Figure S5 Reporter-ChIP-nexus recapitulates gene-specific Pol II pausing stability

Wanqing Shao(was@stowers.org)

Contents

Description

To test if gene-specific paused Pol II stability is also recapitulated on the reporter, we treated the transfected Kc167 cell and *Drosophila pseudoobscura* ML83-63 cells with DMSO or Triptolide (TRI). TRI blocks transcription initiation, leading to the loss of Pol II signal at the pausing position. The degree of Pol II loss at the pausing position is propotional to the stability of paused Pol II.

Enviroment setup

```
library(GenomicRanges, warn.conflicts=F)
library(magrittr)

setwd("/data/analysis_code")
options(knitr.figure_dir =
          "FigureS5_reporter_chipnexus_recapitulates_gene_specific_polii_pausing_stability"
        )

source("shared_code/knitr_common.r")
source("shared_code/ggplot_common.r")
source("shared_code/granges_common.r")
source("shared_code/metapeak_common.r")
source("shared_code/sample_common.r")
```

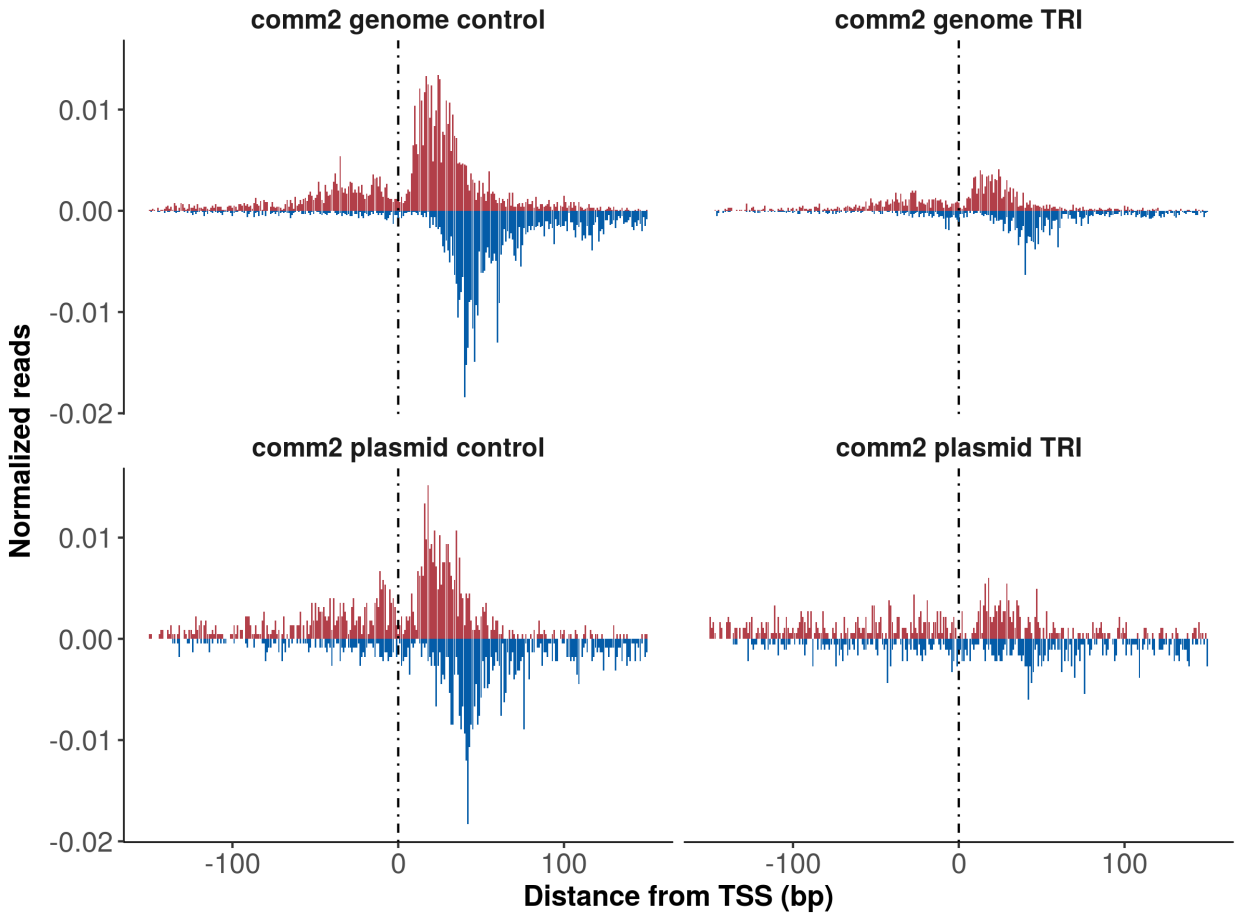
Analysis

Reporter-ChIP-nexus recapitulates gene-specific Pol II pausing stability

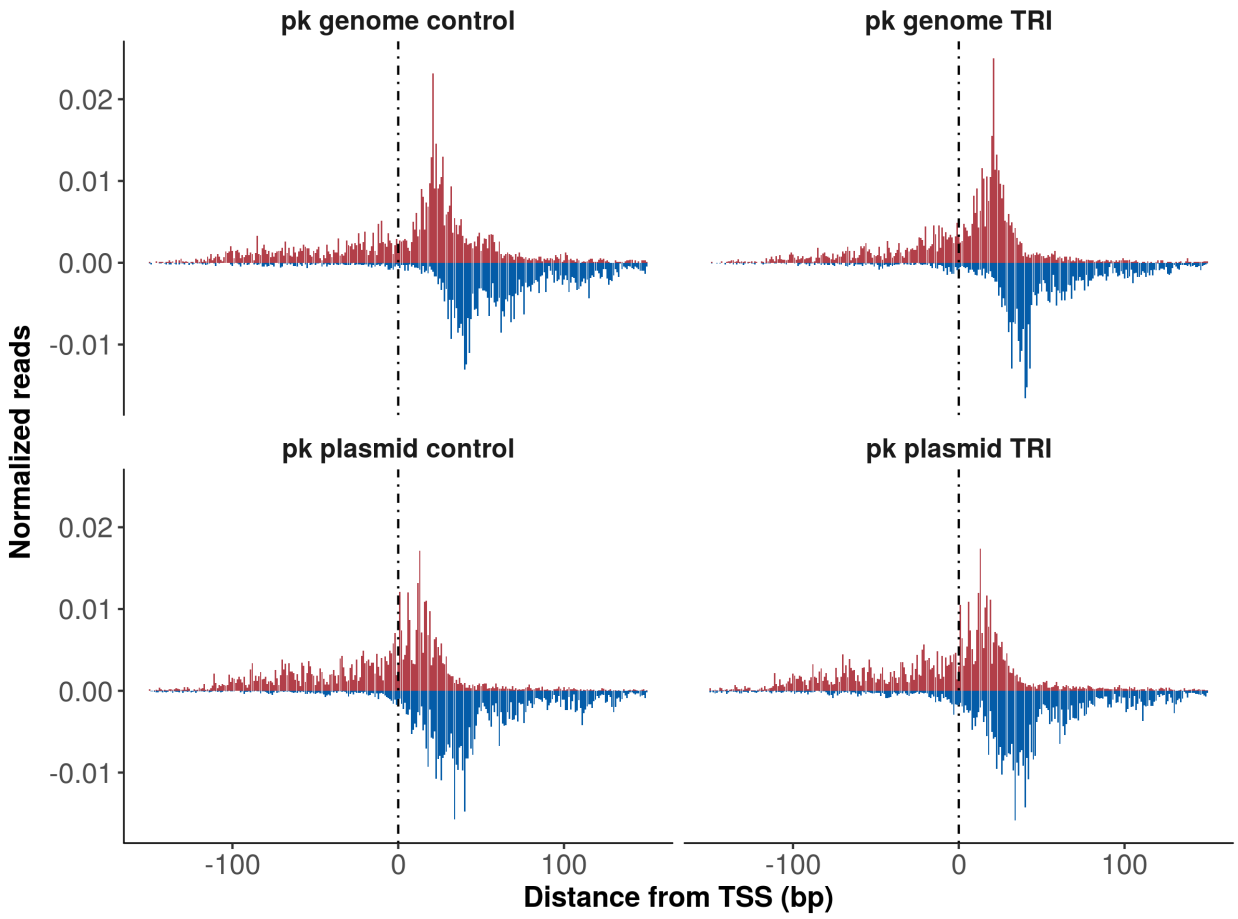
```
plasmid_annotations <- import("./plasmid_annotation.bed")
genome_annotations <- import("./dps_genome_annotation.bed")
```

```
dps_dmso_path <- load_bigwig("genome_dps_dmso_1h_rpb3_chipnexus")
dps_tri_path <- load_bigwig("genome_dps_triptolide_1h_rpb3_chipnexus")

get_exo_metapeak <- function(sample, upstream=100, downstream=101,
                             smooth=NA, endogeneous = F, dps_sample_path=NULL){

  gene <- as.character(subset(sample_list, sample_name == sample)$gene)
    chromosome <- as.character(subset(sample_list, sample_name == sample )$chromosome)

    if(endogeneous ==F){

        sample_path <- load_bigwig(sample)
        region <- plasmid_annotations[seqnames(plasmid_annotations) == chromosome &
                                      plasmid_annotations$name == gene] %>%
                  resize(., 1, "start")
        seqlevels(region) <- chromosome
      metapeak <- exo_metapeak(region, sample_path,
                               upstream=upstream, downstream=downstream,
                               sample_name=gene, smooth=smooth)
      metapeak$sample <- paste(metapeak$sample_name, metapeak$strand)
      metapeak

    }else{

      region <- genome_annotations[grep(gene, genome_annotations$name, ignore.case = T)]
      seqlevels(region) <- as.character(seqnames(region))
      metapeak <- exo_metapeak(region, dps_sample_path,
                               upstream=upstream, downstream=downstream,
                               sample_name=gene, smooth=smooth)
      metapeak$sample <- paste(metapeak$sample_name, metapeak$strand)
      metapeak

    }
}

compare_dmso_and_tri <- function(dmso, tri){

  plasmid_dmso <- get_exo_metapeak(dmso, upstream=150, downstream = 151)
  plasmid_tri <- get_exo_metapeak(tri, upstream=150, downstream = 151)

  plasmid_tri$reads <- plasmid_tri$reads / sum(abs(plasmid_dmso$reads))
  plasmid_dmso$reads <- plasmid_dmso$reads / sum(abs(plasmid_dmso$reads))

  genome_dmso <- get_exo_metapeak(dmso, upstream=150, downstream = 151,
                                  endogeneous = T, dps_sample_path = dps_dmso_path)
  genome_tri <- get_exo_metapeak(tri, upstream=150, downstream = 151,
                                 endogeneous = T, dps_sample_path = dps_tri_path)

  genome_tri$reads <- genome_tri$reads / sum(abs(genome_dmso$reads))
  genome_dmso$reads <- genome_dmso$reads / sum(abs(genome_dmso$reads))

  plasmid_dmso$sample_name <- paste(plasmid_dmso$sample_name, "plasmid control")
  plasmid_tri$sample_name <- paste(plasmid_tri$sample_name, "plasmid TRI")
```

```
  genome_dmso$sample_name <- paste(genome_dmso$sample_name, "genome control")
  genome_tri$sample_name <- paste(genome_tri$sample_name, "genome TRI")

  metapeak_c <- rbind(plasmid_dmso, plasmid_tri, genome_dmso, genome_tri)

  plot <- ggplot(subset(metapeak_c, strand == "+"),
                 aes(x = tss_distance, y = reads, fill = strand))+
          geom_bar(fill="#B23F49", stat="identity") +
          geom_vline(xintercept =0, linetype = "dotdash")+
          geom_bar(data=subset(metapeak_c, strand == "-"),
                   aes(x=tss_distance, y=reads), fill="#045CA8", stat="identity")+
          xlab("Distance from TSS (bp)")+ ylab("Normalized reads")+
          facet_wrap(facets = "sample_name", ncol =2 ) +
          theme(axis.text.x = element_text(size=13),
                axis.text.y = element_text(size=13),
                axis.title.x   = element_text(size=13),
                axis.title.y   = element_text(size=13))
  print(plot)
}

compare_dmso_and_tri("reporter_dmso_1h_dps_comm2_rpb3_chipnexus",
                     "reporter_triptolide_1h_dps_comm2_rpb3_chipnexus")
```



```
compare_dmso_and_tri("reporter_dmso_1h_dps_pk_rpb3_chipnexus",
                     "reporter_triptolide_1h_dps_pk_rpb3_chipnexus")
```



Session Info

This analysis was performed with the following R/Bioconductor session:

```
sessionInfo()
```

```
## R version 3.4.4 (2018-03-15)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.4 LTS
##
## Matrix products: default
## BLAS: /usr/lib/libblas/libblas.so.3.6.0
## LAPACK: /usr/lib/lapack/liblapack.so.3.6.0
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
```

```
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] parallel  stats4    stats     graphics  grDevices utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] lattice_0.20-35      reshape2_1.4.3       rtracklayer_1.38.3
##  [4] ggplot2_2.2.1        pander_0.6.1         magrittr_1.5
##  [7] GenomicRanges_1.30.3 GenomeInfoDb_1.14.0  IRanges_2.12.0
## [10] S4Vectors_0.16.0     BiocGenerics_0.24.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.17               compiler_3.4.4
##  [3] pillar_1.2.3               plyr_1.8.4
##  [5] XVector_0.18.0             bitops_1.0-6
##  [7] tools_3.4.4                zlibbioc_1.24.0
##  [9] digest_0.6.15              evaluate_0.10.1
## [11] tibble_1.4.2               gtable_0.2.0
## [13] rlang_0.2.1                Matrix_1.2-14
## [15] DelayedArray_0.4.1         yaml_2.1.19
## [17] GenomeInfoDbData_1.0.0     stringr_1.3.1
## [19] knitr_1.20                 Biostrings_2.46.0
## [21] rprojroot_1.3-2            grid_3.4.4
## [23] Biobase_2.38.0             XML_3.98-1.11
## [25] BiocParallel_1.12.0        rmarkdown_1.10
## [27] matrixStats_0.53.1         backports_1.1.2
## [29] scales_0.5.0               Rsamtools_1.30.0
## [31] htmltools_0.3.6            GenomicAlignments_1.14.2
## [33] SummarizedExperiment_1.8.1 colorspace_1.3-2
## [35] labeling_0.3               stringi_1.2.3
## [37] RCurl_1.95-4.10            lazyeval_0.2.1
## [39] munsell_0.5.0
```