Figure 2. Aggregate signals analysis for sorted ChIP-seq binding sites

Zheng Zuo

04/15/2022

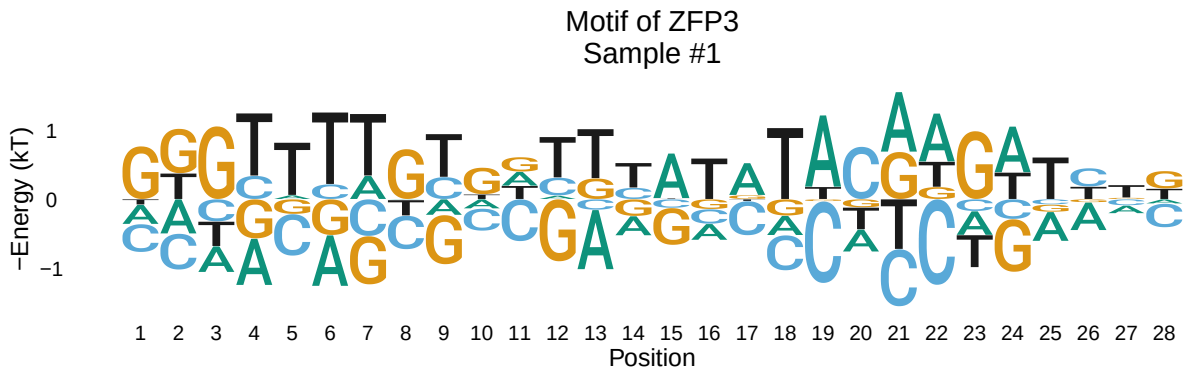
Contents


```
require(dplyr)
require(ggplot2)
require(GenomicRanges)
require(TFCookbook)
```

Build motif models based on all single variants of reference sequence

```
load("ZFP3.RData")

require(dplyr)
Sample1.processed %>%
  dplyr::filter(Mismatch<=1) %>%
  dplyr::rename(Energy=`Relative Energy`) %>%
  TFCookbook::buildEnergyModel() %>%
  as.PEM() -> ZFP3.Full.PEM

ZFP3.Core.PEM <- ZFP3.Full.PEM
ZFP3.Core.PEM[,1:15] <- 0

ZFP3.Upstream.PEM <- ZFP3.Full.PEM
```

```
ZFP3.Upstream.PEM[,16:28] <- 0
#TFCookbook::predictEnergy("GGGTTTTGTGGTTTATATACAAGATCTG", ZFP3.Full.PEM)
require(ggplot2)
ZFP3.Full.PEM %>%
  TFCookbook::plotEnergyLogo() + ggtitle("Motif of ZFP3\nSample #1") + theme(plot.title = element_text(
```



Predicting the binding energy of putative sites within ChIP-seq peaks of H293 cells

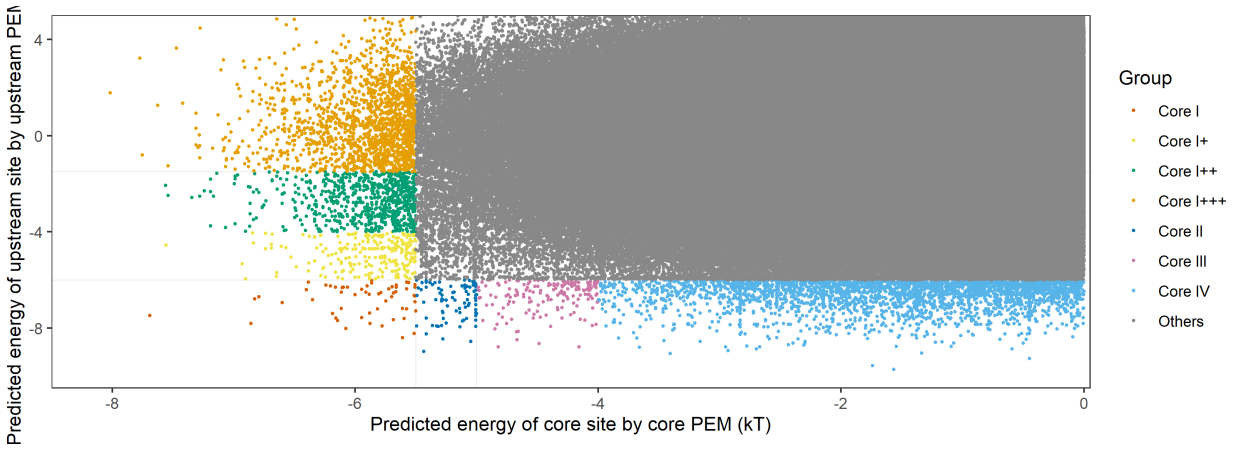
```
require(GenomicRanges)

ZFP3.H293.bigWig <- "ENCFF655DZE.bigWig"
#ZFP3.SKN.bigWig  <- "ENCFF774MKP.bigWig"

ChIP.H293.peaks <- read.table("ENCFF107KSN.bed") %>%
                   GenomicRanges::makeGRangesFromDataFrame(seqnames.field = "V1",
                                                           start.field    = "V2",
                                                           end.field      = "V3") %>% unique()

#ChIP.SKN.peaks <- read.table("ENCFF049VST.bed") %>%
 #                  GenomicRanges::makeGRangesFromDataFrame(seqnames.field = "V1",
  #                                                         start.field    = "V2",
   #                                                        end.field      = "V3") %>% unique()

random.pos = sample.int(2000, size = length(ChIP.H293.peaks), replace = TRUE) %>%
             magrittr::mod(width(ChIP.H293.peaks))

Random.sites <- ChIP.H293.peaks %>%
                as_tibble() %>%
                mutate(start = start+random.pos) %>%
                mutate(end   = start+27,
                       strand= sample(c("+", "-"), size = length(ChIP.H293.peaks), replace = TRUE),
                       Group = "Random") %>%
                GenomicRanges::makeGRangesFromDataFrame(keep.extra.columns = TRUE)

ZFP3.sites <- TFCookbook::matchPEM(PEM = ZFP3.Core.PEM,
                                   subject = ChIP.H293.peaks,
                                   genome  = "hg38",
```

```
                                  out     = "positions",
                                  E.cutoff= 0)

ZFP3.sites$predicted.Core.Energy     <- ZFP3.sites$predicted.Energy
ZFP3.sites$predicted.Energy          <- NULL
ZFP3.sites$predicted.Upstream.Energy <- TFCookbook::predictEnergy(ZFP3.sites$Sequence, ZFP3.Upstream.PE
ZFP3.sites$predicted.Full.Energy     <- ZFP3.sites$predicted.Upstream.Energy + ZFP3.sites$predicted.Cor
```

Figure 2A

```
ZFP3.sites <- as_tibble(ZFP3.sites) %>%
  mutate(Group = case_when(
    predicted.Upstream.Energy < (-6) &  between(predicted.Core.Energy, -10 , -5.5)   ~ "Group I",
    predicted.Upstream.Energy < (-6) &  between(predicted.Core.Energy, -5.5, -5  )   ~ "Group II",
    predicted.Upstream.Energy < (-6) &  between(predicted.Core.Energy, -5  , -4)     ~ "Group III",
    predicted.Upstream.Energy < (-6) &  between(predicted.Core.Energy, -4  , 0   )   ~ "Group IV",
    predicted.Core.Energy     < (-5.5)& between(predicted.Upstream.Energy, -6 , -4)  ~ "Group I+",
    predicted.Core.Energy     < (-5.5)& between(predicted.Upstream.Energy, -4, -1.5) ~ "Group I++",
    predicted.Core.Energy     < (-5.5)& between(predicted.Upstream.Energy, -1.5 , 10)~ "Group I+++",
    TRUE ~ "Others"),
        Group2 = case_when(
    predicted.Full.Energy < (-11) &  between(predicted.Core.Energy, -10 , -5.5) ~ "Group A",
    predicted.Core.Energy < (-5.5)  &  between(predicted.Full.Energy,-11,  -9)  ~ "Group A+",
    predicted.Core.Energy < (-5.5)  &  between(predicted.Full.Energy, -9,  10)  ~ "Group A++",
    predicted.Full.Energy < (-11) &  between(predicted.Core.Energy, -5.5, -4.5) ~ "Group B",
    predicted.Full.Energy < (-11) &  between(predicted.Core.Energy, -4.5, 0   ) ~ "Group C",
    TRUE ~ "Others"
  )) %>%
  GenomicRanges::makeGRangesFromDataFrame(keep.extra.columns = TRUE)
```

```
#as_tibble(ZFP3.sites) %>% subset(Group == "Core I++")
(ZFP3.sites %>% as_tibble() %>%
ggplot(aes(x = predicted.Core.Energy,
           y = predicted.Upstream.Energy,
           color = Group))+
  geom_point(size = 0.6)+
  scale_colour_manual(values=c("#D55E00","#F0E442","#009E73","#E69F00",
                               "#0072B2", "#CC79A7", "#56B4E9","#888888")) +
  xlab("Predicted energy of core site by core PEM (kT)") +
  ylab("Predicted energy of upstream site by upstream PEM (kT)") +
  theme_bw()+
  scale_x_continuous(minor_breaks = c(-5.5, -5, -4),
                     limits = c(-8.5, 0.05), expand = c(0,0)) +
  scale_y_continuous(breaks = c(4, 0, -4, -8), minor_breaks = c(-6, -4, -1.5),
                     limits = c(-10.5, 5), expand = c(0,0)) +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_blank()) -> plot.energy.groups)


#ggsave("Energy.CoreVsUpstream.plots.png", height = 3.5, width = 10)
```



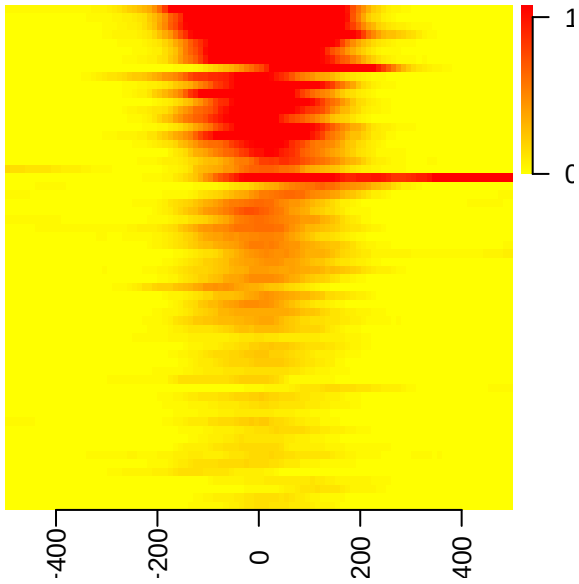

```
as_tibble(ZFP3.sites) %>%
  group_by(Group) %>%
  summarise(Number = n())
```

```
## # A tibble: 8 x 2
##   Group       Number
##   <chr>        <int>
## 1 Group I         60
## 2 Group I+       257
## 3 Group I++      759
## 4 Group I+++    1652
## 5 Group II        87
## 6 Group III      169
## 7 Group IV      3262
## 8 Others     1197803
```

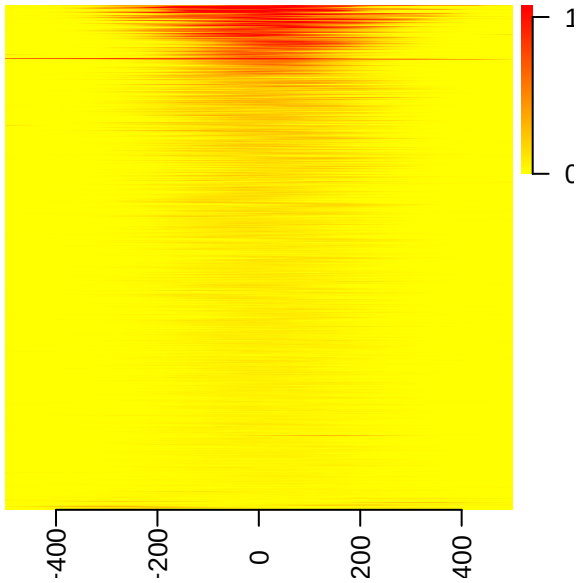
Figure 2B

```
features.GroupI = subset(ZFP3.sites, Group == "Group I") %>%
                  GenomicRanges::flank(500, both = TRUE)

features.GroupI.p3 = subset(ZFP3.sites, Group == "Group I+++") %>%
                  GenomicRanges::flank(500, both = TRUE)

features.GroupIV = subset(ZFP3.sites, Group == "Group IV") %>%
                  GenomicRanges::flank(500, both = TRUE)

features.Random =Random.sites %>%
                  GenomicRanges::flank(500, both = TRUE)


rtracklayer::import.bw(ZFP3.H293.bigWig,
                       selection = rtracklayer::BigWigSelection(features.GroupI),
                       as = "RleList") %>% list() %>%
ChIPpeakAnno::featureAlignedSignal(feature.gr = features.GroupI) %>%
ChIPpeakAnno::featureAlignedHeatmap(features.GroupI,
```

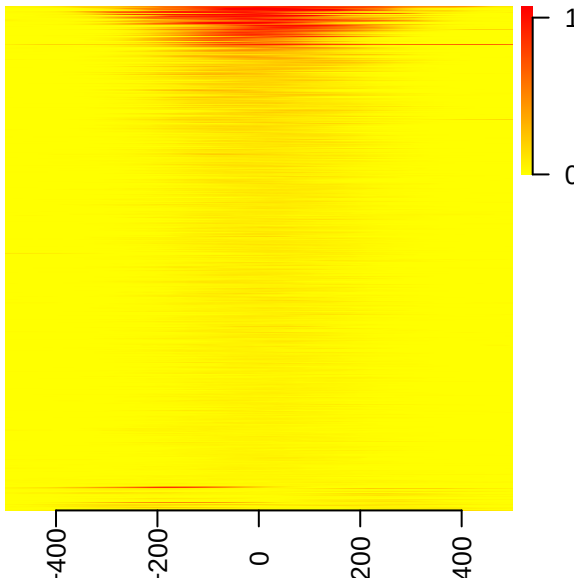
```
                                upstream=500, downstream=500,
                                upper.extreme=c(150, 0.5,4))  -> Heatmap.GroupI
```



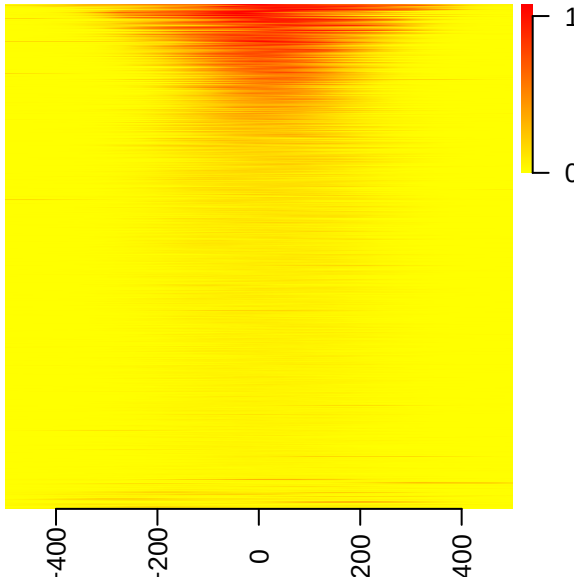

```
rtracklayer::import.bw(ZFP3.H293.bigWig,
                       selection = rtracklayer::BigWigSelection(features.GroupI.p3),
                       as = "RleList") %>% list() %>%
ChIPpeakAnno::featureAlignedSignal(feature.gr = features.GroupI.p3) %>%
ChIPpeakAnno::featureAlignedHeatmap(features.GroupI.p3,
                                upstream=500, downstream=500,
                                upper.extreme=c(150, 0.5,4)) -> Heatmap.GroupI.p3
```



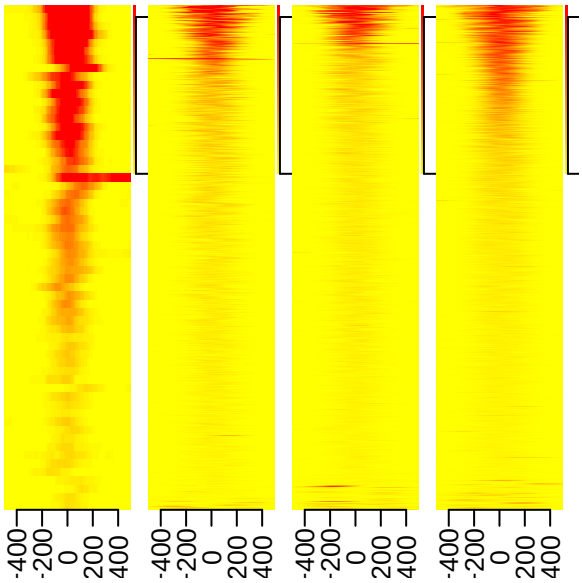

```
rtracklayer::import.bw(ZFP3.H293.bigWig,
                       selection = rtracklayer::BigWigSelection(features.GroupIV),
                       as = "RleList") %>% list() %>%
ChIPpeakAnno::featureAlignedSignal(feature.gr = features.GroupIV) %>%
ChIPpeakAnno::featureAlignedHeatmap(features.GroupIV,
                                    upstream=500, downstream=500,
                                    upper.extreme=c(150, 0.5,4)) -> Heatmap.GroupIV
```



```
rtracklayer::import.bw(ZFP3.H293.bigWig,
                       selection = rtracklayer::BigWigSelection(features.Random),
                       as = "RleList") %>% list() %>%
ChIPpeakAnno::featureAlignedSignal(feature.gr = features.Random) %>%
ChIPpeakAnno::featureAlignedHeatmap(features.Random,
                                    upstream=500, downstream=500,
                                    upper.extreme=c(150, 0.5,4)) -> Heatmap.Random
```




```
cowplot::plot_grid(Heatmap.GroupI,
                   Heatmap.GroupI.p3,
                   Heatmap.GroupIV,
                   Heatmap.Random,
                   ncol = 4)
```

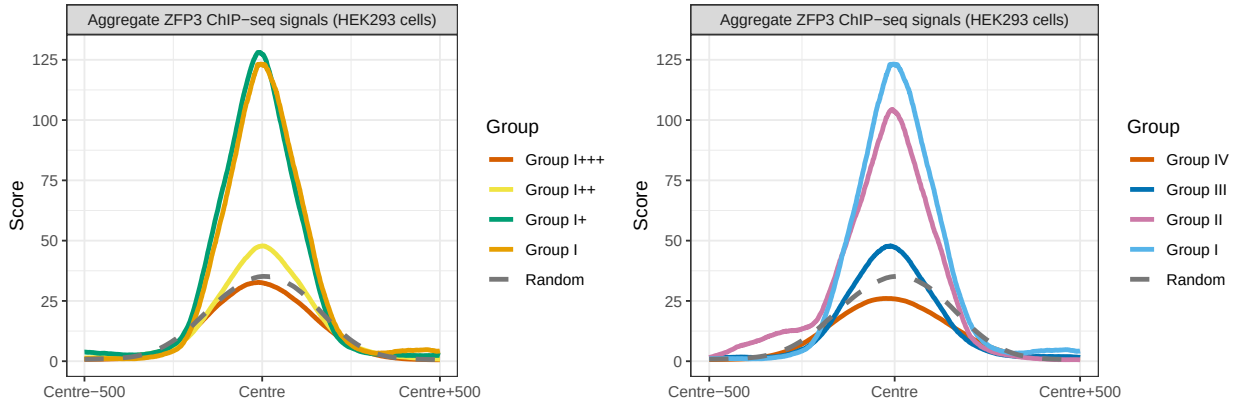



```
#ggsave("SignalsVsFeatures.Heatmaps.svg", height = 3.8, width = 2.6)
```

Figure 2C, 2D

```
c(ZFP3.sites, Random.sites) %>%
  subset(Group %in% c("Group I", "Group I+", "Group I++", "Group I+++", "Random")) %>%
  soGGi::regionPlot(bamFile    = ZFP3.H293.bigWig,
                    testRanges = .,
                    samplename = "Aggregate ZFP3 ChIP-seq signals (HEK293 cells)",
                    format = "bigwig", distanceAround = 500) %>%
  soGGi::plotRegion(summariseBy = "Group",
                    groupBy = "Sample", colourBy = "Group", lineBy = "Group") + theme_bw() + xlab("") +
                    scale_colour_manual(values=c("#D55E00","#F0E442","#009E73","#E69F00", "#777777")) +
                    scale_y_continuous(breaks = seq(0, 125, 25), limits = c(0, 129)) +
                    scale_linetype_manual(values = c("solid", "solid", "solid", "solid", "dashed"))->
  plot.Vary_By_Upstream

c(ZFP3.sites, Random.sites) %>%
  subset(Group %in% c("Group I", "Group II", "Group III", "Group IV", "Random")) %>%
  soGGi::regionPlot(bamFile    = ZFP3.H293.bigWig,
                    testRanges = .,
                    samplename = "Aggregate ZFP3 ChIP-seq signals (HEK293 cells)",
                    format = "bigwig", distanceAround = 500) %>%
  soGGi::plotRegion(summariseBy = "Group",
                    groupBy = "Sample", colourBy = "Group", lineBy = "Group") + theme_bw() + xlab("") +
                    scale_colour_manual(values=c("#D55E00", "#0072B2", "#CC79A7", "#56B4E9","#777777"))
                    scale_y_continuous(breaks = seq(0, 125, 25), limits = c(0, 129))+
                    scale_linetype_manual(values = c("solid", "solid", "solid", "solid", "dashed") ) ->
  plot.Vary_By_Core
```

```
cowplot::plot_grid(plot.Vary_By_Upstream,
                   plot.Vary_By_Core,
                   ncol =2)
```




```
#ggsave("Aggregate.ChIP.CoreVsUpstream.plots.H293.svg", height = 3.5, width = 10)
```

Figure 2E

```
(ZFP3.sites %>% as_tibble() %>%
ggplot(aes(x = predicted.Core.Energy,
           y = predicted.Full.Energy,
           color = Group2))+
  geom_point(size = 0.6)+
  scale_colour_manual(values=c("#D55E00","#F0E442","#009E73",
                               "#0072B2", "#CC79A7","#888888")) +
  xlab("Predicted energy of core site by core PEM (kT)") +
  ylab("Predicted energy of full site by full PEM (kT)") +
  theme_bw()+
  scale_x_continuous(minor_breaks = c(-5.5, -4.5, -4),
                     limits = c(-8.5, 0.05), expand = c(0,0)) +
  scale_y_continuous(breaks = c(2, -2, -6, -10, -14), minor_breaks = c(-11, -9),
                     limits = c(-16, 2.2), expand = c(0,0))  +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_blank()) -> plot.energy.groups2)


#ggsave("Energy.CoreVsFull.plots.png", height = 3.5, width = 10)
```

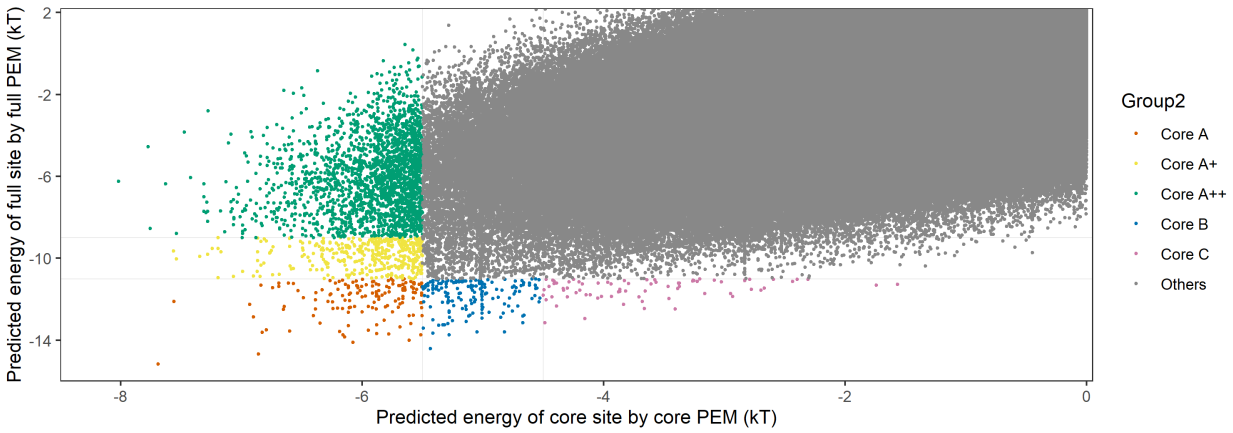
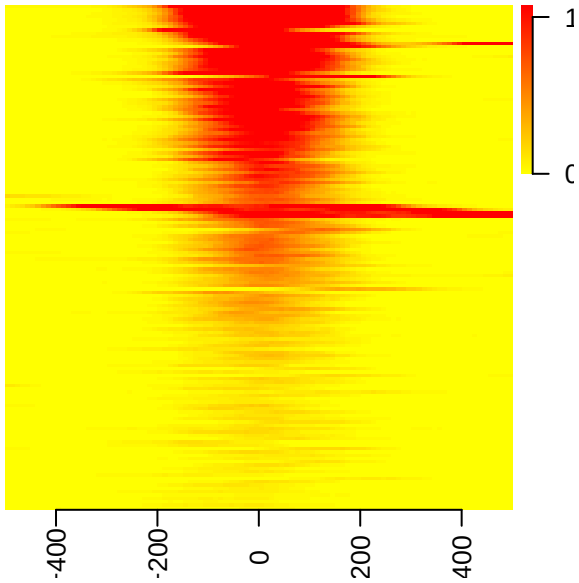
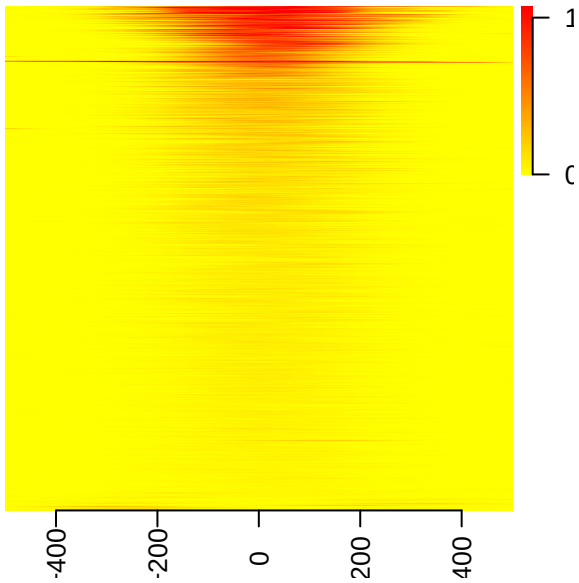


Figure 2F

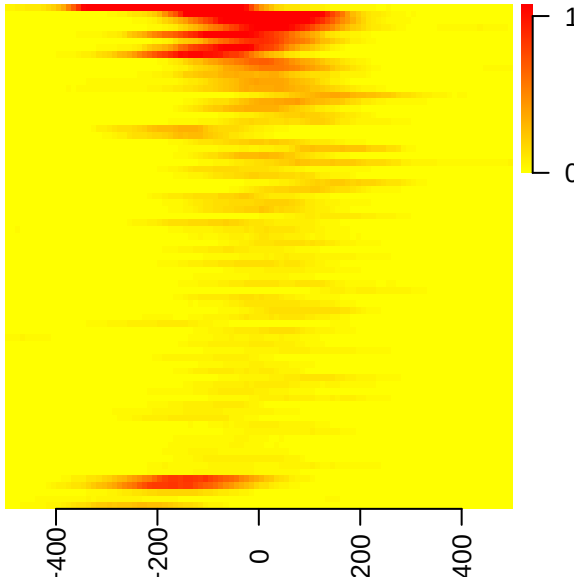
```
ZFP3.sites$Group = ZFP3.sites$Group2


features.GroupA = subset(ZFP3.sites, Group == "Group A") %>%
                  GenomicRanges::flank(500, both = TRUE)

features.GroupA.p2 = subset(ZFP3.sites, Group == "Group A++") %>%
                  GenomicRanges::flank(500, both = TRUE)

features.GroupC = subset(ZFP3.sites, Group == "Group C") %>%
                  GenomicRanges::flank(500, both = TRUE)

features.Random =Random.sites %>%
                  GenomicRanges::flank(500, both = TRUE)


rtracklayer::import.bw(ZFP3.H293.bigWig,
                       selection = rtracklayer::BigWigSelection(features.GroupA),
                       as = "RleList") %>% list() %>%
ChIPpeakAnno::featureAlignedSignal(feature.gr = features.GroupA) %>%
ChIPpeakAnno::featureAlignedHeatmap(features.GroupA,
                                    upstream=500, downstream=500,
                                    upper.extreme=c(150, 0.5,4))  -> Heatmap.GroupA
```



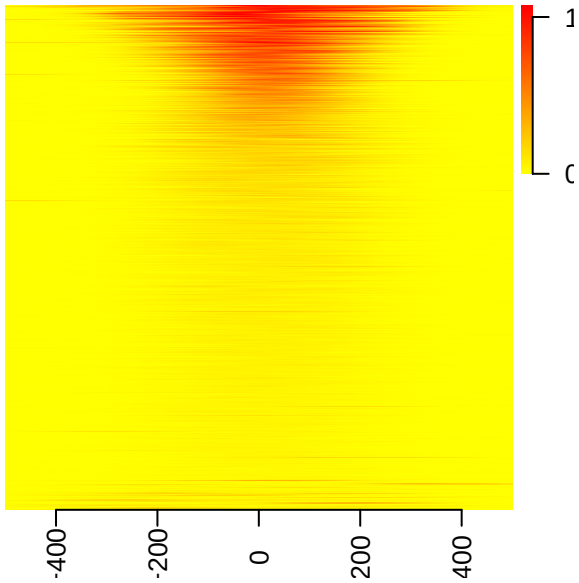

```
rtracklayer::import.bw(ZFP3.H293.bigWig,
                       selection = rtracklayer::BigWigSelection(features.GroupA.p2),
                       as = "RleList") %>% list() %>%
ChIPpeakAnno::featureAlignedSignal(feature.gr = features.GroupA.p2) %>%
ChIPpeakAnno::featureAlignedHeatmap(features.GroupA.p2,
                                    upstream=500, downstream=500,
                                    upper.extreme=c(150, 0.5,4)) -> Heatmap.GroupA.p2
```



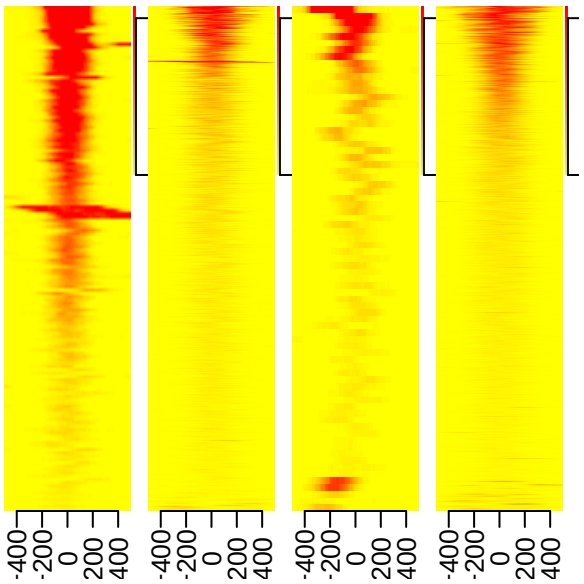
```
rtracklayer::import.bw(ZFP3.H293.bigWig,
                       selection = rtracklayer::BigWigSelection(features.GroupC),
                       as = "RleList") %>% list() %>%
ChIPpeakAnno::featureAlignedSignal(feature.gr = features.GroupC) %>%
ChIPpeakAnno::featureAlignedHeatmap(features.GroupC,
                                    upstream=500, downstream=500,
                                    upper.extreme=c(150, 0.5,4)) -> Heatmap.GroupC
```




```
rtracklayer::import.bw(ZFP3.H293.bigWig,
                       selection = rtracklayer::BigWigSelection(features.Random),
                       as = "RleList") %>% list() %>%
ChIPpeakAnno::featureAlignedSignal(feature.gr = features.Random) %>%
ChIPpeakAnno::featureAlignedHeatmap(features.Random,
                                    upstream=500, downstream=500,
                                    upper.extreme=c(150, 0.5,4)) -> Heatmap.Random
```



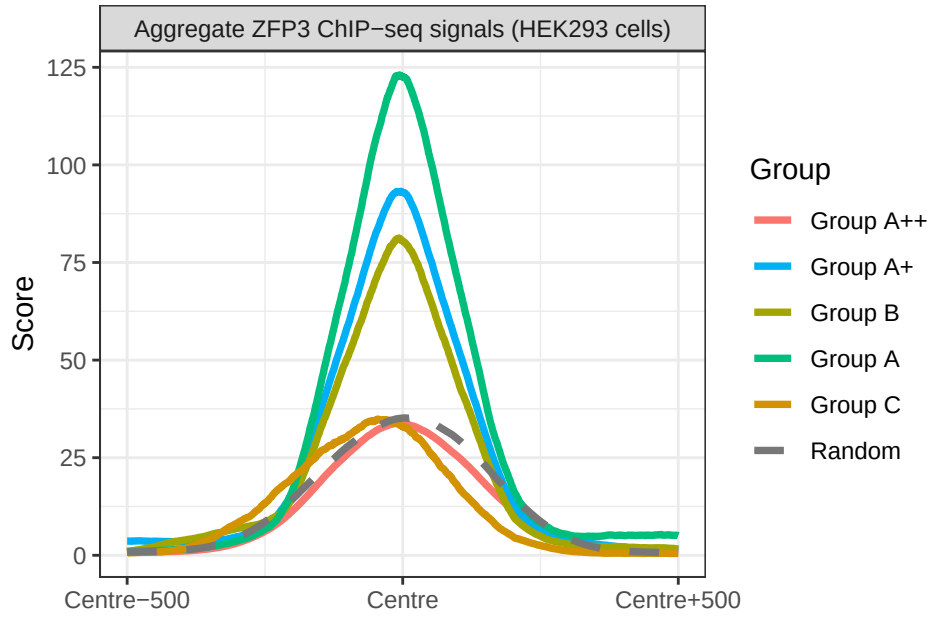

```
cowplot::plot_grid(Heatmap.GroupA,
                   Heatmap.GroupA.p2,
                   Heatmap.GroupC,
                   Heatmap.Random,
                   ncol = 4)
```




```
#ggsave("SignalsVsFeatures.Heatmaps2.svg", height = 3.8, width = 2.6)
```

Figure 2G

```
c(ZFP3.sites, Random.sites) %>%
  subset(Group %in% c("Group A", "Group A+", "Group A++", "Group B", "Group C", "Random")) %>%
  soGGi::regionPlot(bamFile    = ZFP3.H293.bigWig,
                    testRanges = .,
                    samplename = "Aggregate ZFP3 ChIP-seq signals (HEK293 cells)",
                    format = "bigwig", distanceAround = 500) %>%
  soGGi::plotRegion(summariseBy = "Group",
                    groupBy = "Sample", colourBy = "Group", lineBy = "Group") + theme_bw() + xlab("") +
                    scale_colour_manual(values=c("#F8766D", "#00B0F6", "#A3A500", "#00BF7D", "#D39200",
                    scale_linetype_manual(values = c("solid", "solid", "solid", "solid", "solid", "dash
```



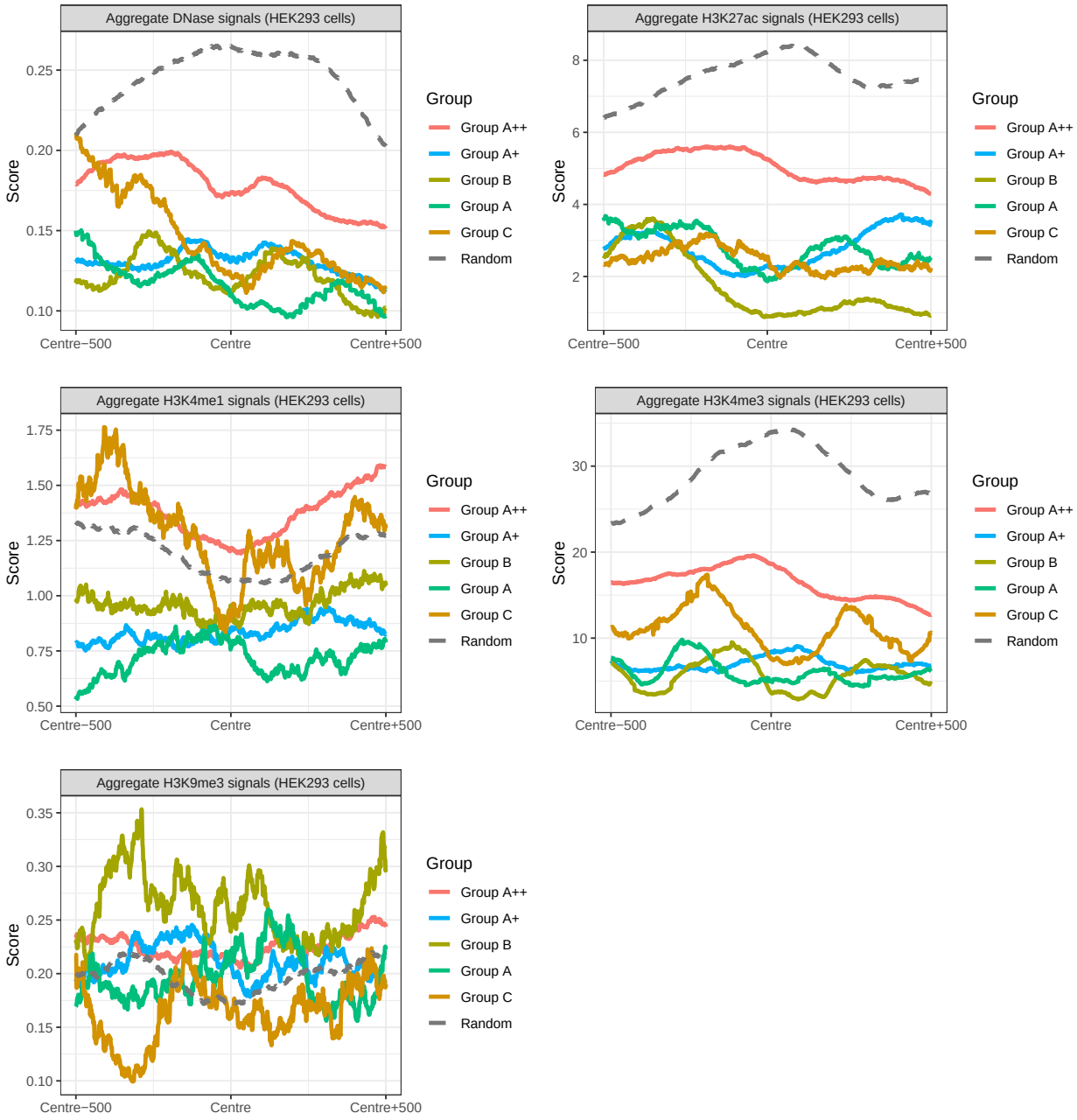

```
#ggsave("Aggregate.ChIP.CoreVsFull.plots.svg", height = 3.5, width = 5)
```

Figure S2

```
DNase.H293T.bigWig  <- "DNase-seq signals/ENCFF529BOG.H293T.bigWig"
H3K27ac.H293.bigWig <- "Chromatin signals/ENCFF157TGK.HEK293.H3K27ac.bigWig"
H3K4me1.H293.bigWig <- "Chromatin signals/ENCFF003LZR.HEK293.H3K4me1.bigWig"
H3K4me3.H293.bigWig <- "Chromatin signals/ENCFF315TAU.HEK293.H3K4me3.bigWig"
H3K9me3.H293.bigWig <- "Chromatin signals/ENCFF758LNF.HEK293.H3K9me3.bigWig"

c(ZFP3.sites, Random.sites) %>%
  subset(Group %in% c("Group A", "Group A+", "Group A++", "Group B", "Group C", "Random")) %>%
  soGGi::regionPlot(bamFile    = DNase.H293T.bigWig,
                    testRanges = .,
                    samplename = "Aggregate DNase signals (HEK293 cells)",
```

```
                    format = "bigwig", distanceAround = 500) %>%
  soGGi::plotRegion(summariseBy = "Group",
                    groupBy = "Sample", colourBy = "Group", lineBy = "Group") + theme_bw() + xlab("") +
                    scale_colour_manual(values=c("#F8766D", "#00B0F6", "#A3A500", "#00BF7D", "#D39200",
                    scale_linetype_manual(values = c("solid", "solid", "solid", "solid", "solid", "dash

c(ZFP3.sites, Random.sites) %>%
  subset(Group %in% c("Group A", "Group A+", "Group A++", "Group B", "Group C", "Random")) %>%
  soGGi::regionPlot(bamFile    = H3K27ac.H293.bigWig,
                    testRanges = .,
                    samplename = "Aggregate H3K27ac signals (HEK293 cells)",
                    format = "bigwig", distanceAround = 500) %>%
  soGGi::plotRegion(summariseBy = "Group",
                    groupBy = "Sample", colourBy = "Group", lineBy = "Group") + theme_bw() + xlab("") +
                    scale_colour_manual(values=c("#F8766D", "#00B0F6", "#A3A500", "#00BF7D", "#D39200",
                    scale_linetype_manual(values = c("solid", "solid", "solid", "solid", "solid", "dash

c(ZFP3.sites, Random.sites) %>%
  subset(Group %in% c("Group A", "Group A+", "Group A++", "Group B", "Group C", "Random")) %>%
  soGGi::regionPlot(bamFile    = H3K4me1.H293.bigWig,
                    testRanges = .,
                    samplename = "Aggregate H3K4me1 signals (HEK293 cells)",
                    format = "bigwig", distanceAround = 500) %>%
  soGGi::plotRegion(summariseBy = "Group",
                    groupBy = "Sample", colourBy = "Group", lineBy = "Group") + theme_bw() + xlab("") +
                    scale_colour_manual(values=c("#F8766D", "#00B0F6", "#A3A500", "#00BF7D", "#D39200",
                    scale_linetype_manual(values = c("solid", "solid", "solid", "solid", "solid", "dash

c(ZFP3.sites, Random.sites) %>%
  subset(Group %in% c("Group A", "Group A+", "Group A++", "Group B", "Group C", "Random")) %>%
  soGGi::regionPlot(bamFile    = H3K4me3.H293.bigWig,
                    testRanges = .,
                    samplename = "Aggregate H3K4me3 signals (HEK293 cells)",
                    format = "bigwig", distanceAround = 500) %>%
  soGGi::plotRegion(summariseBy = "Group",
                    groupBy = "Sample", colourBy = "Group", lineBy = "Group") + theme_bw() + xlab("") +
                    scale_colour_manual(values=c("#F8766D", "#00B0F6", "#A3A500", "#00BF7D", "#D39200",
                    scale_linetype_manual(values = c("solid", "solid", "solid", "solid", "solid", "dash

c(ZFP3.sites, Random.sites) %>%
  subset(Group %in% c("Group A", "Group A+", "Group A++", "Group B", "Group C", "Random")) %>%
  soGGi::regionPlot(bamFile    = H3K9me3.H293.bigWig,
                    testRanges = .,
                    samplename = "Aggregate H3K9me3 signals (HEK293 cells)",
                    format = "bigwig", distanceAround = 500) %>%
  soGGi::plotRegion(summariseBy = "Group",
                    groupBy = "Sample", colourBy = "Group", lineBy = "Group") + theme_bw() + xlab("") +
                    scale_colour_manual(values=c("#F8766D", "#00B0F6", "#A3A500", "#00BF7D", "#D39200",
                    scale_linetype_manual(values = c("solid", "solid", "solid", "solid", "solid", "dash


cowplot::plot_grid(plot.DNase,
                   plot.H3K27ac,
```

```
                plot.H3K4me1, plot.H3K4me3, plot.H3K9me3,
                ncol =2)
```




```
#ggsave("Bias factors search.svg", height = 10.5, width = 10)
```