Supplementary Figure 5E

RAC

10/06/2021

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
#
DF<-read.table("../../data/xiCLIP_all.rel_dist_cigar_3refNT_3readNT_5SS.tab", header = F)


colnames(DF)<-c("Sample","readID", "chr", "read_start", "read_end", "mapQ", "strand", "geneID", "relDis

DF <- filter(DF, !(grepl("CBP20_3", Sample))) %>% unique()

#remove duplicates which could be used
DF<-distinct(DF, readID, .keep_all = TRUE)
```

##Supplementary Figure 5 E #plot cDNA 5' ends around 5'SS. Data grouped based on 5' end of cDNA mutation status

```
annotated_DF<-
DF %>%
  unique() %>%
  mutate(splicing_status = case_when(
    grepl("N",CIGAR) ~ "spliced",
    !grepl("N",CIGAR) ~ "not_spliced"
  )) %>%
  #unite("mutation", refNT, readNT, sep = ">", remove = F) %>%
  gather("readpos","nt",c(refNT_1:readNT_3)) %>%
  mutate(nt = case_when(
    grepl("readNT", readpos) & nt == "A" & strand == '-' ~ "T",
    grepl("readNT", readpos) & nt == "T" & strand == '-' ~ "A",
    grepl("readNT", readpos) & nt == "G" & strand == '-' ~ "C",
    grepl("readNT", readpos) & nt == "C" & strand == '-' ~ "G",
    TRUE ~ nt
  )) %>%
  unique() %>%
  separate(readpos, c("read","readpos"),sep ="NT_") %>%
  spread(read,nt) %>%
  mutate(mapping = case_when(
    read == ref ~ "correct",
    read != ref ~ "mismatch"
```

```
  ))

rRNAFactor<-
read.table("../../data/rRNAFactor.tab") %>%
  setNames(c("Sample", "factor")) %>%
    separate(Sample, into=c("Protein","Rep","Timepoint"))

annotated_wrangled_DF_n<-
annotated_DF %>%
  separate(Sample, into=c("Protein","Rep","Timepoint")) %>%
  mutate(Timepoint = factor(Timepoint, levels=c("negative","PBSDRB","t00","t05", "t10", "t15", "t20", "
  group_by(Protein, Rep, Timepoint, relDist, splicing_status, readpos, mapping) %>%
  summarise(count = n()) %>%
  left_join(rRNAFactor) %>%
  mutate(count_rRNAn = count * factor)
```

```
## `summarise()` has grouped output by 'Protein', 'Rep', 'Timepoint', 'relDist',
## 'splicing_status', 'readpos'. You can override using the `.groups` argument.
## Joining, by = c("Protein", "Rep", "Timepoint")
```

```
#linegraph - excluded spliced reads

plot<-
annotated_wrangled_DF_n %>%
filter(Protein == "RBM7" & readpos == 1 & Timepoint == "DMSO" & splicing_status == "not_spliced") %>%
  mutate(Timepoint = factor(Timepoint, levels=c("negative","PBSDRB","t00","t05", "t10", "t15", "t20", "
  ggplot() +
  stat_summary(aes(x=relDist, y=count_rRNAn, group = mapping),fun.data = mean_cl_boot, geom = "ribbon",
  stat_summary(aes(x=relDist, y=count_rRNAn, col = mapping), geom="line", fun=mean ) +
  facet_grid(Protein~Timepoint) +
  theme_bw() +
  scale_x_continuous(breaks = c(-10,-5,0,+5,+10),
                     labels = c(-10,-5,"5'SS",+5,+10)
                     ) +
  xlab("Relative distance of cDNA to 5'SS") +
  ylab("Coverage") +
  theme(text = element_text(size = 8))

print(plot)
```