

Figure 4A-D & Sup. Fig. 4 A & D

RAC

05/01/2021

Script below was used to uniting bed file of mapped reads (cross-links) with bed file of all TUs exons

```
#!/bin/bash
#script for ubiting bg values with annotation file 200113
#all dir paths should contain / at the end.
#InputVariables

bedAnnotation=$1
inBedDir=$2
outDir=$3

echo "Merging bedgraph counts with $bedAnnotation"
echo "Using bedgraphs in $inBedDir"
echo "Saving outputs in $outDir"

#constructing output suffix for filename.
outFileName=$(basename $bedAnnotation | sed 's/\.bed//g')

#mk output dir
mkdir -p $outDir

#perform on each bed file (mapped tiCLIP cross-link sites)
for inBed in ${inBedDir}*.bed;
do
    #make variables
    ID=$(basename $inBed | sed 's/\.bed//g;s/-/_/g' )

    echo "starting analysis for $ID"

    bedtools intersect -loj -s \
    -a <( sort -k1,1 -k2,2n $inBed | sed 's/^chr//g' ) \
    -b <( sort -k1,1 -k2,2n $bedAnnotation | sed 's/^chr//g' ) \
    | awk '{OFS="\t"} $8 > -1' \
    | awk -v ID=$ID '{OFS="\t"}{print ID,$7,$8,$9,$10,$11,$12,$1,$2,$3,$4,$5,$6}' > ${outDir}${ID}"_"${ID}.bed

    echo "completed for $ID"
done

library(scales)
library(ggformula)
```

```
## Loading required package: ggplot2
## Loading required package: ggstance
##
## Attaching package: 'ggstance'
## The following objects are masked from 'package:ggplot2':
##
##     geom_errorbarh, GeomErrorbarh
## Loading required package: ggribes
##
## New to ggformula? Try the tutorials:
##   learnr::run_tutorial("introduction", package = "ggformula")
##   learnr::run_tutorial("refining", package = "ggformula")
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tibble)
library(tidyr)
library(ggplot2)
library(ggsignif)
library(ggpubr)
knitr::opts_chunk$set(tidy.opts=list(width.cutoff=60), tidy=TRUE)
```

load input files generated from script above annotation bed file, and expression vector

```
dfFilePath = "../data/ALYREF_5primepos_hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.exonNum
```

```
annoFilePath = "../data/hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.exonNumber.sizeRange."
```

```
EXPRESSION_VECTOR_FILEPATH = "../data/log2_mean_cov_RNAseq_TTseq.RData"
```

```
load(EXPRESSION_VECTOR_FILEPATH)
```

```
# wrangle expression vector
```

```
expression_vector <- left_join((as.data.frame(ctrl_RNAseq_expr) %>%
  add_rownames(var = "geneName")), (as.data.frame(ctrl_TTseq_expr) %>%
  add_rownames(var = "geneName"))) %>%
  mutate(ctrl_RNAseq_expr = case_when(ctrl_RNAseq_expr == 0 ~
    min(ctrl_RNAseq_expr[ctrl_RNAseq_expr > 0]), TRUE ~ ctrl_RNAseq_expr))
```

```
## Warning: `add_rownames()` was deprecated in dplyr 1.0.0.
```

```
## i Please use `tibble::rownames_to_column()` instead.
```

```
## Joining, by = "geneName"
```

```
#Functions used for processing input
```

#refined Function finds relative distance of xlinksite across feature. RNA split into 100 bins in total
#Also genes are annotated with mature and gene size.

```
process_distToTSS_1 <- function(dfFilePath, annotationFilePath){

#input file is a bed intersection between annotation file containing exons of genes
df<-read.csv(dfFilePath, sep = "\t", header =F)
col.names<-c("Sample",
             #annotationfile
             "chrAnno", "startAnno", "stopAnno", "IDAnno", "scoreAnno",
             "strandAnno",
             #bed file of cross-link mapped bam file
             "chrRead", "startRead", "stopRead", "IDRead", "scoreRead", "strandRead")

head(df)

print("wrangling data")

df1<-df %>%
  setNames(col.names) %>%
  select(-chrRead, -IDRead, -scoreRead, -scoreAnno) %>%
  separate(IDAnno, c("geneName", "biotype", "exonID", "totalExons", "exonSize", "cumSumExons", "distToTSS", "strandAnno"))
  select(-exonDesc)

#cumSumExons = distance
#distToTSS = the distance
col.numeric<- c("chrAnno", "startAnno", "stopAnno", "totalExons", "exonSize", "cumSumExons", "distToTSS", "strandAnno")
df1[col.numeric] <- sapply(df1[col.numeric], as.numeric)

print("processing rel distance from TSS")
#calculate position of cross-link within the mature RNA. i.e. distance of cross-link to 5' end of RNA exon
df2<-df1 %>%
  mutate(rel.pos = ifelse(strandAnno == "+", ( (startRead-startAnno) + (cumSumExons-exonSize) ),
                          ifelse(strandAnno == "-", ( (stopAnno-stopRead) + (cumSumExons-exonSize) ),
                          "no"))) %>%
  select(Sample, geneName, biotype, geneDesc, rel.pos)

print("loading annotation file")

annoDF<-read.csv(annoFilePath , sep = "\t", header =F)

print("making table with total sizes of RNAs")

totalSizes<-annoDF %>%
  setNames(c("chr", "start", "stop", "ID", "score", "strand")) %>%
  separate(ID, c("geneName", "biotype", "exonID", "totalExons", "exonSize", "cumSumExons", "distToTSS", "strand"))
  filter(exonID == totalExons) %>%
  mutate(matureRNA = as.numeric(cumSumExons),
         geneSize = as.numeric(ifelse(totalExons > 1, distToTSS, exonSize))) %>%
```

```

    select(geneName, biotype, matureRNA, geneSize)

df3<-df2 %>%
  left_join(totalSizes)

return(df3)
}

ScaledNormalise_2_3_RPM_Normalisation <- function(DF, MINSIZE,
  MAXSIZE, annoFilePath, RPMFactorsFilePath) {

  annoDF <- read.csv(annoFilePath, sep = "\t", header = F)

  totalSizes <- annoDF %>%
    setNames(c("chr", "start", "stop", "ID", "score", "strand")) %>%
    separate(ID, c("geneName", "biotype", "exonID", "totalExons",
      "exonSize", "cumSumExons", "distToTSS", "exonDesc",
      "geneDesc"), sep = "::") %>%
    filter(exonID == totalExons) %>%
    mutate(matureRNA = as.numeric(cumSumExons), geneSize = as.numeric(ifelse(totalExons >
      1, distToTSS, exonSize))) %>%
    select(geneName, biotype, matureRNA, geneSize)

  calculated_test <- relDist_ALYREF %>%
    left_join(totalSizes) %>%
    mutate(rel.pos = as.numeric(rel.pos), matureRNA = as.numeric(matureRNA)) %>%
    filter(matureRNA %in% c(MINSIZE:MAXSIZE)) %>%
    mutate(rel.pos.2 = rel.pos/matureRNA)

  # assigning bin numbers to the rel.pos.2

  ALL <- calculated_test %>%
    mutate(rel.pos.2 = ifelse(rel.pos.2 == 0, rel.pos.2 +
      1e-07, rel.pos.2))
  max(ALL$rel.pos.2)
  min(ALL$rel.pos.2)
  ALL$new_distBin <- cut(ALL$rel.pos.2, seq(0, 1, by = 0.01),
    labels = seq("1", "100", by = 1))

  combined <- ALL %>%
    separate(Sample, into = c("Protein", "Rep", "Timepoint",
      "readType"), sep = "_") %>%
    # filter(Timepoint == 'DMSO' &
    # grepl('coding/histone', biotype) &
    # !grepl('\\\\*/non', biotype)) %>%
    group_by(geneName, biotype, Protein, Timepoint, Rep, matureRNA,
      geneSize, new_distBin) %>%
    summarise(tally = n()) %>%
    # new steo to normalise to gene expression level
    left_join(expression_vector) %>%
    mutate(tally_n = (tally/ctrl_RNAseq_expr)/(matureRNA/100)) #step to normalise count to bin siz

  # mutate(tally_n = tally/(matureRNA/100)) #step to

```

```

# normalise count to bin size.

totalLibrarySizes <- read.table(RPMFactorsFilePath) %>%
  setNames(c("Sample", "RPMFactor")) %>%
  separate(Sample, into = c("Protein", "Rep", "Timepoint",
    "readType"), sep = "_") %>%
  mutate(Rep = as.character(Rep)) %>%
  mutate(RPMFactor = 1e+06/RPMFactor)

# this step taken from norm profile 3
combined_2 <- combined %>%
  left_join(totalLibrarySizes) %>%
  mutate(tally_n = tally_n * RPMFactor) %>%
  select(-RPMFactor) %>%
  group_by(geneName, biotype, Protein, Timepoint, Rep,
    matureRNA) %>%
  mutate(totals_unNorm = sum(tally), pct = tally_n/sum(tally_n) *
    100) %>%
  ungroup() %>%
  select(Protein, Timepoint, Rep, matureRNA, geneName,
    biotype, new_distBin, totals_unNorm, pct) %>%
  spread(new_distBin, pct, fill = 0) %>%
  gather("distBin", "value", c("1":"100")) %>%
  mutate(distBin = factor(distBin, levels = c(1:100)))

return(combined_2)
}

```

```

# calculate profiles
kMeansClustering_4 <- function(inputDF, CENTRES, REP, MINCOUNTS) {

  # filter for DMSO and protein Codingg
  DF <- inputDF %>%
    filter(Timepoint == "DMSO") %>%
    spread(distBin, value)

  # filter by rep and only use genes with a minumum of 40
  # counts per gene.
  DF1 <- DF %>%
    filter(Rep == REP & totals_unNorm >= MINCOUNTS) %>%
    column_to_rownames(var = "geneName") %>%
    select("1":"100")

  # kmeans clustering analysis (set seed for
  # reproducibility)
  set.seed(42)
  kClusters <- kmeans(DF1, centers = CENTRES)

  clusters <- as.data.frame(kClusters$cluster) %>%
    rownames_to_column(var = "geneName") %>%
    rename(clusterID = `kClusters$cluster`)

  DF_merge <- DF %>%
    right_join(clusters) %>%

```

```

    group_by(clusterID, Rep) %>%
    mutate(n = n()) %>%
    ungroup() %>%
    gather("distBin", "value", c("1":"100")) %>%
    mutate(distBin = factor(distBin, levels = c(1:100)))

    return(DF_merge)
}

dfFilePath = "../../../data/ALYREF_5primepos_hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.exonNum
annoFilePath = "../../../data/hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.exonNumber.sizeRange."

rRNAFactorsFilePath = "../../../data/rRNAFactor.tab"
RPMFactorsFilePath = "../../../data/RPM_factors.tab"

annoDF <- read.csv(annoFilePath, sep = "\t", header = F)

geneStructures <- annoDF %>%
  setNames(c("chr", "start", "stop", "ID", "score", "strand")) %>%
  separate(ID, c("geneName", "biotype", "exonID", "totalExons",
    "exonSize", "cumSumExons", "distToTSS", "exonDesc", "geneDesc"),
    sep = ":::") %>%
  separate(geneDesc, into = c("geneStructure"), sep = "-") %>%
  select(geneName, geneStructure) %>%
  unique()

## Warning: Expected 1 pieces. Additional pieces discarded in 117503 rows [1, 2, 3,
## 4, 5, 6, 7, 8, 9, 12, 13, 14, 15, 21, 22, 23, 24, 25, 26, 29, ...].

head(geneStructures)

##           geneName      geneStructure
## 1             PRKX  multiExonicGene
## 10          NA.v8937 singleExonicGene
## 11          NA.v8938 singleExonicGene
## 12 RP11-706015.1  multiExonicGene
## 16          NA.v9224 singleExonicGene
## 17          NA.v9223 singleExonicGene

totalSizes <- annoDF %>%
  setNames(c("chr", "start", "stop", "ID", "score", "strand")) %>%
  separate(ID, c("geneName", "biotype", "exonID", "totalExons",
    "exonSize", "cumSumExons", "distToTSS", "exonDesc", "geneDesc"),
    sep = ":::") %>%
  filter(exonID == totalExons) %>%
  mutate(matureRNA = as.numeric(cumSumExons), geneSize = as.numeric(ifelse(totalExons >
    1, distToTSS, exonSize))) %>%
  select(geneName, biotype, matureRNA, geneSize)

head(totalSizes)

##           geneName      biotype matureRNA geneSize
## 1             PRKX  protein_coding      1817  104704
## 2          NA.v8937         nHtH        454      454
## 3          NA.v8938         nTtT       1628     1628

```

```
## 4 RP11-706015.1      lincRNA      11264      25788
## 5      NA.v9224      intergenic      349      349
## 6      NA.v9223      intergenic      736      736
```

```
relDist_ALYREF <- process_distToTSS_1(dfFilePath, annoFilePath)
```

```
## [1] "wrangling data"
## [1] "processing rel distance from TSS"
## [1] "loading annotation file"
## [1] "making table with total sizes of RNAs"

## Joining, by = c("geneName", "biotype")
```

```
head(relDist_ALYREF)
```

```
##           Sample geneName      biotype      geneDesc
## 1 ALYREF_1_DMSO_5primepos NA.v1000 protein_coding multiExonicGene-lastExon
## 2 ALYREF_1_DMSO_5primepos NA.v1000 protein_coding multiExonicGene-lastExon
## 3 ALYREF_1_DMSO_5primepos NA.v1000 protein_coding multiExonicGene-lastExon
## 4 ALYREF_1_DMSO_5primepos NA.v1000 protein_coding multiExonicGene-lastExon
## 5 ALYREF_1_DMSO_5primepos NA.v1000 protein_coding multiExonicGene-lastExon
## 6 ALYREF_1_DMSO_5primepos NA.v1000 protein_coding multiExonicGene-lastExon
##   rel.pos matureRNA geneSize
## 1    1936      1990    2289
## 2    1936      1990    2289
## 3    1916      1990    2289
## 4    1911      1990    2289
## 5    1893      1990    2289
## 6    1889      1990    2289
```

```
ALYREF_100bins_scaled <- ScaledNormalise_2_3_RPM_Normalisation(relDist_ALYREF,
  200, 1e+05, annoFilePath, RPMFactorsFilePath)
```

```
## Joining, by = c("geneName", "biotype", "matureRNA", "geneSize")
## `summarise()` has grouped output by 'geneName', 'biotype', 'Protein',
## 'Timepoint', 'Rep', 'matureRNA', 'geneSize'. You can override using the
## `.groups` argument.
## Joining, by = "geneName"
## Joining, by = c("Protein", "Timepoint", "Rep")
```

```
head(ALYREF_100bins_scaled)
```

```
## # A tibble: 6 x 9
##   Protein Timepoint Rep   matureRNA geneName      biotype total~1 distBin value
##   <chr>    <chr>    <chr>    <dbl> <chr>      <chr>      <int> <fct>    <dbl>
## 1 ALYREF  DMSO      1        1406 5S_rRNA,DTNBP1 rRNA          6 1        0
## 2 ALYREF  DMSO      2        1406 5S_rRNA,DTNBP1 rRNA          4 1        0
## 3 ALYREF  PBSDRB     1        1406 5S_rRNA,DTNBP1 rRNA          2 1        0
## 4 ALYREF  PBSDRB     2        1406 5S_rRNA,DTNBP1 rRNA          2 1        0
## 5 ALYREF  t00        1        1406 5S_rRNA,DTNBP1 rRNA          2 1        0
## 6 ALYREF  t05        2        1406 5S_rRNA,DTNBP1 rRNA          3 1        0
## # ... with abbreviated variable name 1: totals_unNorm
```

```
ALYREF_100bins_scaled_clusters_1 <- kMeansClustering_4(ALYREF_100bins_scaled,
  2, 1, 20)
```

```
## Joining, by = "geneName"
```

```
head(ALYREF_100bins_scaled_clusters_1)
```

```
## # A tibble: 6 x 11
##   Protein Timepoint Rep   mature~1 geneN~2 biotype total~3 clust~4     n distBin
##   <chr>    <chr>    <chr>    <dbl> <chr>    <chr>    <int>    <int> <int> <fct>
## 1 ALYREF  DMSO      1       3254 AACS     protei~    20      2  2794 1
## 2 ALYREF  DMSO      2       3254 AACS     protei~     9      2  2794 1
## 3 ALYREF  DMSO      1      50238 AAK1     protei~    38      1  1361 1
## 4 ALYREF  DMSO      2      50238 AAK1     protei~     9      1  1359 1
## 5 ALYREF  DMSO      1       3382 AARS     protei~    62      2  2794 1
## 6 ALYREF  DMSO      2       3382 AARS     protei~    27      2  2794 1
## # ... with 1 more variable: value <dbl>, and abbreviated variable names
## #   1: matureRNA, 2: geneName, 3: totals_unNorm, 4: clusterID
```

```
ALYREF_100bins_scaled_clusters_2 <- kMeansClustering_4(ALYREF_100bins_scaled,
  2, 2, 20)
```

```
## Joining, by = "geneName"
```

```
head(ALYREF_100bins_scaled_clusters_2)
```

```
## # A tibble: 6 x 11
##   Protein Timepoint Rep   mature~1 geneN~2 biotype total~3 clust~4     n distBin
##   <chr>    <chr>    <chr>    <dbl> <chr>    <chr>    <int>    <int> <int> <fct>
## 1 ALYREF  DMSO      1       3382 AARS     protei~    62      2   865 1
## 2 ALYREF  DMSO      2       3382 AARS     protei~    27      2   865 1
## 3 ALYREF  DMSO      1      8122 ABCA2     protei~   115      2   865 1
## 4 ALYREF  DMSO      2      8122 ABCA2     protei~    48      2   865 1
## 5 ALYREF  DMSO      1     20558 ABCA5     protei~    77      2   865 1
## 6 ALYREF  DMSO      2     20558 ABCA5     protei~    25      2   865 1
## # ... with 1 more variable: value <dbl>, and abbreviated variable names
## #   1: matureRNA, 2: geneName, 3: totals_unNorm, 4: clusterID
```

Join clusters

```
clusters <- full_join((ALYREF_100bins_scaled_clusters_1 %>%
  select(Rep, geneName, clusterID) %>%
  unique() %>%
  mutate(Rep1_clusterID = paste0(clusterID)) %>%
  select(-clusterID)), (ALYREF_100bins_scaled_clusters_2 %>%
  select(Rep, geneName, clusterID) %>%
  unique() %>%
  mutate(Rep2_clusterID = paste0(clusterID)) %>%
  select(-clusterID)))
```

```
## Joining, by = c("Rep", "geneName")
```

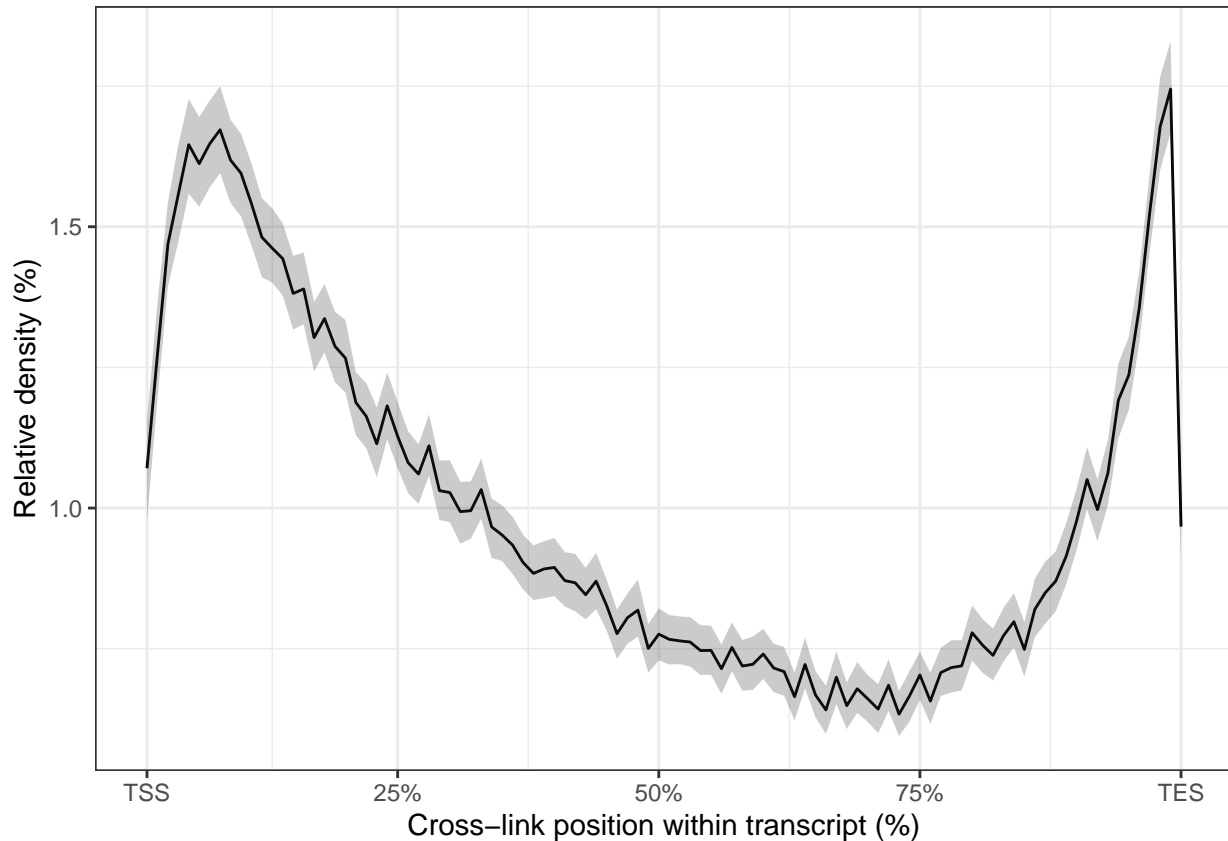
```
clusters[is.na(clusters)] <- "0"
```

#Figure 4 A Plot profile of ALYREF-DMSO cross-links across exonic portions of TUs

```
ALYREF_100bins_scaled %>%
  filter(Timepoint == "DMSO" & grepl("coding", biotype) & !grepl("\\*|non",
    biotype) & totals_unNorm >= 20) %>%
  spread(distBin, value) %>%
  gather("distBin", "value", 8:107) %>%
  ggplot(aes(x = as.numeric(distBin), y = value)) + stat_summary(fun = mean,
    geom = "line", size = 0.5) + stat_summary(fun.data = "mean_cl_boot",
```



```
geom = "ribbon", alpha = 0.25, col = NA) + theme_bw() + xlab("Cross-link position within transcript")
scale_x_continuous(breaks = c(1, 25, 50, 75, 100), labels = c("TSS",
  "25%", "50%", "75%", "TES")) + theme(legend.position = "none") +
ylab("Relative density (%)")
```



#Venn diagram for Supplementary Figure 4 a. Intersect genes identified in kmeans clustering from ALYREF-1-DMSO and ALYREF-2-DMSO

```
Rep1_cluster1 <- ALYREF_100bins_scaled_clusters_1 %>%
  filter(clusterID == 1 & Rep == "1") %>%
  select(geneName, clusterID, Rep) %>%
  unique()

Rep1_cluster2 <- ALYREF_100bins_scaled_clusters_1 %>%
  filter(clusterID == 2 & Rep == "1") %>%
  select(geneName, clusterID, Rep) %>%
  unique()

Rep2_cluster1 <- ALYREF_100bins_scaled_clusters_2 %>%
  filter(clusterID == 1 & Rep == "2") %>%
  select(geneName, clusterID, Rep) %>%
  unique()

Rep2_cluster2 <- ALYREF_100bins_scaled_clusters_2 %>%
  filter(clusterID == 2 & Rep == "2") %>%
  select(geneName, clusterID, Rep) %>%
  unique()
```

```

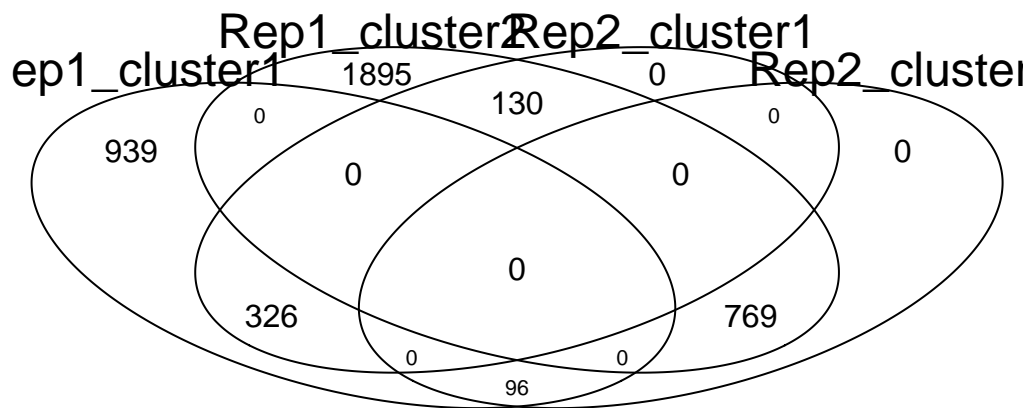
# make venn diagram of cluster id overlaps
library(gplots)

##
## Attaching package: 'gplots'

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

# df('resultsFigs/210119/ven.pdf')
venn(list(Rep1_cluster1 = (Rep1_cluster1 %>%
  select(geneName) %>%
  as.list() %>%
  unlist()), Rep1_cluster2 = (Rep1_cluster2 %>%
  select(geneName) %>%
  as.list() %>%
  unlist()), Rep2_cluster1 = (Rep2_cluster1 %>%
  select(geneName) %>%
  as.list() %>%
  unlist()), Rep2_cluster2 = (Rep2_cluster2 %>%
  select(geneName) %>%
  as.list() %>%
  unlist()))))

```



```

# dev.off()

# Rep1_cluster1 %>% select(geneName) %>% as.list() %>%
# unlist()

# intersect the 2 different genes groups identified
clus_1 <- intersect(Rep1_cluster1$geneName, Rep2_cluster1$geneName) %>%
  as.data.frame() %>%
  mutate(cluster = "1")
clus_2 <- intersect(Rep1_cluster2$geneName, Rep2_cluster2$geneName) %>%
  as.data.frame() %>%
  mutate(cluster = "2")

# combine to one dataframe
clus_all <- rbind(clus_1, clus_2) %>%
  setNames(c("geneName", "clusterID")) %>%

```

```
mutate(geneName = as.character(geneName))
```

```
clust_all
```

##	geneName	clusterID
## 1	ABCC1	1
## 2	ABCC4	1
## 3	AC093838.4	1
## 4	ACACA	1
## 5	ACEA_U3,SNORD3B-2	1
## 6	ADARB2,RP11-398B16.2	1
## 7	AGAP10,BMS1P2,RP11-144G6.12	1
## 8	AMD1	1
## 9	ANAPC1	1
## 10	ANKRD10.v2	1
## 11	ANKRD27	1
## 12	ANKRD32	1
## 13	ANKRD36	1
## 14	ANTXR1	1
## 15	AP1G1	1
## 16	ARHGAP11B	1
## 17	ARID2	1
## 18	ARID5B	1
## 19	ASAP1	1
## 20	ASCC3	1
## 21	ATAD1	1
## 22	ATP2B1	1
## 23	ATP2C1	1
## 24	ATRX	1
## 25	ATXN2	1
## 26	AZIN1	1
## 27	BDP1	1
## 28	BNIP2	1
## 29	BRD4	1
## 30	BRIP1	1
## 31	BRWD1	1
## 32	C5orf42	1
## 33	CCAT1	1
## 34	CCDC88A	1
## 35	CCNL1	1
## 36	CCNT1	1
## 37	CD44,RP1-68D18.2	1
## 38	CDC42BPA	1
## 39	CDK12	1
## 40	CEP72	1
## 41	CHCHD3	1
## 42	CHD9,RP11-295M3.2	1
## 43	CIT	1
## 44	CKAP5	1
## 45	CLASP1	1
## 46	CLEC16A	1
## 47	CLIP1	1
## 48	CLTCL1	1
## 49	COG5	1

## 50	CPS1	1
## 51	CTC-338M12.2	1
## 52	CTD-2340D6.1,VPS13B	1
## 53	CTNNA1	1
## 54	CTPS1	1
## 55	CUX1	1
## 56	DDX5	1
## 57	DENND4A,RP11-16E23.3	1
## 58	DEPDC1	1
## 59	DIAPH3	1
## 60	DMXL1	1
## 61	DNM1L	1
## 62	DOCK1	1
## 63	DOCK7	1
## 64	DOT1L	1
## 65	DPP8	1
## 66	DPY19L4	1
## 67	DYNC2H1	1
## 68	EEF1A1	1
## 69	EIF3A	1
## 70	EIF3E	1
## 71	EIF4EBP3,ANKHD1	1
## 72	ELK4	1
## 73	EP400	1
## 74	EPT1	1
## 75	ERC1,RP5-951N9.2	1
## 76	ERCC6-PGBD3	1
## 77	EXOSC2	1
## 78	EYA4	1
## 79	FAM13B	1
## 80	FASTKD2	1
## 81	FBXW2	1
## 82	FNDC3B	1
## 83	FOXJ3	1
## 84	FTX	1
## 85	FUBP1	1
## 86	FXR1	1
## 87	G3BP1	1
## 88	GAPVD1	1
## 89	GART	1
## 90	GBF1	1
## 91	GDA	1
## 92	GFM1	1
## 93	GMPS	1
## 94	GNL3L	1
## 95	GPC5	1
## 96	GPR126	1
## 97	GSK3B	1
## 98	GTF2I	1
## 99	GTF3C1	1
## 100	HELLS	1
## 101	HELZ	1
## 102	HERC2	1
## 103	HIATL2	1

## 104	HIST1H3G	1
## 105	HLTF	1
## 106	hsa-mir-7706,RP11-815J21.1,AKAP13	1
## 107	HSPA4	1
## 108	IDE	1
## 109	IGF2BP3	1
## 110	ILF3	1
## 111	INADL	1
## 112	INTS7	1
## 113	IP05	1
## 114	IP07	1
## 115	IREB2	1
## 116	ITPR2	1
## 117	KDM5A	1
## 118	KIAA0825	1
## 119	KIAA1033	1
## 120	KIAA1109	1
## 121	KIAA1549	1
## 122	KIAA1841	1
## 123	KIAA1958	1
## 124	KIF20B	1
## 125	KIF2C	1
## 126	KMT2C	1
## 127	KYNU	1
## 128	LARP4	1
## 129	LIMA1,RP3-405J10.2,RP3-405J10.3	1
## 130	LIMCH1	1
## 131	LIMD1	1
## 132	LPGAT1	1
## 133	LPP	1
## 134	LRBA	1
## 135	LRP6	1
## 136	LUC7L3	1
## 137	MAMDC2	1
## 138	MAP4	1
## 139	MARCH6	1
## 140	MAT2A	1
## 141	MBNL2	1
## 142	MDN1	1
## 143	METTL16	1
## 144	MGA	1
## 145	MICAL3,XXbac-B476C20.14,XXbac-B476C20.13,XXbac-B461K10.4	1
## 146	MID1	1
## 147	MIR4444-2	1
## 148	MIR5006,VWA8	1
## 149	MIR6125,USP15	1
## 150	MIR6861,HECTD4	1
## 151	MIR8072,SBN01	1
## 152	MKLN1	1
## 153	MMS22L	1
## 154	MPHOSPH9	1
## 155	MRE11A,RP11-685N10.1	1
## 156	MSI2	1
## 157	MTAP	1

## 158	MTF2	1
## 159	MTPAP	1
## 160	MYBBP1A	1
## 161	MYOF	1
## 162	NA.v3445	1
## 163	NA.v3852	1
## 164	NA.v7958	1
## 165	NA.v8004	1
## 166	NAT10.v1	1
## 167	NBAS	1
## 168	NCBP1	1
## 169	NCOA3	1
## 170	NFIA	1
## 171	NFIB	1
## 172	NFX1	1
## 173	NHLRC2	1
## 174	NLRC5	1
## 175	NPM1	1
## 176	NR3C1	1
## 177	NSD1	1
## 178	NUFIP2	1
## 179	NUMA1	1
## 180	NUP188	1
## 181	NUP214	1
## 182	NUP98	1
## 183	OGDH	1
## 184	OPN3	1
## 185	OSMR	1
## 186	OXCT1	1
## 187	PALLD	1
## 188	PANK3,SLC2A3P1	1
## 189	PAPOLA	1
## 190	PBRM1	1
## 191	PDE3A	1
## 192	PDPR	1
## 193	PDXK	1
## 194	PHC3	1
## 195	PHIP	1
## 196	PI4KA	1
## 197	POLR1A	1
## 198	PPIL2	1
## 199	PPP6R2	1
## 200	PPP6R3	1
## 201	PRKD3	1
## 202	PRKDC	1
## 203	PRMT3	1
## 204	PRRC2B	1
## 205	PSMD1	1
## 206	PSPC1	1
## 207	PTBP2	1
## 208	PTPLB	1
## 209	RACGAP1	1
## 210	RALGAPB	1
## 211	RAPGEF6	1

## 212	RASAL2, Metazoa_SRP	1
## 213	RBL1	1
## 214	RBM25	1
## 215	RBM27	1
## 216	RBMX	1
## 217	RC3H1	1
## 218	RIMKLB	1
## 219	RLIM	1
## 220	RNF169	1
## 221	RNF216	1
## 222	RNU1-1	1
## 223	RNU5E-1	1
## 224	RNU5F-1	1
## 225	RNU6-29P, AC115617.2	1
## 226	RNU6-8	1
## 227	RP1-102G20.2, RABGAP1L	1
## 228	RP11-11N7.5, HNRNPU-AS1	1
## 229	RP11-14206.1, NF1	1
## 230	RP11-274B21.1	1
## 231	RP11-354B3.1	1
## 232	RP11-61102.3, MDM2, RP11-61102.1	1
## 233	RPTOR	1
## 234	RSL1D1	1
## 235	RUNX1	1
## 236	S100PBP	1
## 237	SACS	1
## 238	SBF2	1
## 239	SCAF11	1
## 240	SCD	1
## 241	SDHA	1
## 242	SEC24B	1
## 243	SEC31A	1
## 244	SEC61A1	1
## 245	SEMA3C	1
## 246	SENP5, AC127904.2	1
## 247	SERBP1	1
## 248	SETDB1	1
## 249	SETX	1
## 250	SFI1	1
## 251	SFPQ	1
## 252	SLC1A3	1
## 253	SLC38A1	1
## 254	SLC7A5P2	1
## 255	SMAD5	1
## 256	SMARCC1, RP11-717D12.1	1
## 257	SMEK2	1
## 258	SMPD4	1
## 259	SNORD118	1
## 260	SNORD13	1
## 261	SP1	1
## 262	SPATA5	1
## 263	SPECC1	1
## 264	SPIDR	1
## 265	SRGAP1, RP11-274J7.2	1

## 266	SRP72	1
## 267	SRSF1	1
## 268	SRSF3	1
## 269	SRSF6	1
## 270	SSR3	1
## 271	STAG1	1
## 272	SUGP2	1
## 273	SYNCRIP,RP11-321N4.5	1
## 274	TAF1D	1
## 275	TANC2	1
## 276	TANGO6,RP11-521L9.2	1
## 277	TARDBP	1
## 278	TAS2R43,RP11-785H5.2,PRR4,TAS2R30	1
## 279	THADA	1
## 280	TIMM10B	1
## 281	TIMM23B,LINC00843	1
## 282	TMEM131	1
## 283	TMEM194A	1
## 284	TMOD3	1
## 285	TNPO1	1
## 286	TNPO2	1
## 287	TRAPPC9	1
## 288	TRMT1	1
## 289	TRPM7	1
## 290	TRPS1	1
## 291	TSC1	1
## 292	TUBGCP4	1
## 293	TUBGCP5	1
## 294	U1	1
## 295	U12,RNU12	1
## 296	UBN2	1
## 297	UBR1	1
## 298	UBR2	1
## 299	UGGT2	1
## 300	UHMK1	1
## 301	ULK4	1
## 302	URB1	1
## 303	USP25	1
## 304	USP33	1
## 305	UTP20	1
## 306	UVRAG,RP11-263C24.1	1
## 307	VPS13C	1
## 308	VPS13D	1
## 309	WDR4	1
## 310	WHSC1	1
## 311	WHSC1L1	1
## 312	WRN	1
## 313	XRN1	1
## 314	Y_RNA.v174	1
## 315	YOD1	1
## 316	ZFX	1
## 317	ZMYND8	1
## 318	ZNF121	1
## 319	ZNF148	1

## 320	ZNF207	1
## 321	ZNF562	1
## 322	ZNF587	1
## 323	ZNF638	1
## 324	ZNF644	1
## 325	ZNF800	1
## 326	ZYG11A	1
## 327	AARS	2
## 328	ABCA2	2
## 329	ABCA5	2
## 330	ABCC2	2
## 331	ABL2	2
## 332	AC006014.7	2
## 333	AC006042.8	2
## 334	AC006483.1,ACTB	2
## 335	AC139149.1,ACTG1	2
## 336	ACAD9	2
## 337	ACEA_U3,SNORD3A	2
## 338	ACEA_U3,SNORD3C	2
## 339	ACIN1	2
## 340	ACLY	2
## 341	ACOT9	2
## 342	ACTN4	2
## 343	ADARB1	2
## 344	ADNP	2
## 345	AFF4	2
## 346	AFG3L1P	2
## 347	AGL	2
## 348	AGRN,RP11-5407.14	2
## 349	AHCTF1	2
## 350	AHNAK	2
## 351	AKAP9	2
## 352	AL161626.1	2
## 353	AL442127.2	2
## 354	ANAPC5	2
## 355	ANKFY1	2
## 356	ANKLE2	2
## 357	ANKRD17	2
## 358	ANKRD28	2
## 359	ANLN	2
## 360	AP001469.9,MCM3AP	2
## 361	APC	2
## 362	APOPT1,KLC1	2
## 363	AQR	2
## 364	ARFGEF2	2
## 365	ARHGAP11A	2
## 366	ARHGAP21	2
## 367	ARHGAP29	2
## 368	ARHGAP5	2
## 369	ARID4B	2
## 370	ASPH	2
## 371	ASPM	2
## 372	ASXL1	2
## 373	ATAD2	2

## 374	ATAD5	2
## 375	ATL2	2
## 376	ATM	2
## 377	ATN1	2
## 378	ATP11A	2
## 379	ATP11C	2
## 380	ATP13A3	2
## 381	ATP1B3	2
## 382	ATP2A2	2
## 383	ATP5B	2
## 384	ATR,RP11-383G6.3	2
## 385	ATXN10	2
## 386	ATXN2L	2
## 387	B2M	2
## 388	B4GALT1	2
## 389	BASP1	2
## 390	BAZ1B	2
## 391	BIRC6	2
## 392	BMI1	2
## 393	BPTF	2
## 394	BRCA2	2
## 395	BRD2	2
## 396	BRD8	2
## 397	BRD9	2
## 398	BTBD3	2
## 399	BUB1	2
## 400	BUB3	2
## 401	C16orf72	2
## 402	C20orf24	2
## 403	C4BPA	2
## 404	C4BPB	2
## 405	C5orf24	2
## 406	C6orf100	2
## 407	C6orf48	2
## 408	C6orf62	2
## 409	C7orf55-LUC7L2	2
## 410	CACNB2	2
## 411	CAD	2
## 412	CALM2	2
## 413	CALR	2
## 414	CAMSAP2	2
## 415	CAND1	2
## 416	CANX	2
## 417	CAPN2	2
## 418	CAPN7	2
## 419	CAPRIN1	2
## 420	CAPZA1	2
## 421	CARS2	2
## 422	CASP2	2
## 423	CBX3	2
## 424	CCAR1	2
## 425	CCDC14	2
## 426	CCDC57	2
## 427	CCT2	2

## 428	CCT3	2
## 429	CCT4	2
## 430	CCT5	2
## 431	CCT6A	2
## 432	CCT8,AF129075.5	2
## 433	CD46	2
## 434	CD97	2
## 435	CDC16	2
## 436	CDC42BPB	2
## 437	CDCA7	2
## 438	CDK1	2
## 439	CDK5RAP2	2
## 440	CELF1,RP11-750H9.7	2
## 441	CEP152,RP11-227D13.4	2
## 442	CEP192	2
## 443	CHD1	2
## 444	CHD2	2
## 445	CHD3	2
## 446	CHD4	2
## 447	CLINT1	2
## 448	CLK1	2
## 449	CLPTM1L	2
## 450	CLSTN3	2
## 451	CLUH	2
## 452	CNBP	2
## 453	CNOT10	2
## 454	COL12A1	2
## 455	COL1A1	2
## 456	COL4A5	2
## 457	COPA	2
## 458	COPB1	2
## 459	COPB2	2
## 460	CPSF1,MIR939	2
## 461	CREBBP	2
## 462	CREBZF	2
## 463	CRIM1	2
## 464	CS,RP11-977G19.10	2
## 465	CSAD	2
## 466	CSDE1	2
## 467	CSNK1D	2
## 468	CSPP1	2
## 469	CTB-89H12.4,CSNK1A1	2
## 470	CTD-2006C1.12,ZNF700	2
## 471	CTD-2047H16.3,CTD-2047H16.2,RNF213	2
## 472	CTNNAL1	2
## 473	CTNNB1	2
## 474	CUL1	2
## 475	CUL4A	2
## 476	CUL4B	2
## 477	CYR61	2
## 478	DAB2	2
## 479	DAZAP1	2
## 480	DCAF6	2
## 481	DDX10	2

## 482	DDX12P	2
## 483	DDX17	2
## 484	DDX21	2
## 485	DDX39B,AL662801.1	2
## 486	DDX3X	2
## 487	DDX42	2
## 488	DDX46	2
## 489	DDX47	2
## 490	DDX56	2
## 491	DDX60L	2
## 492	DENND4C	2
## 493	DGKD	2
## 494	DGKH	2
## 495	DHX15	2
## 496	DHX30,uc_338	2
## 497	DHX36	2
## 498	DICER1	2
## 499	DIO2	2
## 500	DIP2B	2
## 501	DIS3L	2
## 502	DKC1	2
## 503	DLG1	2
## 504	DMXL2	2
## 505	DNAJC13	2
## 506	DNAJC2	2
## 507	DNMT1	2
## 508	DROSHA	2
## 509	DSP	2
## 510	DUS1L	2
## 511	DUSP1	2
## 512	DYNC1H1	2
## 513	DYNC1LI2	2
## 514	DYRK1A	2
## 515	E2F3	2
## 516	E2F4	2
## 517	ECT2	2
## 518	EDRF1	2
## 519	EEF2	2
## 520	EIF2S3	2
## 521	EIF3B	2
## 522	EIF4A1	2
## 523	EIF4A2	2
## 524	EIF4G1	2
## 525	EIF4G2	2
## 526	ELAC2	2
## 527	ENO1	2
## 528	ENOSF1	2
## 529	EP300,MIR1281	2
## 530	EPPK1	2
## 531	ERBB2IP	2
## 532	ESPL1	2
## 533	ETF1	2
## 534	EWSR1	2
## 535	FAF1	2

## 536	FAM120A	2
## 537	FAM122B	2
## 538	FAM160B1	2
## 539	FAM208B	2
## 540	FAM46A	2
## 541	FANCI	2
## 542	FASN	2
## 543	FAT1	2
## 544	FBRSL1	2
## 545	FBX038	2
## 546	FBX09	2
## 547	FLNA	2
## 548	FMR1	2
## 549	FNBP1L	2
## 550	FNBP4	2
## 551	FNDC3A	2
## 552	FUBP3	2
## 553	FUS	2
## 554	GABRE	2
## 555	GALNT2	2
## 556	GANAB	2
## 557	GAPDH	2
## 558	GAS5	2
## 559	GBA2	2
## 560	GDI2	2
## 561	GEN1	2
## 562	GIGYF1	2
## 563	GK5	2
## 564	GLG1.v2	2
## 565	GLYR1	2
## 566	GNAS	2
## 567	GOLGA3	2
## 568	GOLGA8A	2
## 569	GOLGA8B	2
## 570	GPBP1	2
## 571	GPBP1L1	2
## 572	GPI	2
## 573	GPR125	2
## 574	GSPT1	2
## 575	GTPBP4	2
## 576	H2AFY	2
## 577	HCFC1	2
## 578	HDLBP	2
## 579	HEG1	2
## 580	HERC2P2	2
## 581	HERC4	2
## 582	HEXIM1	2
## 583	HGS	2
## 584	HIAT1	2
## 585	HIF1A	2
## 586	HIPK1	2
## 587	HIPK3	2
## 588	HIRA	2
## 589	HMGCR	2

## 590	HMGXB3	2
## 591	HNRNPA0	2
## 592	HNRNPA2B1	2
## 593	HNRNPAB	2
## 594	HNRNPD	2
## 595	HNRNPDL	2
## 596	HNRNPH1	2
## 597	HNRNPH3	2
## 598	HNRNPK	2
## 599	HNRNPL	2
## 600	HNRNPM	2
## 601	HNRNPR	2
## 602	HNRNPU,RP11-11N7.5	2
## 603	HP1BP3	2
## 604	hsa-mir-3180-4,PKD1P6,MIR6511B1	2
## 605	hsa-mir-6724-1	2
## 606	HSP90AA1	2
## 607	HSP90AB1	2
## 608	HSP90AB3P	2
## 609	HSP90B1	2
## 610	HSPA8	2
## 611	HSPD1	2
## 612	HUWE1	2
## 613	IARS	2
## 614	IBTK	2
## 615	ICE1	2
## 616	IKBKAP	2
## 617	INCENP	2
## 618	INO80	2
## 619	INPPL1	2
## 620	INTS10	2
## 621	IP09	2
## 622	ITGB1	2
## 623	ITPR1	2
## 624	ITPR3	2
## 625	JAK1	2
## 626	JMJD1C	2
## 627	KANSL1	2
## 628	KANSL2	2
## 629	KAT2A	2
## 630	KHSRP,MIR3940	2
## 631	KIAA0020	2
## 632	KIAA0100	2
## 633	KIAA0368	2
## 634	KIAA1468	2
## 635	KIAA1524	2
## 636	KIAA1731	2
## 637	KIF14	2
## 638	KIF18B	2
## 639	KIF1B	2
## 640	KIF23	2
## 641	KIF5B	2
## 642	KLHL5	2
## 643	KMT2A	2

## 644	KMT2B	2
## 645	KMT2D	2
## 646	KNTC1	2
## 647	KRIT1,AC000120.7	2
## 648	KRT7	2
## 649	KTN1	2
## 650	LAMA1	2
## 651	LAMA5	2
## 652	LAMC1	2
## 653	LARP1	2
## 654	LARS	2
## 655	LBR	2
## 656	LDHA	2
## 657	LDHB	2
## 658	LGR4	2
## 659	LIFR	2
## 660	LINC00324	2
## 661	LINC00472	2
## 662	LINC00657	2
## 663	LINC00969	2
## 664	LMAN1	2
## 665	LMO4	2
## 666	LMO7	2
## 667	LPCAT1	2
## 668	LRIG2	2
## 669	LRP8,RP4-784A16.3	2
## 670	LRPPRC	2
## 671	LSMEM1,IFRD1	2
## 672	MACF1	2
## 673	MADD	2
## 674	MAFK	2
## 675	MALAT1	2
## 676	MAN2A1	2
## 677	MAP3K2	2
## 678	MAP4K5	2
## 679	MAPK14	2
## 680	MARC1	2
## 681	MARK3	2
## 682	MATR3	2
## 683	MBNL3	2
## 684	MBTD1	2
## 685	MBTPS1	2
## 686	MCM4	2
## 687	MCM7	2
## 688	MCMBP	2
## 689	MED14	2
## 690	MED15	2
## 691	MED17	2
## 692	MET	2
## 693	MGEA5	2
## 694	MIR3916	2
## 695	MIR4442,TOP2B	2
## 696	MIR612,mascrNA-menRNA	2
## 697	MIRLET7BHG	2

## 698	MKI67	2
## 699	MLLT1	2
## 700	MLXIP	2
## 701	MMS19	2
## 702	MON2	2
## 703	MORF4L2	2
## 704	MPP6	2
## 705	MSH2,HCG2040054	2
## 706	MSL1	2
## 707	MST4	2
## 708	MTCH1	2
## 709	MTCL1	2
## 710	MTHFD1L	2
## 711	MTRR	2
## 712	MUC16	2
## 713	MYC	2
## 714	MYCBP2	2
## 715	MYO19	2
## 716	MYO1E	2
## 717	MYPN	2
## 718	NA.v1539	2
## 719	NA.v1540	2
## 720	NA.v326	2
## 721	NA.v5750	2
## 722	NA.v7259	2
## 723	NA.v7454	2
## 724	NA.v8191	2
## 725	NA.v8558	2
## 726	NAA16.v1	2
## 727	NAA25.v1	2
## 728	NAP1L1.v1	2
## 729	NAP1L4.v1	2
## 730	NBEAL2	2
## 731	NBPF12	2
## 732	NBPF8P	2
## 733	NCAPD3	2
## 734	NCAPG2	2
## 735	NCL	2
## 736	NEAT1	2
## 737	NEAT1_2,NEAT1_1,NEAT1	2
## 738	NEAT1_3,NEAT1	2
## 739	NEK7	2
## 740	NFAT5	2
## 741	NFE2L2	2
## 742	NFXL1	2
## 743	NKTR	2
## 744	NLGN2	2
## 745	NOL11	2
## 746	NOL8	2
## 747	NOLC1	2
## 748	NOM1	2
## 749	NONO	2
## 750	NOP2	2
## 751	NOP56	2

## 752	NOP58	2
## 753	NOTCH2	2
## 754	NPEPPS	2
## 755	NPIP3,SMG1P3	2
## 756	NPL0C4	2
## 757	NPTX1	2
## 758	NR2C1	2
## 759	NRD1	2
## 760	NSUN2	2
## 761	NUP107	2
## 762	NUP133	2
## 763	NUP153	2
## 764	NUP155	2
## 765	NUP160	2
## 766	NUP205	2
## 767	NUP210	2
## 768	NVL	2
## 769	OBSCN	2
## 770	OCRL	2
## 771	OGT	2
## 772	OLR1	2
## 773	OSER1	2
## 774	P4HB	2
## 775	PABPC1	2
## 776	PABPC4	2
## 777	PAN3	2
## 778	PAPD7	2
## 779	PAXBP1	2
## 780	PCBP1	2
## 781	PCBP2	2
## 782	PCF11,RP11-727A23.4	2
## 783	PCID2	2
## 784	PCM1	2
## 785	PCNA	2
## 786	PCNT	2
## 787	PCNX	2
## 788	PCNXL3	2
## 789	PDCD11	2
## 790	PDCD6	2
## 791	PDE7A	2
## 792	PDS5A	2
## 793	PDS5B	2
## 794	PFAS	2
## 795	PGAM1P11,UBE4B	2
## 796	PHB2	2
## 797	PHF12	2
## 798	PHKA2	2
## 799	PHTF2	2
## 800	PICALM	2
## 801	PIEZ01	2
## 802	PIEZ02	2
## 803	PIK3C2B	2
## 804	PIK3R4	2
## 805	PIP5K1A	2

## 806	PKD1,MIR3180-5,RP11-304L19.3,MIR6511B1	2
## 807	PKD2	2
## 808	PKM	2
## 809	PKN2	2
## 810	PLAGL1	2
## 811	PLEC	2
## 812	PLEKHA1	2
## 813	PLK2	2
## 814	PLOD2	2
## 815	PLXNB2	2
## 816	PMPCB	2
## 817	POGZ	2
## 818	POLG	2
## 819	POLR1B	2
## 820	POLR2A	2
## 821	POLR2B	2
## 822	POLR2J2	2
## 823	PON2	2
## 824	PPP1R15B	2
## 825	PPP1R3C	2
## 826	PPP2R1B	2
## 827	PRC1	2
## 828	PRDX1	2
## 829	PRKAA1	2
## 830	pRNA.v3	2
## 831	PRPF3	2
## 832	PRPF38B	2
## 833	PRPF39	2
## 834	PRPF4B	2
## 835	PRPF8	2
## 836	PRR12	2
## 837	PRRC2A	2
## 838	PSMA1.v2	2
## 839	PSMA5	2
## 840	PSME4	2
## 841	PTBP1	2
## 842	PTCD3	2
## 843	PTP4A1	2
## 844	PTPN14	2
## 845	PTPN4	2
## 846	PTPRF	2
## 847	PTPRM	2
## 848	PTPRS	2
## 849	PUM1	2
## 850	PXN	2
## 851	QARS	2
## 852	QRICH1	2
## 853	QSER1	2
## 854	QSOX2	2
## 855	RABGGTB,ACADM	2
## 856	RAD21	2
## 857	RAD23A	2
## 858	RAD54L	2
## 859	RAF1	2

## 860	RAN	2
## 861	RANBP2	2
## 862	RANGAP1	2
## 863	RAP1GAP2	2
## 864	RAPGEF2	2
## 865	RB1CC1	2
## 866	RBBP7	2
## 867	RBM14	2
## 868	RBM17	2
## 869	RBM26	2
## 870	RBM3	2
## 871	RBM33	2
## 872	RBM39	2
## 873	RBM5	2
## 874	REV1	2
## 875	REV3L	2
## 876	RFWD3	2
## 877	RHOBTB3	2
## 878	RIF1	2
## 879	RMRP,RNase_MRP	2
## 880	RN7SK	2
## 881	RNF145	2
## 882	RNU1-27P	2
## 883	RNU1-28P	2
## 884	ROCK2	2
## 885	RP11-231L11.3,SLC25A36	2
## 886	RP11-298E9.7,PITRM1-AS1,PFKP	2
## 887	RP11-396K3.1	2
## 888	RP11-872D17.8,SLC43A3	2
## 889	RP13-735L24.1	2
## 890	RP4-620E11.4,CHD6	2
## 891	RP5-1172N10.4,USP9X	2
## 892	RPL13A	2
## 893	RPL21P5,HECTD1	2
## 894	RPL5	2
## 895	RPRD1B	2
## 896	RRP1B	2
## 897	SACM1L	2
## 898	SCARB1	2
## 899	SCARNA2	2
## 900	SCML1	2
## 901	SEC16A	2
## 902	SENP6	2
## 903	SETD1A	2
## 904	SETD1B,RP11-347I19.7	2
## 905	SETD2	2
## 906	SETD5	2
## 907	SF1	2
## 908	SF3B1	2
## 909	SFMBT2	2
## 910	SGK1	2
## 911	SH3PXD2A	2
## 912	SHC1,PYG02	2
## 913	SHPRH	2

## 914	SKIL	2
## 915	SKIV2L2	2
## 916	SLC12A6	2
## 917	SLC16A1	2
## 918	SLC20A1	2
## 919	SLC25A3	2
## 920	SLC30A1	2
## 921	SLC38A2	2
## 922	SLC3A2	2
## 923	SLC4A7	2
## 924	SLC7A5	2
## 925	SLC7A6	2
## 926	SLMAP	2
## 927	SLM02	2
## 928	SLTM	2
## 929	SMC1A	2
## 930	SMC3	2
## 931	SMC4	2
## 932	SMC5	2
## 933	SMCHD1	2
## 934	SMG1	2
## 935	SMG1P2,RP11-368N21.5	2
## 936	SMG1P4	2
## 937	SMG7	2
## 938	SMS	2
## 939	SNHG1	2
## 940	SNHG12	2
## 941	SNHG17	2
## 942	SNORA76C	2
## 943	SNORD3D	2
## 944	SNRNP200	2
## 945	SNX5	2
## 946	SON	2
## 947	SPAG5	2
## 948	SPDL1	2
## 949	SPIN1	2
## 950	SPTAN1	2
## 951	SPTBN1	2
## 952	SPTLC1	2
## 953	SQSTM1	2
## 954	SRBD1	2
## 955	SRCAP	2
## 956	SREK1	2
## 957	SRRM2	2
## 958	SRRT	2
## 959	SRSF11	2
## 960	SRSF2	2
## 961	SRSF5	2
## 962	SSFA2	2
## 963	SSR1	2
## 964	STT3B	2
## 965	SUN1	2
## 966	SUPT6H	2
## 967	SUZ12	2

## 968	SVEP1	2
## 969	SYNE2	2
## 970	SZT2	2
## 971	TAF1	2
## 972	TAF15	2
## 973	TAF2	2
## 974	TARBP1	2
## 975	TBC1D8	2
## 976	TBK1	2
## 977	TBX3	2
## 978	TCERG1	2
## 979	TCF25	2
## 980	TCP1	2
## 981	TDRD9	2
## 982	TEAD1,RP11-47J17.1	2
## 983	TEL02	2
## 984	TEX10	2
## 985	TFAP2A	2
## 986	TFG	2
## 987	TFRC	2
## 988	TGFBR1	2
## 989	TGFBR3	2
## 990	THAP9-AS1	2
## 991	THOC2	2
## 992	TIAL1	2
## 993	TIAM2,SCAF8	2
## 994	TICRR	2
## 995	TIMELESS	2
## 996	TJP1	2
## 997	TM4SF1	2
## 998	TM9SF3	2
## 999	TMEM259	2
## 1000	TMEM87A	2
## 1001	TMP0	2
## 1002	TNFRSF10B,RP11-875011.3	2
## 1003	TNKS2	2
## 1004	TNPO3	2
## 1005	TNRC6A	2
## 1006	TOB1	2
## 1007	TOP2A	2
## 1008	TPCN1	2
## 1009	TPP2	2
## 1010	TPR	2
## 1011	TRA2A	2
## 1012	TRAP1	2
## 1013	TRIM28	2
## 1014	TRIM33	2
## 1015	TRIM37	2
## 1016	TRIM44	2
## 1017	TRIM66	2
## 1018	TRIO	2
## 1019	TRIP12	2
## 1020	TRIP13	2
## 1021	TROAP	2

## 1022	TSC22D1	2
## 1023	TSPYL1	2
## 1024	TSR1	2
## 1025	TTC37	2
## 1026	TTI1	2
## 1027	TXLNG	2
## 1028	U2SURP	2
## 1029	UBA2	2
## 1030	UBAP2	2
## 1031	UBAP2L	2
## 1032	UBE2I	2
## 1033	UBE3C	2
## 1034	UBP1	2
## 1035	UBR4	2
## 1036	UBR5	2
## 1037	UBXN4	2
## 1038	ULK1	2
## 1039	ULK3	2
## 1040	UPF1	2
## 1041	UPF2	2
## 1042	UQCRC2	2
## 1043	URI1	2
## 1044	USP1	2
## 1045	USP10	2
## 1046	USP22	2
## 1047	USP24	2
## 1048	USP3	2
## 1049	USP34	2
## 1050	USP47	2
## 1051	USP7	2
## 1052	VCP	2
## 1053	VIM	2
## 1054	VMP1,MIR21	2
## 1055	VPRBP	2
## 1056	VPS13A	2
## 1057	WDR11	2
## 1058	WDR26	2
## 1059	WDR44	2
## 1060	WDR6	2
## 1061	WDR90	2
## 1062	WEE1	2
## 1063	WLS	2
## 1064	WNK1	2
## 1065	WSB1	2
## 1066	XP01	2
## 1067	XP06	2
## 1068	XPOT	2
## 1069	XRN2	2
## 1070	YARS	2
## 1071	YBX3	2
## 1072	YLPM1	2
## 1073	YTHDC2	2
## 1074	YTHDF3	2
## 1075	YWHAG	2

## 1076	ZBED5	2
## 1077	ZC3H11A	2
## 1078	ZC3H7A	2
## 1079	ZCCHC11	2
## 1080	ZDHHC11	2
## 1081	ZDHHC17	2
## 1082	ZDHHC5	2
## 1083	ZDHHC6	2
## 1084	ZFAND5	2
## 1085	ZFAS1,ZNFX1-AS1_2	2
## 1086	ZFC3H1	2
## 1087	ZFP36L1	2
## 1088	ZFR	2
## 1089	ZMYM2	2
## 1090	ZNF106	2
## 1091	ZNF131	2
## 1092	ZNF146	2
## 1093	ZNF195	2
## 1094	ZZEF1	2
## 1095	ZZZ3	2

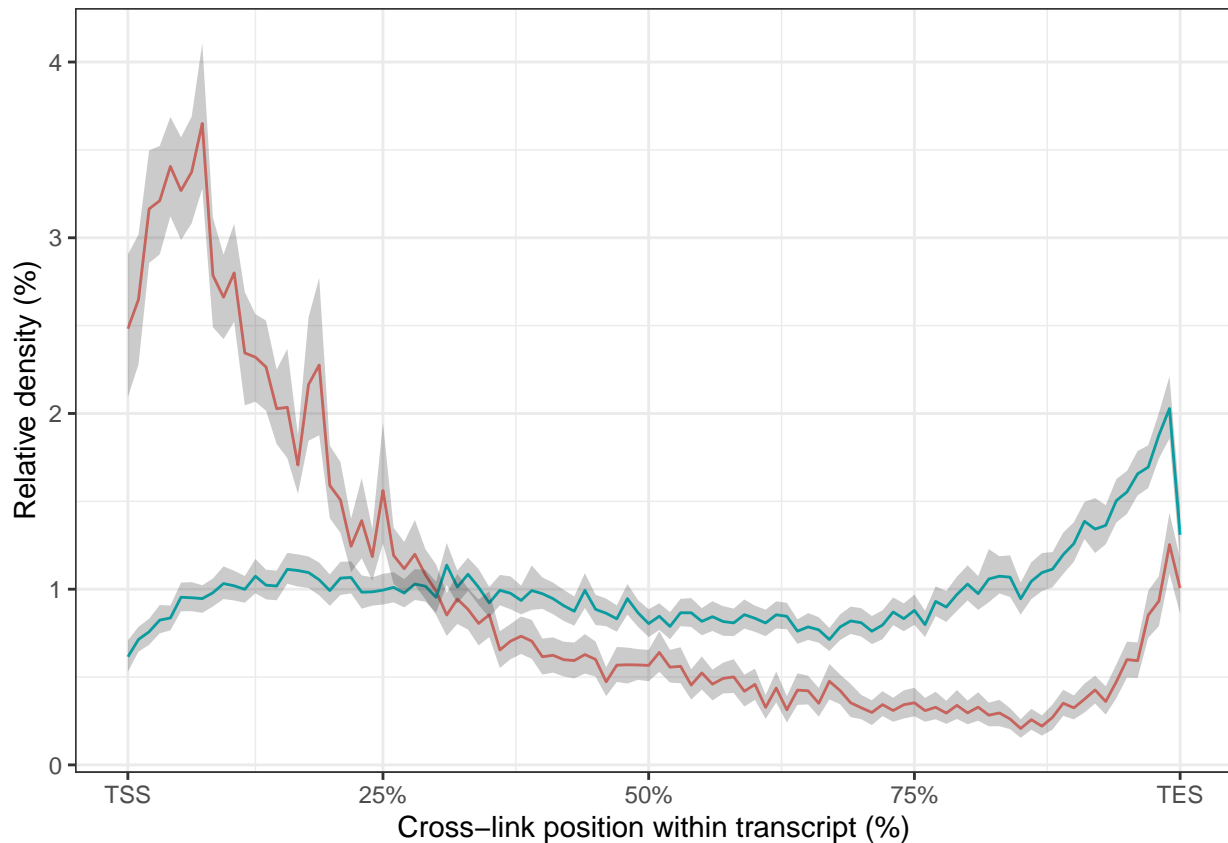
#Figure 4B Plot profile of ALYREF-DMSO cross-links across group 1 and 2 genes

```

ALYREF_100bins_scaled %>%
  filter(geneName %in% clust_all$geneName) %>%
  left_join(clust_all) %>%
  filter(Timepoint == "DMSO" & totals_unNorm >= 20) %>%
  spread(distBin, value) %>%
  gather("distBin", "value", "1":"100") %>%
  ggplot(aes(x = as.numeric(distBin), y = value, col = as.factor(clusterID),
    group = as.factor(clusterID))) + stat_summary(fun = mean,
    geom = "line", size = 0.5) + stat_summary(fun.data = "mean_cl_boot",
    geom = "ribbon", alpha = 0.25, col = NA) + theme_bw() + xlab("Cross-link position within transcript")
  scale_x_continuous(breaks = c(1, 25, 50, 75, 100), labels = c("TSS",
    "25%", "50%", "75%", "TES")) + theme(legend.position = "none") +
  ylab("Relative density (%)")

```

```
## Joining, by = "geneName"
```



#Figure 4 C analyse expression, gene size, exonic size (cDNA size) and number of exons associated with group 1 and 2 genes

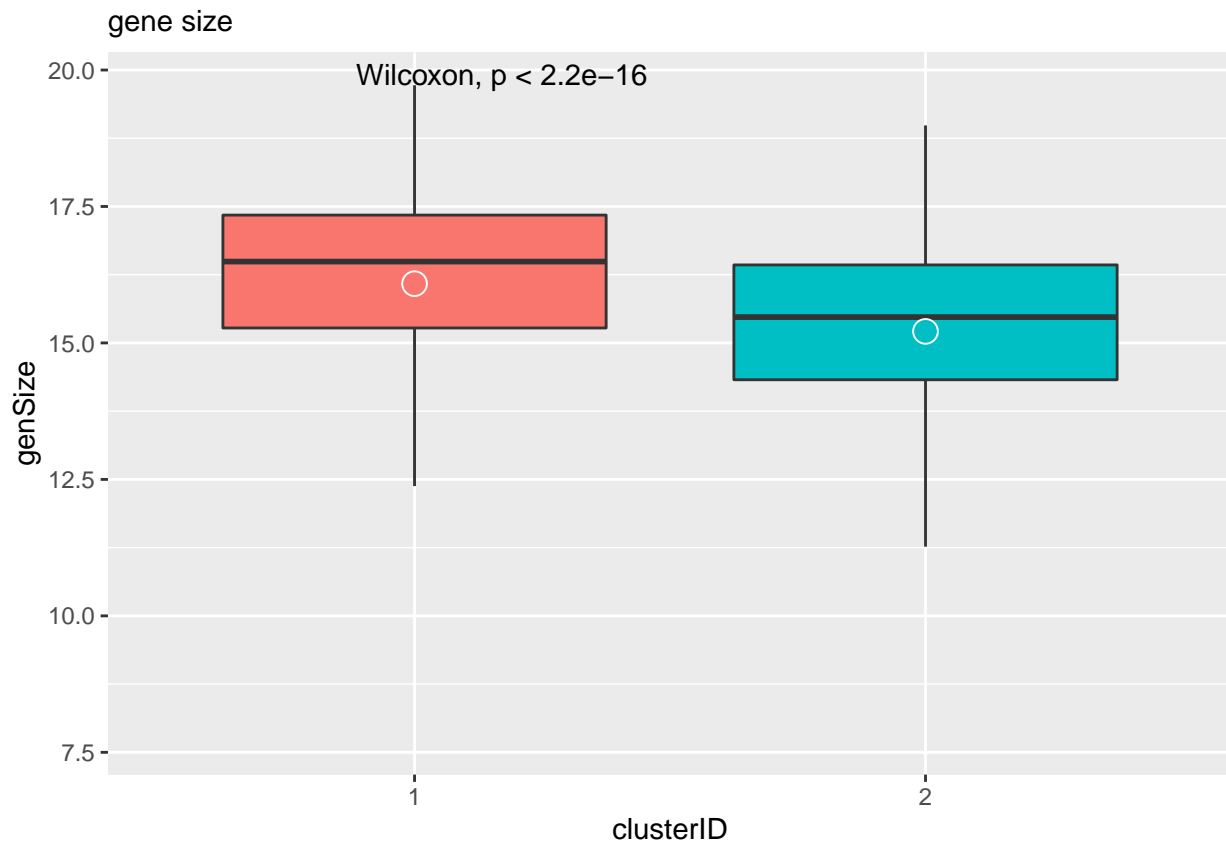
```
annotation_metaData <- read.table("../data/hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.ex
sep = "\t", header = T)
```

```
data_for_boxplots <- merge(annotation_metaData, clust_all) %>%
  filter(exonID == totalExons) %>%
  mutate(genSize = exonSize + genDistToTSS) %>%
  select(-exonID, -exonSize, -stop, -start, -chr, -strand,
        -score, -genDistToTSS) %>%
  select(c("geneName", "clusterID", "totalExons", "relSize",
          "genSize")) %>%
  mutate_at(vars("totalExons", "relSize", "genSize"), log2) %>%
  left_join(expression_vector)
```

```
## Joining, by = "geneName"
```

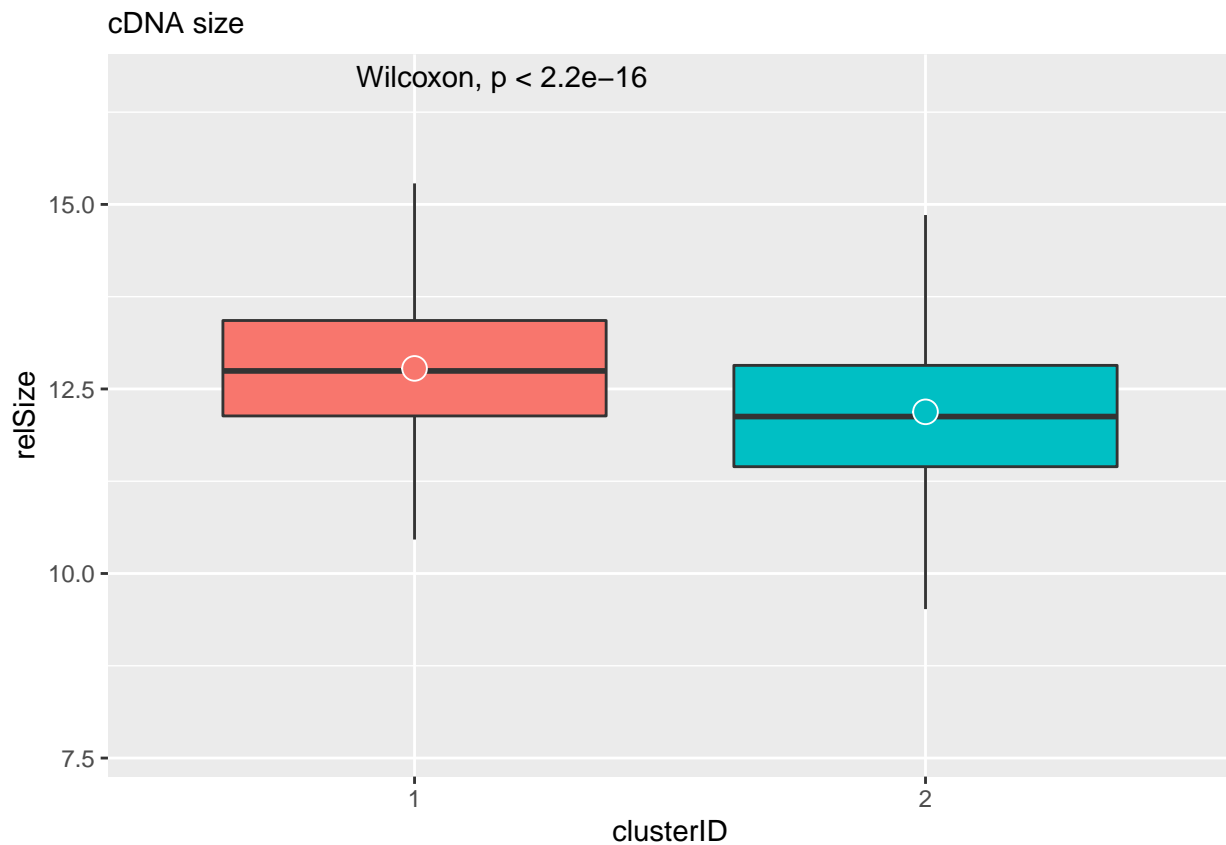
```
data_for_boxplots %>%
  ggplot(aes(x = clusterID, y = genSize, fill = as.factor(clusterID),
            group = clusterID)) + geom_boxplot(outlier.shape = NA) +
  geom_point(stat = "summary", shape = 21, col = "white", size = 4) +
  labs(subtitle = "gene size") + stat_compare_means(method = "wilcox.test") +
  theme(legend.position = "none")
```

```
## No summary function supplied, defaulting to `mean_se()`
```

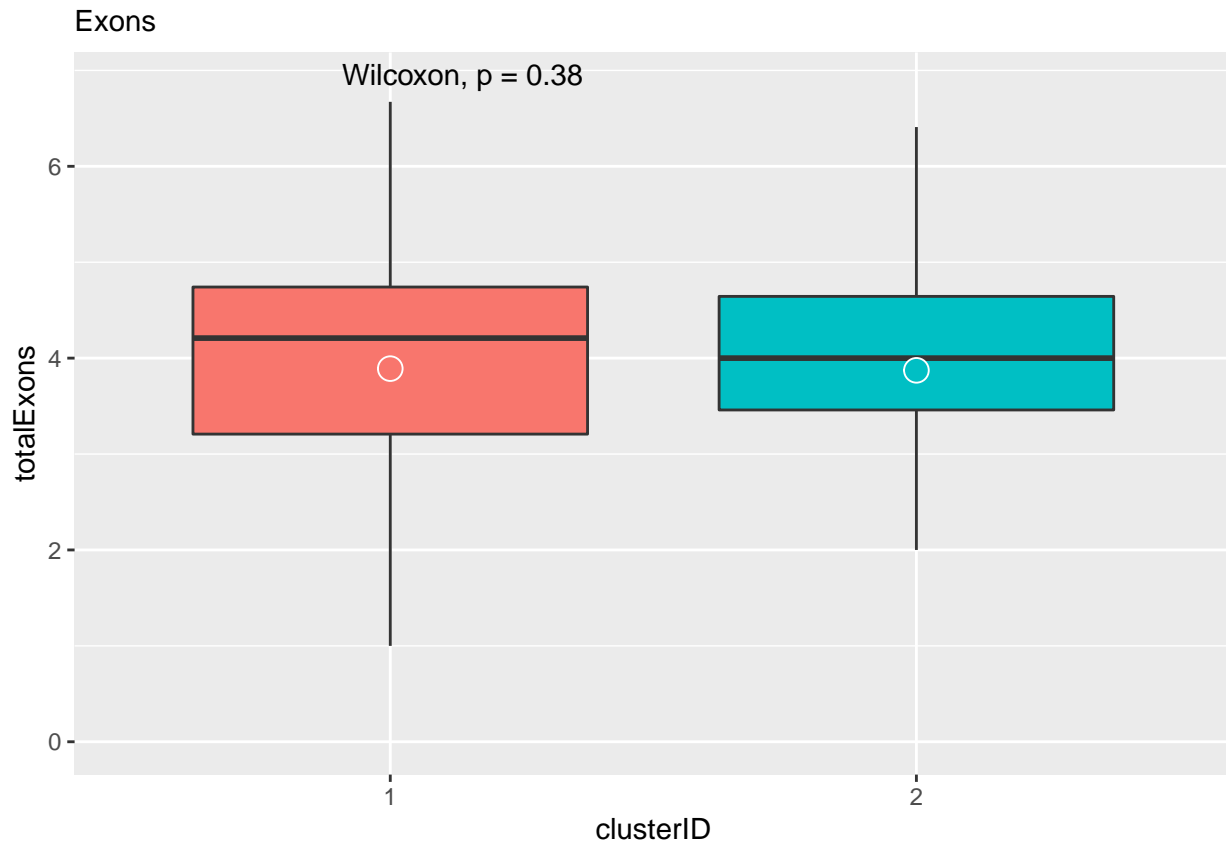
```
data_for_boxplots %>%
  ggplot(aes(x = clusterID, y = relSize, fill = as.factor(clusterID),
    group = clusterID)) + geom_boxplot(outlier.shape = NA) +
  geom_point(stat = "summary", shape = 21, col = "white", size = 4) +
  labs(subtitle = "cDNA size") + stat_compare_means(method = "wilcox.test") +
  theme(legend.position = "none")
```

No summary function supplied, defaulting to `mean_se()`



```
data_for_boxplots %>%  
  ggplot(aes(x = clusterID, y = totalExons, fill = as.factor(clusterID),  
    group = clusterID)) + geom_boxplot(outlier.shape = NA) +  
  geom_point(stat = "summary", shape = 21, col = "white", size = 4) +  
  labs(subtitle = "Exons") + stat_compare_means(method = "wilcox.test") +  
  theme(legend.position = "none")
```

```
## No summary function supplied, defaulting to `mean_se()`
```



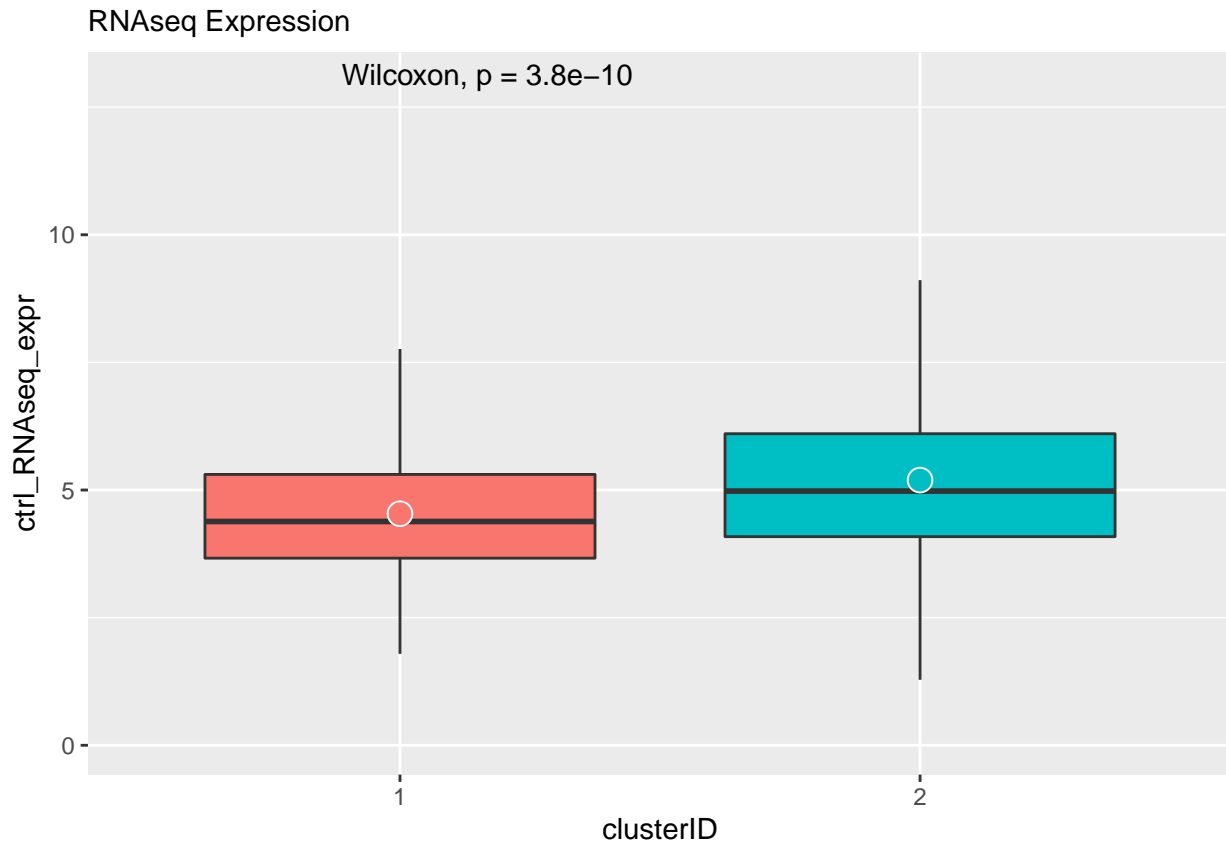
```
data_for_boxplots %>%
  ggplot(aes(x = clusterID, y = ctrl_RNAseq_expr, fill = as.factor(clusterID),
    group = clusterID)) + geom_boxplot(outlier.shape = NA) +
  geom_point(stat = "summary", shape = 21, col = "white", size = 4) +
  labs(subtitle = "RNAseq Expression") + stat_compare_means(method = "wilcox.test") +
  theme(legend.position = "none")
```

```
## Warning: Removed 2 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 2 rows containing non-finite values (stat_summary).
```

```
## No summary function supplied, defaulting to `mean_se()`
```

```
## Warning: Removed 2 rows containing non-finite values (stat_compare_means).
```



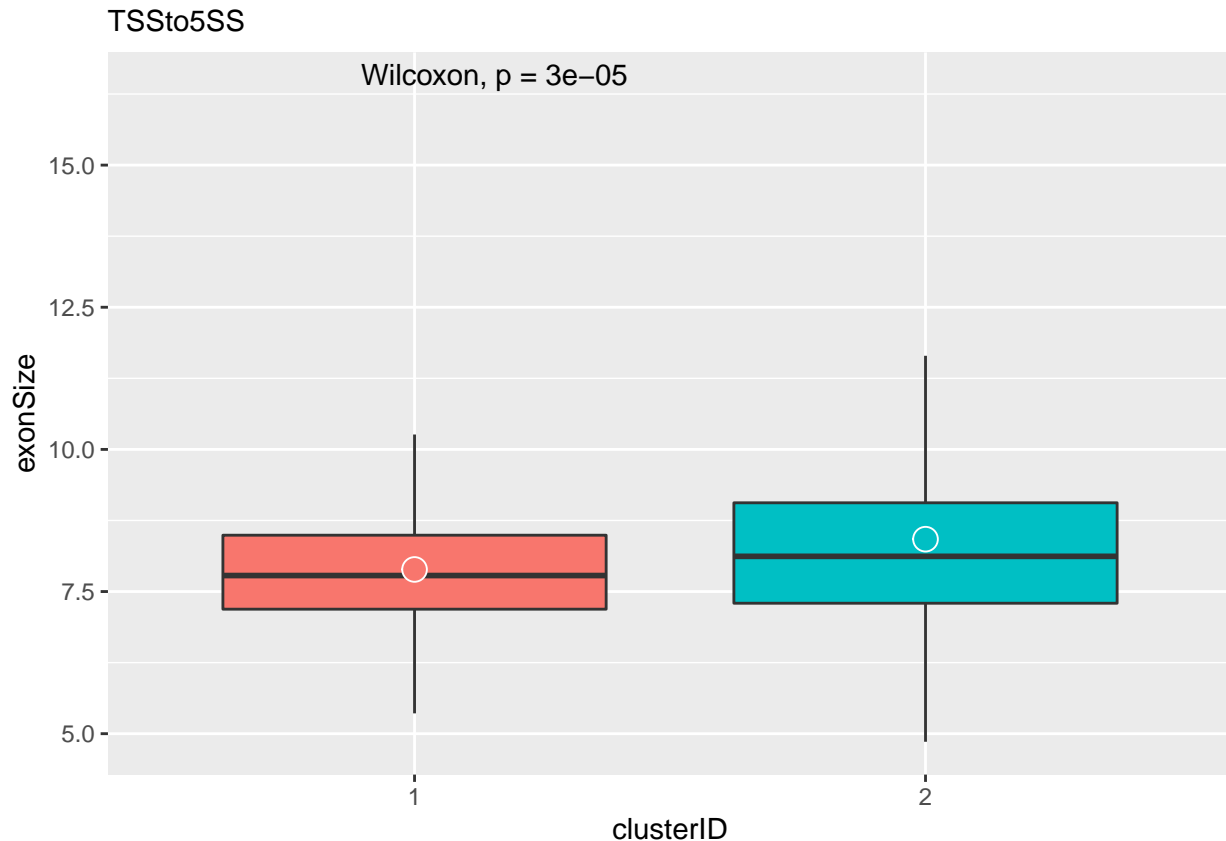
#Supplementary Figure 4 D boxplot of 1st exons size in group 1 and 2 genes

```
annotation_metadata <- read.table("../data/hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.ex
sep = "\t", header = T)
```

```
data_for_boxplots <- merge(annotation_metadata, clust_all) %>%
  filter(exonID == 1) %>%
  select(c("geneName", "clusterID", "exonSize")) %>%
  mutate_at(vars("exonSize"), log2)
```

```
data_for_boxplots %>%
  ggplot(aes(x = clusterID, y = exonSize, fill = as.factor(clusterID),
    group = clusterID)) + geom_boxplot(outlier.shape = NA) +
  geom_point(stat = "summary", shape = 21, col = "white", size = 4) +
  labs(subtitle = "TSS to 5SS") + stat_compare_means(method = "wilcox.test") +
  theme(legend.position = "none")
```

```
## No summary function supplied, defaulting to `mean_se()`
```



#Figure 4 D plot profile of exon junction density across group 1 and 2 genes

make profile of positions of exon junctions first prepair
all genes

```
annoDF.WholeGene <- read.csv("../data/hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.exonNum
sep = "\t", header = F) %>%
setNames(c("chr", "start", "stop", "IDAnno", "score", "strand")) %>%
separate(IDAnno, c("geneName", "biotype", "WholeGene.exonID",
  "WholeGene.totalExons", "WholeGene.exonSize", "WholeGene.relSize",
  "WholeGene.distToTSS", "WholeGene.geneDesc", "WholeGene.distToDist5SS",
  "WholeGene.distToDist3SS", "WholeGene.relSizeof5UTR",
  "WholeGene.genSizeof5UTR", "WholeGene.relSize3UTR", "WholeGene.genSizeof3UTR",
  "WholeGene.relSizeofCDS", "WholeGene.genSizeofCDS", "WholeGene.pAtoUp3ss",
  "WholeGene.pAtoUp5ss", "WholeGene.exonDesc"), sep = ":::")
```

```
all_genes <- annoDF.WholeGene
```

```
totalSizes_WholeGene.mRNASize <- annoDF.WholeGene %>%
  filter(WholeGene.exonID == WholeGene.totalExons) %>%
  mutate(WholeGene.matureRNASize = as.numeric(WholeGene.relSize),
    WholeGene.geneSize = as.numeric(ifelse(WholeGene.totalExons >
      1, (as.numeric(WholeGene.distToTSS) + as.numeric(WholeGene.exonSize)),
      WholeGene.exonSize))) %>%
  select(geneName, WholeGene.matureRNASize)
```

```
totalSizes_relSize <- annoDF.WholeGene %>%
  select(geneName, WholeGene.relSize, WholeGene.exonID, WholeGene.totalExons)
```

```

# annotation for multiexonic genes for profiles
exonic_geneName <- all_genes %>%
  mutate(exonic = case_when(WholeGene.totalExons > 1 ~ "multiexonic",
    TRUE ~ "monoexonic")) %>%
  select(geneName, exonic) %>%
  unique()

# generate profiles for group 1 and 2

EJ_profiles_cluster <- clust_all %>%
  # add total mRNA sizes, exon ids and relative size of
  # mRNA up to exon displayed
  mutate(geneName = as.character(geneName)) %>%
  left_join(totalSizes_WholeGene.mRNASize) %>%
  left_join(totalSizes_relSize) %>%
  mutate(WholeGene.relSize = as.integer(WholeGene.relSize),
    WholeGene.exonID = as.integer(WholeGene.exonID), WholeGene.matureRNASize = as.integer(WholeGene
  # calculate relative positions of junction
  mutate(rel_position_of_EJ = WholeGene.relSize/WholeGene.matureRNASize) %>%
  # remove last exon as this does not end with an
  # exonjunction
  filter(WholeGene.exonID != WholeGene.totalExons)

## Joining, by = "geneName"
## Joining, by = "geneName"
EJ_profiles_1 <- EJ_profiles_cluster

# add bins
EJ_profiles_1$new_distBin <- cut(EJ_profiles_1$rel_position_of_EJ,
  seq(0, 1, by = 0.01), labels = seq("1", "100", by = 1))

EJ_profiles_2 <- EJ_profiles_1 %>%
  group_by(clusterID, geneName, WholeGene.matureRNASize, new_distBin) %>%
  summarise(tally = n()) %>%
  # normalise to gene expression level
  left_join(expression_vector) %>%
  mutate(tally_n = (tally/ctrl_RNAseq_expr)/(WholeGene.matureRNASize/100),
    tally_nr = tally/(WholeGene.matureRNASize/100)) #step to normalise count to bin size.

## `summarise()` has grouped output by 'clusterID', 'geneName',
## 'WholeGene.matureRNASize'. You can override using the `.groups` argument.
## Joining, by = "geneName"

fig <- EJ_profiles_2 %>%
  mutate(totals_unNorm = sum(tally), pct = tally_nr/sum(tally_nr) *
    100) %>%
  spread(new_distBin, pct, fill = 0) %>%
  gather("distBin", "value", c("1":"100")) %>%
  mutate(distBin = factor(distBin, levels = c(1:100))) %>%
  ggplot(aes(x = as.numeric(distBin), y = value)) + stat_summary(fun = mean,
  geom = "line", size = 0.5, aes(col = clusterID)) + stat_summary(fun.data = "mean_cl_boot",
  geom = "ribbon", alpha = 0.2, col = NA, aes(group = clusterID)) +
  theme_bw()

```

fig

