Figure 3 Regulatory elements annotations

Zheng Zuo

04/15/2022

Contents

Figure 3A

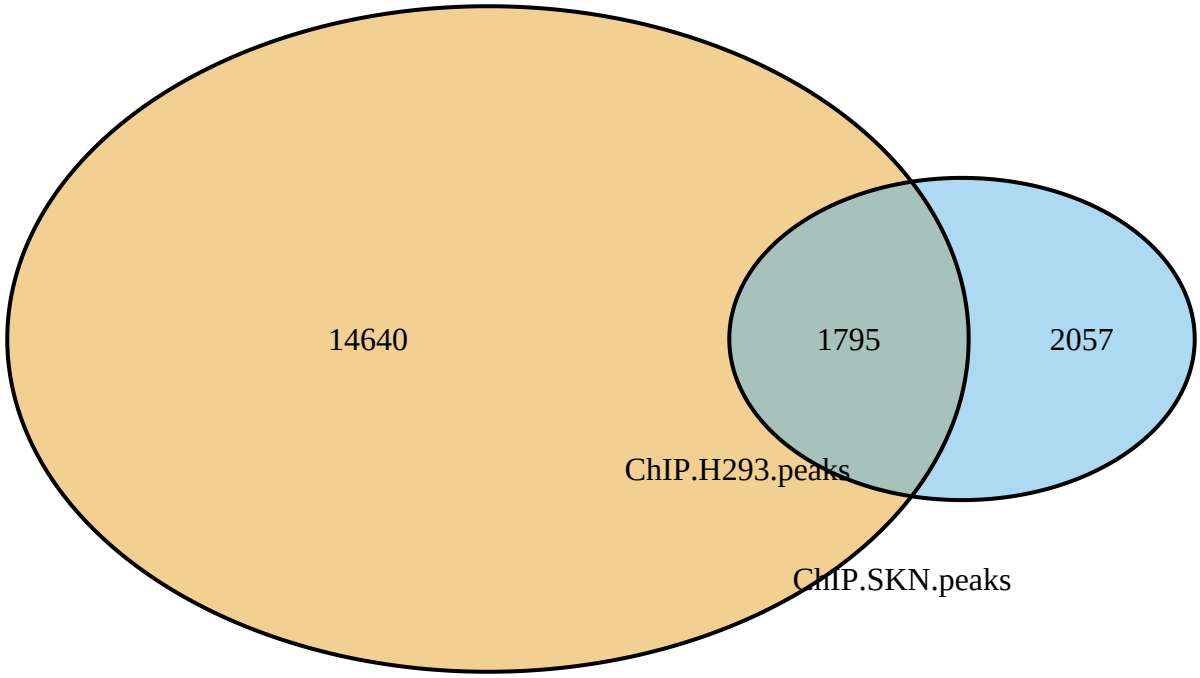
```
require(dplyr)
require(GenomicRanges)

ChIP.H293.peaks <- read.table("ENCFF107KSN.bed") %>%
                   GenomicRanges::makeGRangesFromDataFrame(seqnames.field = "V1",
                                                           start.field    = "V2",
                                                           end.field      = "V3",
                                                           keep.extra.columns = TRUE) %>% unique()

ChIP.SKN.peaks <- read.table("./ENCFF049VST.bed")  %>%
                  GenomicRanges::makeGRangesFromDataFrame(seqnames.field = "V1",
                                                          start.field   = "V2",
                                                          end.field     = "V3",
                                                         keep.extra.columns = TRUE) %>% unique()

ChIP.overlapping.peaks <- ChIPpeakAnno::findOverlapsOfPeaks(ChIP.H293.peaks,
                                                            ChIP.SKN.peaks,
                                                            minoverlap = 0.85,
                                                            connectedPeaks = "merge")

ChIPpeakAnno::makeVennDiagram(ChIP.overlapping.peaks,
                              fill=c("#5BB4E5", "#E6A024"))
```




```
## $p.value
##      ChIP.H293.peaks ChIP.SKN.peaks pval
## [1,]               1              1    0
##
## $vennCounts
##      ChIP.H293.peaks ChIP.SKN.peaks Counts
## [1,]               0              0      0
## [2,]               0              1  14640
## [3,]               1              0   2057
## [4,]               1              1   1795
## attr(,"class")
## [1] "VennCounts"
```

Figure 3C

```
require(GenomicRanges)
load("ZFP3.motif.RData")
ZFP3.merged.sites <- TFCookbook::matchPEM(PEM = ZFP3.Core.PEM,
                                  subject = ChIP.overlapping.peaks$mergedPeaks,
                                  genome  = "hg38",
                                  out     = "positions",
                                  E.cutoff= -5)
```

```
##
```

```
## Attaching package: 'Biostrings'

## The following object is masked from 'package:base':
##
##     strsplit
```

```
ZFP3.unique.sites <- TFCookbook::matchPEM(PEM = ZFP3.Core.PEM,
                                  subject = ChIP.overlapping.peaks$uniquePeaks,
                                  genome  = "hg38",
                                  out     = "positions",
                                  E.cutoff= -5)

ZFP3.merged.sites$Property = "merged"
ZFP3.unique.sites$Property = "unique"
ZFP3.sites = c(ZFP3.merged.sites, ZFP3.unique.sites) %>% unique()


ZFP3.sites$predicted.Core.Energy     <- ZFP3.sites$predicted.Energy
ZFP3.sites$predicted.Energy          <- NULL
ZFP3.sites$predicted.Upstream.Energy <- TFCookbook::predictEnergy(ZFP3.sites$Sequence, ZFP3.Upstream.PE
ZFP3.sites$predicted.Full.Energy     <- ZFP3.sites$predicted.Upstream.Energy+ZFP3.sites$predicted.Core.

ZFP3.sites
```

```
## GRanges object with 37161 ranges and 5 metadata columns:
##           seqnames              ranges strand |               Sequence
##              <Rle>           <IRanges>  <Rle> |            <character>
##       [1]     chr1       863435-863462      + | AGCAAATATCCGGAATATAC..
##       [2]     chr1       863733-863760      + | GATCCCTTCGTGAGCAATAA..
##       [3]     chr1       863689-863716      - | AGCATTATTCCGGTTTACAC..
##       [4]     chr1       931489-931516      + | GGGCCCTGAAGGGTGTGTAG..
##       [5]     chr1     1430070-1430097      - | TTGCTTGTGCCTAGGAGTTC..
##       ...      ...                 ...    ... .                   ...
##   [37157]     chrX 149272780-149272807      - | AACCTCTGTCTCCTGGATTC..
##   [37158]     chrX 153512593-153512620      - | GTGTGTGTGAGTGAGAGTAT..
##   [37159]     chrX 153937271-153937298      + | TCTTTTGAGTTACATTGTTC..
##   [37160]     chrX 153937320-153937347      + | CATGACACAGGAAGGTAAAG..
##   [37161]     chrX 154303162-154303189      - | TAAATAAATAAAAATTTTAA..
##              Property predicted.Core.Energy predicted.Upstream.Energy
##           <character>             <numeric>                 <numeric>
##       [1]      merged              -5.08427                  0.967365
##       [2]      merged              -6.00113                  0.700694
##       [3]      merged              -5.10183                 -0.749548
##       [4]      merged              -5.68745                 -3.121366
##       [5]      merged              -5.17562                  0.137914
##       ...         ...                   ...                       ...
##   [37157]      unique              -5.06732                 -2.263216
##   [37158]      unique              -5.09077                 -2.438773
##   [37159]      unique              -6.09899                 -0.213538
##   [37160]      unique              -5.01703                  2.929134
##   [37161]      unique              -5.04572                  1.662693
##           predicted.Full.Energy
##                       <numeric>
```

```
##       [1]              -4.11691
##       [2]              -5.30044
##       [3]              -5.85138
##       [4]              -8.80882
##       [5]              -5.03771
##       ...                   ...
##   [37157]              -7.33053
##   [37158]              -7.52955
##   [37159]              -6.31253
##   [37160]              -2.08790
##   [37161]              -3.38303
##   -------
##   seqinfo: 23 sequences from an unspecified genome; no seqlengths
```

```
Ciliated.Genes.Tissues <- readr::read_delim("expressionclustertissue_81_Ciliated.tsv",
                                            delim = "\t", escape_double = FALSE, trim_ws = TRUE)
```

```
## Rows: 337 Columns: 3
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (3): Ensembl, Gene, Gene description
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
Ciliated.Genes.Cells   <- readr::read_delim("expressionclustersinglecell_49_Ciliated.tsv",
                                            delim = "\t", escape_double = FALSE, trim_ws = TRUE)
```

```
## Rows: 457 Columns: 3
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr (3): Ensembl, Gene, Gene description
```

```
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

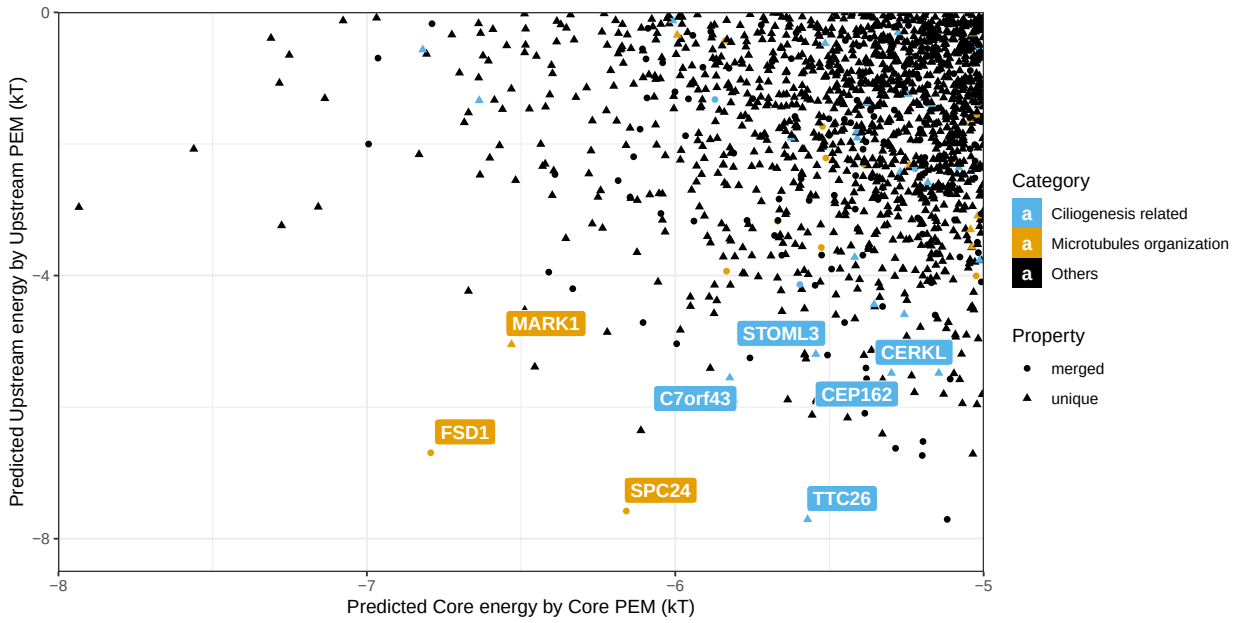
```
Ciliated.Genes <- merge(Ciliated.Genes.Tissues, Ciliated.Genes.Cells,  all = TRUE)

MT.related.genes <- c("MARK1","SPC24","MAP1S","FSD1", "CEP295", "ERBB2", "RABGAP1", "MAPT", "KAT2B")
Cilia.related.genes <- c("CEP95", "STOML3", "WDR19", "WDR38", "TTC26", "CEP162", "CERKL", "PKN2", "C7or
```

```
require(EnsDb.Hsapiens.v86) ##(hg38)
## create annotation file from EnsDb or TxDb
require(ChIPpeakAnno)
annoData             <- ChIPpeakAnno::toGRanges(EnsDb.Hsapiens.v86, feature="gene")
ZFP3.annotated.sites <- ChIPpeakAnno::annotatePeakInBatch(ZFP3.sites, AnnotationData=TSS.human.GRCh38)
```

```
ZFP3.annotated.sites$gene_name <- annoData$gene_name[match(ZFP3.annotated.sites$feature, names(annoData

require(ggplot2)
ZFP3.annotated.sites %>% as_tibble() %>%
  dplyr::filter(abs(distancetoFeature) < 500) %>%
  arrange(predicted.Full.Energy) %>%
  mutate(Category = case_when(gene_name %in% MT.related.genes ~ "Microtubules organization",
                              gene_name %in% c(Ciliated.Genes$Gene, Cilia.related.genes) ~ "Ciliogenesis
                              TRUE ~ "Others")) %>%
 # arrange(predicted.Full.Energy)
  mutate(Label = if_else(Category!="Others" & (predicted.Full.Energy<(-10)), gene_name, NULL)) %>%
  ggplot(aes(x = predicted.Core.Energy,
             y = predicted.Upstream.Energy,
             fill = Category,
             shape = Property,
             color = Category,
             label = Label))+
  geom_point() +
  ggrepel::geom_label_repel(aes(fill = Category), fontface = "bold", color = "white") +
  xlab("Predicted Core energy by Core PEM (kT)") +
  ylab("Predicted Upstream energy by Upstream PEM (kT)") +
  theme_bw() + scale_fill_manual(values = c("#56B4E9", "#E69F00", "#000000"))+
  scale_color_manual(values = c("#56B4E9", "#E69F00", "#000000"))+
  scale_x_continuous(limits = c(-8, -5), expand = c(0,0)) +
  scale_y_continuous(breaks = c(4, 0, -4, -8), limits = c(-8.5, 0), expand = c(0,0))
```

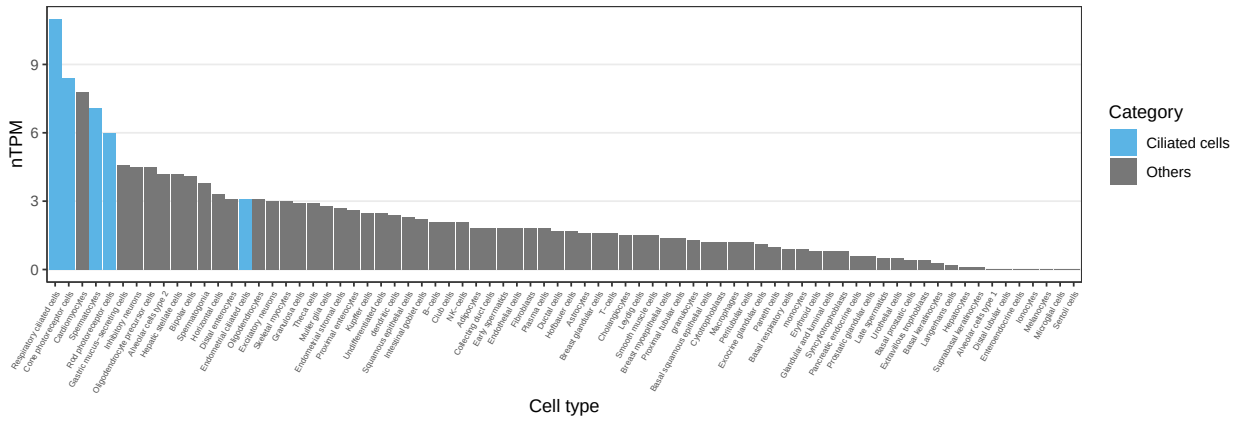



```
#ggsave("Regulatory elements annotations.svg", width = 10, height = 5)
```

Figure 3D

```
#require(extrafont)
#extrafont::loadfonts(device = "win")

  readr::read_delim("rna_single_cell_type.tsv",
                    delim = "\t", escape_double = FALSE, trim_ws = TRUE) %>%
  dplyr::filter(`Gene name` == "ZFP3") %>%
  mutate(`Cell type` = forcats::fct_reorder(`Cell type`, nTPM, .desc = TRUE),
         Category    = case_when(`Cell type` %in% c("Respiratory ciliated cells",
                                                    "Spermatocytes",
                                                    "Endometrial ciliated cells",
                                                  "Cone photoreceptor cells", "Rod photoreceptor cells"
                                                  ) ~ "Ciliated cells",
                                 TRUE ~ "Others")) %>%
  arrange(desc(nTPM)) %>%
  ggplot(aes(`Cell type`, nTPM, fill = Category))+
  geom_col() + theme_bw()+
  scale_fill_manual(values = c("#5BB4E5", "#777777")) +
  #theme(axis.text.x = element_text(angle = 60, family = "TT Arial",  size = 4.5, hjust = 1),
  theme(axis.text.x = element_text(angle = 60, size = 4.5, hjust = 1),
        panel.grid.major.x = element_blank()) +
  scale_y_continuous(minor_breaks = NULL)
```




```
#ggsave("Gene expressions of ZFP3.svg", width = 7, height = 2.4)
```

```
#save.image("ZFP3.annotated.sites.RData")
```