Analysis for ZNF10's association with LINE1 repeats

Zheng Zuo

07/2023

Contents

Mapping ChIP-exo signals along ORF2 repeat coordinates

Sequence alignment and liftOver operations

The Makefile used to process raw sequencing data to bedgraph fies on repeat coordinates

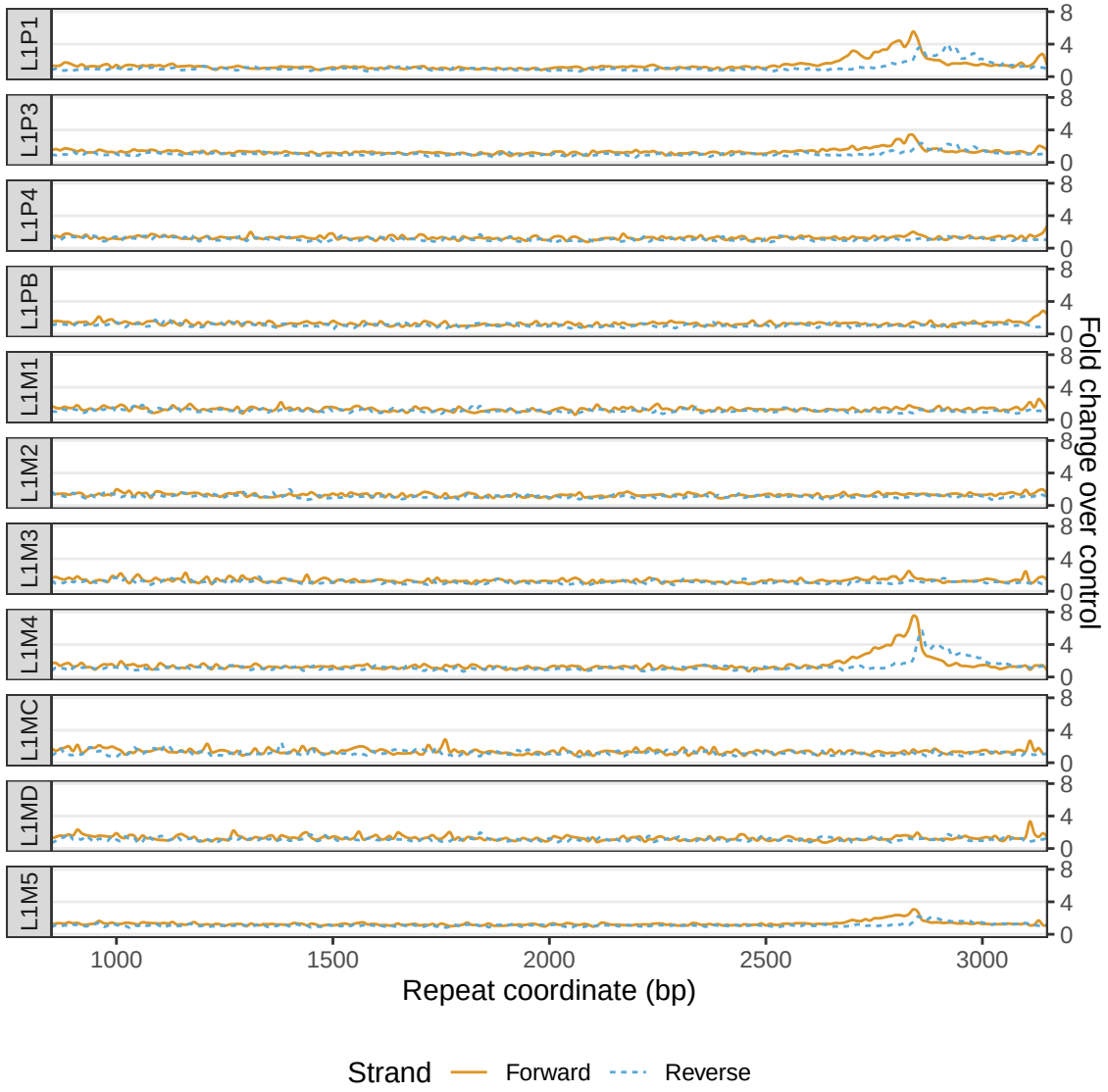
```
aim: ZNF10.Repeat.plus.bedgraph ZNF10.Repeat.minus.bedgraph Control.Repeat.plus.bedgraph Control.Repeat
clean:
        rm *Repeat*
ZNF10.Repeat.plus.bedgraph:ZNF10.Repeat.bed
        bedtools genomecov -i ZNF10.Repeat.bed -g hg38.Repeat.sizes -bg -strand + -5|LC_COLLATE=C sort
ZNF10.Repeat.minus.bedgraph:ZNF10.Repeat.bed
        bedtools genomecov -i ZNF10.Repeat.bed -g hg38.Repeat.sizes -bg -strand - -5|LC_COLLATE=C sort
ZNF10.Repeat.bed:ZNF10.bed
        liftOver ZNF10.bed Hg38ToRepeat.over.chain ZNF10.Repeat.bed ZNF10.noRepeat.bed -minMatch=0.5
        bedtools sort -i ZNF10.Repeat.bed > ZNF10.Repeat.sorted.bed
        mv ZNF10.Repeat.sorted.bed ZNF10.Repeat.bed
        rm ZNF10.noRepeat.bed
ZNF10.bed:SRR5197054.fasta
        bowtie2 -x ../../reference-genomes/hg38/GRCh38_noalt_as --very-sensitive-local -f SRR5197054.fas
        samtools view -bS -o ZNF10.bam ZNF10.sam
        samtools sort ZNF10.bam -o ZNF10.sorted.bam
        samtools index ZNF10.sorted.bam
        bamToBed -i ZNF10.sorted.bam>ZNF10.bed
SRR5197054.fasta:
        fastq-dump --fasta SRR5197054

Control.Repeat.plus.bedgraph:Control.Repeat.bed
        bedtools genomecov -i Control.Repeat.bed -g hg38.repeat.sizes -bg -strand + -5|LC_COLLATE=C sor
Control.Repeat.minus.bedgraph:Control.Repeat.bed
        bedtools genomecov -i Control.Repeat.bed -g hg38.repeat.sizes -bg -strand - -5|LC_COLLATE=C sor
Control.Repeat.bed:Control.bed
```

```
	liftOver Control.bed ../../../Development/TECookbook/Hg38ToRepeat.over.chain Control.Repeat.bed
	bedtools sort -i Control.Repeat.bed > Control.Repeat.sorted.bed
	mv Control.Repeat.sorted.bed Control.Repeat.bed
	rm Control.noRepeat.bed
Control.bed:
	bowtie2 -x ../../reference-genomes/hg38/GRCh38_noalt_as --very-sensitive-local -f SRR5197033.fa
	samtools view -bS -o Control.bam Control.sam
	samtools sort Control.bam -o Control.sorted.bam
	samtools index Control.sorted.bam
	bamToBed -i Control.sorted.bam>Control.bed
	rm *.sam *.bam
	rm *.fasta
SRR5197033.fasta:
	fastq-dump --fasta SRR5197033
```

Plotting ChIP-exo signals

```
TotalReads.ZNF10   = 16933834
TotalReads.Control = 58447968
windowSize = 10

Names  = c("L1P1","L1P3","L1P4","L1PB",
           "L1M1","L1M2","L1M3","L1M4","L1MC","L1MD", "L1M5")

ZNF10.forward.signals <-
  read.table("../data/ZNF10.Repeat.plus.bedgraph",  col.names = c("Repeat", "start", "end", "Signal")) %
  mutate(Strand = "Forward")

ZNF10.reverse.signals <-
  read.table("../data/ZNF10.Repeat.minus.bedgraph", col.names = c("Repeat", "start", "end", "Signal")) %
  mutate(Strand = "Reverse")

ZNF10.signals <- rbind(ZNF10.forward.signals, ZNF10.reverse.signals) %>%
  dplyr::filter(endsWith(Repeat, "orf2")) %>%
  mutate(start = as.integer(start/windowSize)*windowSize,
         Signal = Signal*1e6/TotalReads.ZNF10) %>%
  group_by(start, Strand, Repeat) %>%
  summarise(Signal = mean(Signal))

Control.forward.signals <-
  read.table("../data/Control.Repeat.plus.bedgraph", col.names = c("Repeat", "start", "end", "Signal"))
  mutate(Strand = "Forward")

Control.reverse.signals <-
  read.table("../data/Control.Repeat.minus.bedgraph",col.names = c("Repeat", "start", "end", "Signal"))
  mutate(Strand = "Reverse")

Control.signals <- rbind(Control.forward.signals, Control.reverse.signals) %>%
  dplyr::filter(endsWith(Repeat, "orf2")) %>%
  mutate(start = as.integer(start/windowSize)*windowSize,
         Signal = Signal*1e6/TotalReads.Control) %>%
  group_by(start, Strand, Repeat) %>%
```

```
  summarise(Signal = mean(Signal))


inner_join(ZNF10.signals, Control.signals,
           by = c("Repeat", "start", "Strand"),
           suffix = c(".ZNF10", ".Control")) %>%
  mutate(Repeat = factor(Repeat, levels = paste0(Names, "_orf2")),
         FCC    = Signal.ZNF10/Signal.Control) %>%
  dplyr::filter(Repeat %in% levels(Repeat)) %>%
  ggplot(aes(x = start, y = FCC, color = Strand, linetype = Strand)) +
  ggalt::geom_xspline(spline_shape = 0.4)+
  theme_bw()+
  theme(legend.position = "bottom", panel.grid.minor = element_blank(), panel.grid.major.x = element_bla
  scale_color_manual(values = c("#DC9627", "#59A9D7"))+
  scale_x_continuous(limits = c(850,3150), expand = c(0,0))+
  scale_y_continuous(limits = c(0, 8), breaks = c(0,4,8), position = "right")+
  facet_wrap(~Repeat, ncol = 1, strip.position = "left", labeller = as_labeller(function(x) substr(x, st
  xlab("Repeat coordinate (bp)") + ylab("Fold change over control")
```




```
#ggsave("ZNF10 ChIP-exo profies on repeat coordinates.svg", height = 6, width = 6)
```

Lifting out ZNF10 binding sites from LINE1 ORF2 tracks at L1-ZNF10 locus

Lift out operations

```
ZNF10.L1P1.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1P1_orf2",
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1P3.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1P3_orf2",
                                       start_pos = 2849, end_pos = 2868)
```

```
ZNF10.L1P4.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1P4_orf2",
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1PB.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1PB_orf2",
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1M1.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M1_orf2",
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1M2.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M2_orf2",
                                       start_pos = 2846, end_pos = 2865)

ZNF10.L1M3.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M3_orf2",
                                       start_pos = 2843, end_pos = 2862)

ZNF10.L1M4.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M4_orf2",
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1M5.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M5_orf2",
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1MC.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1MC_orf2",
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1MD.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1MD_orf2",
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1P1.sites$Lineage = "L1P1"
ZNF10.L1P3.sites$Lineage = "L1P3"
ZNF10.L1P4.sites$Lineage = "L1P4"
ZNF10.L1PB.sites$Lineage = "L1PB"
ZNF10.L1M1.sites$Lineage = "L1M1"
ZNF10.L1M2.sites$Lineage = "L1M2"
ZNF10.L1M3.sites$Lineage = "L1M3"
ZNF10.L1M4.sites$Lineage = "L1M4"
ZNF10.L1MC.sites$Lineage = "L1MC"
ZNF10.L1MD.sites$Lineage = "L1MD"
ZNF10.L1M5.sites$Lineage = "L1M5"

ZNF10.sites =       c(ZNF10.L1P1.sites, ZNF10.L1P3.sites, ZNF10.L1P4.sites, ZNF10.L1PB.sites,
                      ZNF10.L1M1.sites, ZNF10.L1M2.sites, ZNF10.L1M3.sites, ZNF10.L1M4.sites, ZNF10.L1M5
                      ZNF10.L1MC.sites, ZNF10.L1MD.sites)

require(BSgenome.Hsapiens.UCSC.hg38)
```

```
ZNF10.sites$Sequence = getSeq(Hsapiens, ZNF10.sites, as.character=TRUE)
#ZNF10.sites$Translation = Biostrings::translate(DNAStringSet(substr(ZNF10.sites$Sequence, 3, 20)))
save(list = "ZNF10.sites", file = "ZNF10.sites.RData")
```

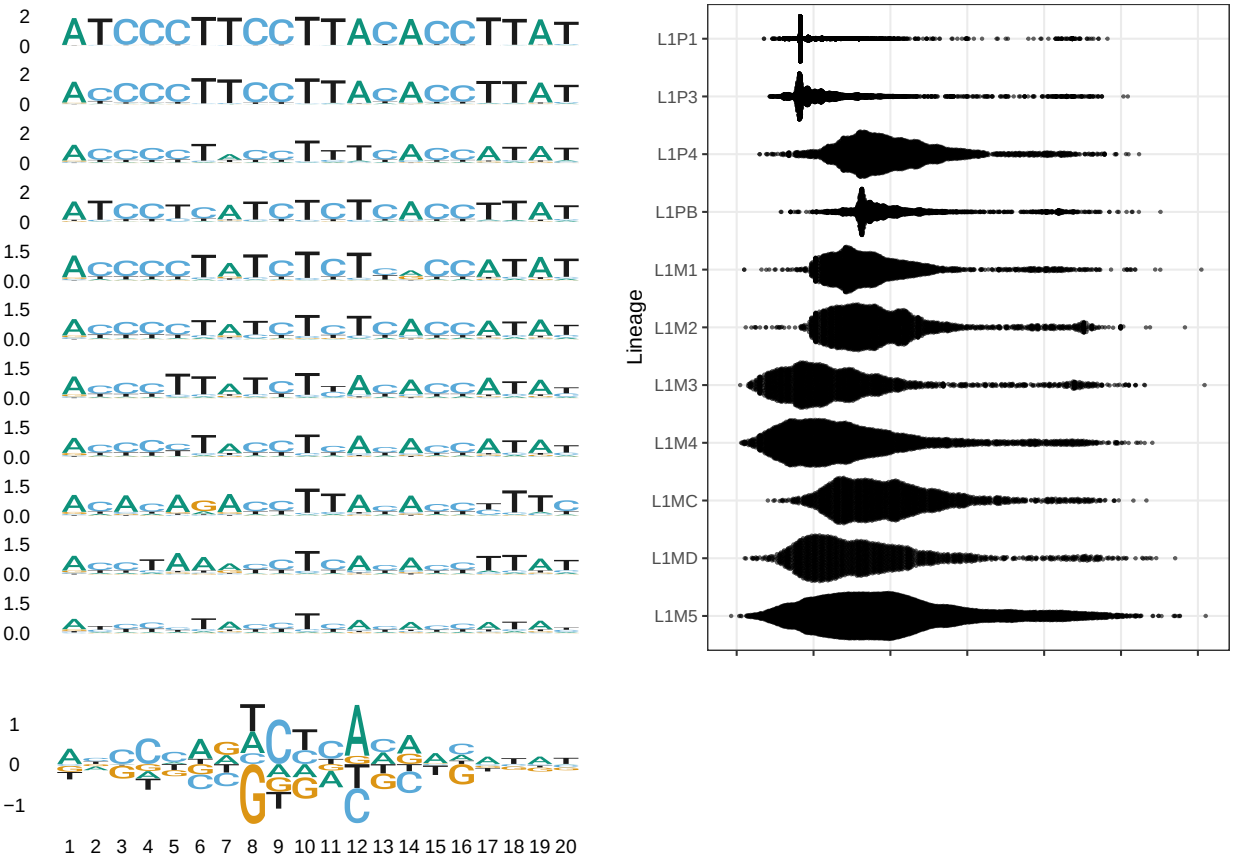
```
load("../data/ZNF10.sites.RData")

ZNF10.sites %>%
  as_tibble()
```

```
## # A tibble: 68,049 x 9
##    seqnames   start     end width strand Sequence        Repea~1 predi~2 Lineage
##    <fct>      <int>   <int> <int> <fct>  <chr>           <chr>     <dbl> <chr>
##  1 chr1       76545   76564    20 +      ATCCCTTCCTTACA~ 86        -6.34 L1P1
##  2 chr1     2364737 2364756    20 -      ATCCCTTCCTTACA~ 3239      -6.34 L1P1
##  3 chr1     3255124 3255143    20 +      ATCCCTTCCTTACA~ 3963      -6.34 L1P1
##  4 chr1     4103855 4103874    20 +      ATCCCTTCCTTACA~ 4937      -6.34 L1P1
##  5 chr1     4245561 4245580    20 -      ATCCCTTCCTTACA~ 5144      -6.34 L1P1
##  6 chr1     4705204 4705223    20 +      ATCCCTTCCTTACA~ 5850      -6.34 L1P1
##  7 chr1     4861939 4861958    20 -      ATCTGTTCCTTACA~ 6063      -5.02 L1P1
##  8 chr1     4876042 4876061    20 -      ATCCCTTCCTTACA~ 6077      -6.34 L1P1
##  9 chr1     5824692 5824711    20 +      ATCCCTTCCTTACA~ 7580      -6.16 L1P1
## 10 chr1     7243404 7243423    20 -      ATCCCTTCCTTACA~ 9944      -6.34 L1P1
## # ... with 68,039 more rows, and abbreviated variable names 1: RepeatID,
## #   2: predicted.Energy
```

Making sequence logo for each sub-group of LINE1 and plotting the predicted energy distribution for all variants

```
load("../data/ZNF10.sites.RData")
col_scheme = ggseqlogo::make_col_scheme(chars=c('A', 'C', 'G', 'T'),
                                        #cols=c('darkgreen', 'blue', 'orange', 'red'))
                                        cols=c("#0E927B", "#59A9D8", "#DC9514", "#1A1A1A"))

for(x in c("L1P1", "L1P3", "L1P4", "L1PB")){
  assign(paste0("ZNF10.",x,".logo"),
         subset(ZNF10.sites, Lineage==x & width==20)$Sequence %>%
           ggseqlogo::ggseqlogo(col_scheme=col_scheme) + scale_y_continuous(limits = c(0,2),breaks = c(0
           theme(axis.text.x = element_blank(), axis.title = element_blank(), axis.ticks.x = element_bla

for(x in c("L1M1", "L1M2", "L1M3", "L1M4", "L1MC", "L1MD", "L1M5")){
  assign(paste0("ZNF10.",x,".logo"),
         subset(ZNF10.sites, Lineage==x & width==20)$Sequence %>%
           ggseqlogo::ggseqlogo(col_scheme=col_scheme) + scale_y_continuous(limits = c(0,1.5),breaks =
           theme(axis.text.x = element_blank(), axis.title = element_blank(), axis.ticks.x = element_bla

cowplot::plot_grid(ZNF10.L1P1.logo, ZNF10.L1P3.logo, ZNF10.L1P4.logo, ZNF10.L1PB.logo,
                   ZNF10.L1M1.logo, ZNF10.L1M2.logo, ZNF10.L1M3.logo, ZNF10.L1M4.logo,
                   ZNF10.L1MC.logo, ZNF10.L1MD.logo, ZNF10.L1M5.logo,
                   ncol=1, align = "v") -> plot.Logos

load("../data/ZNF10.motif.RData")
```

```
ZNF10.motif = TFCookbook::reverseComplement(ZNF10.motif)

TFCookbook::plotEnergyLogo(ZNF10.motif) +
  ylim(-1.5, 1.5)+ theme(axis.title = element_blank())-> ZNF10.Spec.logo

ZNF10.sites$predicted.Energy = TFCookbook::predictEnergy(ZNF10.sites$Sequence, ZNF10.motif)

subset(ZNF10.sites, width==20) %>%
  as_tibble() %>%
  mutate(Lineage = forcats::fct_rev(factor(Lineage, levels = Names))) %>%
  ggplot(aes(x = predicted.Energy, y = Lineage), alpha = 0.5)+
  ggbeeswarm::geom_quasirandom(groupOnX = FALSE, size = 0.4, alpha = 0.6)+
  scale_x_continuous(breaks = seq(-8, 5, 2), minor_breaks = NULL)+
  theme_bw() +
  theme(axis.title.x = element_blank(), axis.text.x = element_blank())-> plot.Energy

cowplot::plot_grid(plot.Logos, plot.Energy, nrow = 2,
                   ZNF10.Spec.logo, align = "h", rel_heights = c(1, 0.26))
```




```
#ggsave("ZNF10 logos and energy distribution.svg", width = 9, height = 6.3)
```