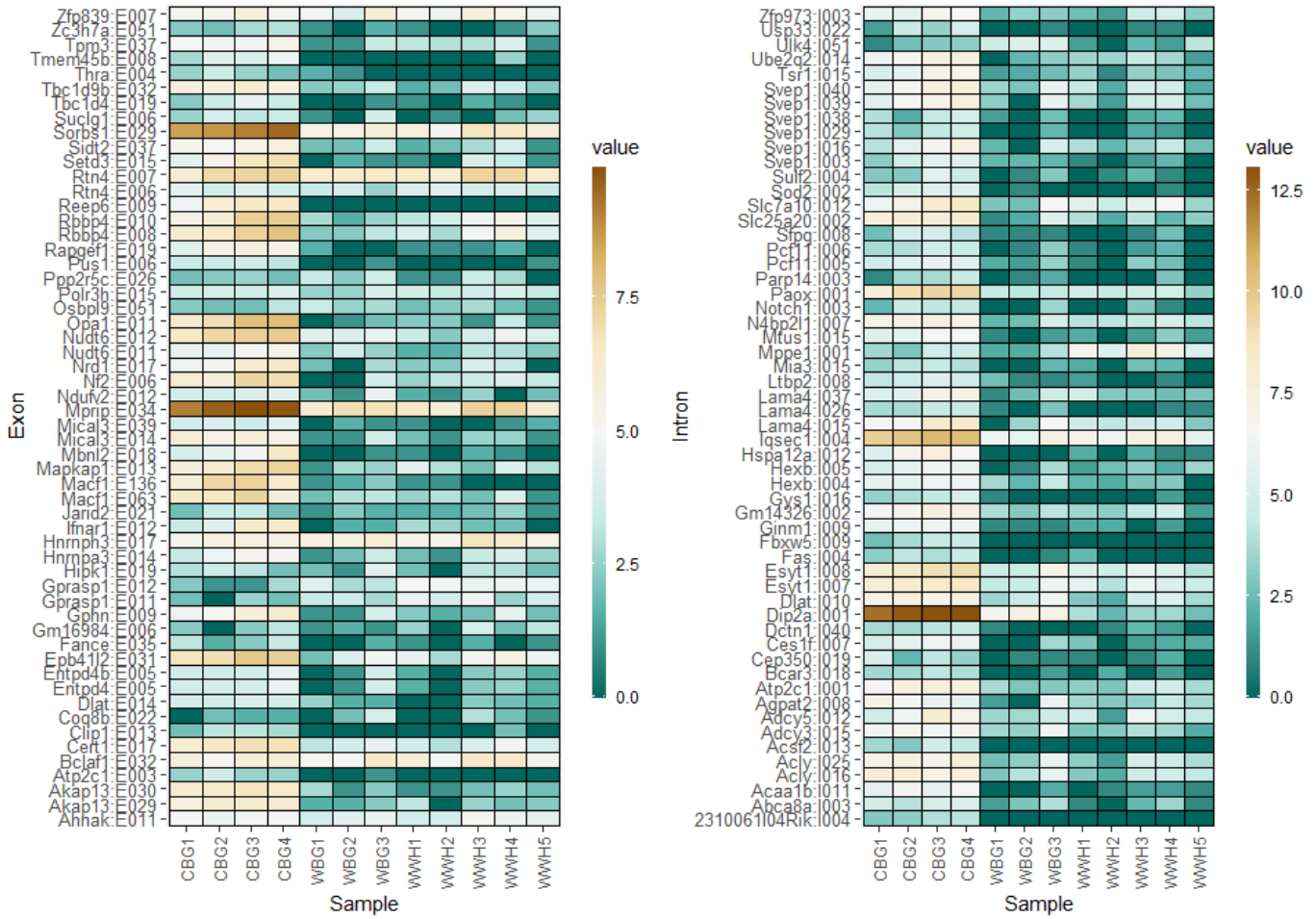
```
Warning in type.convert.default(X[[i]], ...) :
  'as.is' should be specified by the caller; using TRUE
Warning in type.convert.default(X[[i]], ...) :
  'as.is' should be specified by the caller; using TRUE
```

Hide

```
colnames(df2) <- c("Intron", "Sample", "value")
selected_ir_genes <- str_split_fixed(df2$Intron , ":" , 2)
selected_ir_genes[,1] <- vlookup(selected_ir_genes[,1] ,dict = gs_heatdata , result_column = 1 , look
up_column = 2)
df2$Intron <- paste(selected_ir_genes[,1] , selected_ir_genes[,2] , sep = ":")
```

Hide

```
a <- ggplot(df, aes(x = Sample, y = Exon, fill = value)) + geom_tile(color = "black")  +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1) , axis.text.y = element_text(siz
e = 10)) + scale_fill_distiller(palette = "BrBG") +
  guides(fill = guide_colourbar(barwidth = 0.5,
                                barheight = 20))

b <- ggplot(df2, aes(x = Sample, y = Intron, fill = value)) + geom_tile(color = "black")  +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1) , axis.text.y = element_text(siz
e = 10) ) + scale_fill_distiller(palette = "BrBG") +
  guides(fill = guide_colourbar(barwidth = 0.5,
                                barheight = 20))

grid.arrange(a, b, ncol=2)
```



3.6 Correlation Based Analyses

Ucp-1 (ENSMUSG00000031710) is a well-known marker of thermogenesis. As known, the fat-burning capacity increases of an adipocyte is increases with Ucp-1 expression. This protein is considered as a potential drug target for obesity patients.

The correlation of exon usage and Ucp-1 expression is a potentially useful analyses to show the alterations in exon usage in termogenesis process. exons with correlation (R value) is higher than 0.9 or lower than -0.9 are considered as *UCP1-correlated Exons and Introns*.

Correlation of Introns

Hide

```
ucp1_exp <- as.numeric(counts(dds_counts, normalized=TRUE)["ENSMUSG00000031710",])
correlations <- c()
normalized_intron <- log2(intron_matrix +1)
for (i in 1:nrow(normalized_intron)) {
  correlations[i] <- cor(x = normalized_intron[i,], y = ucp1_exp , method = "spearman")
}

high_corr_introns <- normalized_intron[(correlations > 0.9 | correlations < -0.9),]
high_corr_introns <- as.data.frame(rownames(normalized_intron[(correlations > 0.9 | correlations < -
0.9),]))
high_corr_introns <- as.data.frame(str_split_fixed(high_corr_introns[,1], ":", 2))

go_res_corr_intron <- enrichGO(as.character(high_corr_introns[,1]), OrgDb = "org.Mm.eg.db", keyType =
"ENSEMBL" , ont = "all")
```

454 introns out of 47804 were identified as termogenesis-correlated introns. These introns are belong to 350 genes. To identify the common functions of these genes, enrichment analysis (using *clusterProfiler* library) were applied. GeneOntology “*Biological Process, Cellular Component and Molecular Function*” databases were used in analyses.

Correlation of Exons

Hide

```
correlations <- c()
normalized_exons <- log2(exon_matrix + 1)
for (i in 1:nrow(normalized_exons)) {
  correlations[i] <- cor(x = normalized_exons[i,], y = ucp1_exp , method = "spearman")
}

high_corr_exons <- normalized_exons[(correlations > 0.9 | correlations < -0.9),]
high_corr_exons <- as.data.frame(rownames(normalized_exons[(correlations > 0.9 | correlations < -0.
9),]))
high_corr_exons <- as.data.frame(str_split_fixed(high_corr_exons[,1], ":", 2)) # Exon names are in th
e structure as "Gene:ExonIndex". They were splitted for the enrichment analysis

go_res_corr_exon <- enrichGO(as.character(high_corr_exons[,1]), OrgDb = "org.Mm.eg.db", keyType = "EN
SEMBL" , ont = "all")
```

15457 exons out of 148425 were identified as thermogenesis-correlated exons. These exons are belong to 4658 genes. To identify the common functions of these genes, enrichment analysis (using *clusterProfiler* library) were applied. GeneOntology “*Biological Process, Cellular Component and Molecular Function*” databases were used in analyses.

highly correlated introns and exons identified. For unique genes in correlated exons and introns, enrichment analysis applied as well.

Enrichment analysis for genes associated with unique exons and introns in variant features lists.

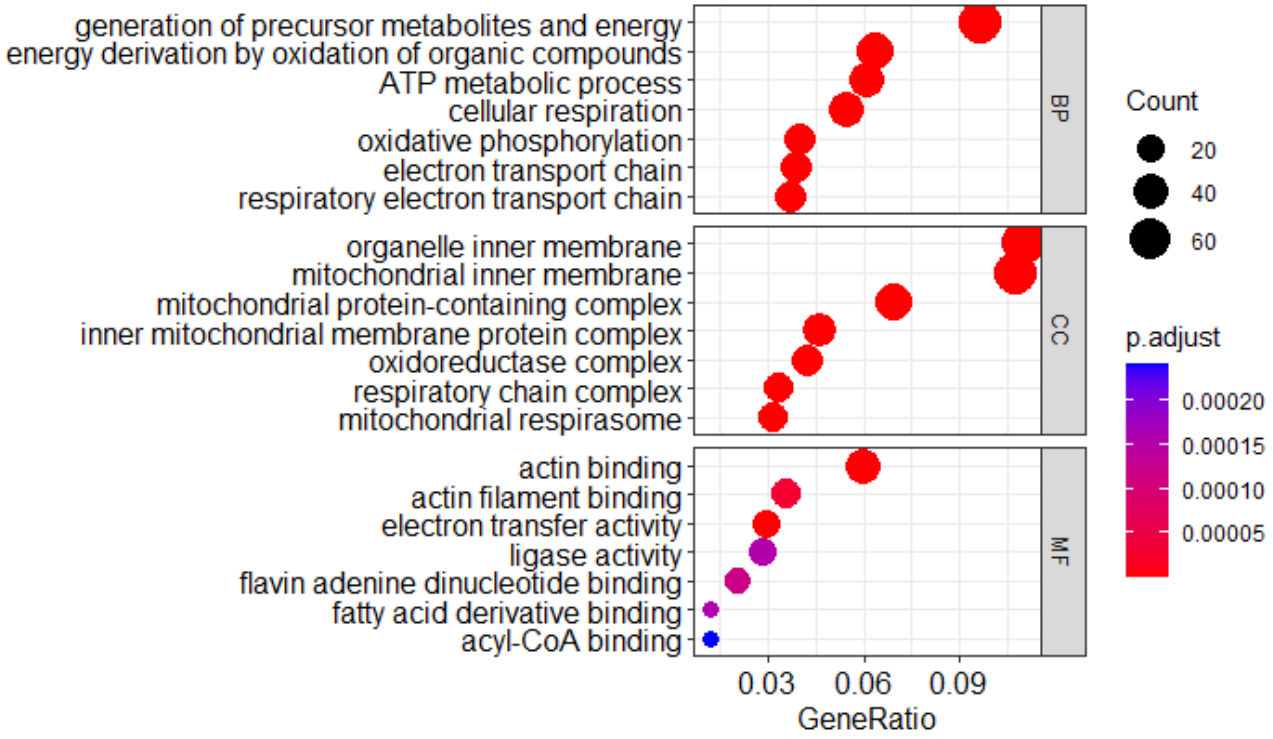
3.6.1 Dotplots of Enrichment Analysis Results of Ucp1-Correlated Exons and Introns.

Plots correspond to BP, CC and MF terms of GeneOntology respectively (If there is any term enriched)

3.6.1.1 Exons | 3.6.1.2 Introns

Hide

```
dotplot(go_res_corr_exon, split="ONTOLOGY" , showCategory = 7 , label_format = 70) + facet_grid(ONTOLOGY~., scale="free")
```



4 Comparison with DE, RREA and All AS Events.

RREA uses gene p-values to calculate p-values for regulators. Basically it is an implementation of Reporter Metabolite Analysis (https://www.pnas.org/doi/10.1073/pnas.0406811102) and Reporter Pathway Analysis (https://www.nature.com/articles/srep14563). As mentioned, analysis calculated P-Values for regulators (transcription factors) but differential expression analysis also gives p-values for these regulators since they also expressed as mRNAs. The difference between two analyses is that, differential expression calculates the change of gene expression between conditions while RREA calculates the function/activity changes. So we hypothesized that transcription factors that are significantly altered in RREA but not signficantly changed in the context of expression may have any change in post-translational modifications that alters the function/activity.

First step of this analysis is to identify transcription factors that signficantly altered in RREA (p-value < 0.05) but did not signficiantly changed in differential expression (p-adjusted > 0.01).

4.1 Potentially Post-Translationally Modificated Transcription Factors

Hide

```
rra_results <- read.table("reporters.txt" , header = T)
rra_results$ensembls <- as.character(vlookup(rra_results$TF , dict = gs_heatdata , result_column = 2
, lookup_column = 1))
rra_results <- na.omit(rra_results)
rownames(rra_results) <- rra_results$ensembls

rra_sig <- rra_results[rra_results$P.Value < 0.05,]
rra_sig <- rra_sig[rra_sig$Edge > 4,]
rra_sig <- rra_sig[rra_sig$Edge < 400,]
```

Hide

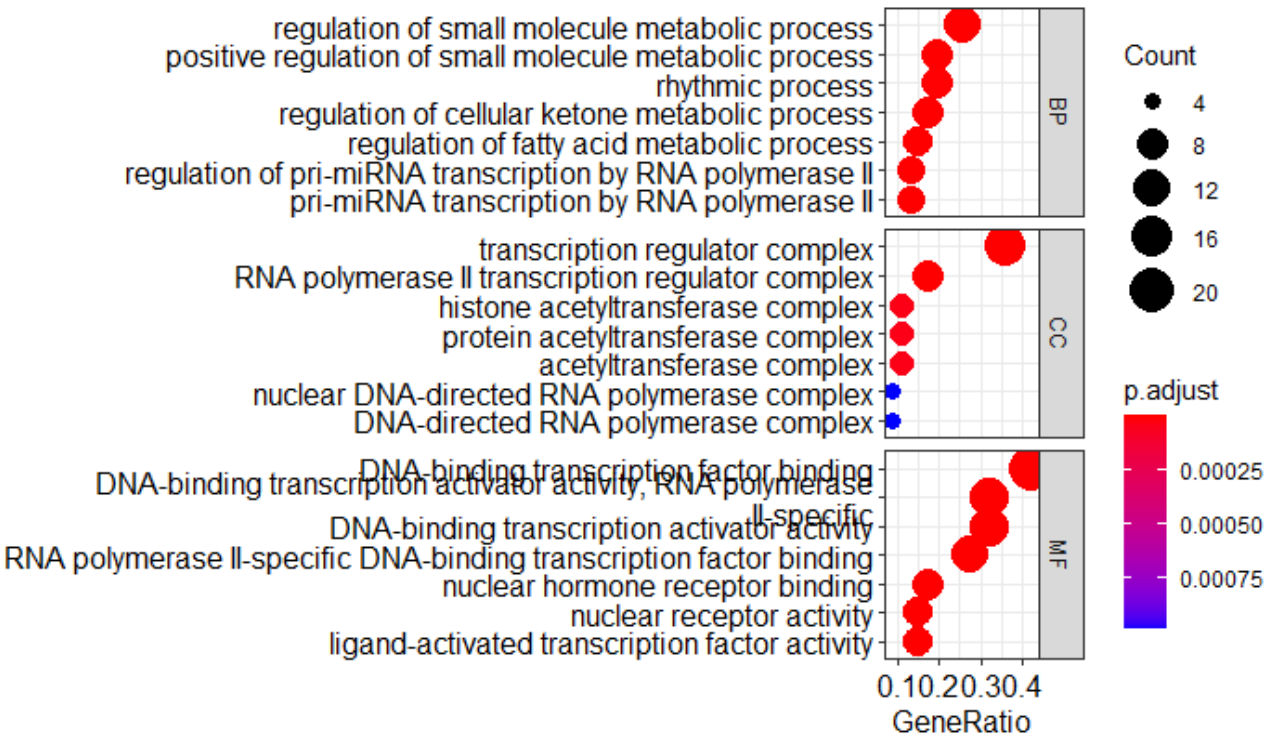
```
rra_go <- enrichGO(rownames(rra_sig), OrgDb = "org.Mm.eg.db", keyType = "ENSEMBL" , ont = "all")

rra_kegg <- enrichKEGG(bitr(rownames(rra_sig) , fromType = "ENSEMBL" , toType = "ENTREZID" , OrgDb="o
rg.Mm.eg.db")[,2] , organism = "mmu")
```

```
Reading KEGG annotation online:

Warning in utils::download.file(url, quiet = TRUE, method = method, ...) :
  URL 'https://rest.kegg.jp/link/mmu/pathway': status was 'Failure when receiving data from the peer'
fail to download KEGG data...
Error in download.KEGG.Path(species) :
  'species' should be one of organisms listed in 'http://www.genome.jp/kegg/catalog/org_list.html'...
```

4.1.1 Dotplots of Reporter Regulators.

Plots correspond to BP, CC and MF terms of GeneOntology respectively (If there is any term enriched)

4.1.1.1 GeneOntology | 4.1.1.2 KEGG



4.1.2 Dotplots of Regulators with Potential Post-translational

Modification.

Plots correspond to BP, CC and MF terms of GeneOntology respectively (If there is any term enriched)

4.1.2.1 GeneOntology | 4.1.2.2 KEGG

Hide

```
dotplot(rra_pm_go, split="ONTOLOGY" , showCategory = 7 , label_format = 70) + facet_grid(ONTOLOGY~., scale="free")
```

4.2 Combination of Alternative Splicing Events and Post-Translationally Modificated Transcription Factors

Hide

```
all_as <- as.data.frame(cbind(signals$locus,signals$bin.event))
possible_post_modification$in_as_signal <- possible_post_modification$ensembls %in% all_as$V1

target_regs_with_as <- possible_post_modification[possible_post_modification$in_as_signal == TRUE,]
```

This Analyses showed that Usf2, upstream transcription factor 2, have a differential intron retention event and also a reporter regulator while it is not differentially expressed. Potentially, statistically significant expression of this particular intron region may affect the activity of Usf2 but does not effect the expression.

In literature, a prior study (https://pubmed.ncbi.nlm.nih.gov/12611894/) that Usf2 represses the induction of Carnitine Palmitoyltransferase I Beta, which is the first step of carnitine palmitoyltransferase system and generating ATP from fat in mitochondria. Also a previous study (https://www.nature.com/articles/s41467-021-25674-5) indicated that CLOCK, regulates ABCA1 expression using an indirect mechanism involving the transcription factor USF2. ABCA1 protein mediates the secretion of free cholesterol into apolipoprotein A-1 to form high-density lipoprotein, thereby playing a critical role in cholesterol homeostasis. Interestingly, a study (https://www.sciencedirect.com/science/article/pii/S2213231720309551) that shows a possible role of Usf2 as a tumor supressor also identified its novel role for mitochondrial function and energy homeostasis thereby linking USF2 to conditions such as insulin resistance, type-2 diabetes mellitus, and the metabolic syndrome.

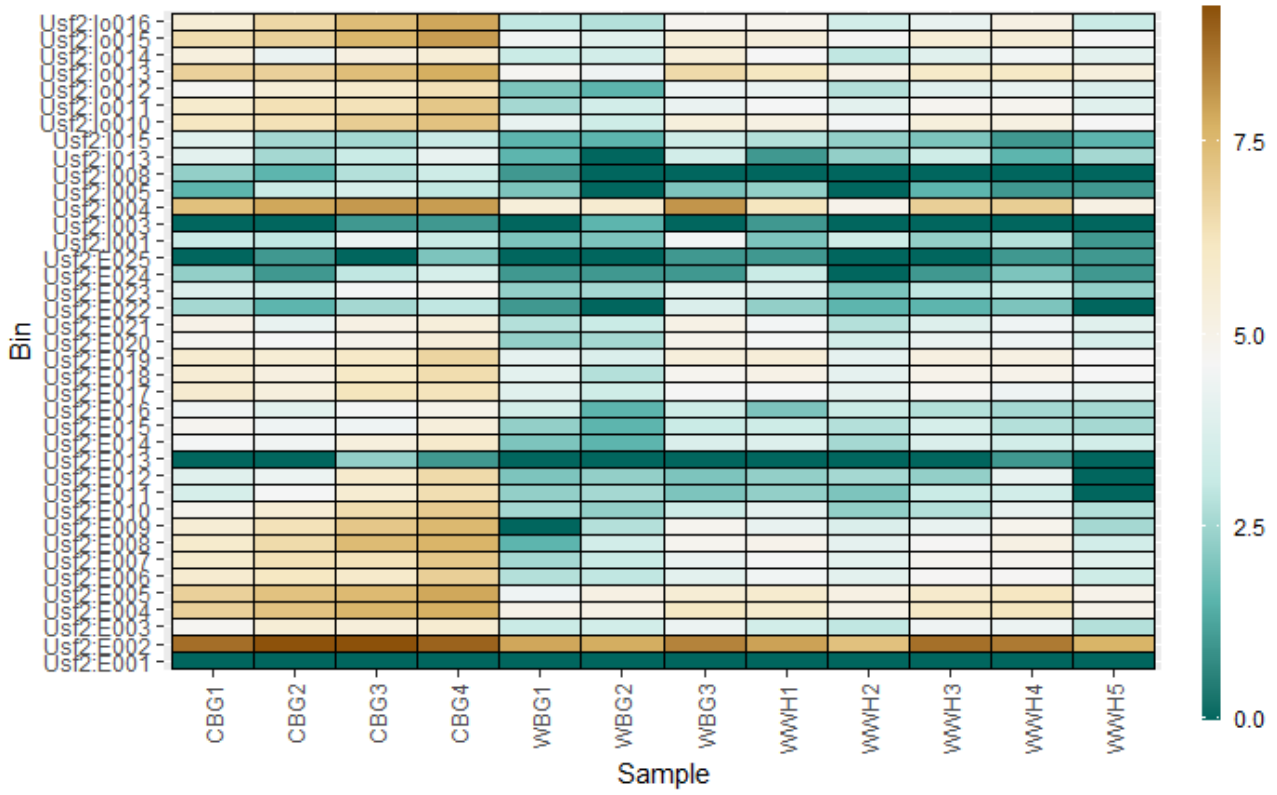
Hide

```
Usf2_bin_counts <- log2(as.matrix(countsb(gbcounts)[grep("ENSMUSG00000058239",x = rownames(BinCounts)),][,c(10:21)]) + 1)
colnames(Usf2_bin_counts) <- c(paste("CBG" , c(1:4) , sep = "") , paste("WBG" , c(1:3) , sep = "") , paste("WWH" , c(1:5) , sep = ""))

df3 <- melt(Usf2_bin_counts)
```

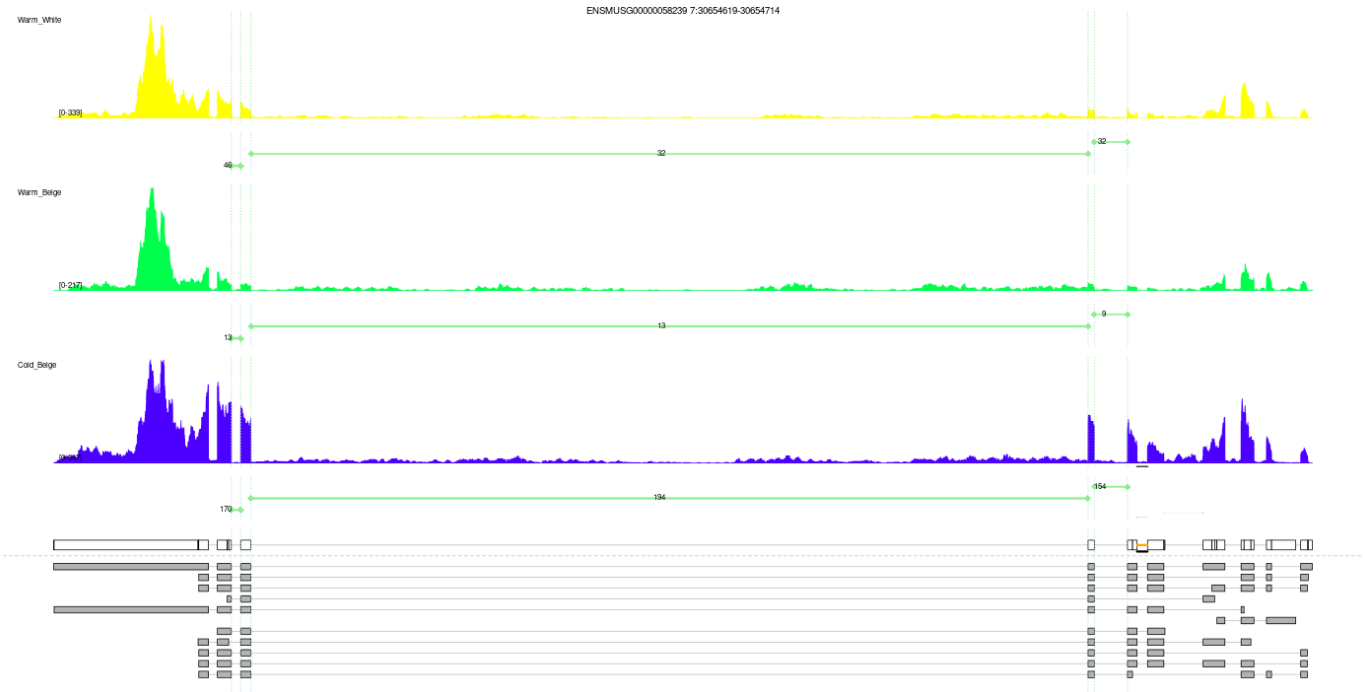
```
Warning in type.convert.default(X[[i]], ...) :
  'as.is' should be specified by the caller; using TRUE
Warning in type.convert.default(X[[i]], ...) :
  'as.is' should be specified by the caller; using TRUE
```

Hide

```
colnames(df3) <- c("Bin", "Sample", "value")
selected_usf2_genes <- str_split_fixed(df3$Bin , ":" , 2)
selected_usf2_genes[,1] <- vlookup(selected_usf2_genes[,1] ,dict = gs_heatdata , result_column = 1 ,
lookup_column = 2)
df3$Bin <- paste(selected_usf2_genes[,1] , selected_usf2_genes[,2] , sep = ":")


ggplot(df3, aes(x = Sample, y = Bin, fill = value)) + geom_tile(color = "black")  +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) + scale_fill_distiller(palette
= "BrBG") +
  guides(fill = guide_colourbar(barwidth = 0.5,
                                barheight = 20))
```

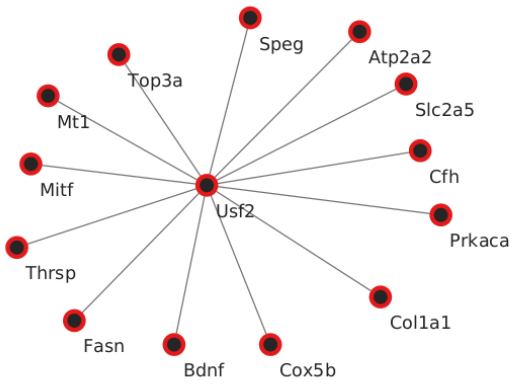


The IR Event is shown in Figure below



Usf2 IR Event.

The Usf2 and Its Connected Differentially Expressed Genes are shown in Network Figure below



Usf2 and Its Connected Differentially Expressed Genes.

Hide

```
usf2_connected_de_genes <- read.table("usf2_network.txt" , header = F)
```

Hide

```
usf2_associated_genes <- queryMany(as.character(usf2_connected_de_genes[,1]) , scopes = "symbol", spe
cies = "mouse")
```

```
Finished
Pass returnall=TRUE to return lists of duplicate or missing query terms.
```

Hide

```
usf2_associated_genes <- as.data.frame(cbind(usf2_associated_genes@listData$query,usf2_associated_gen
es@listData$name))
```

V1	V2
Mitf	melanogenesis associated transcription factor
Fasn	fatty acid synthase
Slc2a5	solute carrier family 2 (facilitated glucose transporter), member 5
Cox5b	cytochrome c oxidase subunit 5B
Bdnf	brain derived neurotrophic factor
Atp2a2	ATPase, Ca++ transporting, cardiac muscle, slow twitch 2
Prkaca	protein kinase, cAMP dependent, catalytic, alpha
Thrsp	thyroid hormone responsive
Speg	SPEG complex locus
Col1a1	collagen, type I, alpha 1
Cfh	complement component factor h
Top3a	topoisomerase (DNA) III alpha
Usf2	upstream transcription factor 2
Mt1	metallothionein 1

V1	V2
Ctcf	CCCTC-binding factor
Usf2	upstream transcription factor 2

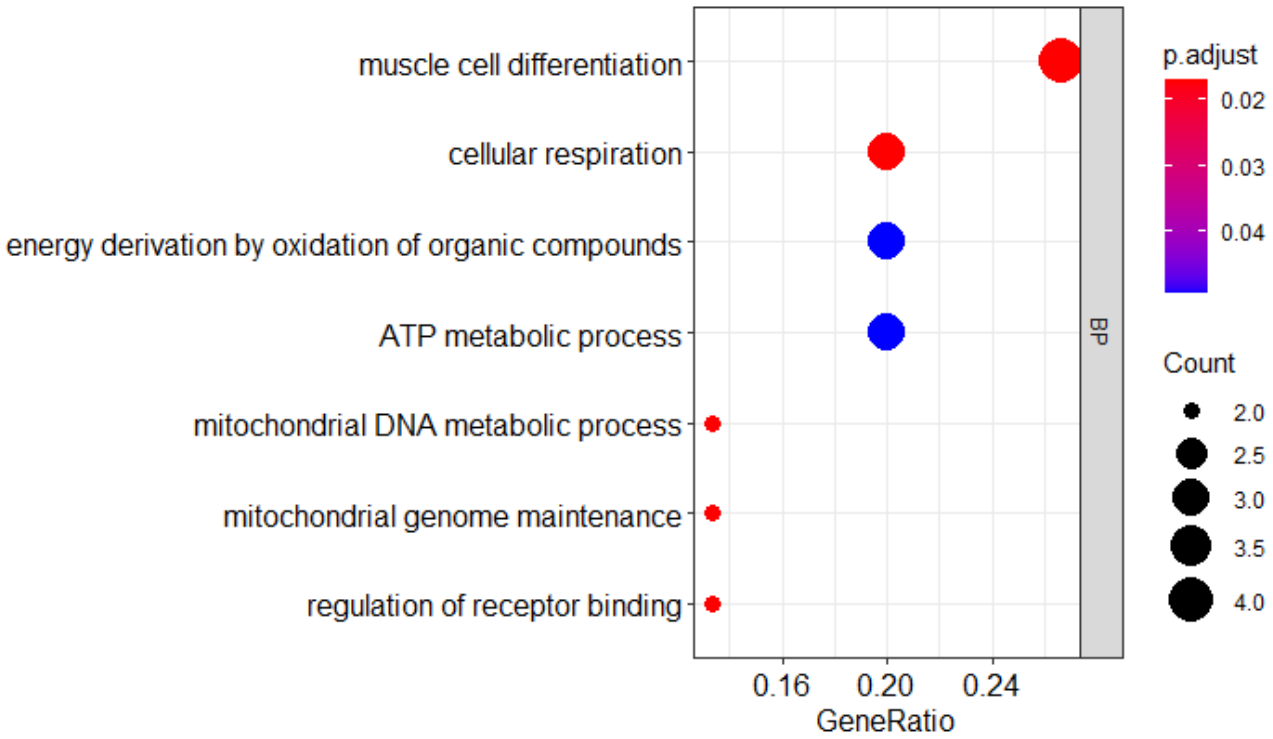
Hide

```
usf_associated_go <- enrichGO(as.character(usf2_connected_de_genes[,1]), OrgDb = "org.Mm.eg.db", keyT
ype = "SYMBOL" , ont = "all")
```

4.2.1 Dotplots of Differentially Expressed and Connected to Usf2 Genes.

Plots correspond to BP, CC and MF terms of GeneOntology respectively (If there is any term enriched)

4.2.1.1 GeneOntology



Hide

```
png(width = 1280 , height = 720 , file= "figures/Enrichment_Results_of_Ucp1_Correlated_Exons.png")
dotplot(go_res_corr_exon, title = "a) Ucp1 Correlated Exons" , split="ONTOLOGY" , showCategory = 7 ,
label_format = 70) + facet_grid(ONTOLOGY~., scale="free")
dev.off()
```

```
null device
          1
```

Hide

```
png(width = 1280 , height = 720 , file= "figures/Enrichment_Results_of_Ucp1_Correlated_Introns.png")
dotplot(go_res_corr_intron, title = "b) Ucp1 Correlated Introns" , split="ONTOLOGY" , showCategory =
7 , label_format = 70) + facet_grid(ONTOLOGY~., scale="free")
```

Hide

```
dev.off()
```

```
null device
          1
```