Sup Fig. 3 b

RAC

24/08/2020

Load packages and assign filepaths

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

COUNTS="../../data/xiCLIP_read2_mono_multi_exonic.calculate_coverage_gene_structures_exonsover99nt_LINC

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

ANNOTATION_BED_FILEPATH="../../data/hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.exonNumber.s
```

```
# beow code was ran on cluster to process large count files
# produced when analysing whole reads !/usr/bin/env Rscript
# suppressMessages(library(dplyr))
# suppressMessages(library(tidyr)) args =
# commandArgs(trailingOnly=TRUE)
# ANNOTATION_BED_FILEPATH<-args[1]
# EXPRESSION_VECTOR_FILEPATH<-args[2]
# IN_FILE_PATH_3_END<-args[3] IN_FILE_PATH_5_END<-args[4]
# OUTFILE_PATH<-args[5] Load expression vector
# ------------------------------------------------------------------------------------
# load(EXPRESSION_VECTOR_FILEPATH) print('load expression
# file') expression_vector<-left_join(
# (as.data.frame(ctrl_RNAseq_expr) %>% add_rownames(var =
# 'geneID')), (as.data.frame(ctrl_TTseq_expr) %>%
# add_rownames(var = 'geneID')) ) %>%
# select(-ctrl_TTseq_expr) %>% mutate(ctrl_RNAseq_expr =
# case_when( ctrl_RNAseq_expr ==0 ~
# min(ctrl_RNAseq_expr[ctrl_RNAseq_expr > 0]), TRUE ~
# ctrl_RNAseq_expr )) load annobed
# ------------------------------------------------------------------------------------------------------
```

```
# print('load annotation file')
# annoBed<-read.table(ANNOTATION_BED_FILEPATH, sep = '\t',
# header = F) %>% setNames(c('chr', 'start', 'end',
# 'geneID', 'score', 'strand')) %>% separate(geneID, into =
# c('geneID', 'Biotype', 'ExonNumber',
# 'TotalNumberOfExons', 'ExonSize', 'ExonicDistance',
# 'ExonDistFromTSS','ExonStature', 'GeneStructure'),
# sep=':::') %>%
# mutate_at(vars(ExonDistFromTSS,ExonicDistance,ExonSize,TotalNumberOfExons,ExonNumber),
# .funs = as.numeric) identify genes for analysis
# ------------------------------------------------------------------------------------
# print('extract top genes') top_genes<- annoBed %>%
# left_join(expression_vector) %>%
# filter(TotalNumberOfExons > 1 &
# !grepl('snRNA|rRNA|TR_C_gene|IG_C_pseudogene|miRNA|misc_RNA',
# Biotype) & ExonSize > 99 ) %>% select(geneID,
# ctrl_RNAseq_expr) %>% unique() %>%
# mutate(ctrl_RNAseq_expr = as.numeric(ctrl_RNAseq_expr))
# %>% arrange(desc(ctrl_RNAseq_expr)) %>%
# filter(ctrl_RNAseq_expr > 1 & geneID != 'LINC00324')
# calculate number of annotations (must have same filter as
# wrangle bed counts)
# ------------------------------------------------------------
# print('count number of annotations')
# number_of_intron_annotations<- annoBed %>%
# filter(TotalNumberOfExons > 1 &
# !grepl('snRNA|rRNA|TR_C_gene|IG_C_pseudogene|miRNA|misc_RNA',
# Biotype) & ExonSize > 99 ) %>% filter(geneID %in%
# top_genes$geneID) %>% group_by(GeneStructure) %>%
# summarise(intron_count =n()) gene count
# ------------------------------------------------------------------
# print('count genes') GENECOUNT<- annoBed %>%
# filter(geneID %in% top_genes$geneID) %>% select(geneID)
# %>% unique() %>% summarise(geneCount =n()) function to
# process count files
# ------------------------------------------------------------------------------
# wrangle_bed_counts <- function(dataframe) { dataframe %>%
# setNames(c('Sample','chr', 'start', 'end', 'geneID',
# 'DistToLandmark', 'strand', 'count')) %>%
# separate(geneID, into = c('geneID', 'Biotype',
# 'ExonNumber', 'TotalNumberOfExons', 'ExonSize',
# 'ExonicDistance', 'ExonDistFromTSS','ExonStature',
# 'GeneStructure'), sep=':::') %>%
# mutate_at(vars(ExonDistFromTSS,ExonicDistance,ExonSize,TotalNumberOfExons,ExonNumber),
# .funs = as.numeric) %>% \tfilter(!grepl('CBP20_3',
# Sample)) %>% \tleft_join(expression_vector) %>% #this
# replaces NAs introduced by no value present in
# expression_vector, and replaces them with min value in
# expression_vector \tmutate_at(vars(ctrl_RNAseq_expr),
# ~replace(., is.na(.),
# min(expression_vector$ctrl_RNAseq_expr))) %>%
# mutate(norm_count = count/ctrl_RNAseq_expr) %>%
# filter(TotalNumberOfExons > 1 &
```

```
# !grepl('snRNA|rRNA|TR_C_gene|IG_C_pseudogene|miRNA|misc_RNA',
# Biotype) & ExonSize > 99 ) %>% filter(geneID %in%
# top_genes$geneID) %>% group_by(Sample,GeneStructure,
# DistToLandmark) %>%
# summarise(sum_RNAseq_norm_count_norm_annotation_number =
# sum(norm_count)) %>%
# left_join(number_of_intron_annotations) %>%
# mutate(sum_RNAseq_norm_count_norm_annotation_number =
# sum_RNAseq_norm_count_norm_annotation_number/intron_count)
# %>% separate(Sample,
# c('Protein','Rep','Timepoint','readType'), sep = '_') %>%
# filter(Timepoint != 'negative') %>% mutate(Timepoint_f =
# case_when( Timepoint == 'PBSDRB' ~ 't00', TRUE ~
# Timepoint )) %>% mutate(region = factor(region, levels =
# c('5end','3end')), Timepoint = factor(Timepoint, levels =
# c('PBSDRB', 't00', 't05', 't10', 't15', 't20', 't40',
# 't60', 'DMSO')), Timepoint_f = factor(Timepoint_f, levels
# = c('t00', 't05', 't10', 't15', 't20', 't40', 't60',
# 'DMSO')), gene_n = GENECOUNT$geneCount ) } load 5end
# count file ----------------- print('load bed counts for
# 3' end')
# int_exon_junction_coverage_five_end<-read.table(IN_FILE_PATH_5_END,
# sep = '\t', header = F) print('wrangle bed counts for 3'
# end') int_exon_junction_coverage_five_end<-
# int_exon_junction_coverage_five_end %>%
# wrangle_bed_counts() %>% mutate(Position = '5end') load
# 3end count file ----------------- print('load bed counts
# for 5' end')
# int_exon_junction_coverage_three_end<-read.table(IN_FILE_PATH_3_END,
# sep = '\t', header = F) print('wrangle bed counts for 5'
# end') int_exon_junction_coverage_three_end <-
# int_exon_junction_coverage_three_end %>%
# wrangle_bed_counts() %>% mutate(Position = '3end') rbind
# the files together and write to output filename
# OUTFILE<-rbind(int_exon_junction_coverage_three_end,int_exon_junction_coverage_five_end)
# write.table(OUTFILE, OUTFILE_PATH, quote = F, sep = '\t',
# row.names = F)
```

Make graph for Sup Fig3 B

```
annotate_rect <- data.frame(XMIN = c(0, -Inf), XMAX = c(Inf,
    0), YMAX = c(Inf, Inf), YMIN = c(-Inf, -Inf), region = c("5end",
    "3end"))

# wrangle data table
DF <- read.table(COUNTS, header = F) %>%
    setNames(c("Protein", "Rep", "Timepoint", "readType", "rem",
        "geneDescription", "DistToLandmark", "meanCoverage",
        "nAnno", "Timepoint_f", "nANNO2", "region")) %>%
    select(-rem, -nANNO2) %>%
    separate(geneDescription, into = c("TU", "ExonPosition"),
        sep = "-") %>%
    mutate_at(vars(DistToLandmark, meanCoverage, nAnno), .funs = as.numeric) %>%
    mutate(region = factor(region, levels = c("5end", "3end")),
```

```r
        Timepoint = factor(Timepoint, levels = c("PBSDRB", "t00",
            "t05", "t10", "t15", "t20", "t40", "t60", "DMSO")),
        Timepoint_f = factor(Timepoint_f, levels = c("t00", "t05",
            "t10", "t15", "t20", "t40", "t60", "DMSO")), GeneStructure = factor(ExonPosition,
            levels = c("first", "internal", "last"))) %>%
    filter(!(Protein == "CBP20" & Rep == "3")) %>%
    filter(TU == "multiExonicGene" & Timepoint != "negative") %>%
    filter(Protein %in% c("ALYREF", "CBP20", "CBP80"))
```

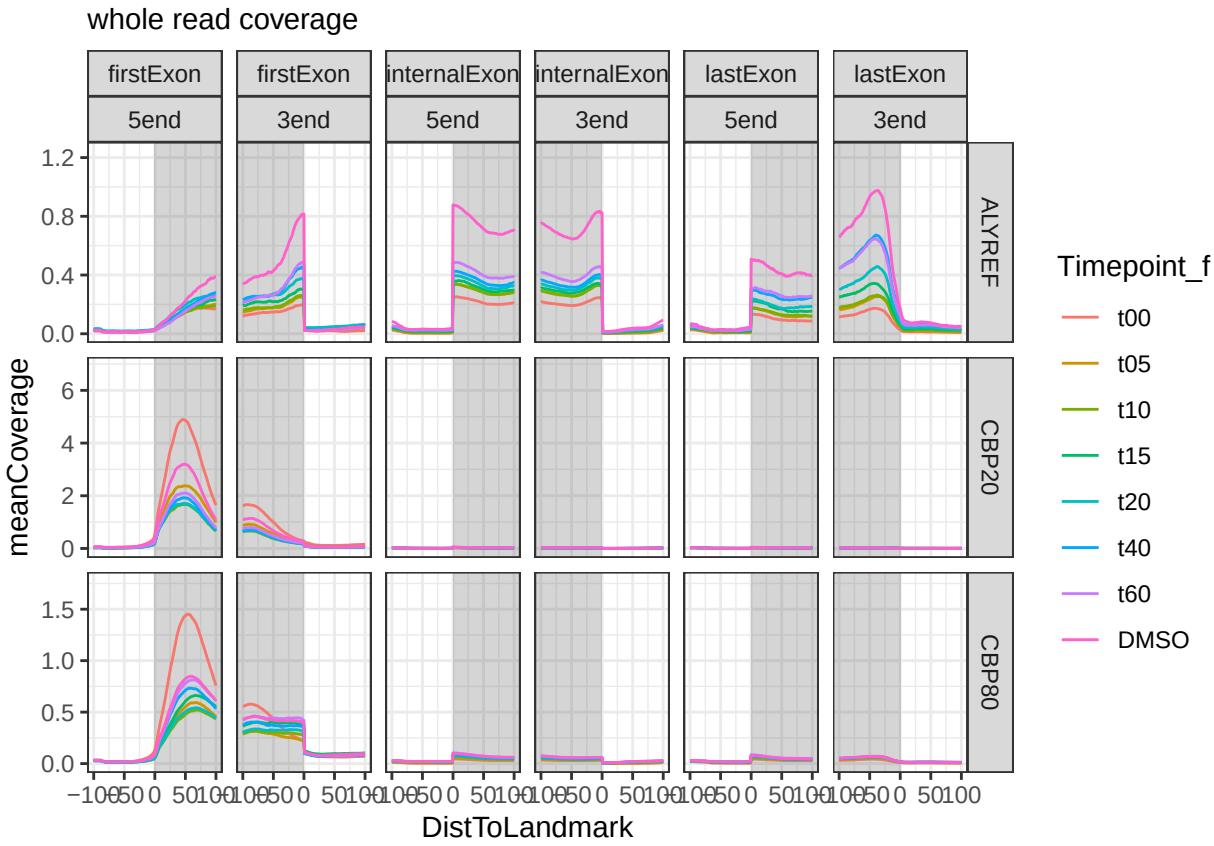
```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 41808 rows [1, 2,
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```r
head(DF)
```

```
##   Protein Rep Timepoint readType              TU ExonPosition DistToLandmark
## 1  ALYREF   1      DMSO    read2 multiExonicGene    firstExon           -100
## 2  ALYREF   1      DMSO    read2 multiExonicGene    firstExon            -99
## 3  ALYREF   1      DMSO    read2 multiExonicGene    firstExon            -98
## 4  ALYREF   1      DMSO    read2 multiExonicGene    firstExon            -97
## 5  ALYREF   1      DMSO    read2 multiExonicGene    firstExon            -96
## 6  ALYREF   1      DMSO    read2 multiExonicGene    firstExon            -95
##   meanCoverage nAnno Timepoint_f region GeneStructure
## 1    0.2746175  9154        DMSO   3end          <NA>
## 2    0.2744483  9154        DMSO   3end          <NA>
## 3    0.2763815  9154        DMSO   3end          <NA>
## 4    0.2789950  9154        DMSO   3end          <NA>
## 5    0.2799805  9154        DMSO   3end          <NA>
## 6    0.2806658  9154        DMSO   3end          <NA>
```

```r
DF %>%
    ggplot() + geom_rect(data = annotate_rect, aes(xmin = XMIN,
    xmax = XMAX, ymin = YMIN, ymax = YMAX), alpha = 0.25) + geom_line(aes(x = DistToLandmark,
    y = meanCoverage, col = Timepoint_f), stat = "summary") +
    facet_grid(Protein ~ ExonPosition + region, scale = "free") +
    theme_bw() + labs(subtitle = "whole read coverage")
```

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


whole read coverage
firstExon
5end
firstExon
3end
internalExon
5end
internalExon
3end
lastExon
5end
lastExon
3end
ALYREF
CBP20
CBP80
meanCoverage
DistToLandmark
Timepoint_f
t00
t05
t10
t15
t20
t40
t60
DMSO