Figure S5 Larger promoter region insertion is required for recapitulating Pol II pausing at RpL13A on the plasmid

Wanqing Shao(was@stowers.org)

Contents

Description

Promoters with TFIID-bound core promoter elements typically do not have strong +1 promoter nucleosomes and may less likely to be dependent on a natural chromatin context for Pol II pausing. To test the versatility of our assay, we therefore also cloned the promoter of a **Drosophila pseudoobscura** ribosomal gene, **RpL13A**, into our plasmid. This promoter belongs to the group of promoters that uses TCT as initiator element and undergoes focused initiation, but unlike promoters with TFIID-bound core promoter elements, it has a strong +1 nucleosome with high levels of H3K4me3. To test if genomic context is important in establishing Pol II pausing, we built reporter construct with 300 bp or 2 kb insertion and we will compare our result **RpL13A** and **Act5C**.

Enviroment setup

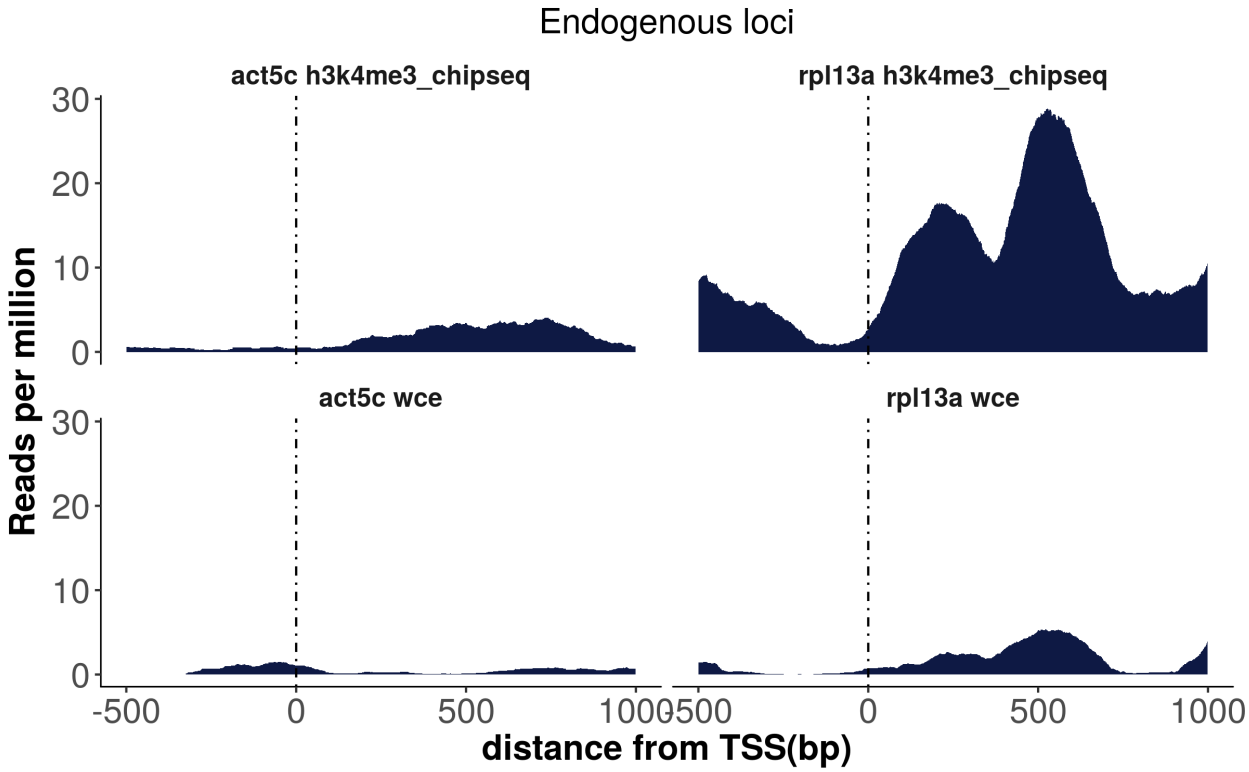
```
library(GenomicRanges, warn.conflicts=F)
library(magrittr)

setwd("/data/analysis_code")
options(knitr.figure_dir =
          "FigureS5_larger_promoter_region_required_for_recapitulating_polii_pausing_at_rpl13a"
        )

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

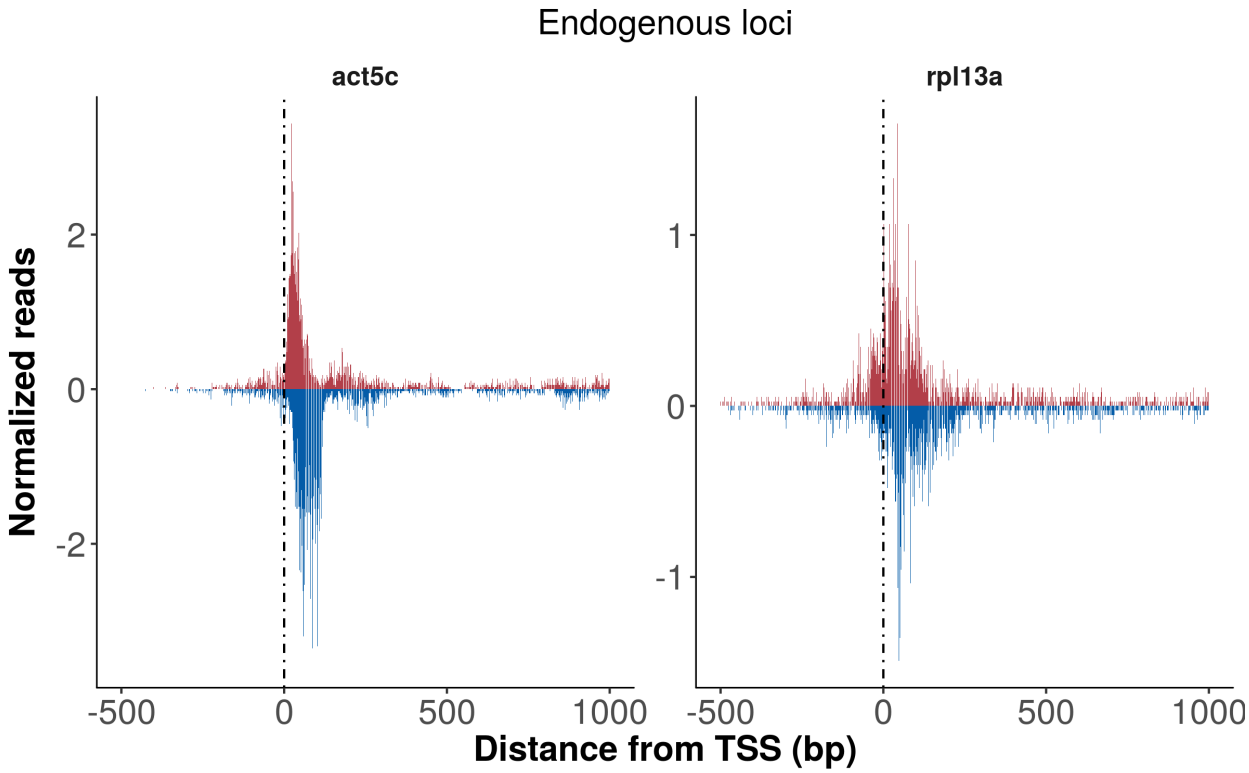
Endogenous profile

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

selected_genes <- c("act5c",  "rpl13a")
chip_samples <- c("h3k4me3_chipseq", "wce")

chipseq_metapeak <-
  mclapply(selected_genes, function(x){
  gene_gr <- genome_annotations[genome_annotations$name == x]
  metapeak <- lapply(chip_samples,
          function(y)standard_metapeak(gene_gr, load_bigwig(paste0("genome_dps_", y))$pos,
                  upstream = 500, downstream = 1000,
                  sample_name = paste(x, y))) %>% do.call(rbind, .)
  metapeak
}, mc.cores =3) %>% do.call(rbind, .)


chipnexus_metapeak <-
  mclapply(selected_genes, function(x){
  gene_gr <- genome_annotations[genome_annotations$name == x]
  metapeak <- exo_metapeak(gene_gr, load_bigwig("genome_dps_dmso_1h_rpb3_chipnexus"),
                           upstream = 500, downstream = 1000, sample_name = x)
  metapeak$sample <- paste(metapeak$sample_name, metapeak$strand)
  metapeak
}, mc.cores =2) %>% do.call(rbind, .)



chipseq_metapeak$sample_name <-
  factor(chipseq_metapeak$sample_name, levels = unique(chipseq_metapeak$sample_name))

chipnexus_metapeak$sample_name <-
  factor(chipnexus_metapeak$sample_name, levels = unique(chipnexus_metapeak$sample_name))

plot_chipseq_single_gene <- function(metapeak, name){
  x <- ggplot(metapeak, aes(x=tss_distance, y=reads)) + geom_area(fill="#0E1944") +
    ggtitle(name) + xlab("distance from TSS(bp)") +
    ylab("Reads per million") +  geom_vline(xintercept=0, linetype=4)+
    facet_wrap("sample_name", nrow =2, dir = "v")
  print(x)
}

plot_chipnexus_single_gene <- function(metapeak, name){
  x <- ggplot(subset(metapeak, strand == "+"),
              aes(x = tss_distance, y = reads, fill = strand))+
    geom_bar(fill="#B23F49", stat="identity") +
    geom_bar(data=subset(metapeak, strand == "-"), aes(x=tss_distance, y=reads),
             fill="#045CA8", stat="identity")+
    ggtitle(name) + xlab("Distance from TSS (bp)")+ ylab("Normalized reads")+
```

```
    facet_wrap(facets = "sample_name", nrow =1, scales = "free_y") +
    geom_vline(xintercept=0, linetype=4)
  print(x)
}


nothing <- plot_chipseq_single_gene(chipseq_metapeak, "Endogenous loci")
```



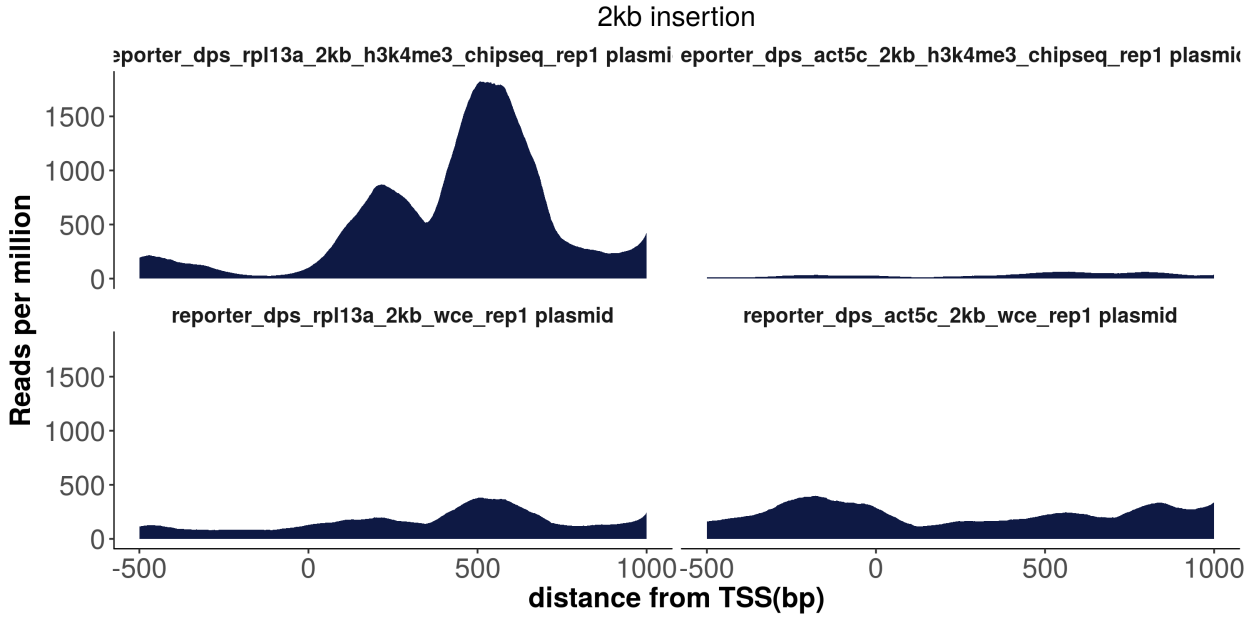

```
nothing <- plot_chipnexus_single_gene(chipnexus_metapeak, "Endogenous loci")
```



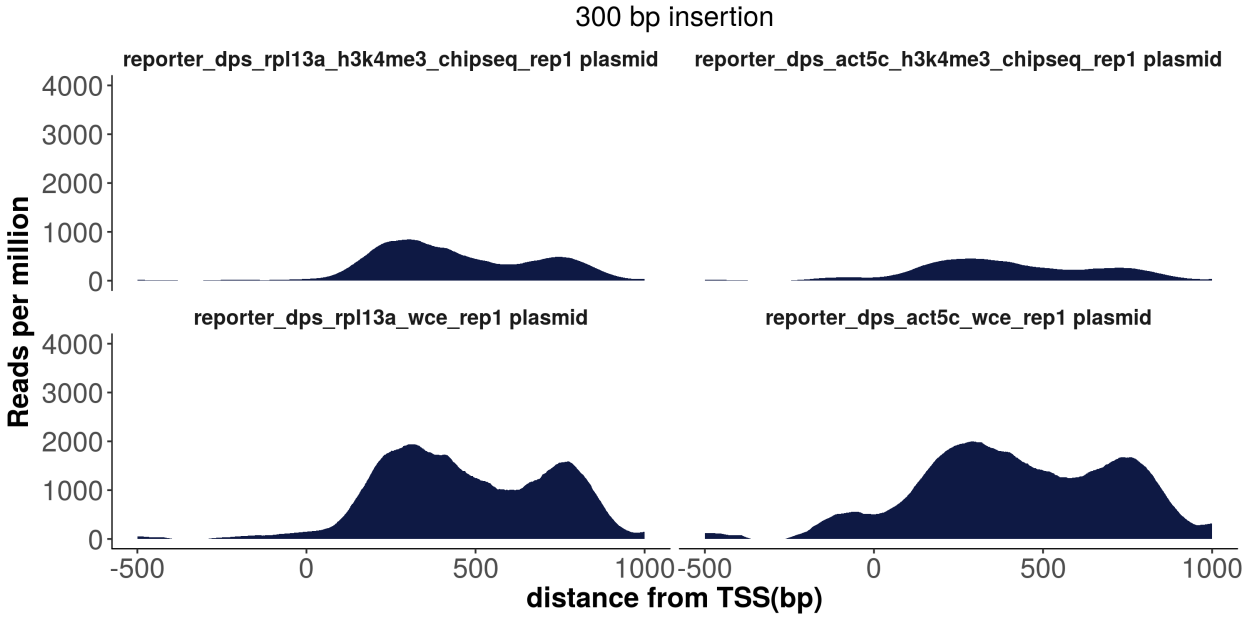
300 bp vs. 2k insertion profile

```
get_exo_metapeak <- function(sample, upstream=100, downstream=101, smooth=NA, gfp_norm=F){
    gene <- as.character(subset(sample_list, sample_name == sample)$gene)
    chromosome <- as.character(subset(sample_list, sample_name == sample )$chromosome)
    sample.path <- load_bigwig(sample)
    region <- resize(plasmid_annotations[seqnames(plasmid_annotations) == chromosome &
                                         plasmid_annotations$name == gene], 1, "start")
    seqlevels(region) <- chromosome
  metapeak <- exo_metapeak(region, sample.path, upstream=upstream, downstream=downstream,
                           sample_name=paste(sample, "plasmid"), smooth=smooth)
  metapeak$sample <- paste(metapeak$sample_name, metapeak$strand)

  if(gfp_norm){
        gfp_gr <- resize(plasmid_annotations[seqnames(plasmid_annotations) == chromosome &
                                             plasmid_annotations$name == gene], 1, "end")
        gfp_gr <- resize(gfp_gr, 701, "end")
        gfp_sig <- nexus_regionSums(gfp_gr,sample.path) / width(gfp_gr)
        metapeak$reads <- metapeak$reads / gfp_sig
  }
  metapeak
}


get_chipseq_metapeak <- function(sample, upstream=100, downstream=101, smooth=NA){
    gene <- as.character(subset(sample_list, sample_name == sample)$gene)
    chromosome <- as.character(subset(sample_list, sample_name == sample )$chromosome)
```

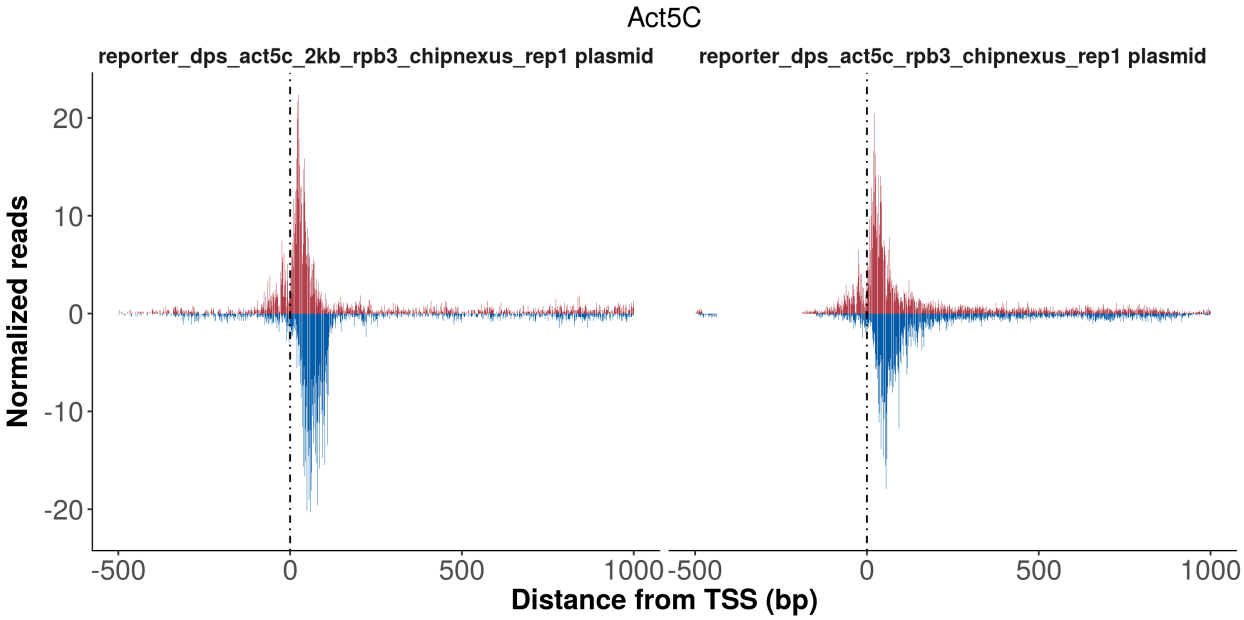
```
    sample.path <- load_bigwig(sample)$pos
    region <- resize(plasmid_annotations[seqnames(plasmid_annotations) == chromosome & plasmid_annotati
    seqlevels(region) <- chromosome
  metapeak <- standard_metapeak(region, sample.path, upstream=upstream, downstream=downstream, sample_na
  metapeak$sample <- paste(metapeak$sample_name, metapeak$strand)
  metapeak
}

histone_modification <-  function(h3k4me3, wce){
  h3k4me3_metapeak <-
    get_chipseq_metapeak(h3k4me3, upstream=500, downstream = 1001,  smooth=NA)
  wce_metapeak <-
    get_chipseq_metapeak(wce, upstream=500, downstream = 1001,  smooth=NA)
  metapeak <- rbind(h3k4me3_metapeak, wce_metapeak)
  metapeak
}

rpl13a_2kb <- histone_modification("reporter_dps_rpl13a_2kb_h3k4me3_chipseq_rep1", "reporter_dps_rpl13a_
act5c_2kb <- histone_modification("reporter_dps_act5c_2kb_h3k4me3_chipseq_rep1", "reporter_dps_act5c_2kb
metapeak_2kb <- rbind(rpl13a_2kb, act5c_2kb)

rpl13a_300 <- histone_modification("reporter_dps_rpl13a_h3k4me3_chipseq_rep1", "reporter_dps_rpl13a_wce_
act5c_300 <- histone_modification("reporter_dps_act5c_h3k4me3_chipseq_rep1", "reporter_dps_act5c_wce_rep
metapeak_300 <- rbind(rpl13a_300, act5c_300)

ggplot(metapeak_2kb, aes(x=tss_distance, y=reads)) +
  geom_area(fill="#0E1944") +
  xlab("distance from TSS(bp)") +
  ylab("Reads per million") +
  ggtitle("2kb insertion")+
  facet_wrap(facets = "sample_name", dir = "v")
```



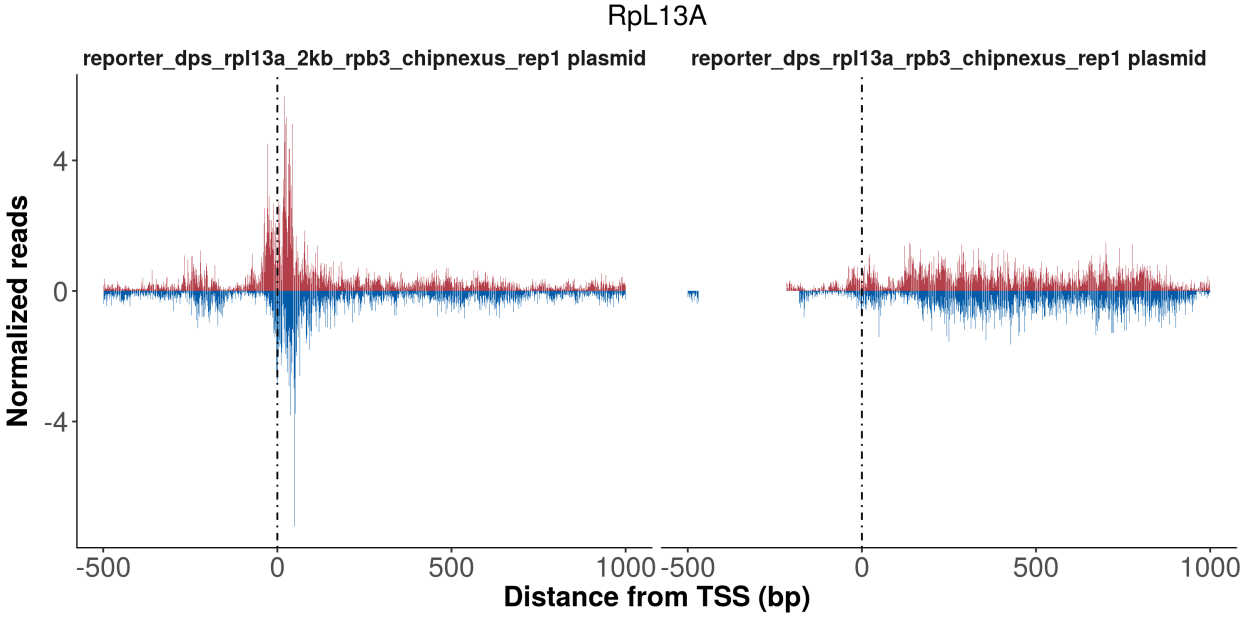
```
ggplot(metapeak_300, aes(x=tss_distance, y=reads)) +
  geom_area(fill="#0E1944") +
  xlab("distance from TSS(bp)") +
  ylab("Reads per million") +
  ggtitle("300 bp insertion")+
  ylim(0, 4000)+
  facet_wrap(facets = "sample_name", dir = "v")
```




```
act5c_chipnexus <- c("reporter_dps_act5c_2kb_rpb3_chipnexus_rep1",
                     "reporter_dps_act5c_rpb3_chipnexus_rep1")
rpl13a_chipnexus <- c("reporter_dps_rpl13a_2kb_rpb3_chipnexus_rep1",
                     "reporter_dps_rpl13a_rpb3_chipnexus_rep1")

chipnexus_metapeak_act5c <- lapply(act5c_chipnexus, function(x)get_exo_metapeak(x, upstream=500,
          downstream = 1001,  smooth=NA, gfp_norm=T)) %>% do.call(rbind, .)

chipnexus_metapeak_rpl13a <- lapply(rpl13a_chipnexus, function(x)get_exo_metapeak(x, upstream=500,
          downstream = 1001,  smooth=NA, gfp_norm=T)) %>% do.call(rbind, .)

plot_chipnexus_single_gene2 <- function(metapeak, name){
  x <- ggplot(subset(metapeak, strand == "+"),
              aes(x = tss_distance, y = reads, fill = strand))+
    geom_bar(fill="#B23F49", stat="identity") +
    geom_bar(data=subset(metapeak, strand == "-"), aes(x=tss_distance, y=reads),
             fill="#045CA8", stat="identity")+
    ggtitle(name) + xlab("Distance from TSS (bp)")+ ylab("Normalized reads")+
    facet_wrap(facets = "sample_name", nrow =1) +
    geom_vline(xintercept=0, linetype=4)
  print(x)
}

plot_chipnexus_single_gene2(chipnexus_metapeak_act5c, "Act5C")
```




```
plot_chipnexus_single_gene2(chipnexus_metapeak_rpl13a, "RpL13A")
```



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
```