mir181 binding sites - union of mir181 enriched binding sites and Ago binding sites targeted by mir181

Melina Klostermann

12 September, 2023

Contents

1 Libraries and settings

```
# ---------------------------------------
# libraries
# ---------------------------------------
library(tidyverse)
library(GenomicRanges)
library(colorspace)
library(eulerr)
library(gghalves)
library(ggpubr)

# ---------------------------------------
# settings
# ---------------------------------------
here <- here::here()
out <- paste0(here,"/Figure2+SF1h-j/01_mir181-enriched_binding_site_definition/")

source(paste0(here,"/Supporting_scripts/themes/theme_paper.R"))
source(paste0(here,"/Supporting_scripts/themes/CustomThemes.R"))

# #nikita
```

```
# out <- "D:/Krueger_Lab/Publications/miR181_paper/Figure1/mir181_binding_sites__venn_types/"
# source("D:/Krueger_Lab/Publications/miR181_paper/Supporting_scripts/themes/theme_paper.R")
# source("D:/Krueger_Lab/Publications/miR181_paper/Supporting_scripts/themes/CustomThemes.R")

# farben
farbeneg <- "#B4B4B4"
farbe1 <- "#0073C2FF" #WT farbe
farbe2 <- "#EFC000FF"
farbe4 <- "#7AA6DCFF"
farbe6 <- "#003C67FF"
farbe14 <- "#8A4198FF"
```

2 What was done?

mir181 binding sites are defined as the union of - AGO binding sites that contain at least 2 chimirc mir181 crosslinks (from the IP_WT chimeric reads or the IP_mir181_WT chimeric reads) in a window from 10nt before till 10nt after a the AGO binding site - binding sites defined on enriched mir181 data (IP_mir181_WT)

- the two subgroups are plotted as a venn diagram (figure 1 XX)
- this is compared to the differntially regulated AGO binding sites from the mir181 KO condition

3 Files

```
# ----------------------------------------
# mir181 enriched binding sites
# ----------------------------------------
mir181_enriched <- readRDS(paste0(here,"/Figure2+SF1h-j/01_mir181-enriched_binding_site_definition/BS_mi

# #nikita
# mir181_enriched <- readRDS("D:/Krueger_Lab/Publications/miR181_paper/Methods/mir181-enriched_binding_

# ----------------------------------------
# chimeric reads
# ----------------------------------------

chimeric_reads <- readRDS(paste0(here,"/Figure1+SF1a-g/03_Ago_targetome/mir_chimeric_crosslinks.rds"))

#nikita
# chimeric_reads <- readRDS("D:/Krueger_Lab/Publications/miR181_paper/Figure1/Ago_targetome/mir_chimeri

# ----------------------------------------
# AGO binding sites
# ----------------------------------------
ago_bs <- readRDS(paste0(here,"/Figure1+SF1a-g/02_AGO_binding_site_definition/AGO_BS.rds"))

#nikita
# ago_bs <- readRDS("D:/Krueger_Lab/Publications/miR181_paper/Figure1/AGO_binding_site_definition/2023-

# ----------------------------------------
```

```
# BS downregulated in mir181 KO
# ---------------------------------------
diff <- readRDS(paste0(here,"/Figure2+SF1h-j/03_Differential_Binding/BsDifferentialResult.rds"))

#nikita
# diff <- readRDS("D:/Krueger_Lab/Publications/miR181_paper/Figure1/Differential_Binding/BsDifferential
```

4 mir181 binding sites

4.1 Get AGO binding sites with chimeric mir181

Here we define mir181 AGO binding sites by overlapping the AGO binding sites (see script Methods/02_AGO_binding_site_definition) with the chimeric mir181 reads (see script Figure1/Ago_targetome). AGO binding sites that contained at least 2 chimeric mir181 crosslinks in the binding site or within 10nt proximity to the binding site are selected as mir181 Ago binding sites.

```
# use region of bs +-10nt for overlaps
ago_bs_10 <- ago_bs + 10

# use chimeric reads from both mir181 enriched and non-enriched data
chimeric_reads <- c(makeGRangesFromDataFrame(chimeric_reads$IP_WT, keep.extra.columns = T), makeGRangesF

# find overlaps of mirt and AGO bs
idx <- findOverlaps(ago_bs_10, chimeric_reads )

# make a data frame from the ago bs
names(ago_bs)<- 1:NROW(ago_bs)
ago_bs <- as.data.frame(ago_bs)
ago_bs$BS_ID <- rownames(ago_bs)

# add mir info to ago bs
ago_bs_mir181_chi <- cbind(ago_bs[queryHits(idx),], mir_IP = chimeric_reads [subjectHits(idx),]$Name)

ago_bs_mir181_chi <- ago_bs_mir181_chi[grepl(ago_bs_mir181_chi$mir_IP,
                                     pattern = "miR-181"),]

# count chimerics
mir181_chi <- ago_bs_mir181_chi %>% group_by(BS_ID) %>%
  summarize(n_mir181 = sum(grepl(mir_IP,pattern = "miR-181")),
            n_mir181a = sum(grepl(mir_IP,pattern = "miR-181a")),
            n_mir181b = sum(grepl(mir_IP,pattern = "miR-181b")),
            n_mir181c = sum(grepl(mir_IP,pattern = "miR-181c")),
            n_mir181d = sum(grepl(mir_IP,pattern = "miR-181d")),
            .groups = "keep") %>% subset (n_mir181 >0)

ago_bs_mir181_chi <- ago_bs_mir181_chi %>%
  subset(!duplicated(ago_bs_mir181_chi$BS_ID)) %>%
  left_join(., mir181_chi, by ="BS_ID") %>% makeGRangesFromDataFrame(keep.extra.columns = T)
```

4.2 Combine AGO binding sites with chimeric mir181 with mir181 enriched binding sites

I combine the mir181 Ago binding sites that we obtained above with the binding sites from the mir181 enriched Ago-eCLIP (see Methods/mir181-enriched_binding_site_definition). In order to do that, I first select binding sites from both conditions that do not overlap with any binding site from the other set. For the binding sites that overlap between the two conditions, I select the AGO mir181 binding sites and tag them as occuring in both sets. Then I combine the three subsets sets. The obtained union of mir181 binding sites from both conditions are our final mir181 binding sites.

```
# ---------------------------
# combine mir181 Ago BS and mir181 enriched Bs
# ---------------------------
# get only Ago mir181 Bs with now overlaps to enriched mir181 BS
only_ago_bs_mir181_chi <- subsetByOverlaps(ago_bs_mir181_chi, mir181_enriched, type = "any", invert = T)
only_ago_bs_mir181_chi$set <- "ago_bs_mir181_chi"

# get only enriched mir181 BS with now overlaps to Ago mir181 Bs
only_mir181_enriched <- subsetByOverlaps(mir181_enriched, ago_bs_mir181_chi,  type = "any", invert = T)
only_mir181_enriched$set <- "mir181_enriched"

# get only Ago mir181 BS overlapping with mir181 enriched BS
both_mir181_enriched_chi <- subsetByOverlaps(ago_bs_mir181_chi, mir181_enriched, type = "any")
both_mir181_enriched_chi$set <- "ago_bs_mir181_chi&mir181_enriched"

# combine all three sets
mir181_bs <- c(only_ago_bs_mir181_chi, only_mir181_enriched, both_mir181_enriched_chi)
mir181_bs$BS_ID <- NULL
mir181_bs$mir181BS_ID <- 1:NROW(mir181_bs)
```

5 Combine mir181 binding sites with differntial binding sites

Next, I combine the obtained mir181 binding sites with the results we obtained from the differntial binding between AGO binding sites and AGO binding sites with mir181 KO (see script Figure1/Differntial_Binding_AGO_BS_mir181_KO).

```
# ---------------------------
# combine with differential BS
# ---------------------------
# get overlaps with diff binding
# diff_overlap <- findOverlaps( mir181_bs, makeGRangesFromDataFrame(diff, keep.extra.columns = T) , typ
# # add differential information to mir181 binding sites
# d <- diff[,9:48]
# mcols(mir181_bs) <- cbind(mcols(mir181_bs), d[diff_overlap,])

# add only diff bs (these are Ago binding sites but not mir181 Binding sites)
# diff_only <- subsetByOverlaps( makeGRangesFromDataFrame(diff, keep.extra.columns = T), mir181_bs , ty
# mir181_bs_diff <- c(mir181_bs, diff_only)

#
# # ---------------------------
# # make venn diagram
# # ---------------------------
#
# # select downregulated BS from differential BS
```

```
# mir181_bs_diff_anygroup <- mir181_bs_diff %>% subset((down == T) | !is.na(set))
#
# # make venn
# venn_df <- data.frame(ago_bs_mir181_chi =
#                         ((mir181_bs_diff_anygroup$set == "ago_bs_mir181_chi") | (mir181_bs_diff_anygr
#                       mir181_enriched =
#                         ((mir181_bs_diff_anygroup$set == "mir181_enriched") | (mir181_bs_diff_anygrou
#                       mir181_ko_downregulated = (mir181_bs_diff_anygroup$down == T )
#                       ) %>%
#   mutate(ago_bs_mir181_chi = case_when(is.na(ago_bs_mir181_chi) ~ F, T~ago_bs_mir181_chi),
#          mir181_enriched = case_when(is.na(mir181_enriched) ~ F, T ~ mir181_enriched),
#          mir181_ko_downregulated = case_when(is.na(mir181_ko_downregulated) ~ F, T ~ mir181_ko_downre
#
#
#
# venn <- eulerr::euler(venn_df)
# p <- plot(venn, quantities = T, fills = c( lighten(farbe1, amount = 0.4), lighten(farbe2, amount = 0.
# p
#
# pdf(paste0(out, "Figure2b_Venn_overlaps_mirBs.pdf"), height = 3, width = 4 )
# p
# dev.off()

#Nikitas changed version
# make venn
# venn_dfnv <- data.frame(ago_bs_mir181_chi =
#                         ((mir181_bs$set == "ago_bs_mir181_chi") | (mir181_bs$set == "ago_bs_mir181_ch
#                       mir181_enriched =
#                         ((mir181_bs$set == "mir181_enriched") | (mir181_bs$set == "ago_bs_mir181_chi&
#                       ) %>%
#   mutate(ago_bs_mir181_chi = case_when(is.na(ago_bs_mir181_chi) ~ F, T~ago_bs_mir181_chi),
#          mir181_enriched = case_when(is.na(mir181_enriched) ~ F, T ~ mir181_enriched))
#
#
#
# vennnv <- eulerr::euler(venn_dfnv)
# pnv <- plot(vennnv, quantities = T, fills = c( "#0073C2", "#8D0391"))
# pnv
#
# pdf(paste0(out, "Figure2b_Venn_overlaps_mirBs_nv.pdf"), height = 3, width = 4 )
# pnv
# dev.off()
```

5.1 Differential binding both sets

Here I look at the overall binding changes between mir181 KO and WT for the three subsets of the mir181 binding sites.

```
# # ----------------------------
# # Compare regulation of BS during mir181 KO of the different subgroups of mir181 binding sites
# # ----------------------------
# names(mir181_bs_diff) <- 1:NROW(mir181_bs_diff)
#
# mir181_bs_diff <- as.data.frame(mir181_bs_diff)
```

```
#
#
# p1 <- ggplot(mir181_bs_diff, aes(x = resBs.log2FoldChange, color = set))+
#   geom_vline(xintercept = 0, color = "grey")+
#   stat_ecdf()+
#   scale_color_manual(values = c(farbe1, farbe14, farbe2, farbeneg))+
#   theme_paper()+
#   theme(legend.position = "top")
#
#
# p2 <- ggplot(mir181_bs_diff, aes(y = resBs.log2FoldChange, x = set, fill = set))+
#   geom_half_violin()+
#   geom_half_boxplot(side = "r")+
#   theme_paper()+
#   scale_fill_manual(values = c(farbe1, farbe14, farbe2, farbeneg))+
#   theme(legend.position = "top")+
#   scale_x_discrete(guide = guide_axis(angle = 25))
#
# p1
#
# p2
#
# ggsave(p1, filename =  paste0(out, "Figure_1J_ecdf_differntial_binding_vs_mir181BS.pdf"), width = 6,
# ggsave(p2, filename = paste0(out, "violin_differntial_binding_vs_mir181BS.pdf"), width = 10, height =
```

5.1.1 Statistical tests for differential binding changes

```
# t.tests of 3 sets against not bound
# -------------------------------------
# t1 <- t.test(x = mir181_bs_diff %>% subset(set == "mir181_enriched") %>% pull(resBs.lfcSE),
#        y = mir181_bs_diff %>% subset(is.na(set)) %>% pull(resBs.lfcSE))
# t1
#
# t.test(x = mir181_bs_diff %>% subset(set == "ago_bs_mir181_chi") %>% pull(resBs.lfcSE),
#        y = mir181_bs_diff %>% subset(is.na(set)) %>% pull(resBs.lfcSE))
#
# t.test(x = mir181_bs_diff %>% subset(set == "ago_bs_mir181_chi&mir181_enriched") %>% pull(resBs.lfcSE
#        y = mir181_bs_diff %>% subset(is.na(set)) %>% pull(resBs.lfcSE))
#
# # --> the p-values are driven strongly by the number of binding sites per set, for this reason the hi
#
# # check power of test
# # -------------------------------------
# pwr::pwr.t2n.test(n1 = mir181_bs_diff %>% subset(set == "mir181_enriched") %>% nrow(.),
#                   n2 = mir181_bs_diff %>% subset(is.na(set)) %>% nrow(.),
#                   d = abs(t1$estimate[1] -t1$estimate[2])/t1$stderr)
```

5.1.1.1 T-tests

```
# Kolmogorov-Smirnov Tests
# -------------------------
# ks.test(x = mir181_bs_diff %>% subset(set == "mir181_enriched") %>% arrange(resBs.lfcSE) %>%pull(resB
```

```
#         y = mir181_bs_diff %>% subset(is.na(set) ) %>% arrange(resBs.lfcSE) %>%pull(resBs.lfcSE))
#
# ks.test(x = mir181_bs_diff %>% subset(set == "ago_bs_mir181_chi") %>% arrange(resBs.lfcSE) %>%pull(re
#         y = mir181_bs_diff %>% subset(is.na(set) ) %>% arrange(resBs.lfcSE) %>%pull(resBs.lfcSE))
#
# ks.test(x = mir181_bs_diff %>% subset(set == "ago_bs_mir181_chi&mir181_enriched") %>% arrange(resBs.l
#         y = mir181_bs_diff %>% subset(is.na(set) ) %>% arrange(resBs.lfcSE) %>%pull(resBs.lfcSE))
```

5.1.1.2 Kolmogorov-Smirnov Tests

5.2 Venn bound genes from all sets

6 pie chart mir181 enriched set

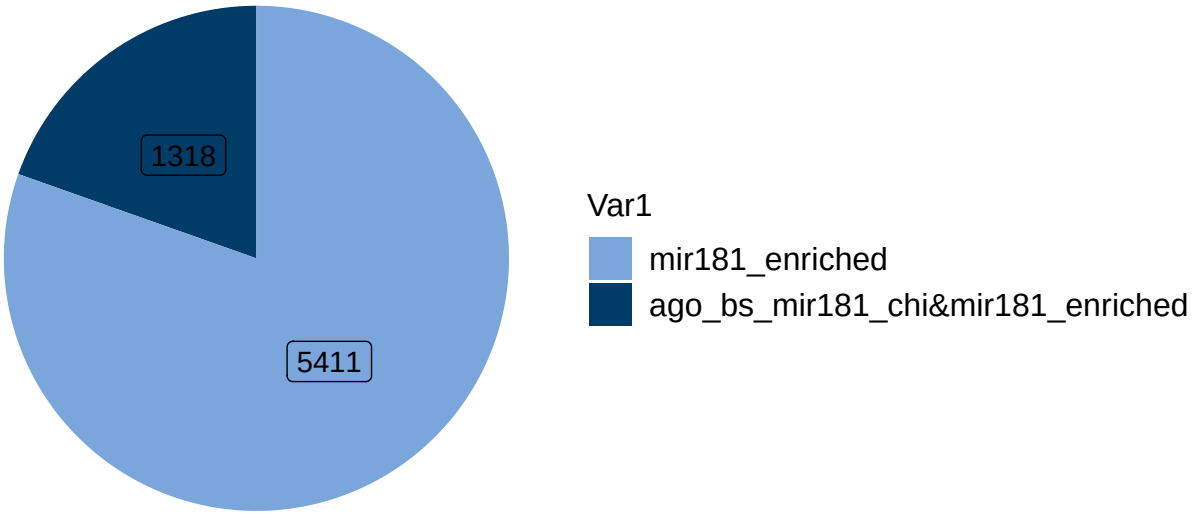
```
# ----------------------------------------
# Compare Ago2 mir181 BS and mir181 enriched BS
# Figure 2 b
# ----------------------------------------

names(mir181_bs) <- 1:NROW(mir181_bs)

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

mir181_enriched_set_df <-   table(mir181_enriched_set$set) %>%
  as.data.frame(.)

p <- ggplot(mir181_enriched_set_df, aes(y=Freq, x="", fill=Var1)) +
     geom_col()+
     coord_polar(theta="y") +
  #    xlim(c(2, 4)) +
  geom_label(aes( fill=Var1, label = Freq),
             position = position_stack(vjust = 0.5),
             show.legend = FALSE) +
  scale_fill_manual(values = c (farbe6, farbe4)) +
  theme_paper() +
  theme_nice_pie() +
  #theme(legend.position = "none") +
  guides(fill = guide_legend(reverse = TRUE)) +
  labs(y = NULL,
       x = NULL)
p
```




```
ggsave(p, filename = paste0(out, "Figure2b_pie_miR181_enriched_BS.pdf"), width = unit(8, "cm"), height =
```

7 Save output

```
saveRDS(mir181_bs, paste0(out, "mir181_bs.rds"))

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

# Supplementary table 2
write_csv(t, paste0(out, "STable2_mir181_enriched_binding_sites.csv"))

table(mir181_bs$set)
```

```
##
##                 ago_bs_mir181_chi ago_bs_mir181_chi&mir181_enriched
##                              7117                              1318
##                   mir181_enriched
##                              5411
```

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] ggpubr_0.6.0         gghalves_0.1.4       eulerr_7.0.0
##  [4] colorspace_2.1-0     GenomicRanges_1.50.2 GenomeInfoDb_1.34.9
##  [7] IRanges_2.32.0       S4Vectors_0.36.2     BiocGenerics_0.44.0
## [10] lubridate_1.9.2      forcats_1.0.0        stringr_1.5.0
## [13] dplyr_1.1.2          purrr_1.0.1          readr_2.1.4
## [16] tidyr_1.3.0          tibble_3.2.1         ggplot2_3.4.2
## [19] tidyverse_2.0.0      knitr_1.43
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.11            here_1.0.1             rprojroot_2.0.3
##  [4] digest_0.6.33          utf8_1.2.3             R6_2.5.1
##  [7] backports_1.4.1        evaluate_0.21          highr_0.10
## [10] pillar_1.9.0           zlibbioc_1.44.0        rlang_1.1.1
## [13] rstudioapi_0.15.0      car_3.1-2              rmarkdown_2.23
## [16] textshaping_0.3.6      labeling_0.4.2         bit_4.0.5
## [19] RCurl_1.98-1.12        munsell_0.5.0          broom_1.0.5
## [22] compiler_4.2.2         xfun_0.39              pkgconfig_2.0.3
## [25] systemfonts_1.0.4      htmltools_0.5.5        tidyselect_1.2.0
## [28] GenomeInfoDbData_1.2.9 fansi_1.0.4            crayon_1.5.2
## [31] tzdb_0.4.0             withr_2.5.0            bitops_1.0-7
## [34] grid_4.2.2             gtable_0.3.3           lifecycle_1.0.3
## [37] magrittr_2.0.3         scales_1.2.1           vroom_1.6.3
## [40] cli_3.6.1              stringi_1.7.12         carData_3.0-5
## [43] farver_2.1.1           XVector_0.38.0         ggsignif_0.6.4
## [46] ragg_1.2.5             generics_0.1.3         vctrs_0.6.3
## [49] tools_4.2.2            bit64_4.0.5            glue_1.6.2
## [52] hms_1.1.3              parallel_4.2.2         abind_1.4-5
## [55] fastmap_1.1.1          yaml_2.3.7             timechange_0.2.0
## [58] rstatix_0.7.2
```