Figure 5 YBX1

Riccardo Mosca

2025-05-13

This markdown show how to generate Figure S5

Loading packages and required data: YBX1 metafile with binding sites of RAPseq Input and T7 input, generated in the script “YBX1_metafile_annotation

```
library(eulerr)
```

```
## Warning: package 'eulerr' was built under R version 4.3.3
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
library(reshape2)
library(gplots)
```

```
## Warning: package 'gplots' was built under R version 4.3.3
```

```
## Registered S3 method overwritten by 'gplots':
##   method    from
##   plot.venn eulerr
```

```
##
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:eulerr':
##
##     venn
```

```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(tidyverse)
```

```
## Warning: package 'purrr' was built under R version 4.3.3
```

```
## Warning: package 'lubridate' was built under R version 4.3.3
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse
2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ lubridate 1.9.4     ✓ tibble    3.2.1
## ✓ purrr     1.0.4     ✓ tidyr     1.3.1
```

```
## ── Conflicts ──────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors
```

```
library(ggpubr)
library(dendextend)
```

```
## Warning: package 'dendextend' was built under R version 4.3.3
```

```
##
## ---------------------
## Welcome to dendextend version 1.19.0
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at:
https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
##   https://stackoverflow.com/questions/tagged/dendextend
##
##  To suppress this message use:
suppressPackageStartupMessages(library(dendextend))
## ---------------------
##
##
## Attaching package: 'dendextend'
##
## The following object is masked from 'package:ggpubr':
##
##     rotate
##
## The following object is masked from 'package:stats':
##
##     cutree
```

```
library(dplyr)
library(gtools)
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.3.3
```

```
## corrplot 0.95 loaded
```

```
library(clusterProfiler)
```

```
## Warning: package 'clusterProfiler' was built under R version 4.3.3
```

```
##
## clusterProfiler v4.10.1  For help: https://yulab-smu.top/biomedical-
knowledge-mining-book/
##
## If you use clusterProfiler in published research, please cite:
## T Wu, E Hu, S Xu, M Chen, P Guo, Z Dai, T Feng, L Zhou, W Tang, L Zhan, X
Fu, S Liu, X Bo, and G Yu. clusterProfiler 4.0: A universal enrichment tool
for interpreting omics data. The Innovation. 2021, 2(3):100141
##
## Attaching package: 'clusterProfiler'
##
## The following object is masked from 'package:purrr':
##
##     simplify
##
## The following object is masked from 'package:stats':
##
##     filter
```

```
library(org.Hs.eg.db)
```

```
## Loading required package: AnnotationDbi
## Loading required package: stats4
## Loading required package: BiocGenerics
##
## Attaching package: 'BiocGenerics'
##
## The following objects are masked from 'package:lubridate':
##
##     intersect, setdiff, union
##
## The following objects are masked from 'package:dplyr':
##
##     combine, intersect, setdiff, union
##
## The following objects are masked from 'package:stats':
##
##     IQR, mad, sd, var, xtabs
##
## The following objects are masked from 'package:base':
##
##     anyDuplicated, aperm, append, as.data.frame, basename, cbind,
##     colnames, dirname, do.call, duplicated, eval, evalq, Filter, Find,
##     get, grep, grepl, intersect, is.unsorted, lapply, Map, mapply,
##     match, mget, order, paste, pmax, pmax.int, pmin, pmin.int,
##     Position, rank, rbind, Reduce, rownames, sapply, setdiff, sort,
##     table, tapply, union, unique, unsplit, which.max, which.min
##
## Loading required package: Biobase
## Welcome to Bioconductor
```

```
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
##
## Loading required package: IRanges
## Loading required package: S4Vectors
##
## Attaching package: 'S4Vectors'
##
## The following object is masked from 'package:clusterProfiler':
##
##     rename
##
## The following objects are masked from 'package:lubridate':
##
##     second, second<-
##
## The following objects are masked from 'package:dplyr':
##
##     first, rename
##
## The following object is masked from 'package:tidyr':
##
##     expand
##
## The following object is masked from 'package:gplots':
##
##     space
##
## The following object is masked from 'package:utils':
##
##     findMatches
##
## The following objects are masked from 'package:base':
##
##     expand.grid, I, unname
##
##
## Attaching package: 'IRanges'
##
## The following object is masked from 'package:clusterProfiler':
##
##     slice
##
## The following object is masked from 'package:lubridate':
##
##     %within%
##
## The following objects are masked from 'package:dplyr':
```

```
##
##     collapse, desc, slice
##
## The following object is masked from 'package:purrr':
##
##     reduce
##
##
## Attaching package: 'AnnotationDbi'
##
## The following object is masked from 'package:clusterProfiler':
##
##     select
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(rrvgo)
```

```
## Warning: package 'rrvgo' was built under R version 4.3.3
```

```
YBX1<-
read.table(file="/Users/riccardomosca/Desktop/RAPseq_PAPER/PEAKs/ANNOTATED/T7
_Fig5/Ybx1_T7_scored_annotated.txt", header=T, stringsAsFactors = F)
head(YBX1)
```

```
##    chr   start     end                peak_ID       RBPs strand
## 1 chr1 1043290 1043298 chr1_1043277_1043341_+ Ybx1_final      +
## 2 chr1 1046002 1046010 chr1_1045989_1046039_+ Ybx1_final      +
## 3 chr1 1046657 1046665 chr1_1046652_1046673_+ Ybx1_final      +
## 4 chr1 1049916 1049924 chr1_1049902_1049932_+ Ybx1_final      +
## 5 chr1 1217314 1217322 chr1_1217305_1217329_- Ybx1_final      -
## 6 chr1 1228604 1228612 chr1_1228566_1228658_- Ybx1_final      -
##   Ybx1_rep1_signal Ybx1_rep2_signal halo_signal input_signal
T7Ybx1_rep1_signal
## 1               34               41          16           17
4
## 2               43               24          12           10
10
## 3               27               26          17           18
11
## 4               23               29          10            7
6
## 5               21               21           4           11
5
## 6               27               30          12           14
2
##   T7Ybx1_rep2_signal T7halo_signal T7input_signal
## 1                  4             9       10.00000
## 2                  3             4        3.00000
```

```
## 3                  29             8       12.66667
## 4                   6             6        9.00000
## 5                   2             2        0.50000
## 6                   3             3       13.50000
##
pos_fa
## 1
GAGGGGGGGCTTGTGGGACCACTGAGCCCCTGTGTCCTTCCCAGACCAGGCCCCGTCCCCATGCCTCGGGGTGCAGT
GTGCATTTGGGGCGACGTGTGCTGTGAAGAACGGGCAGGCAGCGTGTGAATGCCTGCAGGCGTGCTCGAGCCTCTAC
GATCCTGTGTGCGGCAGCGACGGCGTCACATACGGCAGCGCGTGCG
## 2
TCATGGGGTGGGGTGGGGTCACCCGAGCCACAGAGGTTTCCCATGCCCGTGCCCCAGACGCTTCTGCGCCTGCGACC
TGTGCGGAGATGCGCTGTGAGTTCGGTGCGCGGTGCGTGGAGGAGTCTGGCTCAGCCCACTGTGTCTGCCCGATGCT
CACCTGTCCAGAGGCCAACGCTACCAAGGTGAGGGGTGTGGGATGT
## 3
CCACTGCCAGCGTCCCCAGGACCACCGTGTGGCCCGTGCTGACGGTGCCCCCCACGGCACCCTCCCCTGCACCCAGC
CTGGTGGCGTCCGCCTTTGGTGAATCTGGCAGCACTGATGGAAGCAGCGATGAGGAACTGAGCGGGGACCAGGAGGC
CAGTGGGGGTGGCTCTGGGGGTGAGCAGGGATCAAGGACTTGGGGT
## 4
GAGTGGGACCCCGGGGCCTGTGGGCGGTACCCAACCGACGCCTCCTGGGACCTCGGTCCCGGTCCCGTCTTCCTCCA
TCCAGGACCAACCTGTGCCGATGAGAAGAGCCCCTGCCAGCCCAACCCCTGCCATGGGGCGGCGCCCTGCCGTGTGC
TGCCCGAGGGTGGTGCTCAGTGCGAGTGCCCCCTGGGGCGTGAGGG
## 5
GAAGCGAGATTTCGTTCTCAGAAGTGAGATGCCACGATTTCGAGGGACCAGGGAGGAGGCGGCGCCGCGGGGCCACA
GCCCAGCCCCGCGCCCCGACCGCGTCACAGCCAAAGCCCCGCCGGGGTGCGCGCTGCAGAGGGGACACGCCGCTCAT
CAACTGGGGCAGCCGCGGTCGGTCGGCCGGTGGCGGGCGGCAGGGA
## 6
CTCCGCGTCCTCATCAAAGCCACCCAGGTCCTTGCCTAGGAAGACCTCCTGGTGGAAGCCGCGATTGAGGTGCCCGT
CCATCTCCAGCTTCACCCCGTTCAGGTGGTCTGGGGGCAGGATCTCATTCTCCTCCCTGTTGGCTACTCTCTCTCGA
GTGGACGAGTGGTTGGCAGGCCGTGCAGACGCGTCCATCAGAAGGA
##
neg_fa
## 1
GAGGACTCGGAGGACGGGCCGTGTGTCTGTGACTTCAGCTGCCAGAGTGTCCCAGGCAGCCCGGTGAGCTCTGTACC
CCTGGCTCTCGGCGGGCGGCGGGGACGGGGCTGCGGCCGCTCACACTGACACCACCCTCCAGGTGTGCGGCTCAGAT
GGGGTCACCTACAGCACCGAGTGTGAGCTGAAGAAGGCCAGGTGTG
## 2
GCTCTGATGGGAAGACGCCCTCGCTGGACGCAGAGGGCTCCAACTGCCCCGGTGAGTGGACGGCTGGGCGAGGGGAG
TGTGAGGATAGCCTGGGCTCGGCCGAGGTGCTGCCCCCTCGCCTGGGCAGCAGGTCAGTGCCGGGGGTTATGGTCTT
GGGACTCGGCCCCCTCAAACATGTGCGTGCCGGGGACCCCACGCCT
## 3
CCCTGCCACAGCTGGACGACCTCTTCCGGAATTCAGACGTCAAGAAGGATTTTCGGAGTGTCCGCTTGCGGGACCTG
GGGCCCGGCAAATCCGTCCGCGCCATTGTGGATGTGCACTTTGACCCCAGTGAGACCTGCACCCTGGACCCTTCCTG
GGAGGCAATGGGTGGGGGATGCCTGGGGCTCTGCCATGCTCAGAGC
## 4
TGTTGGGGGAGTCCCCGGTGAGTGCTCTGGGCCGCGAGGGGACTCCCGCTGCTGCCTGCTCTTCCTCCTCGGGCGGC
AGCCCCGCCCCTGCCGGCGCTCACGGAGCTGTTTTTCTGTCCTGTTCTCTTGGCCGCCTGCCCTGTCCTCTGCCTCC
TCTGCCTCCCTGCTCTCTGCTCTCGCTCTGCAACCCCACCCGCTCT
## 5
AAGGGGCTGAGGATGGAGCCCGGCCAGGCTCGCCTGTCTCTGATCCGTCTGAACCTAAACGCCAACAACGGCCATGC
```

```
AGCCTGGTGGACACCGCTGAGCAAACGCCCCAGTGACCAGCCCAGATGGAGTCTCAGGCCAGACACCAGCACAGCTT
CCTGCCTCAGGCCCAGATCCACCAGCCCCTCCCGCCACAGCACCTC
## 6
GTCTTTTTTTTACAGACACTGAGTTTTCAGCCCTGTCTCTCCGTCTTTCCATCTCTCTCTCGAGCTGTCAGCACCTC
CAGGGGCTAACCTGTCCCCATCGCCCAACACCTGCAGCACAGCTTTCCTGTGGGGCCCGGCTCTGTCCCTCCTGGCA
TCATGCCACTGACCCATGGGGTTAGGGTCACTCTCCAGACCCTGAC
##   FCH_rep1 FCH_rep2 FCI_rep1 FCI_rep2 FCH_T7rep1 FCH_T7rep2 FCI_T7rep1
## 1 2.125000 2.562500 2.000000 2.411765  0.4444444  0.4444444  0.4000000
## 2 3.583333 2.000000 4.300000 2.400000  2.5000000  0.7500000  3.3333333
## 3 1.588235 1.529412 1.500000 1.444444  1.3750000  3.6250000  0.8684211
## 4 2.300000 2.900000 3.285714 4.142857  1.0000000  1.0000000  0.6666667
## 5 5.250000 5.250000 1.909091 1.909091  2.5000000  1.0000000 10.0000000
## 6 2.250000 2.500000 1.928571 2.142857  0.6666667  1.0000000  0.1481481
##   FCI_T7rep2 Mean_FCs_rep1 Mean_FCs_rep2 Mean_FCs_T7rep1 Mean_FCs_T7rep2
## 1  0.4000000      2.062500      2.487132       0.4222222       0.4222222
## 2  1.0000000      3.941667      2.200000       2.9166667       0.8750000
## 3  2.2894737      1.544118      1.486928       1.1217105       2.9572368
## 4  0.6666667      2.792857      3.521429       0.8333333       0.8333333
## 5  4.0000000      3.579545      3.579545       6.2500000       2.5000000
## 6  0.2222222      2.089286      2.321429       0.4074074       0.6111111
##     BS_rep1   BS_rep2  BS_T7rep1  BS_T7rep2        BS     BS_T7 Mean_FCH
## 1  8.600470 11.008706  0.6692064  0.6692064  9.804588 0.6692064 2.343750
## 2 17.705388  8.140967  7.5394740  1.1566871 12.923178 4.3480805 2.791667
## 3  5.957177  5.661263  3.0291117 11.6934950  5.809220 7.3613033 1.558824
## 4 10.176770 13.924420  1.6666667  1.6666667 12.050595 1.6666667 2.600000
## 5 12.612750 12.612750 11.2959683  2.5000000 12.612750 6.8979841 5.250000
## 6  8.060425  9.285714  0.4074074  0.8078449  8.673069 0.6076262 2.375000
##   Mean_FCI Mean_FCH_T7 Mean_FCI_T7   MeanFC MeanFC_T7            gene_ID
## 1 2.205882   0.4444444   0.4000000 2.274816 0.4222222 ENSG00000188157.15
## 2 3.350000   1.6250000   2.1666667 3.070833 1.8958333 ENSG00000188157.15
## 3 1.472222   2.5000000   1.5789474 1.515523 2.0394737 ENSG00000188157.15
## 4 3.714286   1.0000000   0.6666667 3.157143 0.8333333 ENSG00000188157.15
## 5 1.909091   1.7500000   7.0000000 3.579545 4.3750000 ENSG00000078808.19
## 6 2.035714   0.8333333   0.1851852 2.205357 0.5092593 ENSG00000078808.19
##   feature gene_name      gene_type
## 1     CDS      AGRN protein_coding
## 2     CDS      AGRN protein_coding
## 3     CDS      AGRN protein_coding
## 4     CDS      AGRN protein_coding
## 5    3UTR      SDF4 protein_coding
## 6     CDS      SDF4 protein_coding
```

Figure S5A

```
# Only the RAPseq YBX1 binding sites are considered and compared with the
# signal on the T7RAPseq
YBX1$T7input_signal[YBX1$T7Input_signal == 0] <- min(YBX1[YBX1$T7Input_signal
!=
    0, "T7Input_signal"])
YBX1$T7halo_signal[YBX1$T7halo_signal == 0] <- min(YBX1[YBX1$T7halo_signal !=
```

```
0,
    "T7halo_signal"])



RAP_enrichments <- YBX1$peak_ID
T7RAP_enrichments <- YBX1[YBX1$T7Ybx1_rep1_signal/YBX1$T7input_signal > 1 &
YBX1$T7Ybx1_rep2_signal/YBX1$T7input_signal >
    1 & YBX1$T7Ybx1_rep1_signal/YBX1$T7halo_signal > 1 &
YBX1$T7Ybx1_rep2_signal/YBX1$T7halo_signal >
    1, "peak_ID"]

AB <- list(RAP_enrichments, T7RAP_enrichments)
names(AB) <- c("with modif", "without modif")
FigS5A <- euler(AB, shape = "circle")

plot(FigS5A, fills = c("#93ADD0", "#8CBDA0"), quantities = TRUE, edges = F)
```

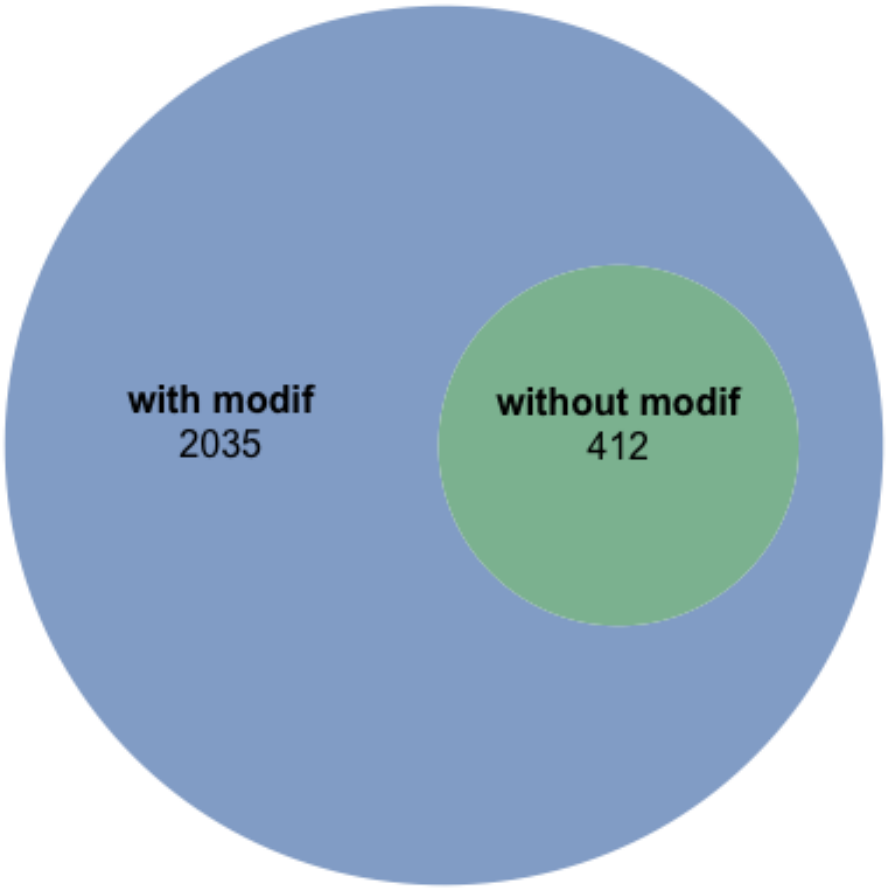
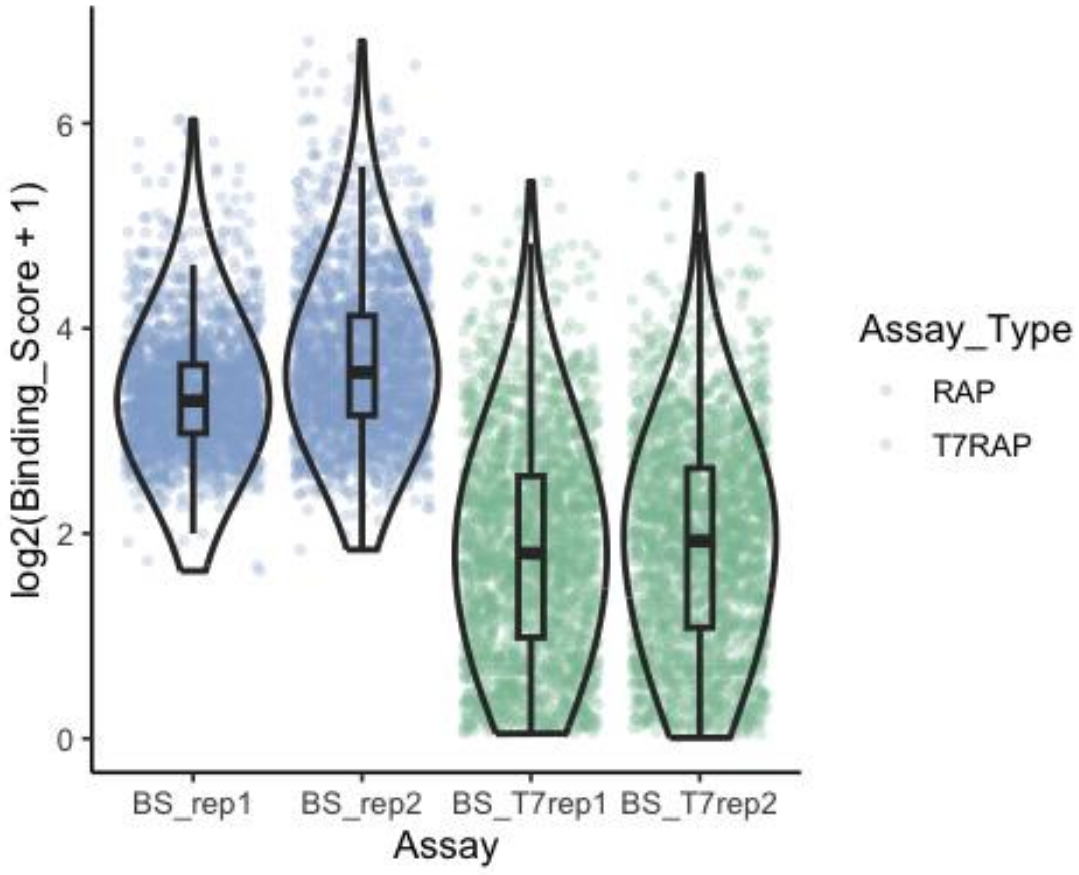


Figure S5B

```
FCs <- YBX1[, c("BS_rep1", "BS_rep2", "BS_T7rep1", "BS_T7rep2")]
FCs <- reshape2::melt(FCs, value.name = "Binding_Score", variable.name =
```

```
"Assay")
RAP <- rep("RAP", nrow(FCs)/2)
T7RAP <- rep("T7RAP", nrow(FCs)/2)
FCs$Assay_Type <- c(RAP, T7RAP)


plot <- ggplot(data = FCs, aes(x = Assay, y = log2(Binding_Score + 1))) +
geom_jitter(aes(color = Assay_Type),
    pch = 16, alpha = 0.25) + scale_color_manual(values = c("#93ADD0",
"#8ABEA0")) +
    theme_classic(base_size = 12.5)

FigS5B <- plot + geom_violin(trim = T, bw = 0.75, scale = "width", lwd = 1,
fill = NA) +
    geom_boxplot(outlier.shape = NA, width = 0.15, lwd = 1, fill = NA)

FigS5B
```




```
list <- list(BS_T71 = YBX1$BS_T7rep1, BS_T72 = YBX1$BS_T7rep2)

# run wilcox.test() for each, comparing to the overlap vector
wt <- lapply(names(list), function(nm) {
    y <- list[[nm]]
```

```
    test <- wilcox.test(YBX1$BS_rep1, y)
    data.frame(comparison = paste0("BS1_vs_", nm), W =
as.numeric(test$statistic),
        p.value = test$p.value)
})

# combine into one df
wt_df <- do.call(rbind, wt)
```

Figure S5C

```
# Looking for the top 10 motifs k=5
k = 5
pos_fa <- str_sub(YBX1$pos_fa, 75, 125)
kmer_table <- as.data.frame("YBX1")
STRINGS <- c((paste(pos_fa, collapse = "NN")))
kmer_table <- cbind(kmer_table, STRINGS)
kmer_table[] <- lapply(kmer_table, as.character)
bases <- c("A", "C", "T", "G")
kmers <- unite(as.data.frame(permutations(n = 4, r = k, v = bases,
repeats.allowed = T)),
    col = kmers, sep = "")

kmers <- cbind(kmers, str_count(kmer_table[1, 2], kmers[,
1])/length(kmer_table[1,
    2]))

rownames(kmers) <- kmers$kmers
pos_kmers_5 <- as.data.frame(kmers[, -1], row.names = rownames(kmers))
colnames(pos_kmers_5) <- "YBX1"



neg_fa <- str_sub(YBX1$neg_fa, 75, 125)
kmer_table <- as.data.frame("YBX1")
STRINGS <- c(as.character(paste(neg_fa, collapse = "NN")))
kmer_table <- cbind(kmer_table, STRINGS)
kmer_table[] <- lapply(kmer_table, as.character)
bases <- c("A", "C", "T", "G")
kmers <- unite(as.data.frame(permutations(n = 4, r = k, v = bases,
repeats.allowed = T)),
    col = kmers, sep = "")

kmers <- cbind(kmers, str_count(kmer_table[1, 2], kmers[,
1])/length(kmer_table[1,
    2]))

rownames(kmers) <- kmers$kmers
neg_kmers_5 <- as.data.frame(kmers[, -1], row.names = rownames(kmers))
```

```r
colnames(neg_kmers_5) <- "YBX1neg"


pos_kmers_5$KMERS <- rownames(pos_kmers_5)
neg_kmers_5$KMERS <- rownames(neg_kmers_5)
K5s_Ybx1 <- merge(pos_kmers_5, neg_kmers_5, by = "KMERS")

K5s_Ybx1 <- K5s_Ybx1[K5s_Ybx1$YBX1 >= 1, ]
K5s_Ybx1 <- K5s_Ybx1[K5s_Ybx1$YBX1neg >= 1, ]
K5s_Ybx1$Fold_Change <- K5s_Ybx1$YBX1/K5s_Ybx1$YBX1neg
# top10
K5s_Ybx1 <- K5s_Ybx1 %>%
    top_n(10, Fold_Change)
K5s_Ybx1 <- K5s_Ybx1[order(K5s_Ybx1$Fold_Change), ]
K5s_Ybx1$KMERS <- gsub("T", "U", K5s_Ybx1$KMERS)

barplot(K5s_Ybx1$Fold_Change, names.arg = K5s_Ybx1$KMERS, horiz = T, las = 1,
xlab = "Counts: Bound sites / Control sites",
    col = "#93ADD0", xlim = c(0, 10), space = 0.1)
abline(v = c(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10), lty = 2, lwd = 1)
```

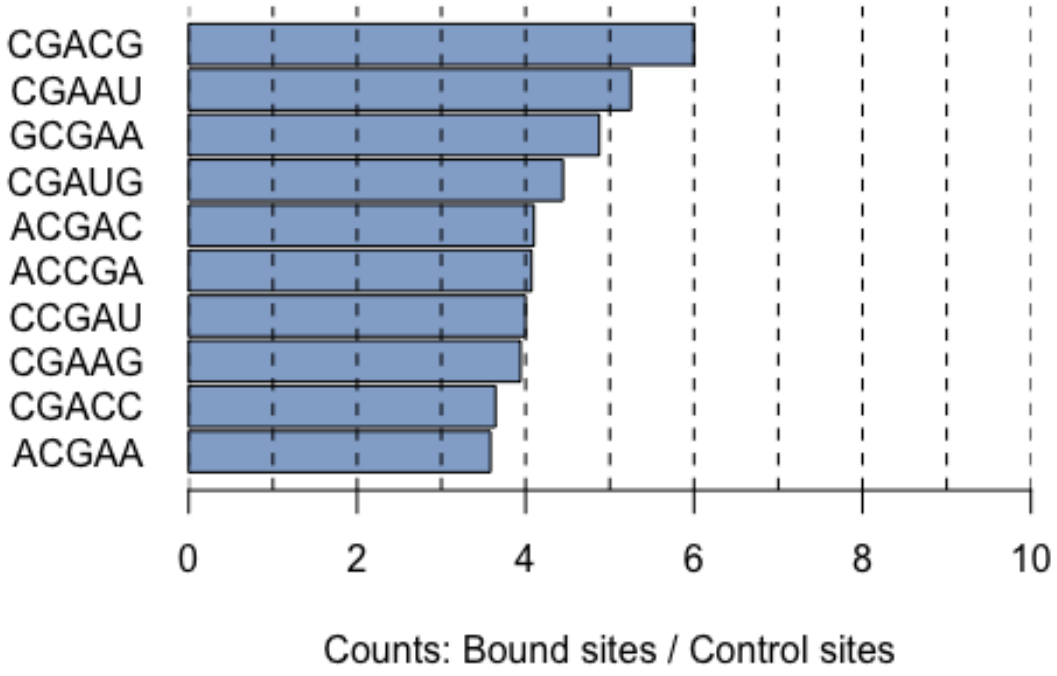
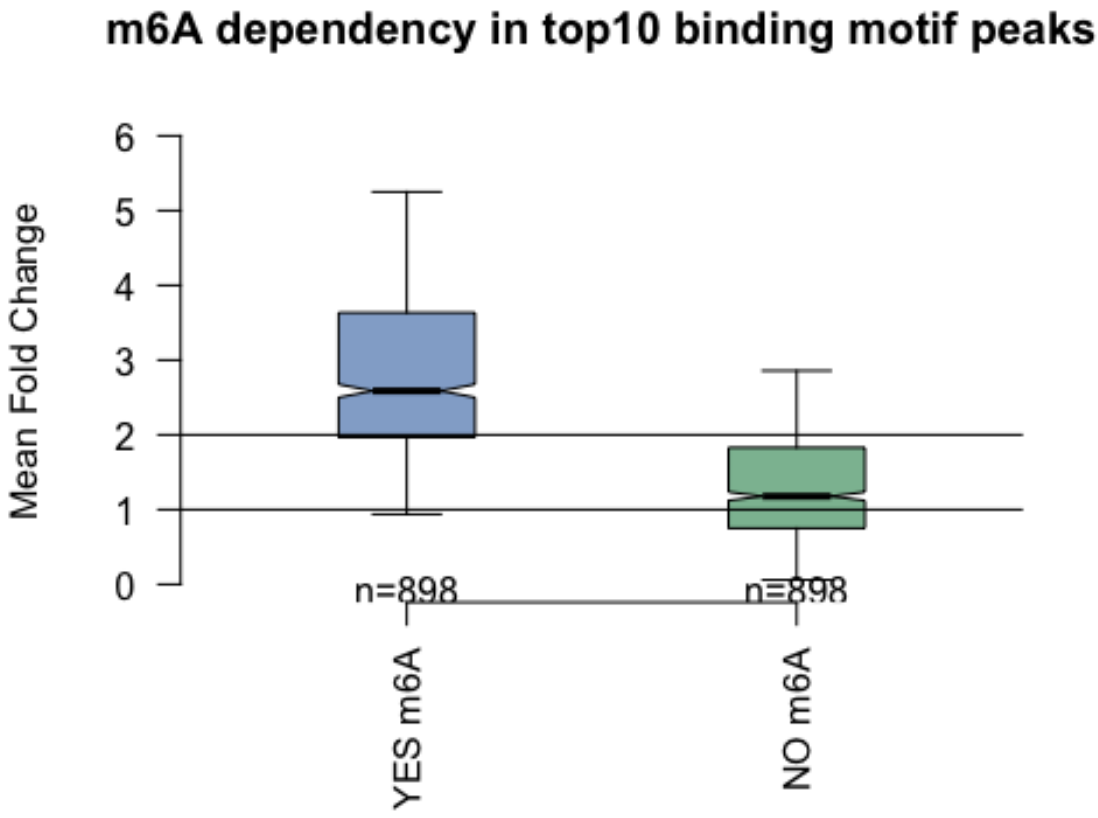


Figure S5D

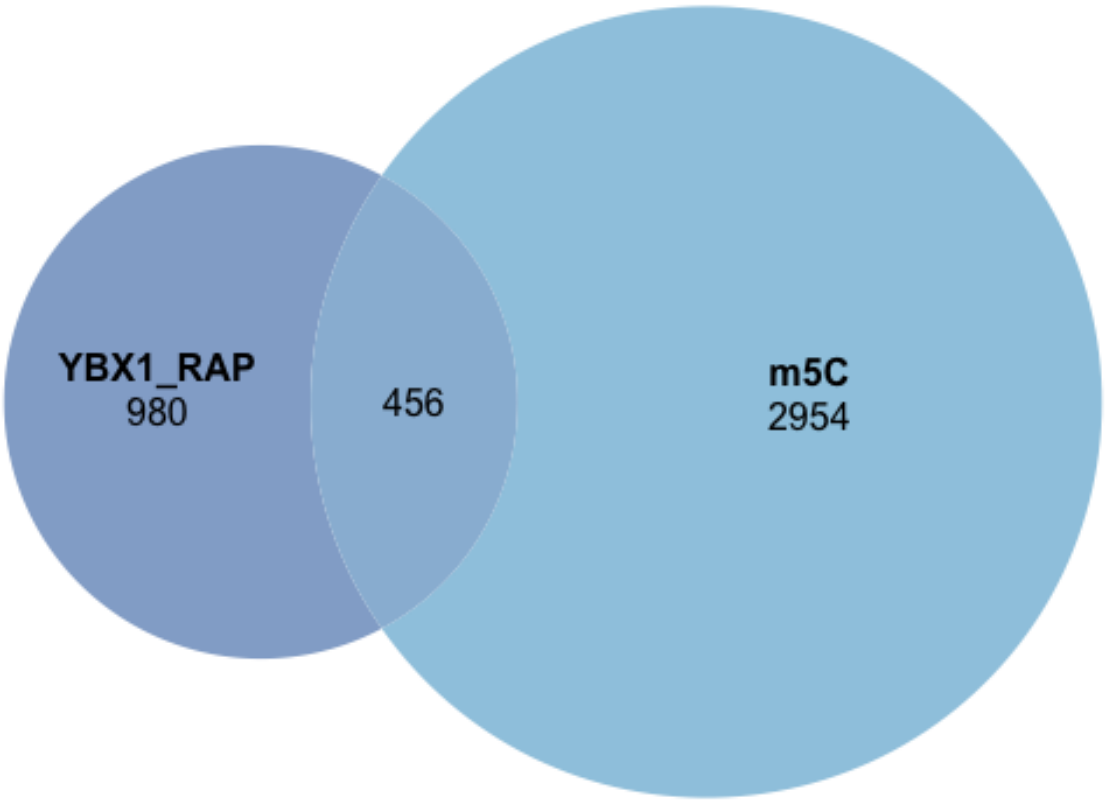
```r
YBX1_sub <-
YBX1[grep("CGACG|CGAAT|GCGAA|CGATG|ACGAC|ACCGA|CCGAT|CGAAG|CGACC|ACGAA",
    str_sub(YBX1$pos_fa)), ]


AB <- list(log2(YBX1_sub$MeanFC + 1), log2(YBX1_sub$MeanFC_T7 + 1))
AB <- list(YBX1_sub$MeanFC, YBX1_sub$MeanFC_T7)

names(AB) <- c("YES m6A", "NO m6A")



par(bty = "n")
boxplot2(AB, outline = F, range = 1, boxwex = 0.35, notch = T, las = 2, ylab
= "Mean Fold Change",
    ylim = c(0, 6), lty = 1, main = "m6A dependency in top10 binding motif
peaks",
    col = c("#93ADD0", "#8CBDA0"))
abline(h = c(1, 2))
```




```r
wt <- wilcox.test(AB$`YES m6A`, AB$`NO m6A`, alternative = "greater")
wt <- wt$p.value
```

Figure S5E

```
# Loading the YBX1_19 metafile and the annotated m5C generated in the
markdown
# 'YBX1_liftover'

m5C_annotated <-
read.table("/Users/riccardomosca/Desktop/RAPseq_PAPER/FIGUREs/FIGURE5/m5C_ann
otated.txt",
    sep = "\t", header = T)
YBX1_19 <-
read.table("/Users/riccardomosca/Desktop/RAPseq_PAPER/FIGUREs/FIGURE5/Ybx1_me
tafile_19.txt",
    sep = "\t", header = T)

# Considering only the genes dependent on the modifications
YBX1_19 <- YBX1_19 %>%
    filter(!peak_ID %in% T7RAP_enrichments)

gene_IDs <- YBX1_19 %>%
    mutate(gene_ID = strsplit(gene_ID, "\\.") %>%
        lapply(., function(x) x[1]) %>%
        unlist())
gene_IDs_Ybx1 <- gene_IDs %>%
    distinct(gene_ID, .keep_all = FALSE)

gene_IDs_m5C <- m5C_annotated %>%
    mutate(gene_ID = strsplit(gene_ID, "\\.") %>%
        lapply(., function(x) x[1]) %>%
        unlist())
gene_IDs_m5C <- gene_IDs_m5C %>%
    distinct(gene_ID, .keep_all = FALSE)

YBX1_RAP <- gene_IDs_Ybx1$gene_ID
m5C <- gene_IDs_m5C$gene_ID

AB <- list(YBX1_RAP, m5C)
names(AB) <- c("YBX1_RAP", "m5C")
v <- euler(AB, shape = "circle")


plot(v, fills = c("#93ADD0", "#9ECAE1"), quantities = TRUE, edges = F)
```


YBX1_RAP
980
456
m5C
2954