Seed motifs on transcripts

Melina Klostermann

28 September, 2023

Contents

1 Libraries and settings

```
# ----------------------------------------
# libraries
# ----------------------------------------
library(tidyverse)
library(GenomicRanges)
library(colorspace)
library(gghalves)
library(BSgenome.Mmusculus.UCSC.mm10)
library(Biostrings)
library(plyranges)

here <- here::here()

# ----------------------------------------
# settings
# ----------------------------------------

out <- paste0(here,"/Figure5/01_Seed_motifes/")
source(paste0(here,"/Supporting_scripts/themes/theme_paper.R"))
source(paste0(here,"/Supporting_scripts/themes/CustomThemes.R"))
```

2 What was done?

- I count different versions of the miR181 seed in the 200nt before and after mir181 binding sites.
- I use the seed 6mer, 7mers with one adjecent nt, and a 8mer with two adjecent nts.

3 Files

```
# ------------------------------------
# transcript sequences
# ------------------------------------
transcript_fasta <- readDNAStringSet("/Users/melinaklostermann/Documents/projects/anno/gencodevM23/genco

transcript_anno_meta <- names(transcript_fasta)
transcript_anno_meta <- data.frame(all = transcript_anno_meta) %>%
  tidyr::separate(., col = all,
                  into = c("transcript_id", "gene_id", "a", "b", "isoform_name", "gene_name", "entrez_g

names_transcript_fasta <- sub("\\..*", "", transcript_anno_meta$transcript_id)

# add N in beginning in end to not run out of transcripts when search motif
n200 <- c(rep("N",200)) %>%
  paste(., collapse = "") %>%
  RNAStringSet()

transcript_fasta <- xscat(n200, transcript_fasta, n200)
names(transcript_fasta) <- names_transcript_fasta

transcript_fasta_df <- data.frame(tx_name = names(transcript_fasta), width = width(transcript_fasta))

# ----------------------------------------
# MREs
# ----------------------------------------

mir181_bs <- readRDS(paste0(here,"/Figure4/03_assign_transcripts/mir181_bs_on_transcripts.rds"))

mir_crosslinks <- readRDS("/Users/melinaklostermann/Documents/projects/AgoCLIP_miR181/xx_down_stream_R/


# move bs annotation because transcripts got 200N in beginning
# Achtung! this needs to be shifted back in the end of the script!!!
mir181_bs <- makeGRangesFromDataFrame(mir181_bs, keep.extra.columns = T) %>%
  shift(., 200) %>%
  as.data.frame(.)

mir181_enriched_set <- mir181_bs %>%
  subset(set %in% c("ago_bs_mir181_chi&mir181_enriched", "mir181_enriched"))

nrow(mir181_enriched_set)
```

```
## [1] 4519
```

4 XSTREME de novo motif discovery

```
# get seqeunce 200nt around binding sites

mir181_bs_200_both_sides <- as.data.frame(mir181_enriched_set) %>%
  left_join(transcript_fasta_df, by= c(seqnames = "tx_name"), suffix = c(".bs", ".tx")) %>%
  mutate(end = end + 197, start = start -197) %>%
  dplyr::filter((end <  width.tx)  & (start > 0)) %>%
  makeGRangesFromDataFrame(., keep.extra.columns = T)

mir181_bs_200_both_sides_seq <- getSeq(mir181_bs_200_both_sides, x = transcript_fasta) %>%
  RNAStringSet()

names(mir181_bs_200_both_sides_seq) <- 1:NROW(mir181_bs_200_both_sides_seq)

# write fasta file for XSTREME
writeXStringSet(mir181_bs_200_both_sides_seq, filepath = paste0(out, "mirBS_200_both_sides_transcripts.
```

XSTREME is executed on the fasta file from above via the MEME SUITE webpage (https://meme-suite.org/meme/tools/xstreme) with the following parameters:

- E-value <= 0.05
- Width 5-10
- background: model control sequences
- STREME limit: Number of motifes = 20
- MEME options: Default E-value, Zero or one occurence per sequence
- SEA: Output the matching sequences in a TSV file

```
##################
# the mir181 seed and interesting seed variations
##################

seed_8mer <- "UGAAUGUA"
seed_7mer_m8 <- "UGAAUGU"
seed_7mer_a1 <- "GAAUGUA"
seed_6mer <- "GAAUGU"
seed_6mer_wobble <- "GAUUGU"
seed_8mer_wobble <- "UGAUUGUA"
seed_7mer_m8_wobble <- "UGAUUGU"
seed_7mer_a1_wobble <- "GAUUGUA"

seed_alt_38 <- "UGAAUG"
seed_alt_38_wobble <- "UGAUUG"


# make a list of all seeds
seed_list <- list(seed_8mer, seed_7mer_m8, seed_7mer_a1,seed_6mer, seed_6mer_wobble, seed_8mer_wobble, s

seed_names_list <- list( "seed_8mer",  "seed_7mer_m8", "seed_7mer_a1",  "seed_6mer", "seed_6mer_wobble"

# hierarchy order, to decide which seed to use if several ar present
seed_importance_order <- c("seed_8mer",  "seed_7mer_m8", "seed_7mer_a1",  "seed_6mer", "seed_alt_38", "s
```

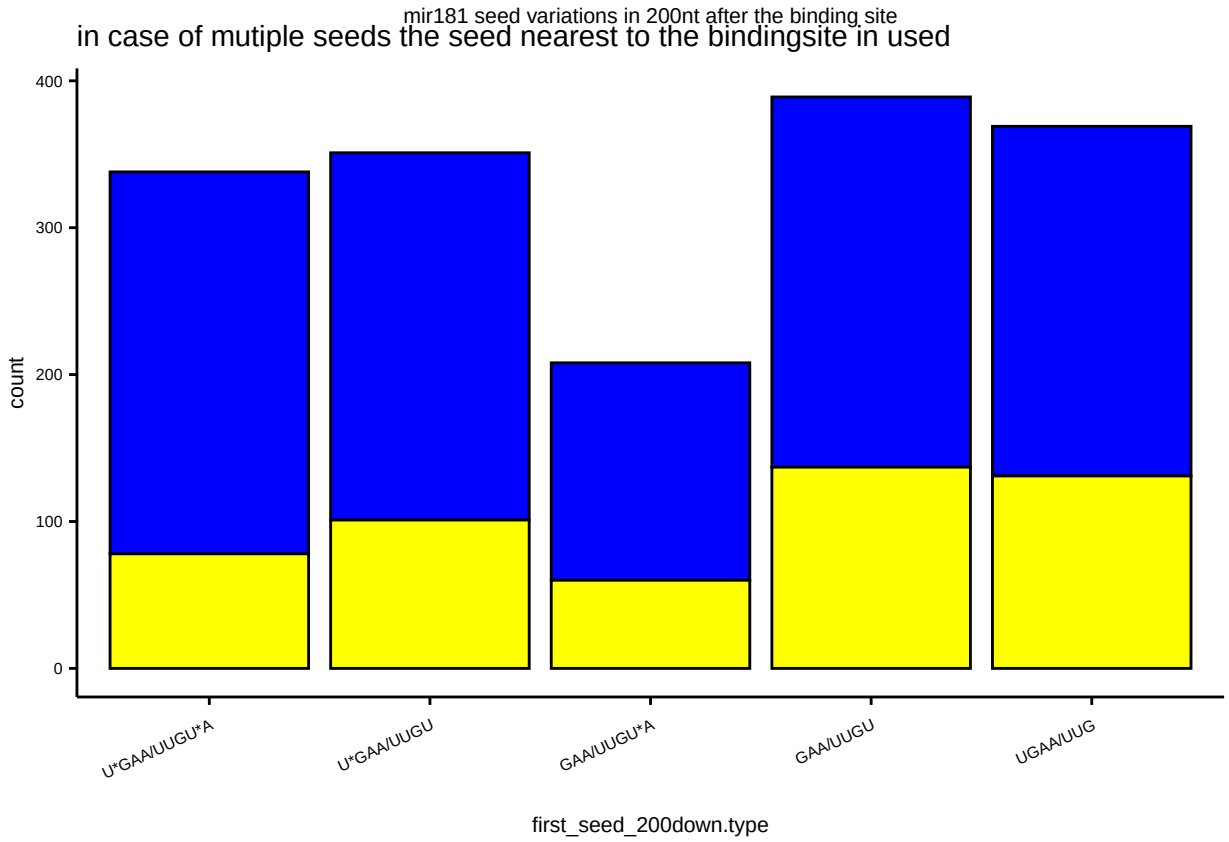
5 Seed position and distribution

5.1 200nt after the binding site

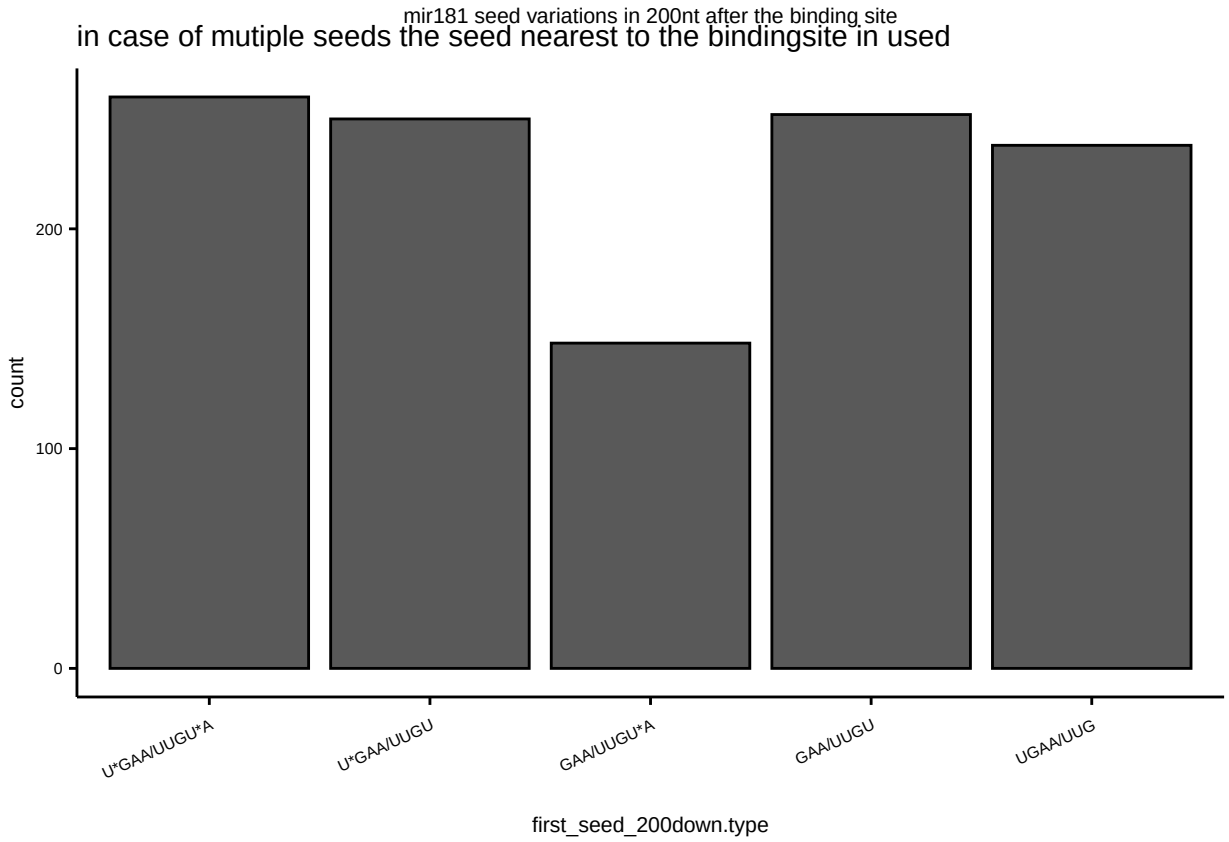
```
#########################
# get seed in 200er window
#########################

mir181_bs_200down <- as.data.frame(mir181_enriched_set) %>%
  left_join(transcript_fasta_df, by= c(seqnames = "tx_name"), suffix = c(".bs", ".tx")) %>%
  mutate(end = end + 200) %>%
  dplyr::filter((end <  width.tx)  & (start > 0)) %>%
  makeGRangesFromDataFrame(., keep.extra.columns =T)

mir181_bs_200down_seq <- getSeq(mir181_bs_200down, x = transcript_fasta) %>%
  RNAStringSet()

# count occurences of all seed variations
Seeds_200down <- lapply(seed_list, function(x) {
  vmatchPattern(pattern = x, mir181_bs_200down_seq) %>%
  lapply(., function(x) as.data.frame(x))})

# add the binding site id to the seeds and make a df per seed type
BS_ID_list <- as.list(mir181_bs_200down$mir181BS_ID)

Seeds_200down <- map(Seeds_200down,
                     ~map2(.x, BS_ID_list, ~mutate(.x, mir181BS_ID = .y) ) %>%
               map_dfr(~.x))


# add the seed type names and make one df of all
Seeds_200down <- map2(Seeds_200down, seed_names_list, ~mutate(.x, seed = .y) ) %>% map_dfr(~.x)


# extract wobble positions
Seeds_1_per_BS <- Seeds_200down %>%
  mutate(wobble = grepl("wobble", seed),
         seed = case_when(wobble  ~ substr(seed, 1, nchar(seed)-7), T ~ seed))


# order seeds by importance
Seeds_1_per_BS$seed <- factor(Seeds_1_per_BS$seed, levels = seed_importance_order )

# select 1 seed per BS --> closest seed with highest importance
Seeds_1_per_BS <- Seeds_1_per_BS %>%
   group_by(mir181BS_ID) %>%
  arrange(start, seed ) %>%
  dplyr::slice(1) %>%
  ungroup(.)

#########################
# combine the closest seed, and all found seeds to the Binding site data.frame
#########################
```

```r
# add all as list column

colnames(Seeds_200down) <- c("Seeds_200down.start",
                             "Seeds_200down.end",
                             "Seeds_200down.width",
                             "mir181BS_ID",
                             "Seeds_200down.type")


mir181_bs <- left_join(mir181_bs, Seeds_200down, by = "mir181BS_ID") %>%
  tidyr::nest(all_seeds_200down = c("Seeds_200down.start",
                             "Seeds_200down.end",
                             "Seeds_200down.width",
                             "Seeds_200down.type"))

# add closest mir
colnames(Seeds_1_per_BS) <- c("first_seed_200down.start",
                             "first_seed_200down.end",
                             "first_seed_200down.width",
                             "mir181BS_ID",
                             "first_seed_200down.type",
                             "first_seed_200down.wobble")

mir181_bs <- left_join(mir181_bs, Seeds_1_per_BS, by = "mir181BS_ID")

mir181_bs <- mir181_bs %>%
  rowwise() %>%
  mutate(seed_repetitions.200down = sum(all_seeds_200down$Seeds_200down.type == "seed_6mer"),
         seed_repetitions.200down.wobble = sum(all_seeds_200down$Seeds_200down.type == "seed_6mer_wobble
```

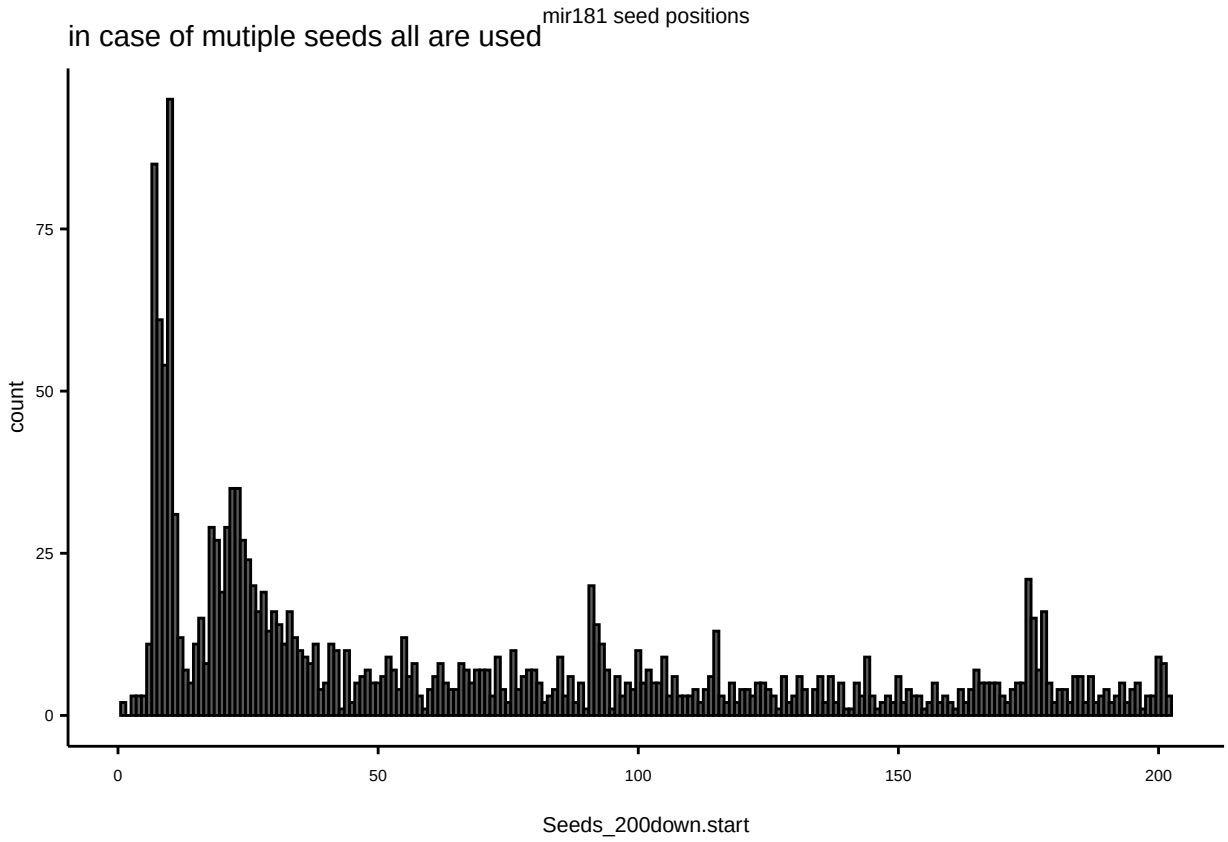
```r
###################
# plots
###################

# plot seed variations SuppFigure5C
####################################
p <- ggplot(mir181_bs %>% subset(set %in% c("ago_bs_mir181_chi&mir181_enriched", "mir181_enriched")) %>%
  geom_bar(color = "black")+
  theme_paper()+
  scale_fill_manual(values = c("blue", "yellow"))+
  theme(legend.position = "None") +
  scale_x_discrete(labels=c(seed_8mer = "U*GAA/UUGU*A",
                            seed_7mer_m8 = "U*GAA/UUGU",
                            seed_7mer_a1 = "GAA/UUGU*A",
                            seed_6mer = "GAA/UUGU",
                            seed_alt_38 = "UGAA/UUG"),
                   guide = guide_axis(angle = 25))

p+
  ggtitle("mir181 seed variations in 200nt after the binding site",
          subtitle = "in case of mutiple seeds the seed nearest to the bindingsite in used")
```



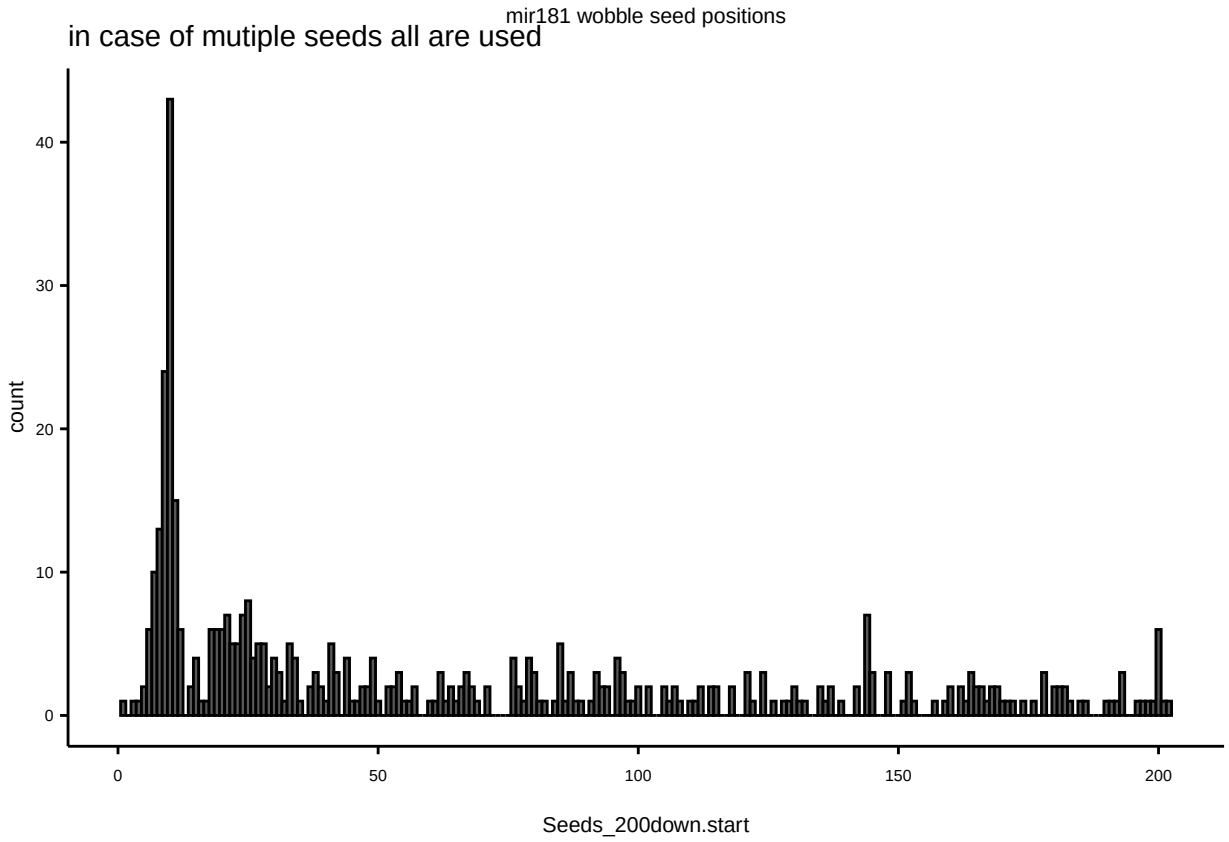

```
# plot seed versions Figure5B
##############################
p1 <- ggplot(mir181_bs %>% subset(set %in% c("ago_bs_mir181_chi&mir181_enriched", "mir181_enriched")) %>%
               subset(first_seed_200down.wobble == F), aes(x = first_seed_200down.type))+
  geom_bar(color = "black")+
  theme_paper()+
  theme(legend.position = "None") +
  scale_x_discrete(labels=c(seed_8mer = "U*GAA/UUGU*A",
                            seed_7mer_m8 = "U*GAA/UUGU",
                            seed_7mer_a1 = "GAA/UUGU*A",
                            seed_6mer = "GAA/UUGU",
                            seed_alt_38 = "UGAA/UUG"),
                   guide = guide_axis(angle = 25))

p1+
  ggtitle("mir181 seed variations in 200nt after the binding site",
          subtitle = "in case of mutiple seeds the seed nearest to the bindingsite in used")
```




```
# plot seed position
####################
p2 <- ggplot(unnest(mir181_bs) %>%
               subset(Seeds_200down.type %in% c("seed_6mer", "seed_6mer_wobble")) %>%
               subset(set %in% c("ago_bs_mir181_chi&mir181_enriched", "mir181_enriched")), aes(x = Seeds
  geom_histogram(color = "black", binwidth = 1)+
  theme_paper()

p2+
  ggtitle("mir181 seed positions",
          subtitle = "in case of mutiple seeds all are used")
```




```
# plot wobble seed position
############################
p3 <- ggplot(unnest(mir181_bs) %>% subset(Seeds_200down.type == "seed_6mer_wobble") %>% subset(set %in%
  geom_histogram(color = "black", binwidth = 1)+
  theme_paper()
p3+
  ggtitle("mir181 wobble seed positions",
          subtitle = "in case of mutiple seeds all are used")
```




```
# get numbers for text
########################
t <- mir181_bs %>% subset(set %in% c("ago_bs_mir181_chi&mir181_enriched", "mir181_enriched")) %>%
  pull(first_seed_200down.wobble) %>%
  table()

t
```

```
## .
## FALSE  TRUE
##  1148   507
```

```
t/sum(t)
```

```
## .
##     FALSE      TRUE
## 0.6936556 0.3063444
```

```
b <- nrow(mir181_bs %>% subset(set %in% c("ago_bs_mir181_chi&mir181_enriched", "mir181_enriched")))

a <- mir181_bs %>% subset(set %in% c("ago_bs_mir181_chi&mir181_enriched", "mir181_enriched")) %>%
  subset(first_seed_200down.wobble == F)

nrow(a)
```

```
## [1] 1148
```

```
nrow(a)/b
```

```
## [1] 0.2540385
```

```
ggsave(p, filename = paste0(out, "SuppFigure5C_seed_versions.pdf"), width = 6, height = 6, units = "cm"
ggsave(p1, filename = paste0(out, "Figure5B_seed_versions.pdf"), width = 6, height = 6, units = "cm"  )
ggsave(p2, filename = paste0(out, "Figure5C_seed_position_after_BS.pdf"), width = 6, height = 6, units =
ggsave(p3, filename = paste0(out, "Figure5H_wobbleseed_position_after_BS.pdf"), width = 6, height = 4,
```

5.1.1 percent binding sites with a seed downstream

```
nrow(mir181_bs %>% subset(set %in% c("ago_bs_mir181_chi&mir181_enriched", "mir181_enriched")) %>% subse
```

```
## [1] 0.3662315
```

6 Make table of 6mers on transcripts

```
# ! shift transcript positions 200nt back!!!
mir181_bs <- makeGRangesFromDataFrame(mir181_bs, keep.extra.columns = T) %>%
  shift(., shift = -200) %>% as.data.frame(.)

# get 6mers per bs
seeds_tx <- mir181_bs %>% unnest(all_seeds_200down) %>%
  subset(!is.na(Seeds_200down.start)) %>%
  makeGRangesFromDataFrame(., keep.extra.columns = T)

seeds_tx <- shift(seeds_tx, seeds_tx$Seeds_200down.start -1)

seeds_tx <- resize(seeds_tx, width = seeds_tx$Seeds_200down.width, fix = "start")
```

7 Save tables

```
saveRDS(mir181_bs, file = paste0(out, "mir181_bs_with_seeds_transcripts.rds"))
saveRDS(seeds_tx, file = paste0(out, "seeds_transcripts.rds"))

t <- mir181_bs %>% as.data.frame() %>%
   subset(set %in% c("mir181_enriched", "ago_bs_mir181_chi&mir181_enriched"))

write_csv(t, paste0(out, "STable6_MREs_transcripts_seeds.csv"))
```

8 Session Info

```
sessionInfo()
```

```
## R version 4.2.2 (2022-10-31)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur ... 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
```

```
##
## attached base packages:
## [1] stats4    stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] plyranges_1.18.0                  BSgenome.Mmusculus.UCSC.mm10_1.4.3
##  [3] BSgenome_1.66.3                   rtracklayer_1.58.0
##  [5] Biostrings_2.66.0                 XVector_0.38.0
##  [7] gghalves_0.1.4                    colorspace_2.1-0
##  [9] GenomicRanges_1.50.2              GenomeInfoDb_1.34.9
## [11] IRanges_2.32.0                    S4Vectors_0.36.2
## [13] BiocGenerics_0.44.0               lubridate_1.9.2
## [15] forcats_1.0.0                     stringr_1.5.0
## [17] dplyr_1.1.2                       purrr_1.0.1
## [19] readr_2.1.4                       tidyr_1.3.0
## [21] tibble_3.2.1                      ggplot2_3.4.2
## [23] tidyverse_2.0.0                   knitr_1.43
##
## loaded via a namespace (and not attached):
##  [1] bitops_1.0-7                matrixStats_1.0.0
##  [3] bit64_4.0.5                 rprojroot_2.0.3
##  [5] tools_4.2.2                 backports_1.4.1
##  [7] utf8_1.2.3                  R6_2.5.1
##  [9] DBI_1.1.3                   withr_2.5.0
## [11] tidyselect_1.2.0            bit_4.0.5
## [13] compiler_4.2.2              textshaping_0.3.6
## [15] cli_3.6.1                   Biobase_2.58.0
## [17] DelayedArray_0.24.0         labeling_0.4.2
## [19] scales_1.2.1                systemfonts_1.0.4
## [21] digest_0.6.33               Rsamtools_2.14.0
## [23] rmarkdown_2.23              pkgconfig_2.0.3
## [25] htmltools_0.5.5             MatrixGenerics_1.10.0
## [27] fastmap_1.1.1               highr_0.10
## [29] rlang_1.1.1                 rstudioapi_0.15.0
## [31] BiocIO_1.8.0                generics_0.1.3
## [33] farver_2.1.1                BiocParallel_1.32.6
## [35] vroom_1.6.3                 car_3.1-2
## [37] RCurl_1.98-1.12             magrittr_2.0.3
## [39] GenomeInfoDbData_1.2.9      Matrix_1.5-4.1
## [41] munsell_0.5.0               fansi_1.0.4
## [43] abind_1.4-5                 lifecycle_1.0.3
## [45] stringi_1.7.12              yaml_2.3.7
## [47] carData_3.0-5               SummarizedExperiment_1.28.0
## [49] zlibbioc_1.44.0             grid_4.2.2
## [51] parallel_4.2.2              crayon_1.5.2
## [53] lattice_0.21-8              hms_1.1.3
## [55] pillar_1.9.0                ggpubr_0.6.0
## [57] rjson_0.2.21                ggsignif_0.6.4
## [59] codetools_0.2-19            XML_3.99-0.14
## [61] glue_1.6.2                  evaluate_0.21
## [63] vctrs_0.6.3                 tzdb_0.4.0
## [65] gtable_0.3.3                xfun_0.39
## [67] broom_1.0.5                 restfulr_0.0.15
```

```
## [69] rstatix_0.7.2               ragg_1.2.5
## [71] GenomicAlignments_1.34.1    timechange_0.2.0
## [73] here_1.0.1
```