histone_stats H3K4me3

```
library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(ggpubr)
library(ggrepel)
library(here); i_am("R/Figure3/Fig3C_histonestats_H3K4me3.Rmd")
```

```
## here() starts at /projects/imb-pkbphil/sp/rnaseq/six_donor_trans/splicing_paper
```

```
## here() starts at /projects/imb-pkbphil/sp/rnaseq/six_donor_trans/splicing_paper
```

```
histone = "H3K4me3"
window = "1000:2000"
tss_sets = c("beige","white","not_sig")

enrich_tables = list()
for (set in tss_sets){
  file = here("31_leafcutter/histone_profile", histone, paste0("window", window), paste0(set,".",window
  annot = read.delim(file, quote="'")
  colnames(annot)[grep("chr", colnames(annot))] = "chr"
  annot$tss_set = set
  enrich_tables[[set]] = annot
}
str(enrich_tables)
```

```
## List of 3
##  $ beige  :'data.frame': 209 obs. of  6 variables:
##   ..$ chr       : chr [1:209] "chr1" "chr1" "chr1" "chr1" ...
##   ..$ start     : int [1:209] 6613856 14923129 23798401 45337954 55213361 87128764 113756062 1137578
##   ..$ end       : int [1:209] 6616857 14926130 23801402 45340955 55216362 87131765 113759063 1137608
##   ..$ White_Adi.: num [1:209] 3.73 2.09 2.14 2.02 3.8 ...
```

```
##   ..$ Beige_Adi.: num [1:209] 4.54 2.55 2.36 2.92 4.59 ...
##   ..$ tss_set   : chr [1:209] "beige" "beige" "beige" "beige" ...
##  $ white  :'data.frame': 213 obs. of  6 variables:
##   ..$ chr       : chr [1:213] "chr1" "chr1" "chr1" "chr1" ...
##   ..$ start     : int [1:213] 6612730 11801676 14944918 17633256 23798780 26431281 33347884 55213363
##   ..$ end       : int [1:213] 6615731 11804677 14947919 17636257 23801781 26434282 33350885 55216364
##   ..$ White_Adi.: num [1:213] 4.591 2.003 0.861 -0.35 2.365 ...
##   ..$ Beige_Adi.: num [1:213] 5.447 1.773 0.637 -0.488 2.537 ...
##   ..$ tss_set   : chr [1:213] "white" "white" "white" "white" ...
##  $ not_sig:'data.frame': 23636 obs. of  6 variables:
##   ..$ chr       : chr [1:23636] "chr1" "chr1" "chr1" "chr1" ...
##   ..$ start     : int [1:23636] 27369 493475 496975 496975 497174 499606 500872 500872 512412 515251
##   ..$ end       : int [1:23636] 30370 496476 499976 499976 500175 502607 503873 503873 515413 518252
##   ..$ White_Adi.: num [1:23636] 0 0.0087 0 0 0 ...
##   ..$ Beige_Adi.: num [1:23636] -0.0124 0 0 0 0 ...
##   ..$ tss_set   : chr [1:23636] "not_sig" "not_sig" "not_sig" "not_sig" ...
```

```
annot <- do.call(rbind, enrich_tables)
table(annot$tss_set)
```

```
##
##   beige not_sig   white
##     209   23636     213
```

```
head(annot); nrow(annot)
```

```
##          chr    start      end White_Adi. Beige_Adi. tss_set
## beige.1 chr1  6613856  6616857   3.727283   4.543351   beige
## beige.2 chr1 14923129 14926130   2.089665   2.546589   beige
## beige.3 chr1 23798401 23801402   2.138708   2.357445   beige
## beige.4 chr1 45337954 45340955   2.018657   2.918385   beige
## beige.5 chr1 55213361 55216362   3.796409   4.585399   beige
## beige.6 chr1 87128764 87131765   1.176095   1.777729   beige
```

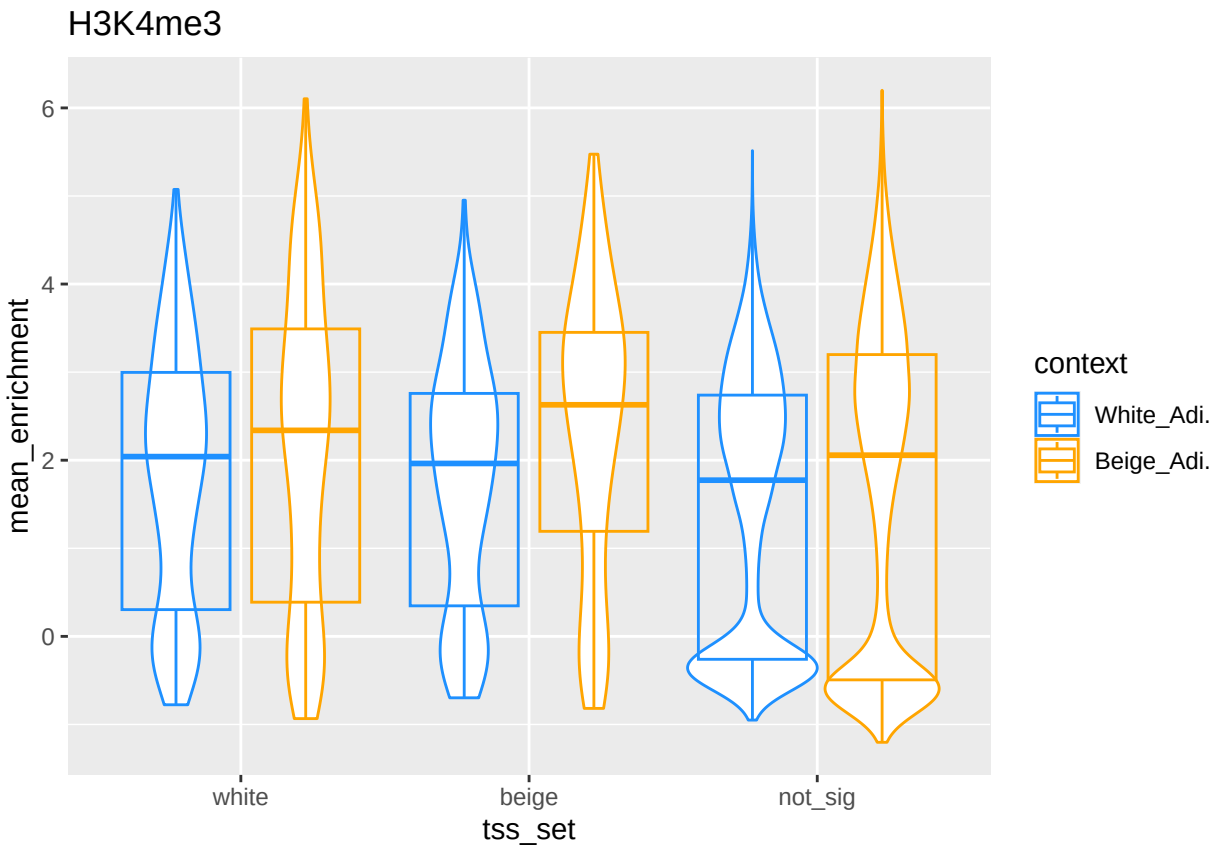
```
## [1] 24058
```

```
long = pivot_longer(annot, grep("Adi.",colnames(annot)), names_to = "context", values_to = "mean_enrichm
long$context = factor(long$context, levels=c("White_Adi.","Beige_Adi."))
long$tss_set = factor(long$tss_set, levels=c("white","beige","not_sig"))
```

```
ggplot(long,
       aes(x=tss_set, y=mean_enrichment, color=context)) +geom_violin()+
    geom_boxplot(fill=NA, position=position_dodge(0.9)) +scale_color_manual(values= c("dodgerblue","ora
```

```
## Warning: Removed 12 rows containing non-finite values (`stat_ydensity()`).
```

```
## Warning: Removed 12 rows containing non-finite values (`stat_boxplot()`).
```



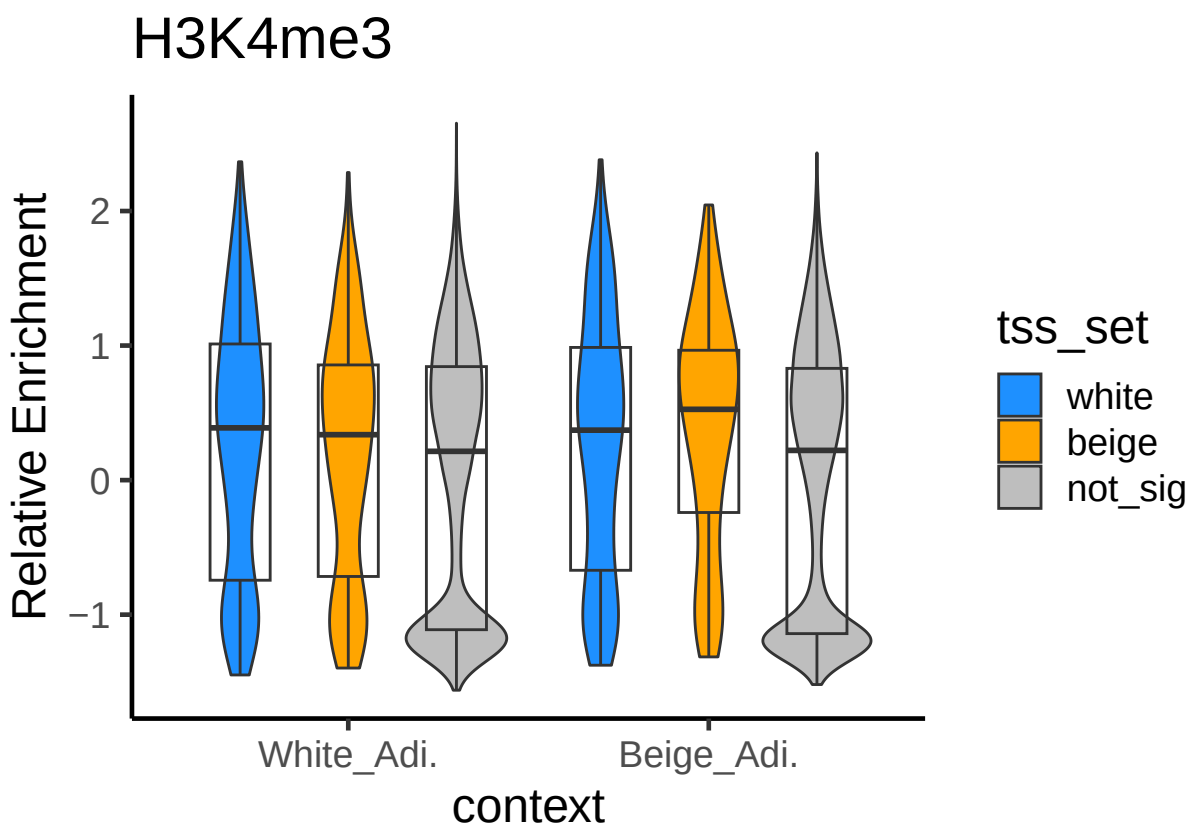
H3K4me3 signal is always higher in the beige sample. To normalise, we could compare the signal to the average of the background and the two sets for each sample. Or we could also calculate a zscore for each ChIP. that makes a lot of sense. Average will be weighted towards the background, but not too heavily.

```
relative = long %>% group_by(context) %>% mutate(zscore = scale(mean_enrichment), group=paste0(context,
relative$group = factor(relative$group, levels=paste0(rep(levels(relative$context),each=3), levels(rela

ggplot(relative,
       aes(fill=tss_set, y=zscore, x=context, group=group)) + geom_violin() +
    geom_boxplot(fill=NA, width=0.5, position= position_dodge(0.9)) + scale_fill_manual(values= c("dodg
    theme_classic(base_size=18) + ylab("Relative Enrichment") + ggtitle(histone)
```

```
## Warning: Removed 12 rows containing non-finite values (‘stat_ydensity()‘).
```

```
## Warning: Removed 12 rows containing non-finite values (‘stat_boxplot()‘).
```



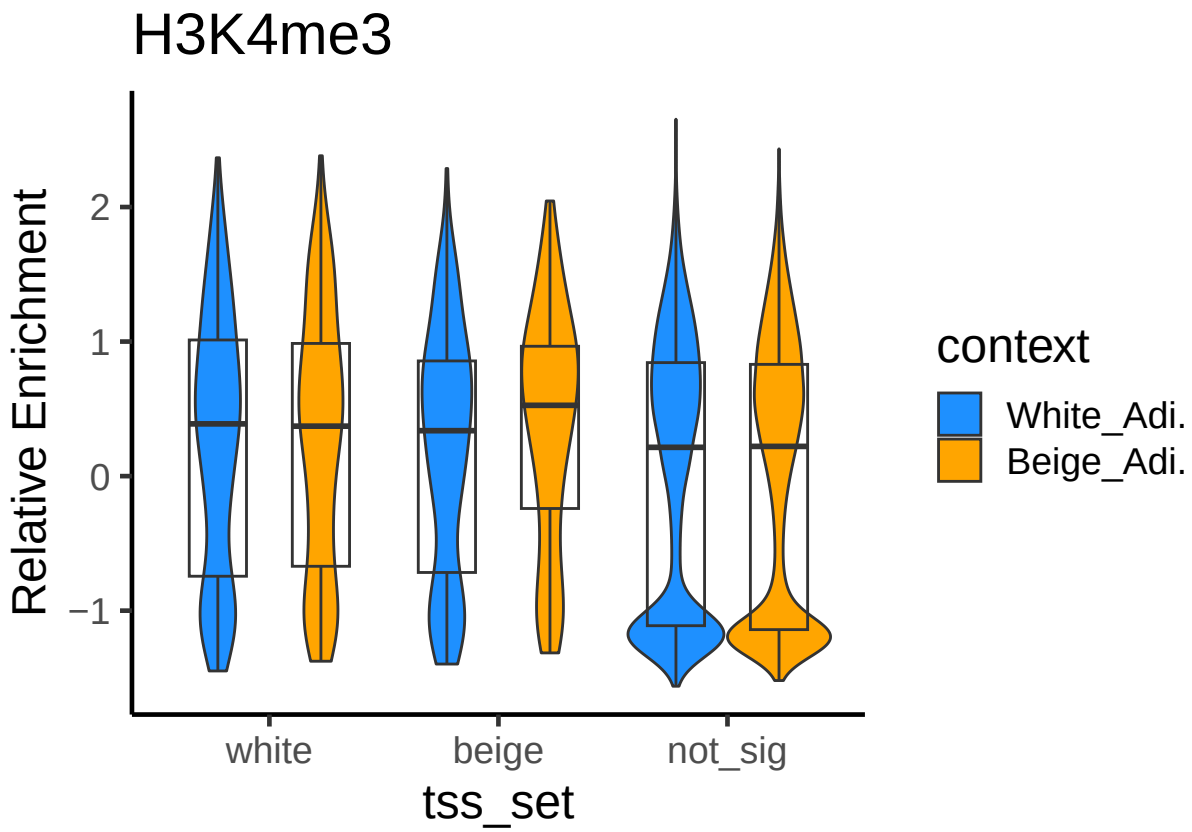

```
ggsave(here("31_leafcutter/histone_profile", histone, paste0(histone, "_violin_plot.pdf")))
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 12 rows containing non-finite values (`stat_ydensity()`).
## Removed 12 rows containing non-finite values (`stat_boxplot()`).
```

```
ggplot(relative,
       aes(x=tss_set, y=zscore, fill=context, group=group)) + geom_violin() +
    geom_boxplot(fill=NA, width=0.5, position= position_dodge(0.9))+scale_fill_manual(values= c("dodgerb
  theme_classic(base_size=18) + ylab("Relative Enrichment")+ ggtitle(histone)
```

```
## Warning: Removed 12 rows containing non-finite values (`stat_ydensity()`).
## Removed 12 rows containing non-finite values (`stat_boxplot()`).
```




```
summary(aov(zscore ~tss_set+context, data=relative))
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)
## tss_set         2     63  31.252   31.29 2.63e-14 ***
## context         1      0   0.000    0.00        1
## Residuals   48100  48039   0.999
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 12 observations deleted due to missingness
```

```
#print(kruskal.test(zscore ~ tss_set * context, data=relative)) #cannot run a kruskal wallis with inter

compare_means(zscore ~ tss_set, data=relative, method="wilcox.test", group.by="context")
```

```
## # A tibble: 6 x 9
##   context     .y.    group1 group2            p   p.adj p.format p.signif method
##   <fct>       <chr>  <chr>  <chr>         <dbl>   <dbl> <chr>    <chr>    <chr>
## 1 White_Adi. zscore white  beige       5.50e-1 5.5 e-1 0.54990  ns       Wilco~
## 2 White_Adi. zscore white  not_sig     1.91e-4 7.6 e-4 0.00019  ***      Wilco~
## 3 White_Adi. zscore beige  not_sig     5.10e-3 1.5 e-2 0.00510  **       Wilco~
## 4 Beige_Adi. zscore white  beige       1.64e-1 3.3 e-1 0.16386  ns       Wilco~
## 5 Beige_Adi. zscore white  not_sig     3.32e-5 1.7 e-4 3.3e-05  ****     Wilco~
## 6 Beige_Adi. zscore beige  not_sig     6.54e-9 3.90e-8 6.5e-09  ****     Wilco~
```

With the subsampled background analysis of last time using a zscore meant the background was different between the two chip conditons. With removing DEG and using a larger background we get a more consistent average H3relative signal between the two conditions.

```
summary(aov(zscore ~tss_set*context, data=relative))
```

```
##                     Df Sum Sq Mean Sq F value   Pr(>F)
## tss_set              2     63  31.252  31.293 2.62e-14 ***
## context              1      0   0.000   0.000    1.000
## tss_set:context      2      4   2.014   2.016    0.133
## Residuals        48098  48035   0.999
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 12 observations deleted due to missingness
```

```
compare_means(zscore ~ context, data=relative, method="t.test", group.by="tss_set") # <- not significan
```

```
## # A tibble: 3 x 9
##   tss_set .y.    group1     group2          p p.adj p.format p.signif method
##   <fct>   <chr>  <chr>      <chr>       <dbl> <dbl> <chr>    <chr>    <chr>
## 1 beige   zscore White_Adi. Beige_Adi. 0.0248 0.074 0.025    *        T-test
## 2 white   zscore White_Adi. Beige_Adi. 0.851  1     0.851    ns       T-test
## 3 not_sig zscore White_Adi. Beige_Adi. 0.838  1     0.838    ns       T-test
```