Figure 3 H-I & Supp Fig. 3 D

RAC

24/08/2020

```
knitr::opts_chunk$set(tidy.opts=list(width.cutoff=60), tidy=TRUE)
suppressMessages(library(dplyr))
suppressMessages(library(tidyr))
library(ggplot2)
COUNTS="../../data/xiCLIP.3endOfRead2.rRNAScaled.hg38_HeLa_trimmed_loci_major_primary_isoform_annotated
EXPRESSION_VECTOR_FILEPATH="../../data/log2_mean_cov_RNAseq_TTseq.RData"
ANNOTATION_BED_FILEPATH="../../data/hg38_HeLa_trimmed_loci_major_primary_isoform_annotated.exonNumber.s
```

```
# Load expression vector
# ----------------------------------

load(EXPRESSION_VECTOR_FILEPATH)

expression_vector <- left_join((as.data.frame(ctrl_RNAseq_expr) %>%
    add_rownames(var = "geneID")), (as.data.frame(ctrl_TTseq_expr) %>%
    add_rownames(var = "geneID"))) %>%
    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 = "geneID"
```

```
# load annobed ----------------------

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)

# load count file -----------------

counts <- read.table(COUNTS, sep = "\t", header = F) %>%
    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) %>%
```

```
    filter(!grepl("CBP20_3", Sample))
```

#calculate the CLIP coverage at nucleotide resolution in 201nt windows centred around the 3' end of the first exons.

```
# normalise data to gene expression

norm_counts_to_gene_expression <- counts %>%
    left_join(expression_vector) %>%
    # this replaces NAs introduced by no value present in
    # expression_vector, and replaces them with min value
    # in expression_vector
mutate_at(vars(ctrl_RNAseq_expr), ~replace(., is.na(.), min(expression_vector$ctrl_RNAseq_expr))) %>%
    mutate(norm_count = count/ctrl_RNAseq_expr)
```

```
## Joining, by = "geneID"
```

```
# filter annotation for exon bed file for genes that are
# expressed, exons > 99nt, and remove LINC00324 and other
# biotypes not of interest
expressed_genes_of_interest <- annoBed %>%
    right_join(expression_vector) %>%
    filter(!grepl("snRNA|rRNA|TR_C_gene|IG_C_pseudogene|miRNA|misc_RNA",
        Biotype) & as.numeric(ExonSize) > 99) %>%
    select(geneID, ctrl_RNAseq_expr, ctrl_TTseq_expr) %>%
    unique() %>%
    mutate(ctrl_RNAseq_expr = as.numeric(ctrl_RNAseq_expr)) %>%
    filter(ctrl_RNAseq_expr > 1 & geneID != "LINC00324")
```

```
## Joining, by = "geneID"
```

```
# count number of genes identified from filtered bed above.
# count exons based on position, i.e. first, last, internal
number_of_exon_annotations <- annoBed %>%
    filter(geneID %in% expressed_genes_of_interest$geneID) %>%
    group_by(GeneStructure) %>%
    summarise(exon_count = n())

# count genes identified from filtered exon bed
GENECOUNT <- annoBed %>%
    filter(geneID %in% expressed_genes_of_interest$geneID) %>%
    select(geneID) %>%
    unique() %>%
    summarise(geneCount = n())
```

#process data for figure #select counts from expressed genes, #aggrigate counts for each nucleotide, and normalise by the total number of exon annotations #wrangle data labels for ggplot

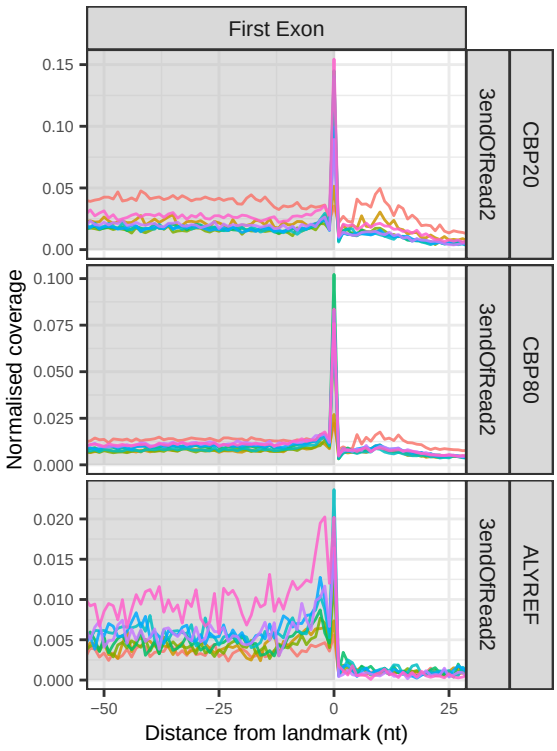
```
for_fig <- norm_counts_to_gene_expression %>%
    filter(geneID %in% expressed_genes_of_interest$geneID & ExonNumber ==
        1 & TotalNumberOfExons > 1) %>%
    group_by(Sample, GeneStructure, DistToLandmark) %>%
    summarise(sum_RNAseq_norm_count_norm_annotation_number = sum(norm_count)) %>%
    left_join(number_of_exon_annotations) %>%
    mutate(sum_RNAseq_norm_count_norm_annotation_number = sum_RNAseq_norm_count_norm_annotation_number/
    # wrangle
```

```
separate(Sample, c("Protein", "Rep", "Timepoint", "readType",
    "region"), sep = "_") %>%
    filter() %>%
    mutate(Timepoint_f = case_when(Timepoint == "PBSDRB" ~ "t00",
        TRUE ~ Timepoint)) %>%
    mutate(region = factor(region, levels = c("5end", "3end")),
        Timepoint_f = factor(Timepoint_f, levels = c("t00", "t05",
            "t10", "t15", "t20", "t40", "t60", "DMSO")), Protein_f = factor(Protein,
            levels = c("CBP20", "CBP80", "ALYREF"))) %>%
    filter(Protein %in% c("CBP20", "CBP80", "ALYREF") & region ==
        "3end" & Timepoint != "negative") %>%
    mutate_at("GeneStructure", ~replace(., GeneStructure == "multiExonicGene-firstExon",
        "First Exon"), "readType", ~replace(., readType == "3endOfRead2",
        "3'CLIP"))
```

```
## `summarise()` has grouped output by 'Sample', 'GeneStructure'. You can override
## using the `.groups` argument.
## Joining, by = "GeneStructure"
```

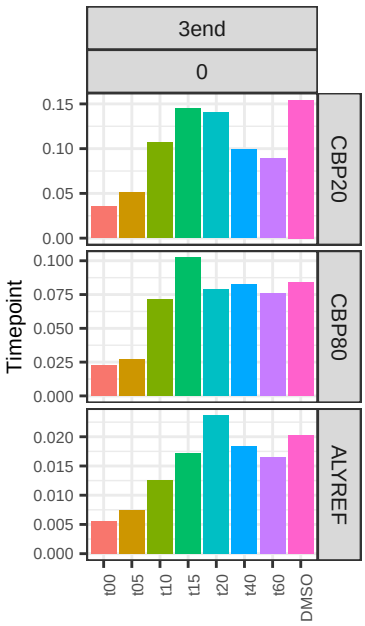
#Figure 3H Coverage plot showing 3'CLIP reads arount 201nt window over the 3' end of the first exon.

```
fig3h <- for_fig %>%
    filter(Protein %in% c("CBP20", "CBP80", "ALYREF") & region ==
        "3end" & GeneStructure == "First Exon") %>%
    ggplot() + geom_rect(data = data.frame(region = "3end"),
    aes(xmin = -100, xmax = 0, ymin = 0, ymax = Inf), alpha = 0.5,
    fill = "grey") + geom_line(aes(x = DistToLandmark, y = sum_RNAseq_norm_count_norm_annotation_number
    col = Timepoint_f), stat = "summary", fun = "mean", alpha = 0.85,
    size = 0.5) + facet_grid(Protein_f + readType ~ GeneStructure,
    scales = "free") + xlab("") + ylab("") + theme_bw() + theme(text = element_text(size = 8),
    legend.position = "none", panel.spacing = unit(0.15, "lines"),
    strip.text.x = element_text(size = 8), strip.text.y = element_text(size = 8)) +
    ylab("Normalised coverage") + xlab("Distance from landmark (nt)") +
    coord_cartesian(xlim = c(-50, 25))


fig3h
```



#Figure 3 I bar plot of induvidual timepoint data at 0nt on x axis from Figure 3H.

```
fig3i <- for_fig %>%
    filter(Protein %in% c("CBP20", "CBP80", "ALYREF") & region ==
        "3end" & GeneStructure == "First Exon" & DistToLandmark ==
        0) %>%
    mutate(zone = case_when(region == "3end" & DistToLandmark ==
        0 ~ "0")) %>%
    ggplot() + geom_bar(aes(x = Timepoint_f, y = sum_RNAseq_norm_count_norm_annotation_number,
    fill = Timepoint_f), stat = "summary", fun = "mean") + facet_grid(Protein_f ~
    region + zone, scale = "free") + xlab("") + ylab("") + theme_bw() +
    theme(axis.text.x = element_text(angle = 90, hjust = 1),
        text = element_text(size = 8), legend.position = "none",
        panel.spacing = unit(0.15, "lines"), strip.text.x = element_text(size = 8),
        strip.text.y = element_text(size = 8)) + ylab("Timepoint") +
    xlab("")

fig3i
```



#Sup Figure 3d Coverage plot showing 3'CLIP reads arount the last 10 nt from the 3' end of exon 1 and the first 10 nt from the 5' end of exon 2. For this plot exon 1 and 2 has been stiched together and the intervening intron has beeen removed.

```
#normalise data to gene expression
norm_counts_to_gene_expression<-
  counts %>%
    left_join(expression_vector) %>%
    #this replaces NAs introduced by no value present in expression_vector, and replaces them with min
    mutate_at(vars(ctrl_RNAseq_expr), ~replace(., is.na(.), min(expression_vector$ctrl_RNAseq_expr))) %
    mutate(norm_count = count/ctrl_RNAseq_expr)
```

```
## Joining, by = "geneID"
```

```
#filter annotation for exon bed file for genes that are expressed, exons > 99nt, and remove LINC00324 a
expressed_genes_of_interest<-
annoBed %>%
  right_join(expression_vector) %>%
  filter(!grepl("snRNA|rRNA|TR_C_gene|IG_C_pseudogene|miRNA|misc_RNA", Biotype)
         & as.numeric(ExonSize) > 25
         & TotalNumberOfExons > 1
         & ExonNumber %in% c(1:2)
         ) %>%
  group_by(geneID) %>%
  filter(n() > 1) %>%
  ungroup() %>%
  select(geneID, ctrl_RNAseq_expr, ctrl_TTseq_expr) %>%
  unique() %>%
  mutate(ctrl_RNAseq_expr = as.numeric(ctrl_RNAseq_expr)) %>%
  filter(ctrl_RNAseq_expr > 1
         & geneID != "LINC00324" )
```

```
## Joining, by = "geneID"
```

```
#count number of genes identified from filtered bed above.
#count exons based on EXON NUMBER, not position. Number of exon 1 == exon 2 number
```

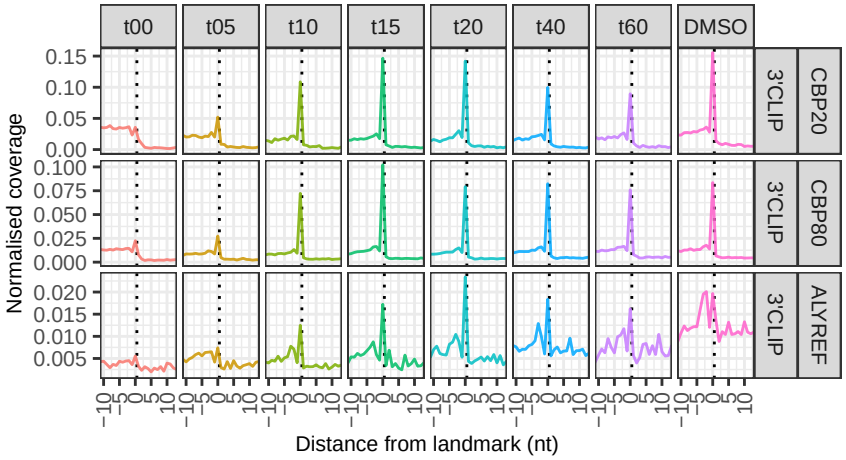
```
number_of_exon_annotations<-
  annoBed %>%
  filter(geneID %in% expressed_genes_of_interest$geneID) %>%
  group_by(ExonNumber) %>%
  summarise(exon_count =n())


#count genes identified from filtered exon bed
GENECOUNT<-
  annoBed %>%
  filter(geneID %in% expressed_genes_of_interest$geneID) %>%
  select(geneID) %>%
  unique() %>%
  summarise(geneCount =n())


for_fig<-
norm_counts_to_gene_expression %>%
  filter(geneID %in% expressed_genes_of_interest$geneID
         & ExonNumber %in% c(1:2)
         & TotalNumberOfExons > 1) %>%
  group_by(Sample, GeneStructure,  ExonNumber, DistToLandmark) %>%

  summarise(sum_RNAseq_norm_count_norm_annotation_number = sum(norm_count)) %>%
  left_join(number_of_exon_annotations) %>%
  mutate(sum_RNAseq_norm_count_norm_annotation_number = sum_RNAseq_norm_count_norm_annotation_number / c
  #wrangle
  separate(Sample, c("Protein", "Rep", "Timepoint", "readType", "region"), sep = "_") %>%
  filter() %>%
  mutate(Timepoint_f = case_when(Timepoint == "PBSDRB" ~ "t00", TRUE ~ Timepoint)) %>%
  mutate(
      region = factor(region, levels = c("5end", "3end")),
      Timepoint_f = factor(Timepoint_f, levels = c("t00", "t05", "t10", "t15", "t20", "t40", "t60", "DMS
      Protein_f = factor(Protein, levels = c("CBP20", "CBP80", "ALYREF")))   %>%
  filter(Protein %in% c("CBP20", "CBP80", "ALYREF")
         #& region == "3end"
         & Timepoint != "negative"
         ) %>%
  mutate_at("GeneStructure", ~replace(., GeneStructure == "multiExonicGene-firstExon", "First Exon")) %>
  mutate_at("readType", ~replace(.,readType == "3endOfRead2", "3'CLIP"))
```

```
## `summarise()` has grouped output by 'Sample', 'GeneStructure', 'ExonNumber'.
## You can override using the `.groups` argument.
## Joining, by = "ExonNumber"
```

```
supFig3d<-
for_fig %>%
  filter(
    (ExonNumber == 1 & DistToLandmark %in% c(-100:0) & region == "3end" )|
    (ExonNumber == 2 & DistToLandmark %in% c(0:100) & region == "5end" )
     ) %>%
  mutate(DistToLandmark = ifelse (ExonNumber == 2, DistToLandmark + 1, DistToLandmark)) %>%
filter(
  Protein %in% c("CBP20", "CBP80", "ALYREF")
```

```
   ) %>%
ggplot() +
  geom_vline(xintercept = 0.5, linetype = "dotted") +
  geom_line(
    aes(x = DistToLandmark, y = sum_RNAseq_norm_count_norm_annotation_number, col = Timepoint_f),
    stat = "summary",
    fun = "mean",
    alpha = 0.85,
    size = 0.5
  ) +
  facet_grid( Protein_f + readType ~ Timepoint_f, scales = "free_y") +
  xlab("") +
  ylab("") +
  theme_bw() +
  theme(
    text = element_text(size = 8),
    axis.title = element_text(size = 8),
    axis.text.y = element_text(size = 8),
    axis.text.x = element_text(size = 8, angle = 90, vjust = 0.5, hjust=1),
    legend.position = "none",
    panel.spacing = unit(0.15, "lines"),
    strip.text.x = element_text(size = 8),
    strip.text.y = element_text(size = 8)
  ) +
  ylab("Normalised coverage") +
  xlab("Distance from landmark (nt)") +
  coord_cartesian(xlim = c(-10,12))
```

supFig3d




```
#  ggsave("Supplemental_Fig_3d.pdf", width = 6, height = 3)
```