

# CUT&RUN BMDM WT vs KO

Domien Vanneste

2025-09-05 16:14:47 +0200

## Contents

<b>Introduction</b>	<b>2</b>
<b>Load Packages</b>	<b>3</b>
<b>nf-core/cutandrun</b>	<b>3</b>
<b>Homer</b>	<b>3</b>
Create “Tag Directory” with makeTagDirectory . . . . .	3
QC: Tag quantification and distribution . . . . .	4
Peak selection (WT vs KO) . . . . .	27
Annotate Peaks . . . . .	30
H3K27Ac at MafB peaks . . . . .	31
MafB target genes . . . . .	40

## Introduction

Despite substantial evidence pointing to MafB as an essential regulator of core macrophage identity, the genes it directly controls and the mechanisms underlying this regulation remain poorly understood. To address this question, we performed cleavage under targets and release using nuclease (CUT&RUN) for MafB on BMDMs from Mafbfl/fl and Lyz2CreMafbfl/fl mice. MafB and H3K27ac CUT&RUN was performed with a CUTANA ChIC/CUT&RUN Kit (EpiCypher, 141048) according to manufacturer's instructions, with modifications. BMDMs from Mafbfl/fl or Lyz2CreMafbfl/fl mice were washed with ice-cold PBS and fixed with 0.1 % formaldehyde in PBS for 2 min at room temperature. Fixation was quenched by adding glycine (Merck, 104691000) to 0.125 M. For each CUT&RUN sample, 1 x 10<sup>6</sup> fixed cells and 0.5 µg antibodies were added:

IgG (EpiCypher, 13-0042)

MafB1 (Sigma, HPA005653) MafB2 (Cell Signaling Technology, 41019) MafB3 (Proteintech, 20189-1-AP)

H3K27ac (Thermo Fisher Scientific, MA5-23516).

CUT&RUN libraries were prepared with a CUTANA™ CUT&RUN Library Prep Kit (EpiCypher, 141001) according to manufacturer's instructions. These libraries were sequenced on an NovaSeq 6000 (Illumina) sequencer on an S4 flow cell at 10 million reads per sample.

## Load Packages

```
suppressMessages({  
  library(ggplot2)  
  library(colorRamp2)  
  library(ComplexHeatmap)  
  library(readxl)  
  library(futile.logger)  
  library(grid)  
  library(rtracklayer)  
  library(dplyr)  
})
```

```
## Warning: package 'colorRamp2' was built under R version 4.4.3
```

## nf-core/cutandrun

The command used to launch the workflow was as follows:

```
nextflow run nf-core/cutandrun --input sample_list_all.csv --gtf genes.gtf --fasta genome.fa --peakcall
```

## Homer

### Create “Tag Directory” with makeTagDirectory

```
#IgG  
makeTagDirectory IgG_WT_R1/ IgG_WT_R1.target.markdup.sorted.bam  
makeTagDirectory IgG_KO_R1/ IgG_KO_R1.target.markdup.sorted.bam  
  
#MafB1  
makeTagDirectory MafB1_WT_R1/ MafB1_WT_R1.target.markdup.sorted.bam  
makeTagDirectory MafB1_WT_R2/ MafB1_WT_R2.target.markdup.sorted.bam  
makeTagDirectory MafB1_KO_R1/ MafB1_KO_R1.target.markdup.sorted.bam  
makeTagDirectory MafB1_KO_R2/ MafB1_KO_R2.target.markdup.sorted.bam  
  
#MafB2  
makeTagDirectory MafB2_WT_R1/ MafB2_WT_R1.target.markdup.sorted.bam  
makeTagDirectory MafB2_WT_R2/ MafB2_WT_R2.target.markdup.sorted.bam  
makeTagDirectory MafB2_KO_R1/ MafB2_KO_R1.target.markdup.sorted.bam  
makeTagDirectory MafB2_KO_R2/ MafB2_KO_R2.target.markdup.sorted.bam  
  
#MafB3  
makeTagDirectory MafB3_WT_R1/ MafB3_WT_R1.target.markdup.sorted.bam  
makeTagDirectory MafB3_WT_R2/ MafB3_WT_R2.target.markdup.sorted.bam  
makeTagDirectory MafB3_KO_R1/ MafB3_KO_R1.target.markdup.sorted.bam  
makeTagDirectory MafB3_KO_R2/ MafB3_KO_R2.target.markdup.sorted.bam
```

## QC: Tag quantification and distribution

MafB1

```
MafB1_peaks_rep1 <- read.table("MafB1_WT_R1.seacr.peaks.stringent.bed",
  header = FALSE, sep = "\t")

MafB1_peaks_rep2 <- read.table("MafB1_WT_R2.seacr.peaks.stringent.bed",
  header = FALSE, sep = "\t")

MafB1_peaks_consensus <- read.table("MafB1_WT.seacr.consensus.peak_counts.bed",
  header = FALSE, sep = "\t")

# peaks rep1
nrow(MafB1_peaks_rep1)

## [1] 11192

# peaks rep2
nrow(MafB1_peaks_rep2)

## [1] 16428

# consensus peaks
nrow(MafB1_peaks_consensus)

## [1] 19103

# shared peaks
length(which(MafB1_peaks_consensus[, 10] == 2))

## [1] 7991

MafB1_peaks_merged <- MafB1_peaks_consensus[MafB1_peaks_consensus[, 10] == 2, ]

split_scores <- strsplit(as.character(MafB1_peaks_merged$V6),
  ", ")

means <- sapply(split_scores, function(x) {
  if (all(is.na(x)))
    return(NA_real_)
  nums <- as.numeric(x)
  mean(nums, na.rm = TRUE)
})

MafB1_peaks_merged$V4 <- means

write.table(MafB1_peaks_merged, "MafB1_peaks_merged.bed", sep = "\t",
  quote = FALSE, row.names = FALSE, col.names = FALSE)
```

```

annotatePeaks.pl MafB1_peaks_merged.bed mm10 -d MafB1_WT_R1/ MafB1_WT_R2/ MafB1_KO_R1/ MafB1_KO_R2/ > MafB1_counts.txt

MafB1_counts <- read.table("MafB1_counts.txt", header = TRUE,
sep = "\t")

colnames(MafB1_counts)[colnames(MafB1_counts) == "PeakID..cmd.annotatePeaks.pl.MafB1_peaks_merged.bed.mm10"] <- "Norm tags"

colnames(MafB1_counts)[colnames(MafB1_counts) == "MafB1_WT_R1..Tag.Count.in.given.bp..5125124.0.Total..0"] <- "Rep. 1 Norm tags"

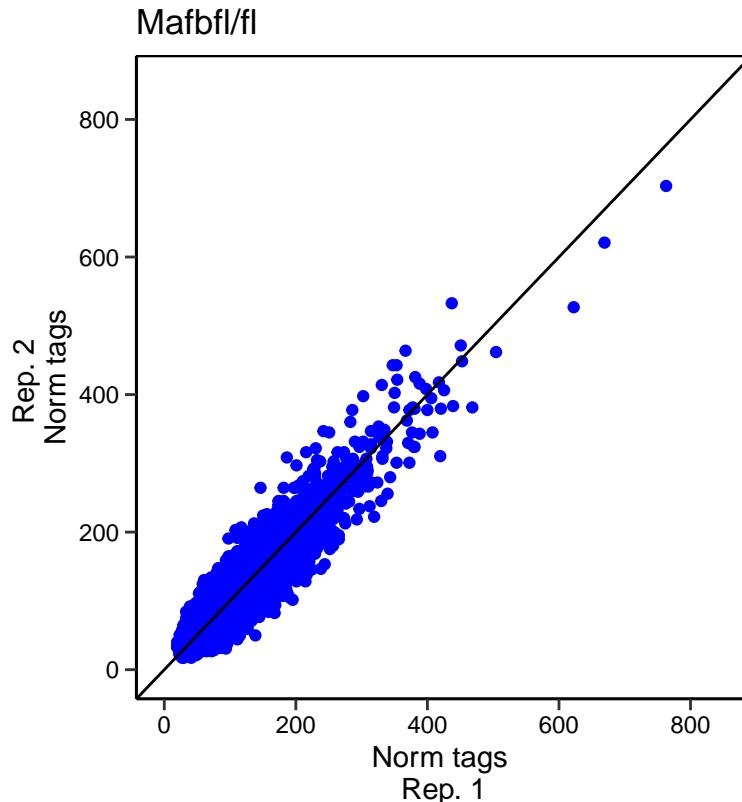
colnames(MafB1_counts)[colnames(MafB1_counts) == "MafB1_WT_R2..Tag.Count.in.given.bp..5218170.0.Total..0"] <- "Rep. 2 Norm tags"

colnames(MafB1_counts)[colnames(MafB1_counts) == "MafB1_KO_R1..Tag.Count.in.given.bp..5819980.0.Total..0"] <- "Rep. 1 KO Norm tags"

colnames(MafB1_counts)[colnames(MafB1_counts) == "MafB1_KO_R2..Tag.Count.in.given.bp..5208759.0.Total..0"] <- "Rep. 2 KO Norm tags"

ggplot(data = MafB1_counts, aes(x = MafB1_WT_R1, y = MafB1_WT_R2)) +
  geom_point(show.legend = FALSE, colour = "blue") + geom_abline(slope = 1) +
  ggtitle("Mafbfl/f1") + xlab("Norm tags \nRep. 1") + ylab("Rep. 2 \nNorm tags") +
  xlim(0, 850) + ylim(0, 850) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
"cm"), panel.border = element_rect(fill = NA, color = "black",
linetype = "solid"))

```



```

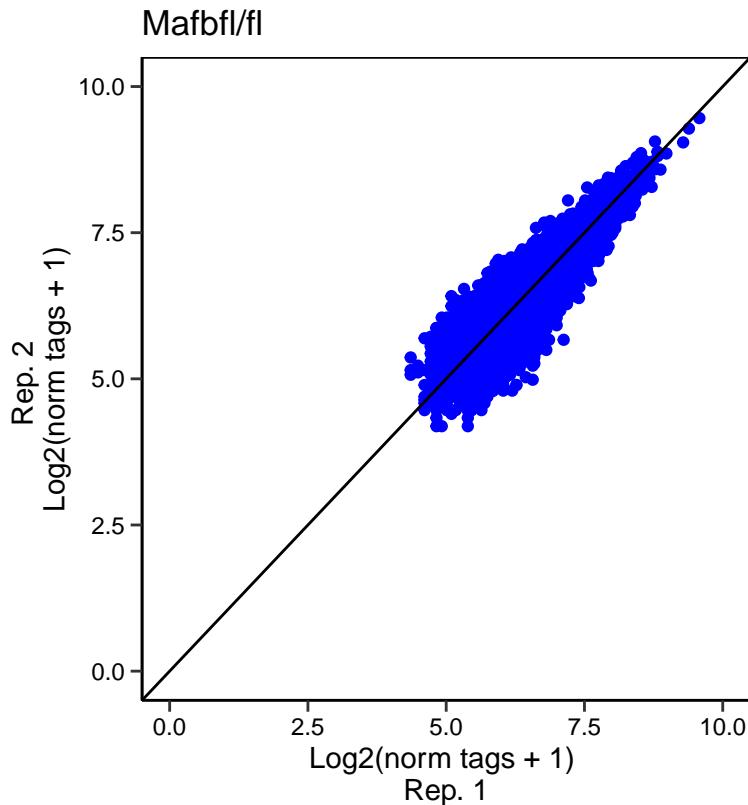
ggplot(data = MafB1_counts, aes(x = log2(MafB1_WT_R1 + 1), y = log2(MafB1_WT_R2 + 1))) + geom_point(show.legend = FALSE, colour = "blue") +

```

```

geom_abline(slope = 1) + ggtitle("Mafbfl/fl") + xlab("Log2(norm tags + 1) \nRep. 1") +
ylab("Rep. 2 \nLog2(norm tags + 1)") + xlim(0, 10) + ylim(0,
10) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
"cm"), panel.border = element_rect(fill = NA, color = "black",
linetype = "solid"))

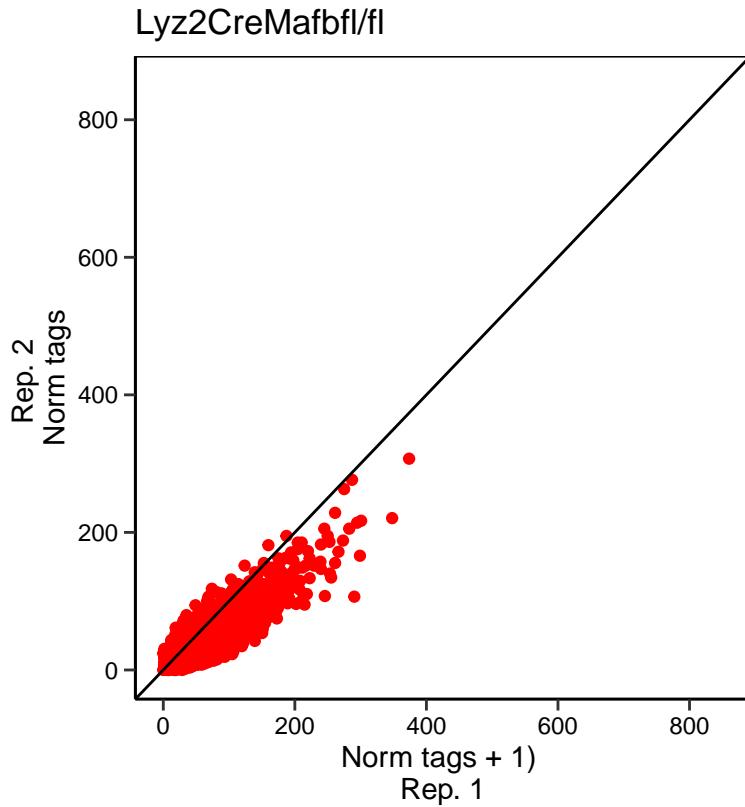
```



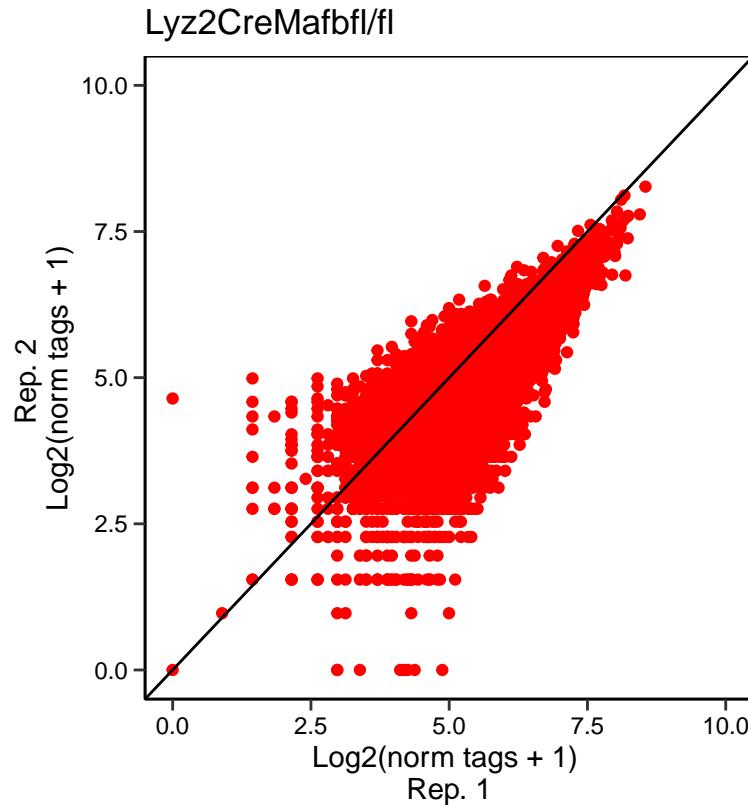
```

ggplot(data = MafB1_counts, aes(x = MafB1_KO_R1, y = MafB1_KO_R2)) +
  geom_point(show.legend = FALSE, colour = "red") + geom_abline(slope = 1) +
  ggtitle("Lyz2CreMafbfl/fl") + xlab("Norm tags + 1) \nRep. 1") +
  ylab("Rep. 2 \nNorm tags") + xlim(0, 850) + ylim(0, 850) +
  theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))

```



```
ggplot(data = MafB1_counts, aes(x = log2(MafB1_KO_R1 + 1), y = log2(MafB1_KO_R2 + 1))) + geom_point(show.legend = FALSE, colour = "red") +
  geom_abline(slope = 1) + ggtitle("Lyz2CreMafbfl/fl") + xlab("Log2(norm tags + 1) \nRep. 1") +
  ylab("Rep. 2 \nLog2(norm tags + 1)") + xlim(0, 10) + ylim(0, 10) +
  theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
annotatePeaks.pl MafB1_peaks_merged.bed mm10 -size 4000 -hist 10 -d MafB1_WT_R1/ MafB1_WT_R2/ MafB1_KO_R1/ MafB1_KO_R2/
```

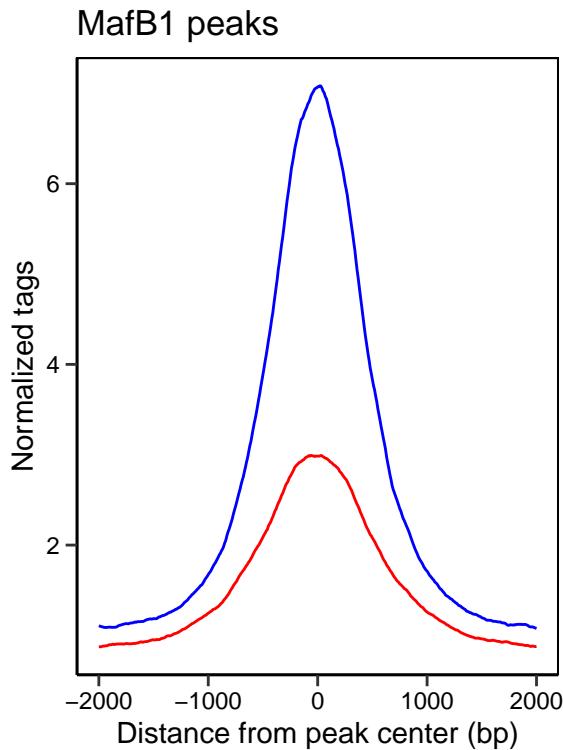
```
MafB1_hist <- read.table("MafB1_hist.txt", header = TRUE, sep = "\t")
```

```
colnames(MafB1_hist)[colnames(MafB1_hist) == "Distance.from.Center..cmd.annotatePeaks.pl.MafB1_peaks_merged"] <- c("Distance.from.Center", "MafB1_hist")
```

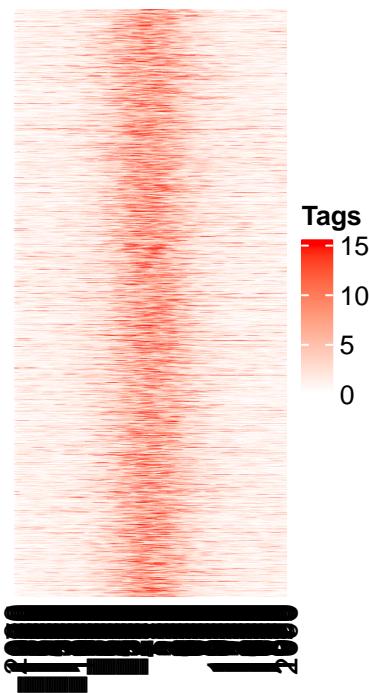
```
MafB1_hist$MafB1_WT <- rowMeans(MafB1_hist[, c("MafB1_WT_R1..Coverage", "MafB1_WT_R2..Coverage")])
```

```
MafB1_hist$MafB1_KO <- rowMeans(MafB1_hist[, c("MafB1_KO_R1..Coverage", "MafB1_KO_R2..Coverage")])
```

```
ggplot(data = MafB1_hist, aes(x = Distance.from.Center, y = MafB1_WT)) +
  geom_line(show.legend = FALSE, colour = "blue") + geom_line(show.legend = FALSE, colour = "red", aes(x = Distance.from.Center, y = MafB1_KO)) +
  ggtitle("MafB1 peaks") + xlab("Distance from peak center (bp)") +
  ylab("Normalized tags") + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



## Mafbfl/fl



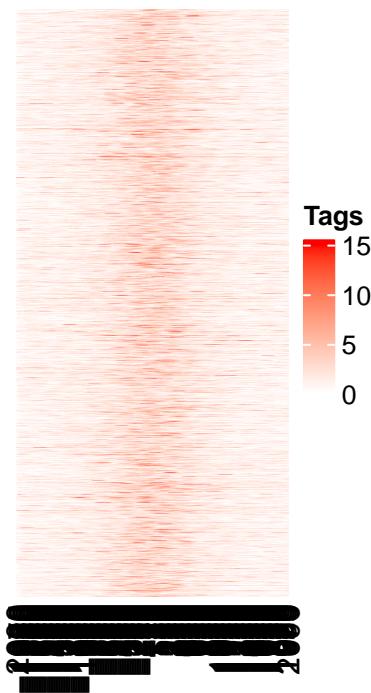
```
col_fun = colorRamp2(c(0, 15), c("white", "red"))

Heatmap(MafB1_K0_heatmap, col_fun, cluster_columns = F, cluster_rows = F,
        heatmap_legend_param = list(title = "Tags"), use_raster = TRUE,
        raster_quality = 10, , column_title = "LyzCreMafbfl/fl")
```

```
## Warning: The input is a data frame-like object, convert it to a matrix.

## 'magick' package is suggested to install to give better rasterization.
##
## Set 'ht_opt$message = FALSE' to turn off this message.
```

## LyzCreMafbfl/fl



## MafB2

```
MafB2_peaks_rep1 <- read.table("MafB2_WT_R1.seacr.peaks.stringent.bed",
  header = FALSE, sep = "\t")

MafB2_peaks_rep2 <- read.table("MafB2_WT_R2.seacr.peaks.stringent.bed",
  header = FALSE, sep = "\t")

MafB2_peaks_consensus <- read.table("MafB2_WT.seacr.consensus.peak_counts.bed",
  header = FALSE, sep = "\t")

# peaks rep1
nrow(MafB2_peaks_rep1)

## [1] 10549

# peaks rep2
nrow(MafB2_peaks_rep2)

## [1] 11478

# consensus peaks
nrow(MafB2_peaks_consensus)

## [1] 14436
```

```

# shared peaks
length(which(MafB2_peaks_consensus[, 10] == 2))

## [1] 7256

MafB2_peaks_merged <- MafB2_peaks_consensus[MafB2_peaks_consensus[,
  10] == 2, ]

split_scores <- strsplit(as.character(MafB2_peaks_merged$V6),
  ",") 

means <- sapply(split_scores, function(x) {
  if (all(is.na(x)))
    return(NA_real_)
  nums <- as.numeric(x)
  mean(nums, na.rm = TRUE)
})

MafB2_peaks_merged$V4 <- means

write.table(MafB2_peaks_merged, "MafB2_peaks_merged.bed", sep = "\t",
  quote = FALSE, row.names = FALSE, col.names = FALSE)

```

annotatePeaks.pl MafB2\_peaks\_merged.bed mm10 -d MafB2\_WT\_R1/ MafB2\_WT\_R2/ MafB2\_KO\_R1/ MafB2\_KO\_R2/ > MafB2\_peaks\_merged.bed

```

MafB2_counts <- read.table("MafB2_counts.txt", header = TRUE,
  sep = "\t")

colnames(MafB2_counts)[colnames(MafB2_counts) == "PeakID..cmd.annotatePeaks.pl.MafB2_peaks_merged.bed.mm10"] <- "MafB2_WT_R1..Tag.Count.in.given.bp..5577784.0.Total.."

colnames(MafB2_counts)[colnames(MafB2_counts) == "MafB2_WT_R1..Tag.Count.in.given.bp..5577784.0.Total.."] <- "MafB2_WT_R1..Tag.Count.in.given.bp..5577784.0.Total.."

colnames(MafB2_counts)[colnames(MafB2_counts) == "MafB2_WT_R2..Tag.Count.in.given.bp..6205913.0.Total.."] <- "MafB2_WT_R2..Tag.Count.in.given.bp..6205913.0.Total.."

colnames(MafB2_counts)[colnames(MafB2_counts) == "MafB2_KO_R1..Tag.Count.in.given.bp..5444540.0.Total.."] <- "MafB2_KO_R1..Tag.Count.in.given.bp..5444540.0.Total.."

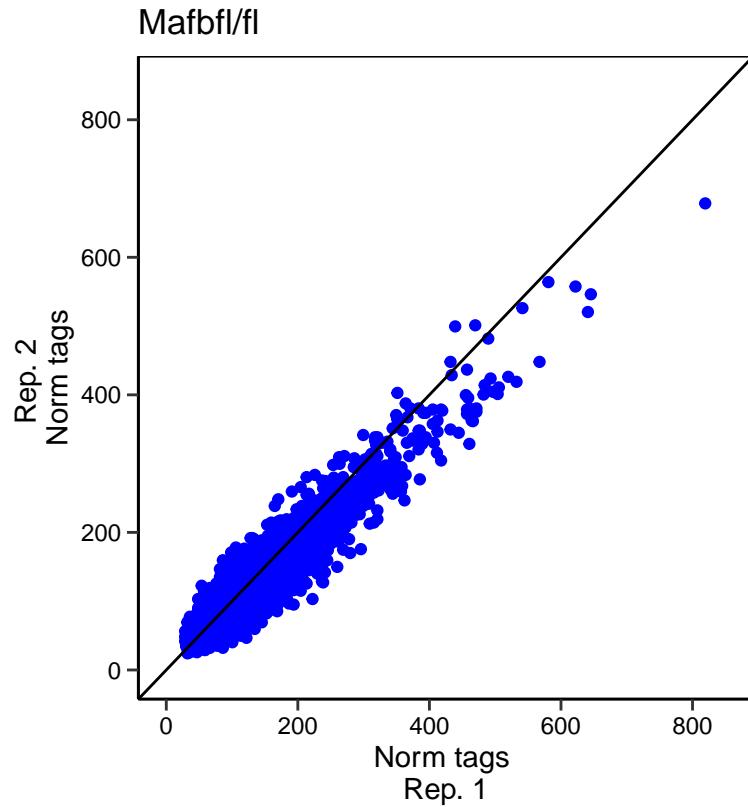
colnames(MafB2_counts)[colnames(MafB2_counts) == "MafB2_KO_R2..Tag.Count.in.given.bp..6810425.0.Total.."] <- "MafB2_KO_R2..Tag.Count.in.given.bp..6810425.0.Total.."

```

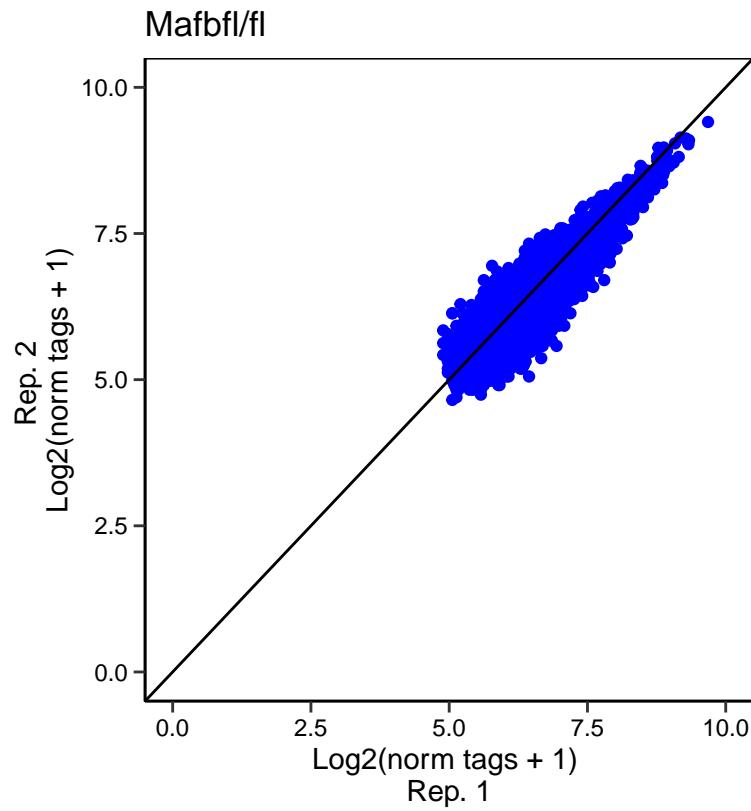
```

ggplot(data = MafB2_counts, aes(x = MafB2_WT_R1, y = MafB2_WT_R2)) +
  geom_point(show.legend = FALSE, colour = "blue") + geom_abline(slope = 1) +
  ggtitle("Mafbfl/f1") + xlab("Norm tags \nRep. 1") + ylab("Rep. 2 \nNorm tags") +
  xlim(0, 850) + ylim(0, 850) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))

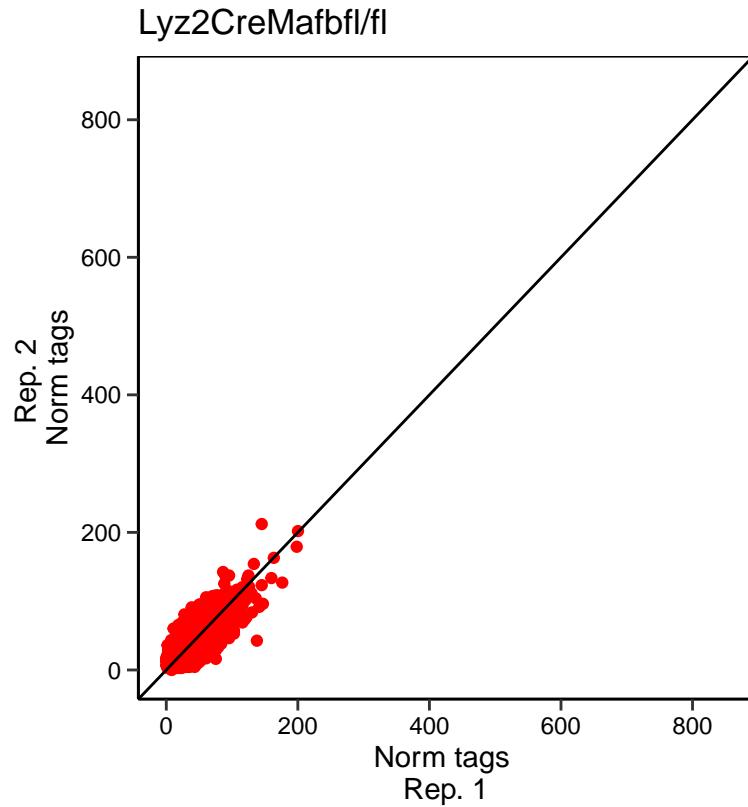
```



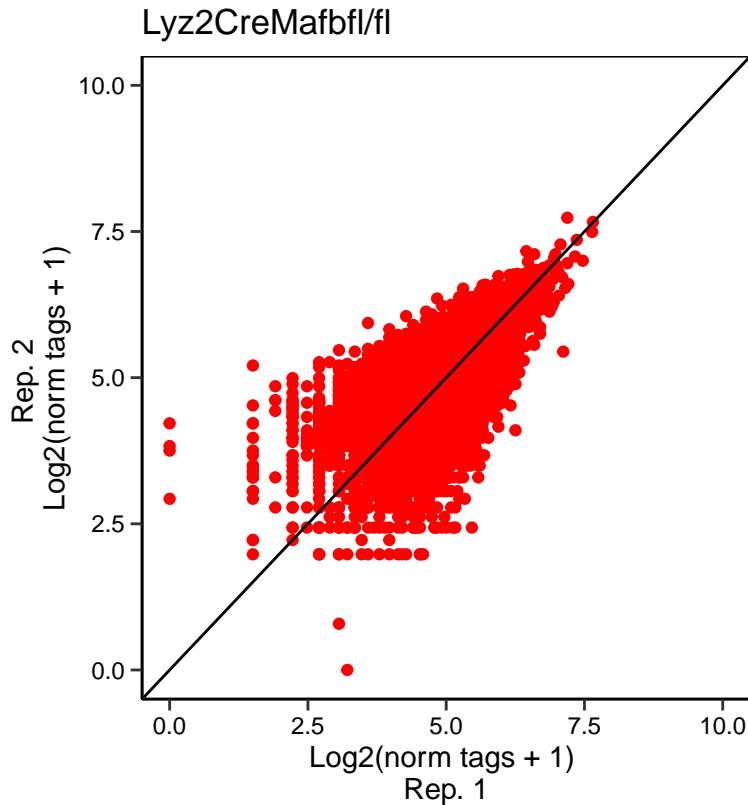
```
ggplot(data = MafB2_counts, aes(x = log2(MafB2_WT_R1 + 1), y = log2(MafB2_WT_R2 + 1))) + geom_point(show.legend = FALSE, colour = "blue") +
  geom_abline(slope = 1) + ggtitle("Mafbfl/fl") + xlab("Log2(norm tags + 1) \nRep. 1") +
  ylab("Rep. 2 \nLog2(norm tags + 1)") + xlim(0, 10) + ylim(0, 10) +
  theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
ggplot(data = MafB2_counts, aes(x = MafB2_KO_R1, y = MafB2_KO_R2)) +
  geom_point(show.legend = FALSE, colour = "red") + geom_abline(slope = 1) +
  ggtitle("Lyz2CreMafbfl/f1") + xlab("Norm tags \nRep. 1") +
  ylab("Rep. 2 \nNorm tags") + xlim(0, 850) + ylim(0, 850) +
  theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
ggplot(data = MafB2_counts, aes(x = log2(MafB2_KO_R1 + 1), y = log2(MafB2_KO_R2 + 1))) + geom_point(show.legend = FALSE, colour = "red") +
  geom_abline(slope = 1) + ggtitle("Lyz2CreMafbfl/fl") + xlab("Log2(norm tags + 1) \nRep. 1") +
  ylab("Rep. 2 \nLog2(norm tags + 1)") + xlim(0, 10) + ylim(0, 10) +
  theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
annotatePeaks.pl MafB2_peaks_merged.bed mm10 -size 4000 -hist 10 -d MafB2_WT_R1/ MafB2_WT_R2/ MafB2_KO_1

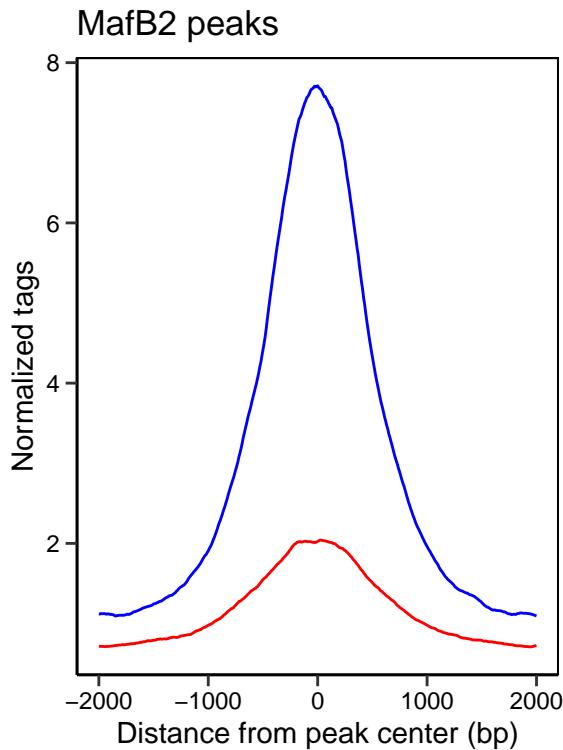
MafB2_hist <- read.table("MafB2_hist.txt", header = TRUE, sep = "\t")

colnames(MafB2_hist)[colnames(MafB2_hist) == "Distance.from.Center..cmd.annotatePeaks.pl.MafB2_peaks_me"]

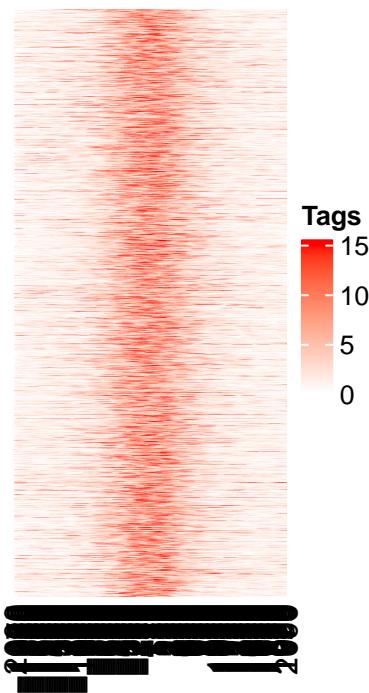
MafB2_hist$MafB2_WT <- rowMeans(MafB2_hist[, c("MafB2_WT_R1..Coverage",
  "MafB2_WT_R2..Coverage")])

MafB2_hist$MafB2_KO <- rowMeans(MafB2_hist[, c("MafB2_KO_R1..Coverage",
  "MafB2_KO_R2..Coverage")])

ggplot(data = MafB2_hist, aes(x = Distance.from.Center, y = MafB2_WT)) +
  geom_line(show.legend = FALSE, colour = "blue") + geom_line(show.legend = FALSE,
  colour = "red", aes(x = Distance.from.Center, y = MafB2_KO)) +
  ggtitle("MafB2 peaks") + xlab("Distance from peak center (bp)") +
  ylab("Normalized tags") + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



## Mafbfl/fl



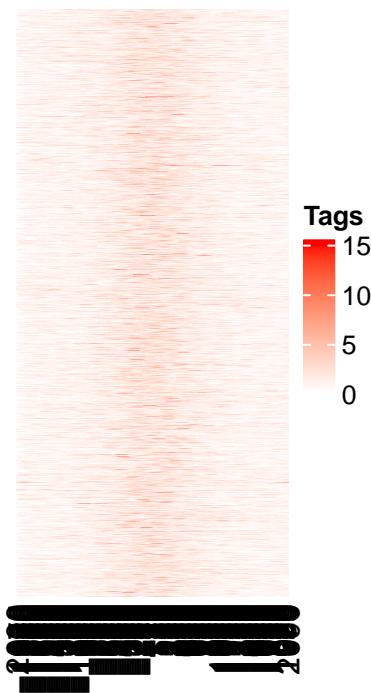
```
col_fun = colorRamp2(c(0, 15), c("white", "red"))

Heatmap(MafB2_K0_heatmap, col_fun, cluster_columns = F, cluster_rows = F,
        heatmap_legend_param = list(title = "Tags"), use_raster = TRUE,
        raster_quality = 10, , column_title = "LyzCreMafbfl/fl")
```

```
## Warning: The input is a data frame-like object, convert it to a matrix.

## 'magick' package is suggested to install to give better rasterization.
##
## Set 'ht_opt$message = FALSE' to turn off this message.
```

## LyzCreMafbfl/fl



## MafB3

```
MafB3_peaks_rep1 <- read.table("MafB3_WT_R1.seacr.peaks.stringent.bed",
  header = FALSE, sep = "\t")

MafB3_peaks_rep2 <- read.table("MafB3_WT_R2.seacr.peaks.stringent.bed",
  header = FALSE, sep = "\t")

MafB3_peaks_consensus <- read.table("MafB3_WT.seacr.consensus.peak_counts.bed",
  header = FALSE, sep = "\t")

# peaks rep1
nrow(MafB3_peaks_rep1)

## [1] 5696

# peaks rep2
nrow(MafB3_peaks_rep2)

## [1] 5254

# consensus peaks
nrow(MafB3_peaks_consensus)

## [1] 7750
```

```

# shared peaks
length(which(MafB3_peaks_consensus[, 10] == 2))

## [1] 3077

MafB3_peaks_merged <- MafB3_peaks_consensus[MafB3_peaks_consensus[,
  10] == 2, ]

split_scores <- strsplit(as.character(MafB3_peaks_merged$V6),
  ", ")

means <- sapply(split_scores, function(x) {
  if (all(is.na(x)))
    return(NA_real_)
  nums <- as.numeric(x)
  mean(nums, na.rm = TRUE)
})

MafB3_peaks_merged$V4 <- means

write.table(MafB3_peaks_merged, "MafB3_peaks_merged.bed", sep = "\t",
  quote = FALSE, row.names = FALSE, col.names = FALSE)

annotatePeaks.pl MafB3_peaks_merged.bed mm10 -d MafB3_WT_R1/ MafB3_WT_R2/ MafB3_KO_R1/ MafB3_KO_R2/ > MafB3_peaks_merged.bed

MafB3_counts <- read.table("MafB3_counts.txt", header = TRUE,
  sep = "\t")

colnames(MafB3_counts)[colnames(MafB3_counts) == "PeakID..cmd.annotatePeaks.pl.MafB3_peaks_merged.bed.mm10"] <- "PeakID..cmd.annotatePeaks.pl.MafB3_peaks_merged.bed.mm10"

colnames(MafB3_counts)[colnames(MafB3_counts) == "MafB3_WT_R1..Tag.Count.in.given.bp..6327645.0.Total..0"] <- "MafB3_WT_R1..Tag.Count.in.given.bp..6327645.0.Total..0"

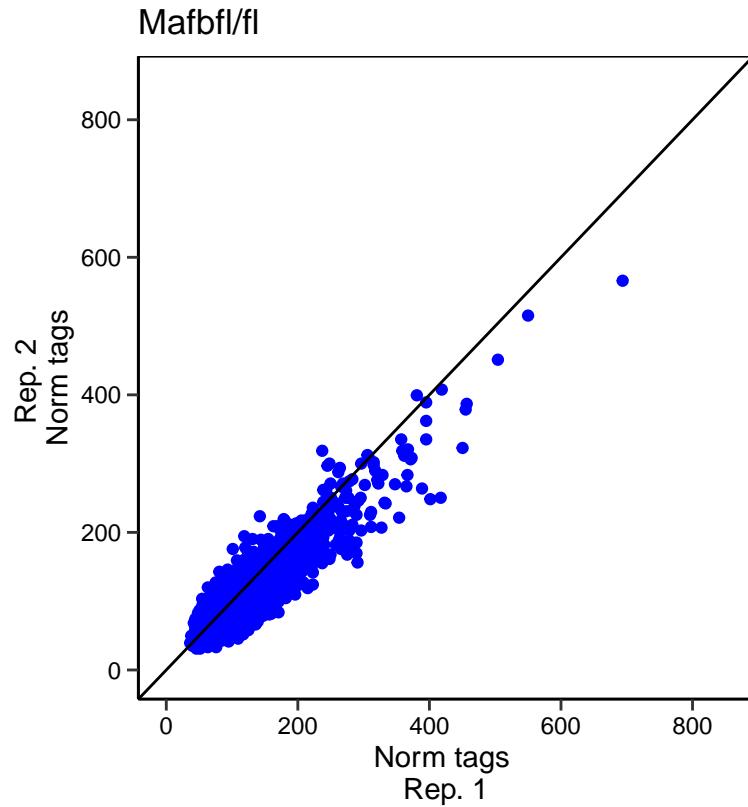
colnames(MafB3_counts)[colnames(MafB3_counts) == "MafB3_WT_R2..Tag.Count.in.given.bp..4832571.0.Total..0"] <- "MafB3_WT_R2..Tag.Count.in.given.bp..4832571.0.Total..0"

colnames(MafB3_counts)[colnames(MafB3_counts) == "MafB3_KO_R1..Tag.Count.in.given.bp..6875061.0.Total..0"] <- "MafB3_KO_R1..Tag.Count.in.given.bp..6875061.0.Total..0"

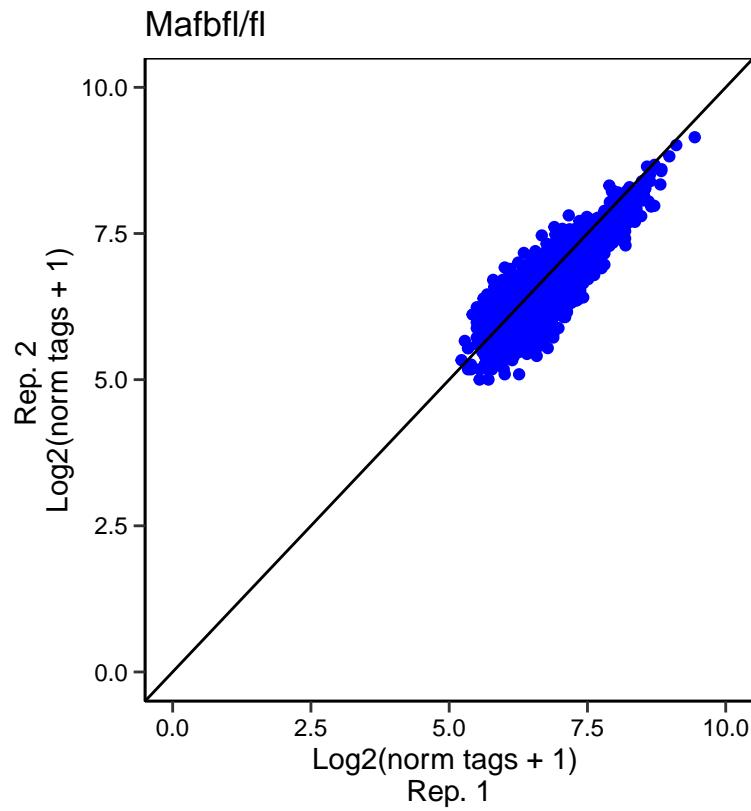
colnames(MafB3_counts)[colnames(MafB3_counts) == "MafB3_KO_R2..Tag.Count.in.given.bp..6527343.0.Total..0"] <- "MafB3_KO_R2..Tag.Count.in.given.bp..6527343.0.Total..0"

ggplot(data = MafB3_counts, aes(x = MafB3_WT_R1, y = MafB3_WT_R2)) +
  geom_point(show.legend = FALSE, colour = "blue") + geom_abline(slope = 1) +
  ggtitle("Mafbfl/f1") + xlab("Norm tags \nRep. 1") + ylab("Rep. 2 \nNorm tags") +
  xlim(0, 850) + ylim(0, 850) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))

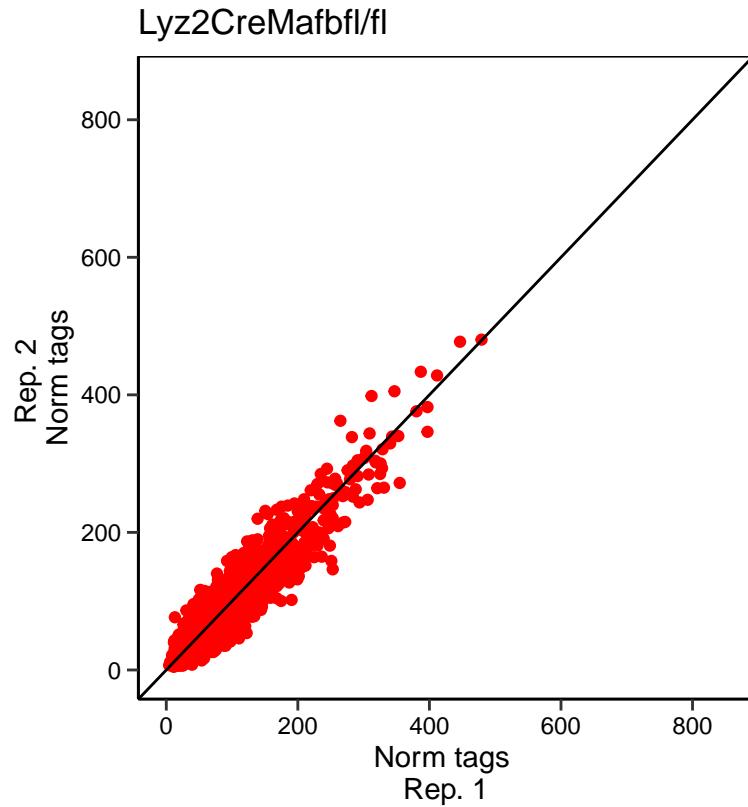
```



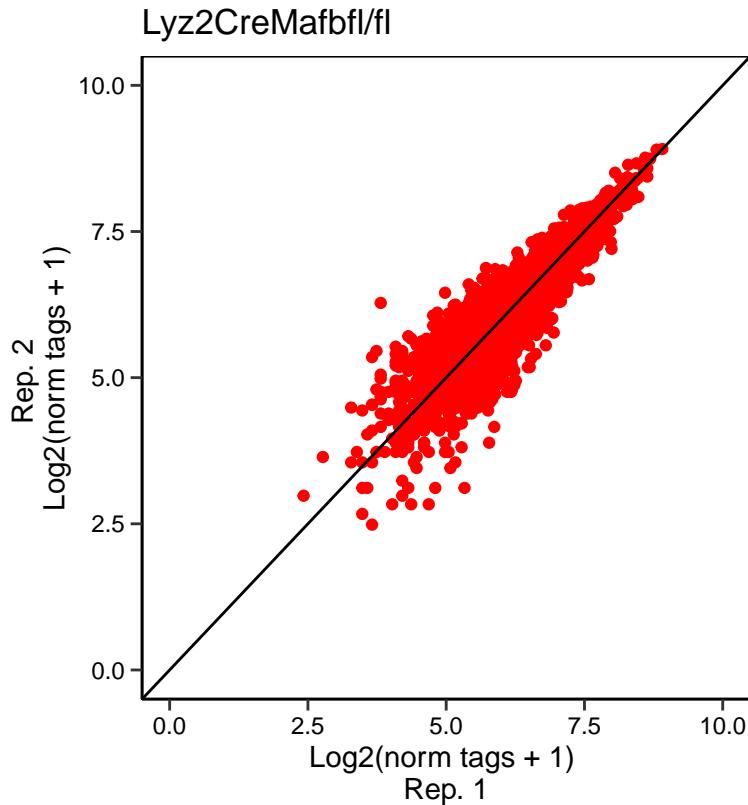
```
ggplot(data = MafB3_counts, aes(x = log2(MafB3_WT_R1 + 1), y = log2(MafB3_WT_R2 + 1))) + geom_point(show.legend = FALSE, colour = "blue") +
  geom_abline(slope = 1) + ggtitle("Mafbfl/fl") + xlab("Log2(norm tags + 1) \nRep. 1") +
  ylab("Rep. 2 \nLog2(norm tags + 1)") + xlim(0, 10) + ylim(0, 10) +
  theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
ggplot(data = MafB3_counts, aes(x = MafB3_KO_R1, y = MafB3_KO_R2)) +
  geom_point(show.legend = FALSE, colour = "red") + geom_abline(slope = 1) +
  ggtitle("Lyz2CreMafbfl/f1") + xlab("Norm tags \nRep. 1") +
  ylab("Rep. 2 \nNorm tags") + xlim(0, 850) + ylim(0, 850) +
  theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
ggplot(data = MafB3_counts, aes(x = log2(MafB3_KO_R1 + 1), y = log2(MafB3_KO_R2 + 1))) + geom_point(show.legend = FALSE, colour = "red") +
  geom_abline(slope = 1) + ggtitle("Lyz2CreMafbfl/fl") + xlab("Log2(norm tags + 1) \nRep. 1") +
  ylab("Rep. 2 \nLog2(norm tags + 1)") + xlim(0, 10) + ylim(0, 10) +
  theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
annotatePeaks.pl MafB2_peaks_merged.bed mm10 -size 4000 -hist 10 -d MafB2_WT_R1/ MafB2_WT_R2/ MafB2_KO_1

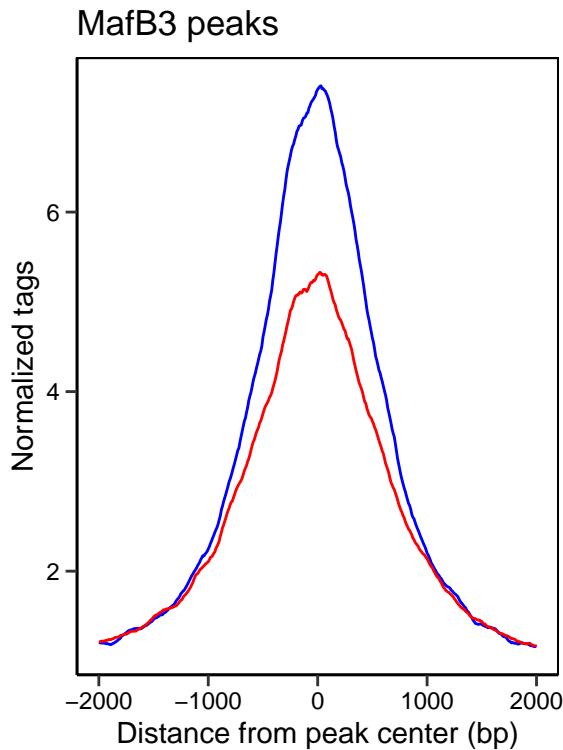
MafB3_hist <- read.table("MafB3_hist.txt", header = TRUE, sep = "\t")

colnames(MafB3_hist)[colnames(MafB3_hist) == "Distance.from.Center..cmd.annotatePeaks.pl.MafB3_peaks_me"]

MafB3_hist$MafB3_WT <- rowMeans(MafB3_hist[, c("MafB3_WT_R1..Coverage",
  "MafB3_WT_R2..Coverage")])

MafB3_hist$MafB3_KO <- rowMeans(MafB3_hist[, c("MafB3_KO_R1..Coverage",
  "MafB3_KO_R2.Coverage")])

ggplot(data = MafB3_hist, aes(x = Distance.from.Center, y = MafB3_WT)) +
  geom_line(show.legend = FALSE, colour = "blue") + geom_line(show.legend = FALSE,
  colour = "red", aes(x = Distance.from.Center, y = MafB3_KO)) +
  ggtitle("MafB3 peaks") + xlab("Distance from peak center (bp)") +
  ylab("Normalized tags") + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
annotatePeaks.pl MafB3_peaks_merged.bed mm10 -size 4000 -hist 25 -ghist -d MafB3_WT_R1/ MafB3_WT_R2/ Ma

MafB3_heatmap <- read.table("MafB3_heatmap.txt", header = TRUE,
  check.names = FALSE, sep = "\t")

MafB3_WT_R1_heatmap <- MafB3_heatmap[, 2:162]
MafB3_WT_R2_heatmap <- MafB3_heatmap[, 163:323]
MafB3_WT_heatmap <- (MafB3_WT_R1_heatmap + MafB3_WT_R2_heatmap)/2

MafB3_KO_R1_heatmap <- MafB3_heatmap[, 324:484]
MafB3_KO_R2_heatmap <- MafB3_heatmap[, 485:645]
MafB3_KO_heatmap <- (MafB3_KO_R1_heatmap + MafB3_KO_R2_heatmap)/2

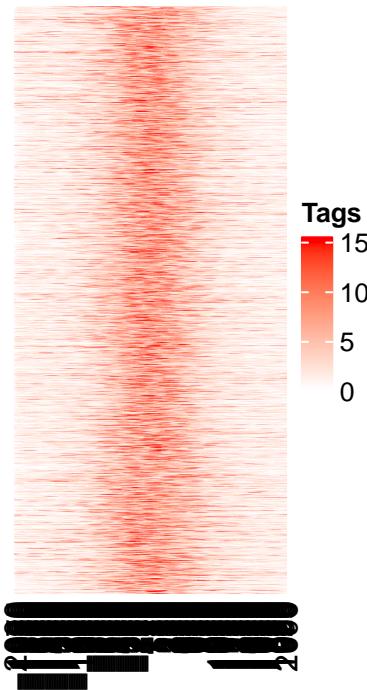
col_fun = colorRamp2(c(0, 15), c("white", "red"))

Heatmap(MafB3_WT_heatmap, col_fun, cluster_columns = F, cluster_rows = F,
  heatmap_legend_param = list(title = "Tags"), use_raster = TRUE,
  raster_quality = 10, column_title = "Mafbfl/fl")

## Warning: The input is a data frame-like object, convert it to a matrix.

## 'magick' package is suggested to install to give better rasterization.
##
## Set 'ht opt$message = FALSE' to turn off this message.
```

## Mafbfl/fl



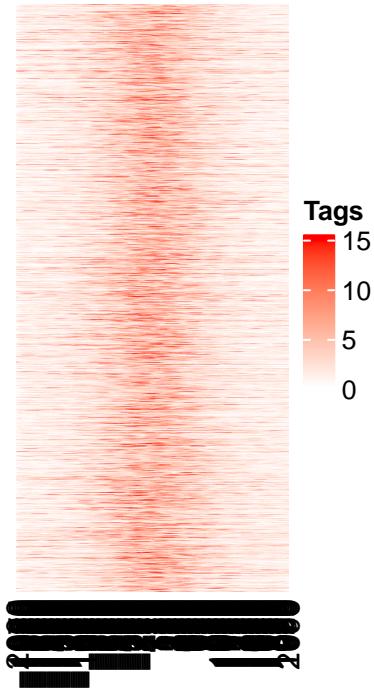
```
col_fun = colorRamp2(c(0, 15), c("white", "red"))

Heatmap(MafB3_K0_heatmap, col_fun, cluster_columns = F, cluster_rows = F,
        heatmap_legend_param = list(title = "Tags"), use_raster = TRUE,
        raster_quality = 10, , column_title = "LyzCreMafbfl/fl")
```

```
## Warning: The input is a data frame-like object, convert it to a matrix.

## 'magick' package is suggested to install to give better rasterization.
##
## Set 'ht_opt$message = FALSE' to turn off this message.
```

## LyzCreMafbfl/fl



## Peak selection (WT vs KO)

### MafB1

```

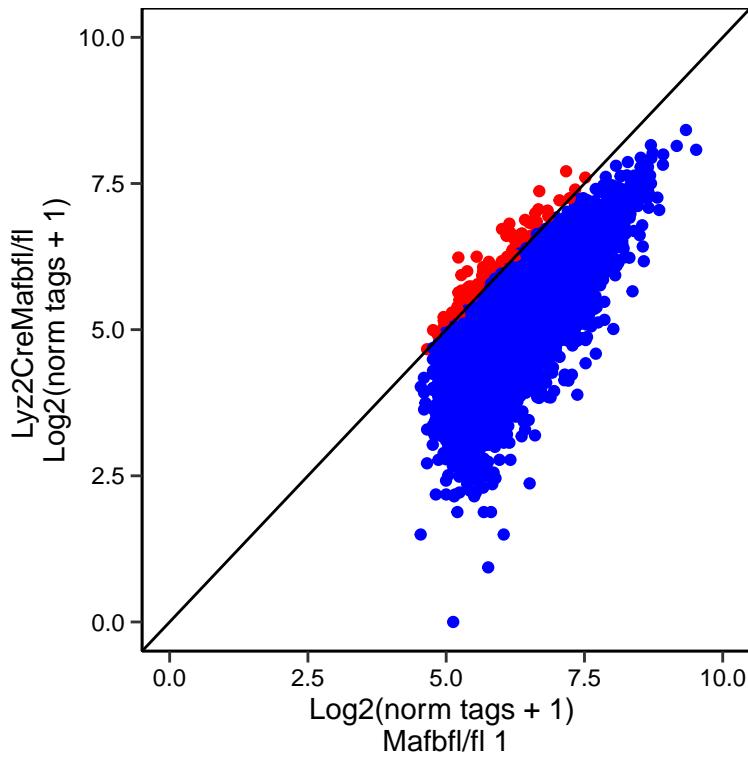
MafB1_counts$MafB1_WT <- rowMeans(MafB1_counts[, c("MafB1_WT_R1",
  "MafB1_WT_R2")])
MafB1_counts$MafB1_KO <- rowMeans(MafB1_counts[, c("MafB1_KO_R1",
  "MafB1_KO_R2")])

MafB1_counts$log2_MafB1_WT <- log2(MafB1_counts$MafB1_WT + 1)
MafB1_counts$log2_MafB1_KO <- log2(MafB1_counts$MafB1_KO + 1)
MafB1_counts$below_diag <- MafB1_counts$log2_MafB1_KO < MafB1_counts$log2_MafB1_WT

ggplot(data = MafB1_counts, aes(x = log2_MafB1_WT, y = log2_MafB1_KO)) +
  geom_point(aes(color = below_diag), show.legend = FALSE) +
  scale_color_manual(values = c(`FALSE` = "red", `TRUE` = "blue")) +
  geom_abline(slope = 1) + ggtitle("MafB1 Peaks") + xlab("Log2(norm tags + 1) \nMafbfl/fl 1") +
  ylab("Lyz2CreMafbfl/fl \nLog2(norm tags + 1)") + xlim(0,
  10) + ylim(0, 10) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))

```

## MafB1 Peaks



```

MafB1_PeakID <- c(MafB1_counts[MafB1_counts[, "below_diag"] ==
  "TRUE", ]$PeakID)
MafB1_peaks <- MafB1_peaks_merged[MafB1_peaks_merged[, 4] %in%
  MafB1_PeakID, ]

write.table(MafB1_peaks, "MafB1_peaks.bed", sep = "\t", quote = FALSE,
  row.names = FALSE, col.names = FALSE)
  
```

## MafB2

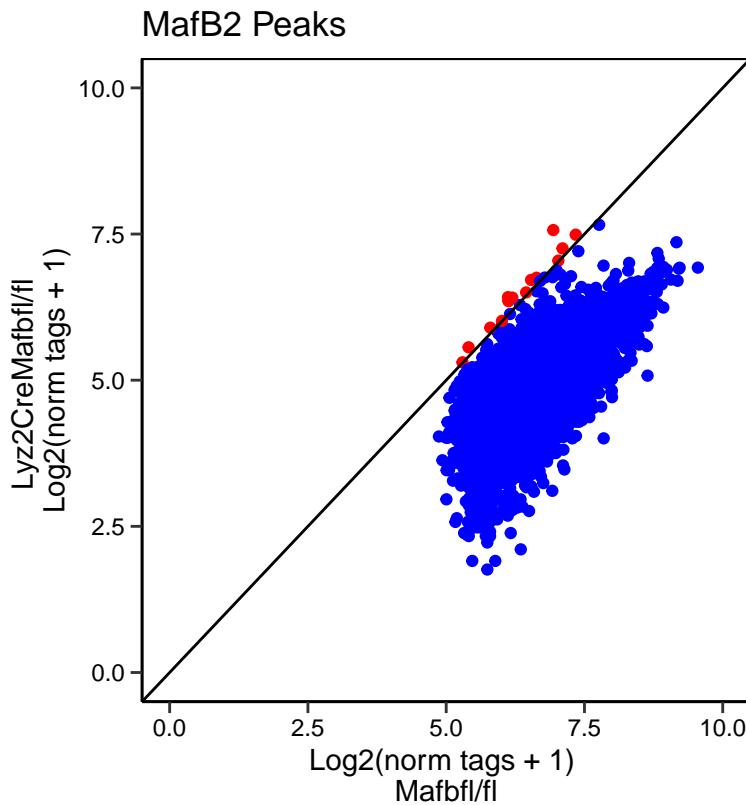
```

MafB2_counts$MafB2_WT <- rowMeans(MafB2_counts[, c("MafB2_WT_R1",
  "MafB2_WT_R2")])
MafB2_counts$MafB2_KO <- rowMeans(MafB2_counts[, c("MafB2_KO_R1",
  "MafB2_KO_R2")])

MafB2_counts$log2_MafB2_WT <- log2(MafB2_counts$MafB2_WT + 1)
MafB2_counts$log2_MafB2_KO <- log2(MafB2_counts$MafB2_KO + 1)
MafB2_counts$below_diag <- MafB2_counts$log2_MafB2_KO < MafB2_counts$log2_MafB2_WT

ggplot(data = MafB2_counts, aes(x = log2_MafB2_WT, y = log2_MafB2_KO)) +
  geom_point(aes(color = below_diag), show.legend = FALSE) +
  scale_color_manual(values = c(`FALSE` = "red", `TRUE` = "blue")) +
  geom_abline(slope = 1) + ggtitle("MafB2 Peaks") + xlab("Log2(norm tags + 1) \nMafbfl/fl") +
  ylab("Lyz2CreMafbfl/fl \nLog2(norm tags + 1)") + xlim(0,
  10) + ylim(0, 10) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  
```

```
axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))
```



```
MafB2_PeakID <- c(MafB2_counts[MafB2_counts[, "below_diag"] ==
  "TRUE", ]$PeakID)
MafB2_peaks <- MafB2_peaks_merged[MafB2_peaks_merged[, 4] %in%
  MafB2_PeakID, ]

write.table(MafB2_peaks, "MafB2_peaks.bed", sep = "\t", quote = FALSE,
  row.names = FALSE, col.names = FALSE)
```

### MafB3

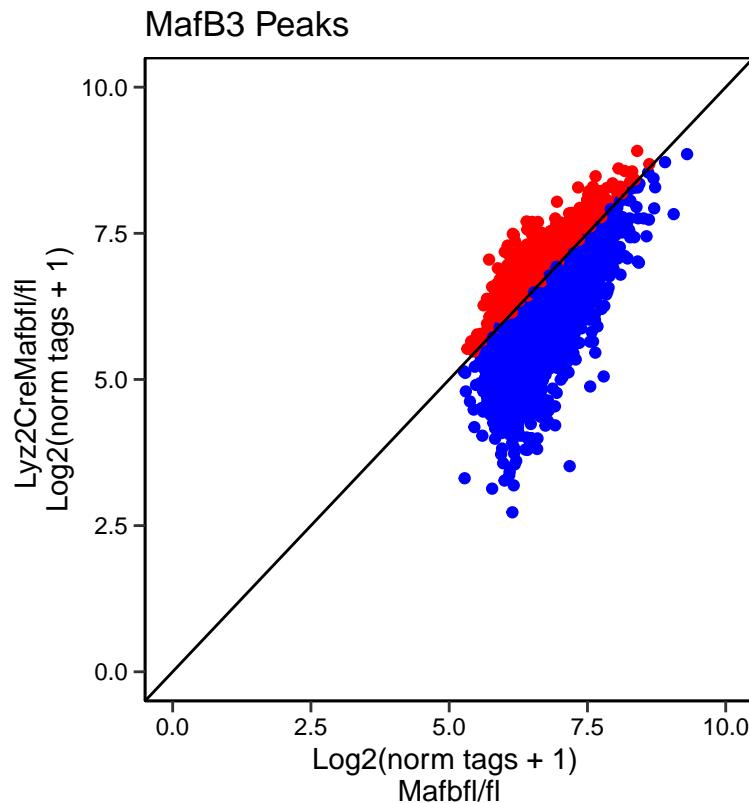
```
MafB3_counts$MafB3_WT <- rowMeans(MafB3_counts[, c("MafB3_WT_R1",
  "MafB3_WT_R2")])
MafB3_counts$MafB3_KO <- rowMeans(MafB3_counts[, c("MafB3_KO_R1",
  "MafB3_KO_R2")])

MafB3_counts$log2_MafB3_WT <- log2(MafB3_counts$MafB3_WT + 1)
MafB3_counts$log2_MafB3_KO <- log2(MafB3_counts$MafB3_KO + 1)
MafB3_counts$below_diag <- MafB3_counts$log2_MafB3_KO < MafB3_counts$log2_MafB3_WT
```

```

ggplot(data = MafB3_counts, aes(x = log2_MafB3_WT, y = log2_MafB3_KO)) +
  geom_point(aes(color = below_diag), show.legend = FALSE) +
  scale_color_manual(values = c(`FALSE` = "red", `TRUE` = "blue")) +
  geom_abline(slope = 1) + ggtitle("MafB3 Peaks") + xlab("Log2(norm tags + 1) \nMafbfl/fl") +
  ylab("Lyz2CreMafbfl/fl \nLog2(norm tags + 1)") + xlim(0,
  10) + ylim(0, 10) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))

```



```

MafB3_PeakID <- c(MafB3_counts[MafB3_counts[, "below_diag"] ==
  "TRUE", ]$PeakID)
MafB3_peaks <- MafB3_peaks_merged[MafB3_peaks_merged[, 4] %in%
  MafB3_PeakID, ]

write.table(MafB3_peaks, "MafB3_peaks.bed", sep = "\t", quote = FALSE,
  row.names = FALSE, col.names = FALSE)

```

## Annotate Peaks

MafB1

```
annotatePeaks.pl MafB1_peaks.bed genome.fa -gtf genes.gtf > MafB1_peaks_annot.txt
```

```

MafB1_peaks_annot <- read.table("MafB1_peaks_annot.txt", header = TRUE,
  sep = "\t")

MafB1_peaks_annot$Annotation.not.detailed <- sub(" .*", "", MafB1_peaks_annot$Annotation)

table(MafB1_peaks_annot$Annotation.not.detailed)

##          exon    Intergenic      intron promoter-TSS       TTS
##        461           2128         2654         2396         213

```

## MafB2

```

annotatePeaks.pl MafB2_peaks.bed genome.fa -gtf genes.gtf > MafB2_peaks_annot.txt

MafB2_peaks_annot <- read.table("MafB2_peaks_annot.txt", header = TRUE,
  sep = "\t")

MafB2_peaks_annot$Annotation.not.detailed <- sub(" .*", "", MafB2_peaks_annot$Annotation)

table(MafB2_peaks_annot$Annotation.not.detailed)

##          exon    Intergenic      intron promoter-TSS       TTS
##        472           1897         2423         2271         178

```

## MafB3

```

annotatePeaks.pl MafB3_peaks.bed genome.fa -gtf genes.gtf > MafB3_peaks_annot.txt

MafB3_peaks_annot <- read.table("MafB3_peaks_annot.txt", header = TRUE,
  sep = "\t")

MafB3_peaks_annot$Annotation.not.detailed <- sub(" .*", "", MafB3_peaks_annot$Annotation)

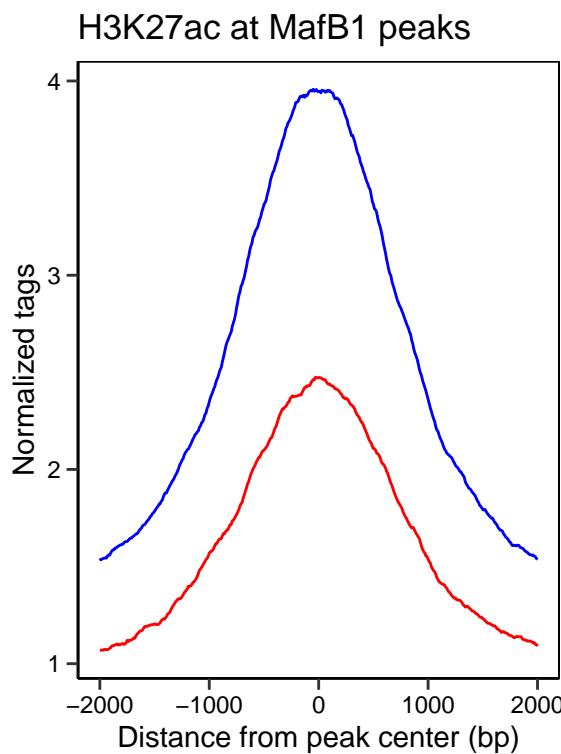
table(MafB3_peaks_annot$Annotation.not.detailed)

##          exon    Intergenic      intron promoter-TSS       TTS
##        195           564          750          780          49

```

## H3K27Ac at MafB peaks

### MafB1



```
annotatePeaks.pl MafB1_peaks.bed mm10 -d tag_directory_WT-H3K27Ac_R1/ tag_directory_WT-H3K27Ac_R2/ tag_
```

```

MafB1_H3K27Ac_counts <- read.table("MafB1_H3K27Ac_counts.txt",
  header = TRUE, sep = "\t")

colnames(MafB1_H3K27Ac_counts)[colnames(MafB1_H3K27Ac_counts) ==
  "PeakID..cmd.annotatePeaks.pl.MafB1_peaks.bed.mm10..d.tag_directory_WT.H3K27Ac_R1..tag_directory_WT"]

colnames(MafB1_H3K27Ac_counts)[colnames(MafB1_H3K27Ac_counts) ==
  "tag_directory_WT.H3K27Ac_R1..Tag.Count.in.given.bp..3449599.0.Total..normalization.factor...2.90..."]

colnames(MafB1_H3K27Ac_counts)[colnames(MafB1_H3K27Ac_counts) ==
  "tag_directory_WT.H3K27Ac_R2..Tag.Count.in.given.bp..3954896.0.Total..normalization.factor...2.53..."]

colnames(MafB1_H3K27Ac_counts)[colnames(MafB1_H3K27Ac_counts) ==
  "tag_directory_KO.H3K27Ac_R1..Tag.Count.in.given.bp..3206024.0.Total..normalization.factor...3.12..."]

colnames(MafB1_H3K27Ac_counts)[colnames(MafB1_H3K27Ac_counts) ==
  "tag_directory_KO.H3K27Ac_R2..Tag.Count.in.given.bp..3641165.0.Total..normalization.factor...2.75..."]

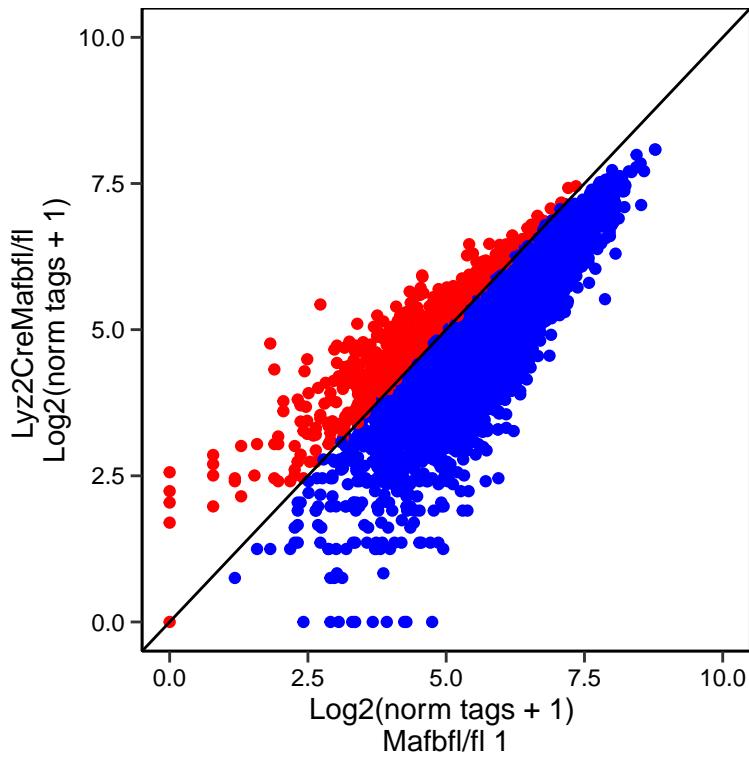
MafB1_H3K27Ac_counts$H3K27Ac_WT <- rowMeans(MafB1_H3K27Ac_counts[,,
  c("H3K27Ac_WT_R1", "H3K27Ac_WT_R2")])
MafB1_H3K27Ac_counts$H3K27Ac_KO <- rowMeans(MafB1_H3K27Ac_counts[,,
  c("H3K27Ac_KO_R1", "H3K27Ac_KO_R2")])

MafB1_H3K27Ac_counts$log2_H3K27Ac_WT <- log2(MafB1_H3K27Ac_counts$H3K27Ac_WT +
  1)
MafB1_H3K27Ac_counts$log2_H3K27Ac_KO <- log2(MafB1_H3K27Ac_counts$H3K27Ac_KO +
  1)
MafB1_H3K27Ac_counts$below_diag <- MafB1_H3K27Ac_counts$log2_H3K27Ac_KO <
  MafB1_H3K27Ac_counts$log2_H3K27Ac_WT

ggplot(data = MafB1_H3K27Ac_counts, aes(x = log2_H3K27Ac_WT,
  y = log2_H3K27Ac_KO)) + geom_point(aes(color = below_diag),
  show.legend = FALSE) + scale_color_manual(values = c(`FALSE` = "red",
  `TRUE` = "blue")) + geom_abline(slope = 1) + ggtitle("H3K27ac at MafB1 Peaks") +
  xlab("Log2(norm tags + 1) \nMafbfl/fl 1") + ylab("Lyz2CreMafbfl/fl \nLog2(norm tags + 1)") +
  xlim(0, 10) + ylim(0, 10) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
  axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
  "cm"), panel.border = element_rect(fill = NA, color = "black",
  linetype = "solid"))

```

### H3K27ac at MafB1 Peaks



```
table(MafB1_H3K27Ac_counts$below_diag)
```

```
##  
## FALSE TRUE  
## 897 6955
```

```
wilcox.test(MafB1_H3K27Ac_counts$log2_H3K27Ac_WT, MafB1_H3K27Ac_counts$log2_H3K27Ac_WTvsKO)
```

```
##  
## Wilcoxon signed rank test with continuity correction  
##  
## data: MafB1_H3K27Ac_counts$log2_H3K27Ac_WT  
## V = 30791628, p-value < 2.2e-16  
## alternative hypothesis: true location is not equal to 0
```

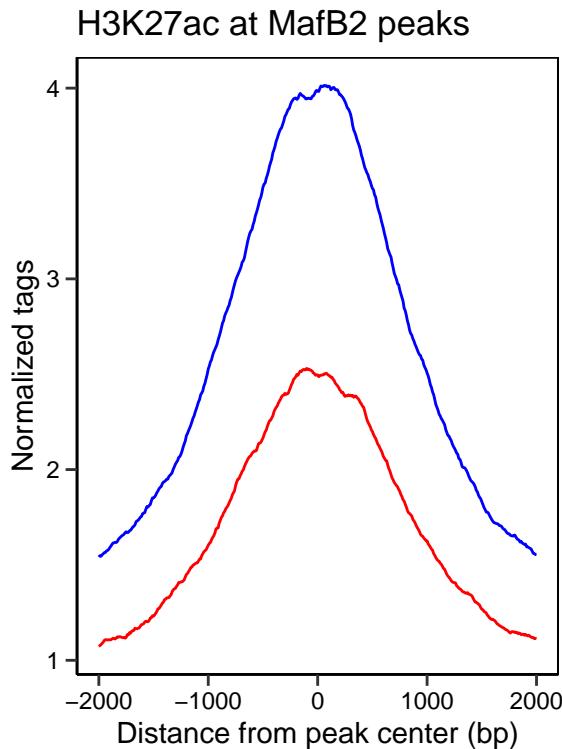
MafB2

```
annotatePeaks.pl MafB2_peaks.bed mm10 -size 4000 -hist 10 -d tag_directory_WT-H3K27Ac_R1/ tag_directory
```

```
MafB2_H3K27Ac_hist <- read.table("MafB2_H3K27Ac_hist.txt", header = TRUE,  
sep = "\t")
```

```
colnames(MafB2_H3K27Ac_hist)[colnames(MafB2_H3K27Ac_hist) ==
```

```
"Distance.from.Center..cmd.annotatePeaks.pl.MafB2_peaks.bed.mm10..size.4000..hist.10..d.tag_directory_WT.H3K27Ac_hist$H3K27Ac_WT <- rowMeans(MafB2_H3K27Ac_hist[, c("tag_directory_WT.H3K27Ac_R1..Coverage", "tag_directory_WT.H3K27Ac_R2..Coverage")])  
MafB2_H3K27Ac_hist$H3K27Ac_KO <- rowMeans(MafB2_H3K27Ac_hist[, c("tag_directory_KO.H3K27Ac_R1..Coverage", "tag_directory_KO.H3K27Ac_R2..Coverage")])  
  
ggplot(data = MafB2_H3K27Ac_hist, aes(x = Distance.from.Center, y = H3K27Ac_WT)) + geom_line(show.legend = FALSE, colour = "blue") +  
  geom_line(show.legend = FALSE, colour = "red", aes(x = Distance.from.Center, y = H3K27Ac_KO)) + ggtitle("H3K27ac at MafB2 peaks") +  
  xlab("Distance from peak center (bp)") + ylab("Normalized tags") +  
  theme_classic() + theme(axis.text.x = element_text(color = "black"),  
    axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15, "cm"), panel.border = element_rect(fill = NA, color = "black", linetype = "solid"))
```



```

>tag_directory_WT.H3K27Ac_R1..Tag.Count.in.given.bp..3449599.0.Total..normalization.factor...2.90...
colnames(MafB2_H3K27Ac_counts) [colnames(MafB2_H3K27Ac_counts) ==
>tag_directory_WT.H3K27Ac_R2..Tag.Count.in.given.bp..3954896.0.Total..normalization.factor...2.53...
colnames(MafB2_H3K27Ac_counts) [colnames(MafB2_H3K27Ac_counts) ==
>tag_directory_KO.H3K27Ac_R1..Tag.Count.in.given.bp..3206024.0.Total..normalization.factor...3.12...
colnames(MafB2_H3K27Ac_counts) [colnames(MafB2_H3K27Ac_counts) ==
>tag_directory_KO.H3K27Ac_R2..Tag.Count.in.given.bp..3641165.0.Total..normalization.factor...2.75...

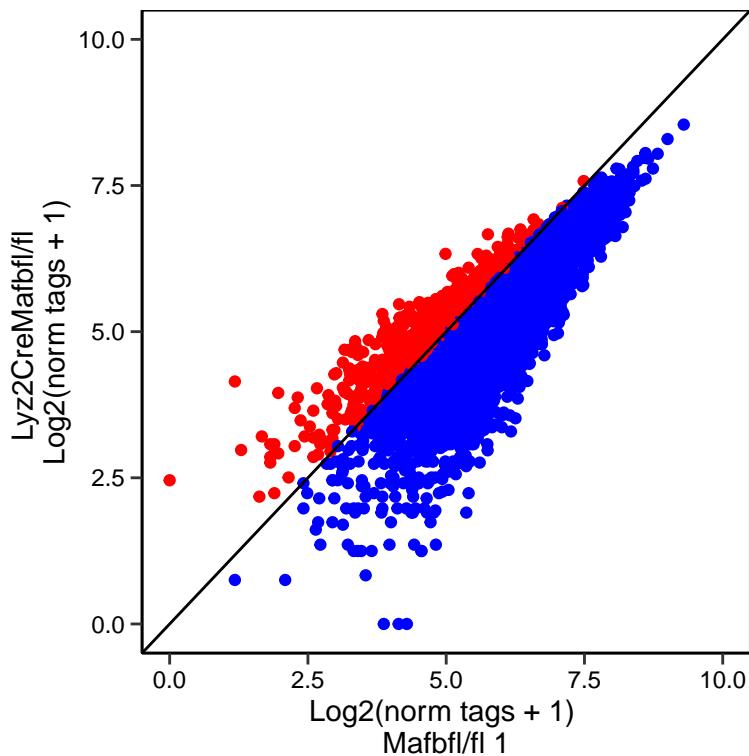
MafB2_H3K27Ac_counts$H3K27Ac_WT <- rowMeans(MafB2_H3K27Ac_counts[,,
c("H3K27Ac_WT_R1", "H3K27Ac_WT_R2")])
MafB2_H3K27Ac_counts$H3K27Ac_KO <- rowMeans(MafB2_H3K27Ac_counts[,,
c("H3K27Ac_KO_R1", "H3K27Ac_KO_R2")])

MafB2_H3K27Ac_counts$log2_H3K27Ac_WT <- log2(MafB2_H3K27Ac_counts$H3K27Ac_WT +
1)
MafB2_H3K27Ac_counts$log2_H3K27Ac_KO <- log2(MafB2_H3K27Ac_counts$H3K27Ac_KO +
1)
MafB2_H3K27Ac_counts$below_diag <- MafB2_H3K27Ac_counts$log2_H3K27Ac_KO <
MafB2_H3K27Ac_counts$log2_H3K27Ac_WT

ggplot(data = MafB2_H3K27Ac_counts, aes(x = log2_H3K27Ac_WT,
y = log2_H3K27Ac_KO)) + geom_point(aes(color = below_diag),
show.legend = FALSE) + scale_color_manual(values = c(`FALSE` = "red",
`TRUE` = "blue")) + geom_abline(slope = 1) + ggtitle("H3K27ac at MafB2 Peaks") +
xlab("Log2(norm tags + 1) \nMafbfl/fl 1") + ylab("Lyz2CreMafbfl/fl \nLog2(norm tags + 1)") +
xlim(0, 10) + ylim(0, 10) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
"cm"), panel.border = element_rect(fill = NA, color = "black",
linetype = "solid"))

```

### H3K27ac at MafB2 Peaks



```
table(MafB2_H3K27Ac_counts$below_diag)
```

```
##  
## FALSE TRUE  
## 778 6464
```

```
wilcox.test(MafB2_H3K27Ac_counts$log2_H3K27Ac_WT, MafB2_H3K27Ac_counts$log2_H3K27Ac_WTvsKO)
```

```
##  
## Wilcoxon signed rank test with continuity correction  
##  
## data: MafB2_H3K27Ac_counts$log2_H3K27Ac_WT  
## V = 26219661, p-value < 2.2e-16  
## alternative hypothesis: true location is not equal to 0
```

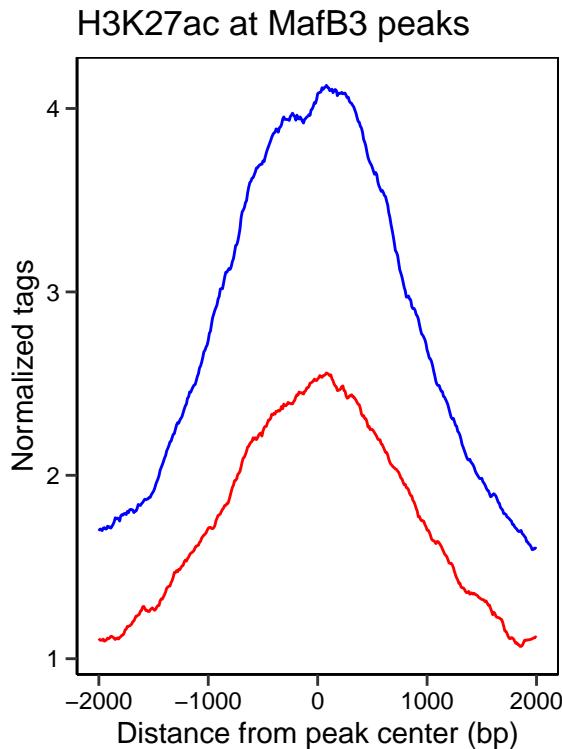
### MafB3

```
annotatePeaks.pl MafB3_peaks.bed mm10 -size 4000 -hist 10 -d tag_directory_WT-H3K27Ac_R1/ tag_directory
```

```
MafB3_H3K27Ac_hist <- read.table("MafB3_H3K27Ac_hist.txt", header = TRUE,  
sep = "\t")
```

```
colnames(MafB3_H3K27Ac_hist)[colnames(MafB3_H3K27Ac_hist) ==
```

```
"Distance.from.Center..cmd.annotatePeaks.pl.MafB3_peaks.bed.mm10..size.4000..hist.10..d.tag_directory_WT.H3K27Ac_hist$H3K27Ac_WT <- rowMeans(MafB3_H3K27Ac_hist[, c("tag_directory_WT.H3K27Ac_R1..Coverage", "tag_directory_WT.H3K27Ac_R2..Coverage")])  
MafB3_H3K27Ac_hist$H3K27Ac_KO <- rowMeans(MafB3_H3K27Ac_hist[, c("tag_directory_KO.H3K27Ac_R1..Coverage", "tag_directory_KO.H3K27Ac_R2..Coverage")])  
  
ggplot(data = MafB3_H3K27Ac_hist, aes(x = Distance.from.Center, y = H3K27Ac_WT)) + geom_line(show.legend = FALSE, colour = "blue") +  
  geom_line(show.legend = FALSE, colour = "red", aes(x = Distance.from.Center, y = H3K27Ac_KO)) + ggtitle("H3K27ac at MafB3 peaks") +  
  xlab("Distance from peak center (bp)") + ylab("Normalized tags") +  
  theme_classic() + theme(axis.text.x = element_text(color = "black"),  
    axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15, "cm"), panel.border = element_rect(fill = NA, color = "black", linetype = "solid"))
```



```

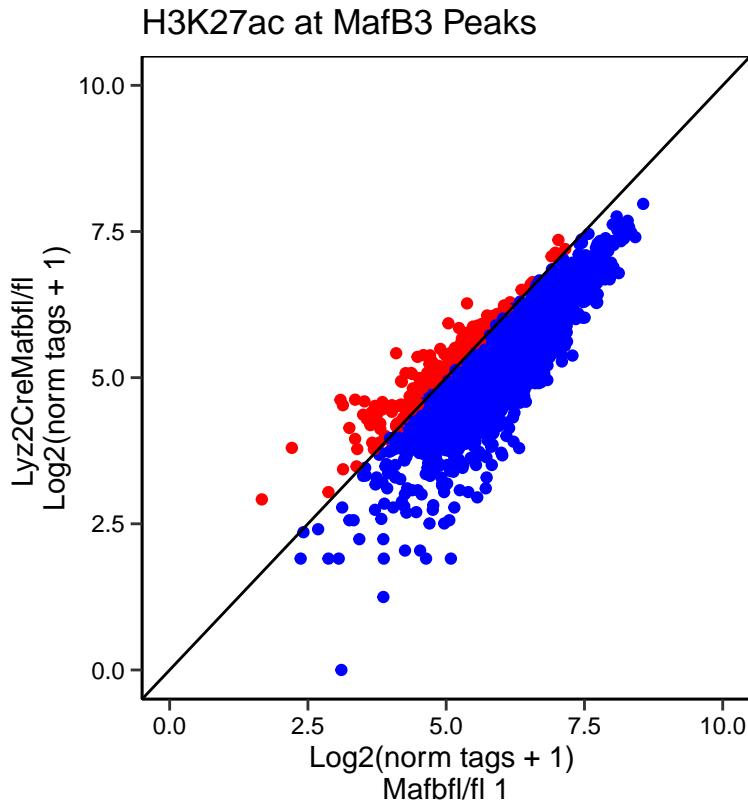
>tag_directory_WT.H3K27Ac_R1..Tag.Count.in.given.bp..3449599.0.Total..normalization.factor...2.90...
colnames(MafB3_H3K27Ac_counts) [colnames(MafB3_H3K27Ac_counts) ==
>tag_directory_WT.H3K27Ac_R2..Tag.Count.in.given.bp..3954896.0.Total..normalization.factor...2.53...
colnames(MafB3_H3K27Ac_counts) [colnames(MafB3_H3K27Ac_counts) ==
>tag_directory_KO.H3K27Ac_R1..Tag.Count.in.given.bp..3206024.0.Total..normalization.factor...3.12...
colnames(MafB3_H3K27Ac_counts) [colnames(MafB3_H3K27Ac_counts) ==
>tag_directory_KO.H3K27Ac_R2..Tag.Count.in.given.bp..3641165.0.Total..normalization.factor...2.75...

MafB3_H3K27Ac_counts$H3K27Ac_WT <- rowMeans(MafB3_H3K27Ac_counts[,,
c("H3K27Ac_WT_R1", "H3K27Ac_WT_R2")])
MafB3_H3K27Ac_counts$H3K27Ac_KO <- rowMeans(MafB3_H3K27Ac_counts[,,
c("H3K27Ac_KO_R1", "H3K27Ac_KO_R2")])

MafB3_H3K27Ac_counts$log2_H3K27Ac_WT <- log2(MafB3_H3K27Ac_counts$H3K27Ac_WT +
1)
MafB3_H3K27Ac_counts$log2_H3K27Ac_KO <- log2(MafB3_H3K27Ac_counts$H3K27Ac_KO +
1)
MafB3_H3K27Ac_counts$below_diag <- MafB3_H3K27Ac_counts$log2_H3K27Ac_KO <
MafB3_H3K27Ac_counts$log2_H3K27Ac_WT

ggplot(data = MafB3_H3K27Ac_counts, aes(x = log2_H3K27Ac_WT,
y = log2_H3K27Ac_KO)) + geom_point(aes(color = below_diag),
show.legend = FALSE) + scale_color_manual(values = c(`FALSE` = "red",
`TRUE` = "blue")) + geom_abline(slope = 1) + ggtitle("H3K27ac at MafB3 Peaks") +
xlab("Log2(norm tags + 1) \nMafbfl/fl 1") + ylab("Lyz2CreMafbfl/fl \nLog2(norm tags + 1)") +
xlim(0, 10) + ylim(0, 10) + theme_classic() + theme(axis.text.x = element_text(color = "black"),
axis.text.y = element_text(color = "black"), axis.ticks.length = unit(0.15,
"cm"), panel.border = element_rect(fill = NA, color = "black",
linetype = "solid"))

```



```
table(MafB3_H3K27Ac_counts$below_diag)
```

```
##  
## FALSE TRUE  
## 190 2148
```

```
wilcox.test(MafB3_H3K27Ac_counts$log2_H3K27Ac_WT, MafB3_H3K27Ac_counts$log2_H3K27Ac_WTvsKO)
```

```
##  
## Wilcoxon signed rank test with continuity correction  
##  
## data: MafB3_H3K27Ac_counts$log2_H3K27Ac_WT  
## V = 2734291, p-value < 2.2e-16  
## alternative hypothesis: true location is not equal to 0
```

## MafB target genes

### MafB1

```
MafB1_genes <- MafB1_peaks_annot[MafB1_peaks_annot[, "Distance.to.TSS"] >= -2000 & MafB1_peaks_annot[, "Distance.to.TSS"] <= 2000, ]
```

```
MafB1_genes <- unique(MafB1_genes$Gene.Name)
```

```

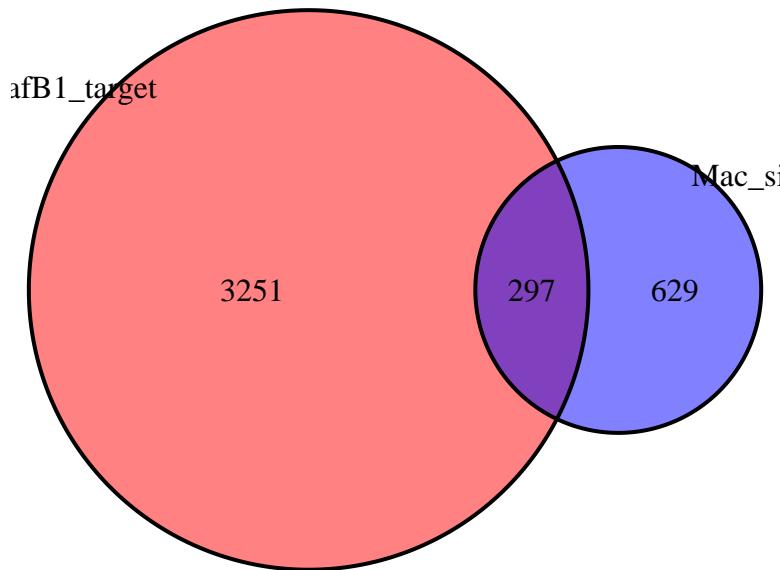
Mac_sign <- read_excel("Mac_sign.xlsx")

Mac_sign_genes <- Mac_sign$Gene_Symbol

# Helper function to display Venn diagram
display_venn <- function(x, ...) {
  library(VennDiagram)
  grid.newpage()
  venn_object <- venn.diagram(x, filename = NULL, ...)
  grid.draw(venn_object)
}

x <- list(MafB1_target = MafB1_genes, Mac_sign = Mac_sign_genes)
# pdf(file =
# 'C:/Users/domie/Documents/CUTandRUN/Venn_MafB1_Mac.pdf',
# width = 4, height = 4)
display_venn(x, fill = c("red", "blue"))

```



```

# dev.off()

g <- readGFF("C:/Users/domie/Documents/CUTandRUN/genes.gtf")
pc <- g %>%
  dplyr::filter(type == "gene")
rm(g)
dim(pc)
# genes 22597

```

```

dat <- matrix(c(297, 3251, 629, 18420), nrow = 2, dimnames = list(Mac_sign = c("YES",
  "NO"), MafB1_target = c("YES", "NO")))
dat

##           MafB1_target
## Mac_sign   YES     NO
##       YES 297    629
##       NO  3251 18420

fisher.test(dat)

##
## Fisher's Exact Test for Count Data
##
## data: dat
## p-value < 2.2e-16
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
## 2.310688 3.091887
## sample estimates:
## odds ratio
## 2.675045

```

## MafB2

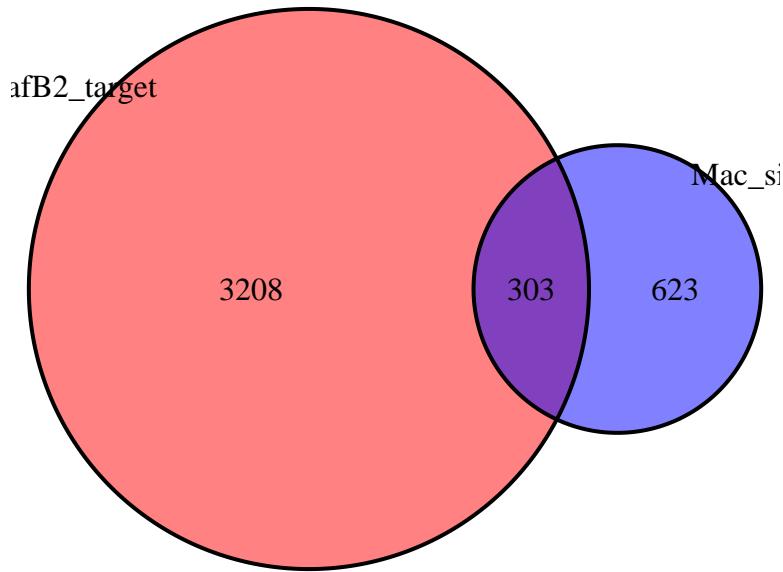
```

MafB2_genes <- MafB2_peaks_annot[MafB2_peaks_annot[, "Distance.to.TSS"] >=
  -2000 & MafB2_peaks_annot[, "Distance.to.TSS"] <= 2000, ]

MafB2_genes <- unique(MafB2_genes$Gene.Name)
MafB2_genes <- na.omit(MafB2_genes)

x <- list(MafB2_target = MafB2_genes, Mac_sign = Mac_sign_genes)
# pdf(file =
# 'C:/Users/domie/Documents/CUTandRUN/Venn_MafB2_Mac.pdf',
# width = 4, height = 4)
display_venn(x, fill = c("red", "blue"))

```



```
# dev.off()

dat <- matrix(c(303, 3208, 623, 18463), nrow = 2, dimnames = list(Mac_sign = c("YES",
  "NO"), MafB2_target = c("YES", "NO")))
dat

##           MafB2_target
## Mac_sign   YES      NO
##       YES  303    623
##       NO   3208 18463

fisher.test(dat)

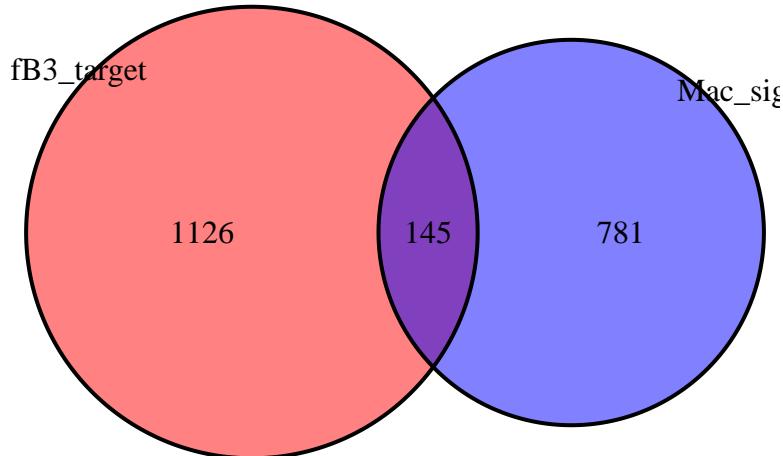
##
## Fisher's Exact Test for Count Data
##
## data: dat
## p-value < 2.2e-16
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  2.419563 3.232976
## sample estimates:
## odds ratio
##  2.798956
```

## MafB3

```
MafB3_genes <- MafB3_peaks_annot[MafB3_peaks_annot[, "Distance.to.TSS"] >=
-2000 & MafB3_peaks_annot[, "Distance.to.TSS"] <= 2000, ]
```

```
MafB3_genes <- unique(MafB3_genes$Gene.Name)
```

```
x <- list(MafB3_target = MafB3_genes, Mac_sign = Mac_sign_genes)
# pdf(file =
# 'C:/Users/domie/Documents/CUTandRUN/Venn_MafB3_Mac.pdf',
# width = 4, height = 4)
display_venn(x, fill = c("red", "blue"))
```



```
# dev.off()
```

```
dat <- matrix(c(145, 1125, 781, 20546), nrow = 2, dimnames = list(Mac_sign = c("YES",
"NO"), MafB1_target = c("YES", "NO")))
dat
```

```
##          MafB1_target
## Mac_sign  YES      NO
##        YES  145    781
##        NO   1125  20546
```

```
fisher.test(dat)

##
## Fisher's Exact Test for Count Data
##
## data: dat
## p-value < 2.2e-16
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
## 2.791837 4.096395
## sample estimates:
## odds ratio
## 3.390408
```