Analysis for ZNF10's association with LINE1 repeats

Zheng Zuo

Contents

Mapping ChIP-exo signals along ORF2 repeat coordinates

Sequence alignment and liftOver operations

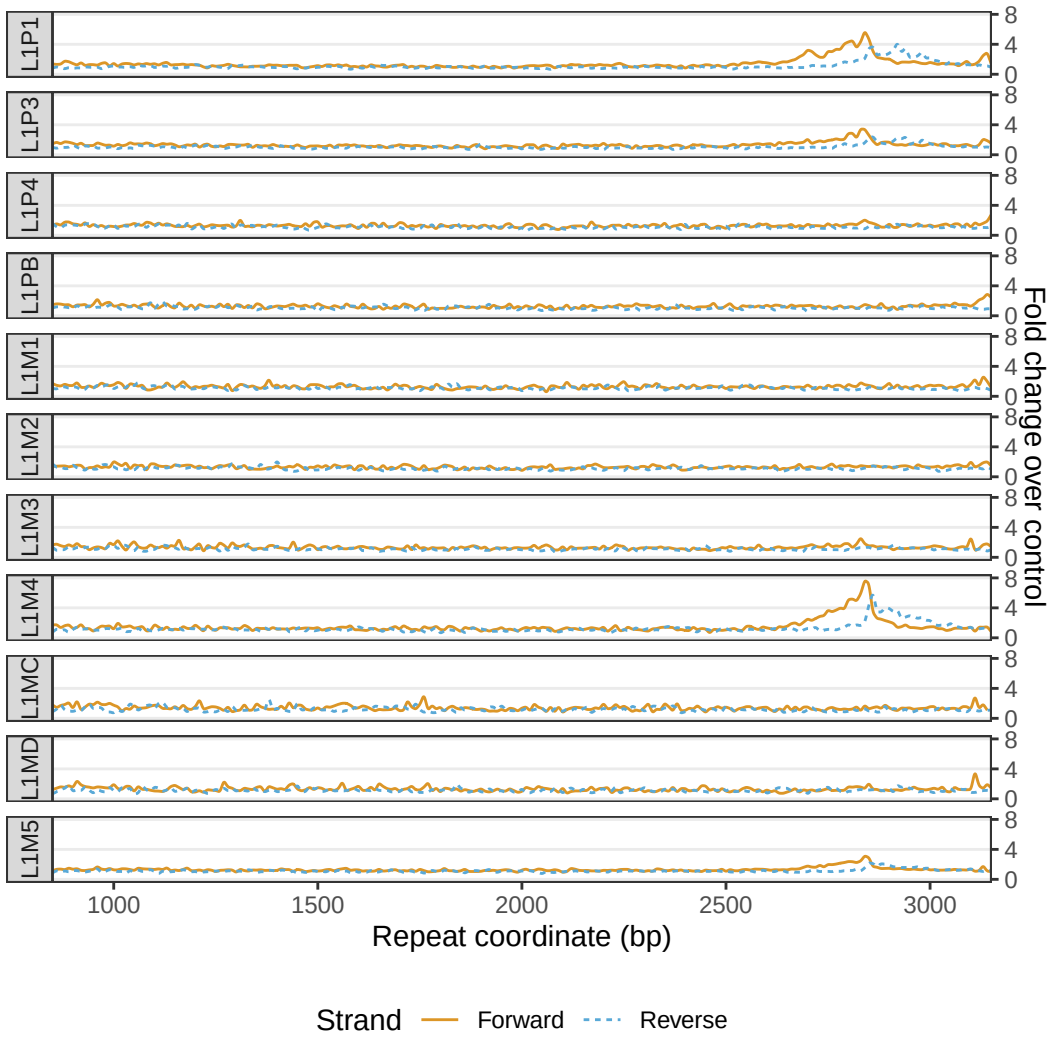
The Makefile used to process raw sequencing data to bedgraph fies on repeat coordinates

```
aim: ZNF10.Repeat.plus.bedgraph ZNF10.Repeat.minus.bedgraph

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 -k1,

ZNF10.Repeat.minus.bedgraph:ZNF10.Repeat.bed
    bedtools genomecov -i ZNF10.Repeat.bed -g hg38.repeat.sizes -bg -strand - -5|LC_COLLATE=C sort -k1,

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.fasta
    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
```

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("../../ZNF10/ChIP-exo data/ZNF10.Repeat.plus.bedgraph",
                                   col.names = c("Repeat", "start", "end", "Signal")) %>%
  mutate(Strand = "Forward",
         Signal = Signal*1e6/TotalReads.ZNF10)
ZNF10.reverse.signals = read.table("../../ZNF10/ChIP-exo data/ZNF10.Repeat.minus.bedgraph",
                                   col.names = c("Repeat", "start", "end", "Signal")) %>%
  mutate(Strand = "Reverse",
         Signal = Signal*1e6/TotalReads.ZNF10)

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

Control.forward.signals = read.table("../../KAP1/ChIP-exo Input/Control.Repeat.plus.bedgraph",
                                     col.names = c("Repeat", "start", "end", "Signal")) %>%
  mutate(Strand = "Forward",
         Signal = Signal*1e6/TotalReads.Control)

Control.reverse.signals = read.table("../../KAP1/ChIP-exo Input/Control.Repeat.minus.bedgraph",
                                     col.names = c("Repeat", "start", "end", "Signal")) %>%
  mutate(Strand = "Reverse",
         Signal = Signal*1e6/TotalReads.Control)

Control.signals <- rbind(Control.forward.signals, Control.reverse.signals) %>%
  dplyr::filter(Repeat %>% endsWith("orf2")) %>%
  mutate(start = as.integer(start/windowSize)*windowSize) %>%
  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, s
  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

Making the correspondence table between RepeatID and RepeatORF identity in alignment file

```
hg38_fa_LINE1 <- readr::read_table2("hg38.fa.LINE1.out", col_names = FALSE) %>%
  dplyr::rename(Repeat = X10, RepeatID = X15)

hg38_fa_LINE1.orf2 <- readr::read_table2("hg38.fa.L1_orf2.out", col_names = FALSE) %>%
  tidyr::separate(X9, into = "RepeatORF", sep = "#") %>%
```

```
  dplyr::rename(RepeatID = X14)

hg38_fa_LINE1 %>%
  dplyr::filter(RepeatID==1509)

inner_join(hg38_fa_LINE1, hg38_fa_LINE1.orf2, by = "RepeatID") %>%
  select(Repeat, RepeatORF, RepeatID) %>%
  unique() %>%
  group_by(Repeat, RepeatORF) %>%
  #dplyr::filter(startsWith(Repeat, "L1HS") & startsWith(RepeatORF, "L1M5"))
  tally() %>%
  arrange(desc(n)) %>%
  dplyr::filter(startsWith(RepeatORF, "L1M4"))

save("Repeat_ORF.correspondence.table", file = "L1 Repeat-ORF correspondence.RData")
```

Lift out operations

```
load("L1 Repeat-ORF correspondence.RData")
L1P1_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1P1_orf2" &
                  Repeat %in% c("L1HS","L1PA2", "L1PA3", "L1PA4", "L1PA5", "L1PA6"))

L1P3_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1P3_orf2" &
                  Repeat %in% c("L1PA7", "L1PA8", "L1PA10", "L1PA11", "L1PA12"))

L1P4_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1P4_orf2" &
                  Repeat %in% c("L1PA13", "L1PA14","L1PA15", "L1PA16", "L1PA17", "L1PA15-16"))

L1PB_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1PB_orf2" & startsWith(Repeat, "L1PB"))

L1M1_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1M1_orf2" & Repeat %in% c("L1MA1", "L1MA2", "L1MA3"))

L1M2_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1M2_orf2" & Repeat %in% c("L1MA4", "L1MA4A", "L1MA5", "L1MA6"))

L1M3_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1M3_orf2" & Repeat %in% c("L1MA7", "L1MA8", "L1MA9", "L1MA10"))

L1M4_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1M4_orf2" & startsWith(Repeat, "L1MB"))

L1MC_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1MC_orf2" & startsWith(Repeat, "L1MC"))

L1MD_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1MD_orf2" & startsWith(Repeat, "L1MD"))
```

```
L1M5_ORF.repeats = Repeat_ORF.correspondence.table %>%
  dplyr::filter(RepeatORF=="L1M5_orf2" & startsWith(Repeat, "L1ME"))


ZNF10.L1P1.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1P1_orf2", RepeatID = L1P1_ORF.repeats$RepeatID,
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1P3.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1P3_orf2", RepeatID = L1P3_ORF.repeats$RepeatID,
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1P4.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1P4_orf2", RepeatID = L1P4_ORF.repeats$RepeatID,
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1PB.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1PB_orf2", RepeatID = L1PB_ORF.repeats$RepeatID,
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1M1.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M1_orf2", RepeatID = L1M1_ORF.repeats$RepeatID,
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1M2.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M2_orf2", RepeatID = L1M2_ORF.repeats$RepeatID,
                                       start_pos = 2846, end_pos = 2865)

ZNF10.L1M3.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M3_orf2", RepeatID = L1M3_ORF.repeats$RepeatID,
                                       start_pos = 2843, end_pos = 2862)

ZNF10.L1M4.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1M4_orf2", RepeatID = L1M4_ORF.repeats$RepeatID,
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1M5.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                            Repeat = "L1M5_orf2", RepeatID = L1M5_ORF.repeats$RepeatID,
                            start_pos = 2849, end_pos = 2868)

ZNF10.L1MC.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1MC_orf2", RepeatID = L1MC_ORF.repeats$RepeatID,
                                       start_pos = 2849, end_pos = 2868)

ZNF10.L1MD.sites = TECookbook::liftOut(alignment = "hg38.fa.align",
                                       Repeat = "L1MD_orf2", RepeatID = L1MD_ORF.repeats$RepeatID,
                                       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"

save(list = 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.sites","ZNF10.L1MC.sites", "ZNF10.L1MD.sites"), file = "ZN
```

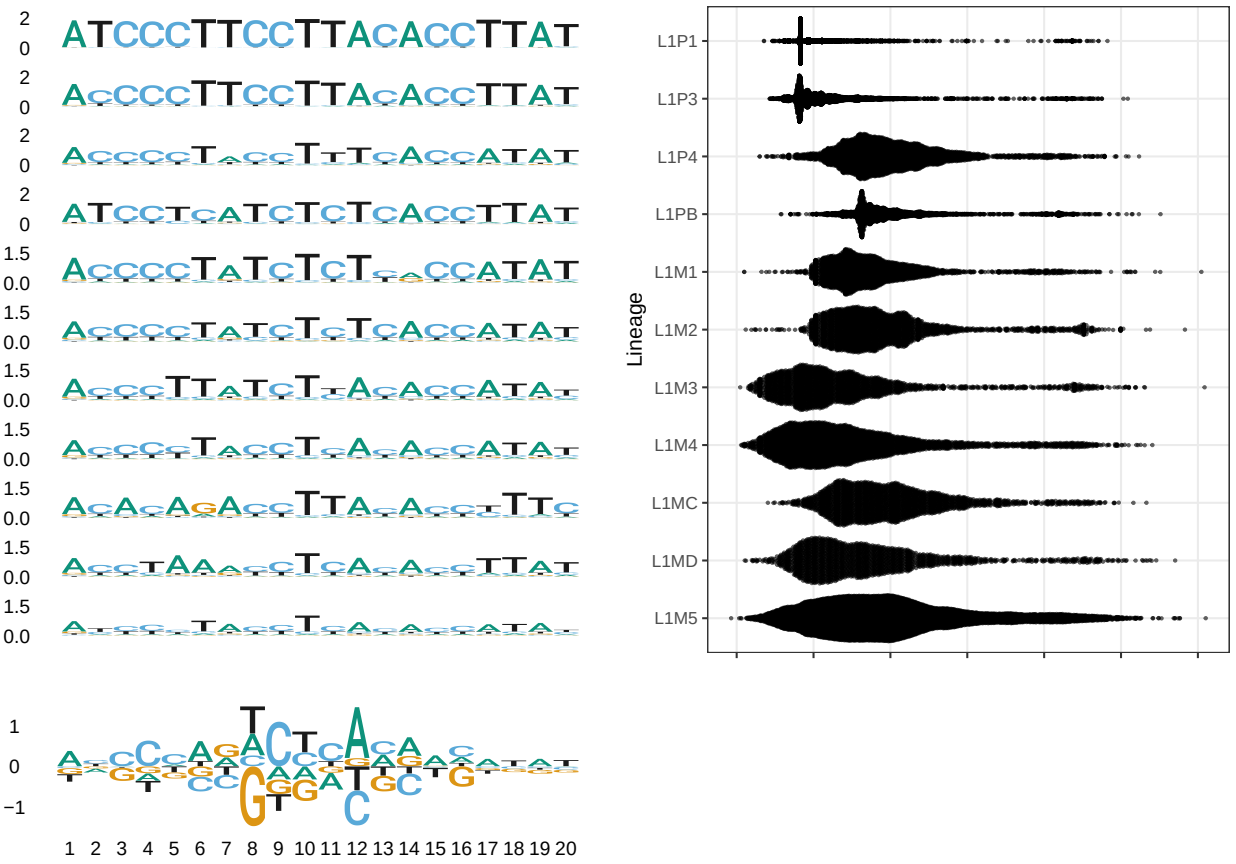
Making sequence logo for each sub-group of LINE1 and plotting the predicted energy distribution for all variants

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

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
```

```
## GRanges object with 68049 ranges and 4 metadata columns:
##           seqnames            ranges strand |             Sequence    RepeatID
##              <Rle>         <IRanges>  <Rle> |          <character> <character>
##       [1]     chr1       76545-76564      + | ATCCCTTCCTTACACCTTAT          86
##       [2]     chr1   2364737-2364756      - | ATCCCTTCCTTACACCTTAT        3239
##       [3]     chr1   3255124-3255143      + | ATCCCTTCCTTACACCTTAT        3963
##       [4]     chr1   4103855-4103874      + | ATCCCTTCCTTACACCTTAT        4937
##       [5]     chr1   4245561-4245580      - | ATCCCTTCCTTACACCTTAT        5144
##       ...      ...               ...    ... .                   ...         ...
##   [68045]     chrY 22548780-22548795      - |     ACATAAATCTTATACT     4828056
##   [68046]     chrY 23646890-23646905      + |     ACATAAATCTTATACT     4829178
##   [68047]     chrY 26023173-26023188      - |     ACATAAATCTTATACT     4831834
##   [68048]     chrY 56924532-56924551      + | ACCTAATTCTCACACCTTGT     4832869
##   [68049]     chrY 57002040-57002059      + | GCTTAACCTCATACCTTATA     4832992
##           predicted.Energy     Lineage
##                  <numeric> <character>
##       [1]         -6.34481        L1P1
##       [2]         -6.34481        L1P1
##       [3]         -6.34481        L1P1
##       [4]         -6.34481        L1P1
##       [5]         -6.34481        L1P1
##       ...              ...         ...
##   [68045]         -4.91234        L1MD
##   [68046]         -4.91234        L1MD
##   [68047]         -4.91234        L1MD
##   [68048]         -6.03994        L1MD
##   [68049]          1.32718        L1MD
```

```
##   -------
##   seqinfo: 312 sequences from an unspecified genome; no seqlengths
```

```
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(
           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

load("../../../Development/ZFPCookbook/ZNF10/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

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


require(ggplot2)
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)+
  #geom_boxplot()+
  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)
```