Mouse methylation level around protein coding genes and CpG islands

Xuanken Tay

```
library(data.table)
library(rtracklayer)
library(mclust)
library(tidyverse)
library(dplyr)
library(viridis)
library(ggridges)
```

Initialise variables

```
cgi_ggf_path <- "/wehisan/bioinf/lab_speed/txk/cpg_data/GRCm38/CGI_coordinates_GRCm38.tsv"

gtf_path <- "/stornext/HPCScratch/home/tay.x/imprinted_methylation/ensembl_GRCm38.98.chr.gtf"

mouse_per_read_stats_path <- "/wehisan/bioinf/lab_speed/txk/methylation_data/mouse/mouse_per_read_stats_
```

Load CGI, CpG data and genes data

```
cgi_ggf <- read_tsv(cgi_ggf_path,
                    col_names=c("chr", "start", "end", "name", "length", "CpGcount", "GCcount", "pctCpG",
                    col_types='ciiciiiddd', skip=1) %>%
  mutate(chr=sub("chr", "", chr)) %>%
  makeGRangesFromDataFrame(keep.extra.columns=TRUE)

gtf <- import(gtf_path)
genes <- gtf %>%
  as_data_frame() %>%
  group_by(gene_name, seqnames, gene_biotype, strand, type) %>%
  summarise(start=min(start),
            end=max(end))
```

```
## Warning: `as_data_frame()` is deprecated, use `as_tibble()` (but mind the new semantics).
## This warning is displayed once per session.
```

```
protein_coding_genes <- genes %>%
  as_data_frame() %>%
  filter(gene_biotype == "protein_coding",
         type == "gene") %>%
  makeGRangesFromDataFrame()
```

Load mouse per-read statistics

```
loadRData <- function(infile) {
          # loads an RData file and returns it
          # assume that the variable saved in same name as the file
          load(infile)
          get(ls()[ls() == gsub(basename(infile), pattern=".RData$",
                                replacement="")])
}

mouse_per_read_stats <- loadRData(mouse_per_read_stats_path) %>%
  dplyr::rename(seqnames=chr, start=pos) %>%
  mutate(end=start+1) %>%
  select(-c(read_id, strain)) %>%
  as.data.table()
```

Function to find overlaps between methylation data and the regions (and their surroundings)

```
find_overlaps <- function(gr, df, overhang=5000, feature_width=2) {

  # modify the group so that it includes surrounding as well
  gr <- gr %>%
    as_data_frame() %>%
    dplyr::rename(feature_start=start, feature_end=end) %>%
    mutate(start=feature_start-overhang,
           end=feature_end+overhang,
           id=row_number()) %>%
    as.data.table()
  setkey(gr, seqnames, start, end)

  # find overlap between df and gr
  overlap <- foverlaps(df, gr, nomatch=0) %>%
    mutate(feature_length=feature_end-feature_start,
           pos=(i.start+i.end)/2,
           pos=ifelse(strand=="+" | strand=="*", pos-feature_start, feature_end-pos),
           pos=ifelse(pos < 0, pos/overhang,
                      ifelse(pos > feature_length,
                             feature_width + (pos-feature_length)/overhang,
                             feature_width * pos/feature_length)),
           pos=round(pos, 2)) %>%
    na.omit()

  overlap
}
```

Functions to plot methylation level

```
plot_single_meth <- function(overlap_df, region_id, feature_width=2, feature_name="CGI") {
  summarised_df <- overlap_df %>%
```

```
    filter(id == region_id) %>%
    group_by(seqnames, feature_start, feature_end, pos) %>%
    summarise(mean=mean(beta))

  chrm <- first(summarised_df$seqnames)
  feature_start <- first(summarised_df$feature_start)
  feature_end <- first(summarised_df$feature_end)

  p <- summarised_df %>%
    ggplot(aes(x=pos, y=mean)) +
    geom_point() +
    geom_smooth(method="loess") +
    coord_cartesian(ylim=c(0, 1)) +
    geom_vline(xintercept=0, linetype="dashed") +
    geom_vline(xintercept=feature_width, linetype="dashed") +
    theme_bw() +
    scale_x_continuous(breaks=c(-1, 0, feature_width, feature_width + 1),
                       labels=c(paste0("-", overhang/1000, "Kb"),
                                paste0(feature_name, " start"),
                                paste0(feature_name, " end"),
                                paste0("+", overhang/1000, "Kb"))) +
    labs(x="Relative Genomic Position",
         y="Methylation (%)",
         title=paste0("Methylation level around ", feature_name, " ",
                      chrm, ":", feature_start, ":", feature_end))

  p
}

plot_meth <- function(overlap_df, num_cluster, feature_width=2, feature_name="CGI") {
  meth_stats <- overlap_df %>%
    group_by(id) %>%
    filter(pos >= 0, pos <= feature_width) %>%
    summarise(mean=mean(beta),
              median=median(beta),
              max=max(beta),
              min=min(beta),
              iqr=IQR(beta),
              sd=sd(beta)) %>%
    ungroup() %>%
    na.omit()

  clustered <- meth_stats %>%
    select(-id) %>%
    as.matrix() %>%
    scale() %>%
    kmeans(num_cluster)

  meth_stats <- meth_stats %>%
    mutate(cluster=clustered$cluster) %>%
    group_by(cluster) %>%
    mutate(count=n(),
           cluster_name=paste0(cluster, " (n=", count, ")")) %>%
```

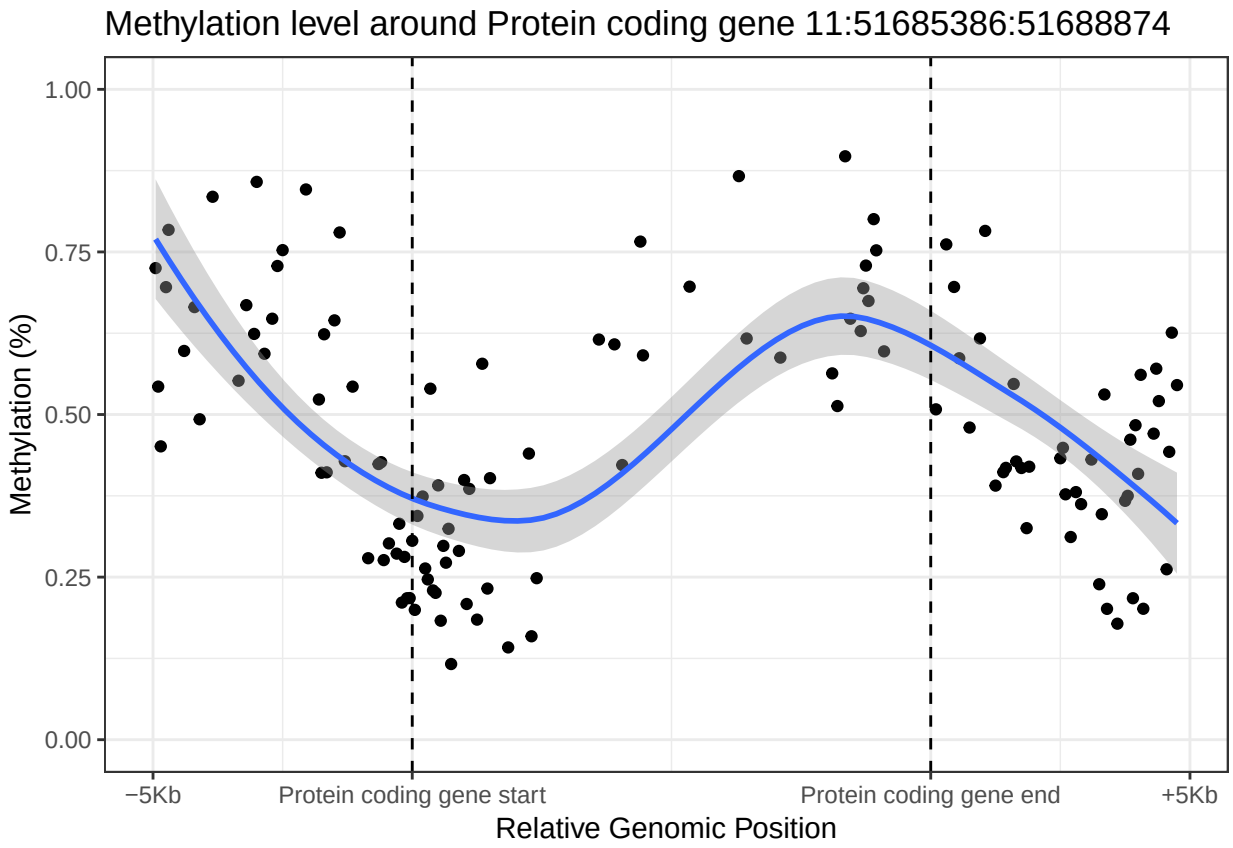
```
    ungroup()

  p <- meth_stats %>%
    select(id, cluster_name) %>%
    right_join(overlap_df, by=c("id"="id")) %>%
    na.omit() %>%
    group_by(cluster_name, pos) %>%
    summarise(mean=mean(beta),
              n_cgi=n()) %>%
    ungroup() %>%
    ggplot(aes(x=pos, y=mean, group=cluster_name, colour=cluster_name)) +
    geom_line(size=1) +
    coord_cartesian(ylim=c(0, 1)) +
    geom_vline(xintercept=0, linetype="dashed") +
    geom_vline(xintercept=feature_width, linetype="dashed") +
    scale_x_continuous(breaks=c(-1, 0, feature_width, feature_width + 1),
                       labels=c(paste0("-", overhang/1000, "Kb"),
                                paste0(feature_name, " start"),
                                paste0(feature_name, " end"),
                                paste0("+", overhang/1000, "Kb"))) +
    labs(x="Relative Genomic Position",
         y="Methylation (%)",
         title=paste0("Average Methylation level around ", feature_name),
         colour="Cluster") +
    theme_bw()

  p
}
```

Find overlaps between methylation data and protein coding regions

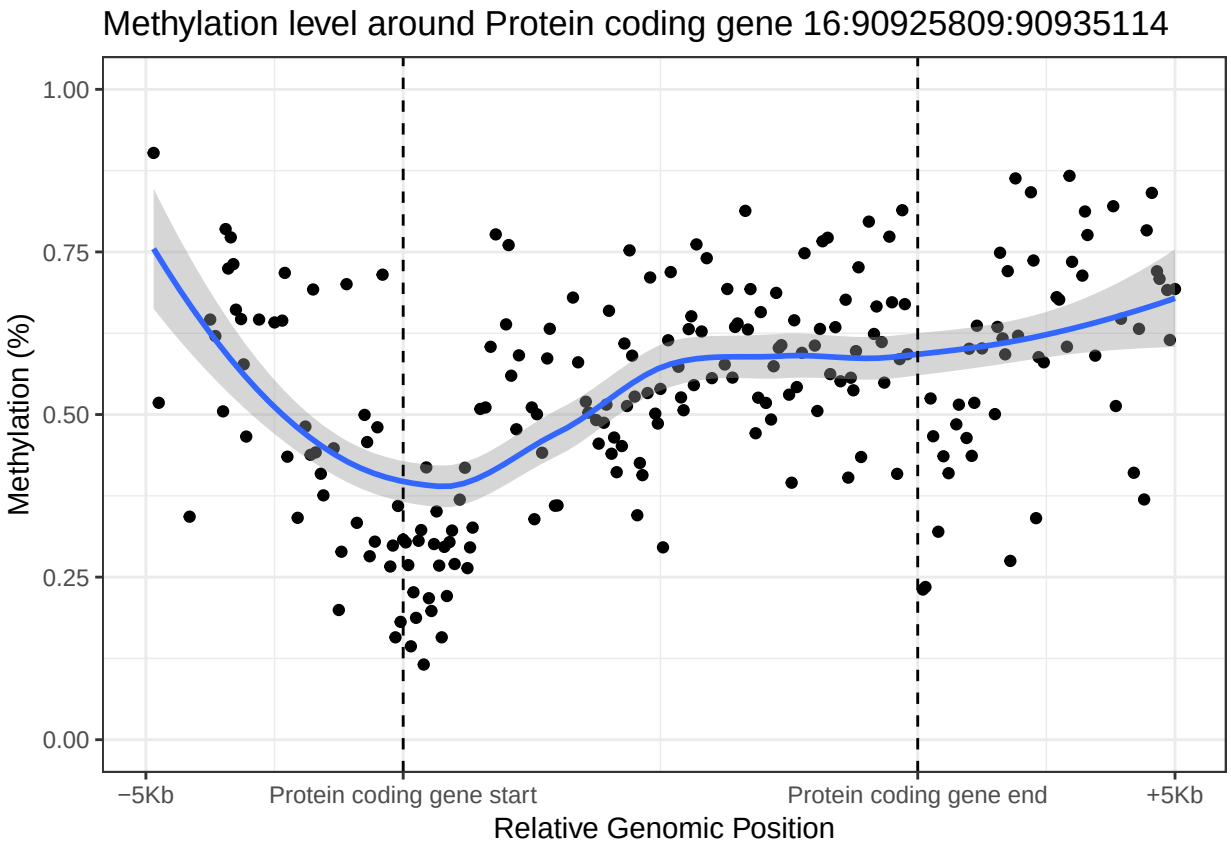
```
overhang <- 5000
feature_width <- 2
overlap_protein <- find_overlaps(protein_coding_genes, mouse_per_read_stats, overhang=overhang,
                                 feature_width=feature_width)
```

Plot gene methylation level

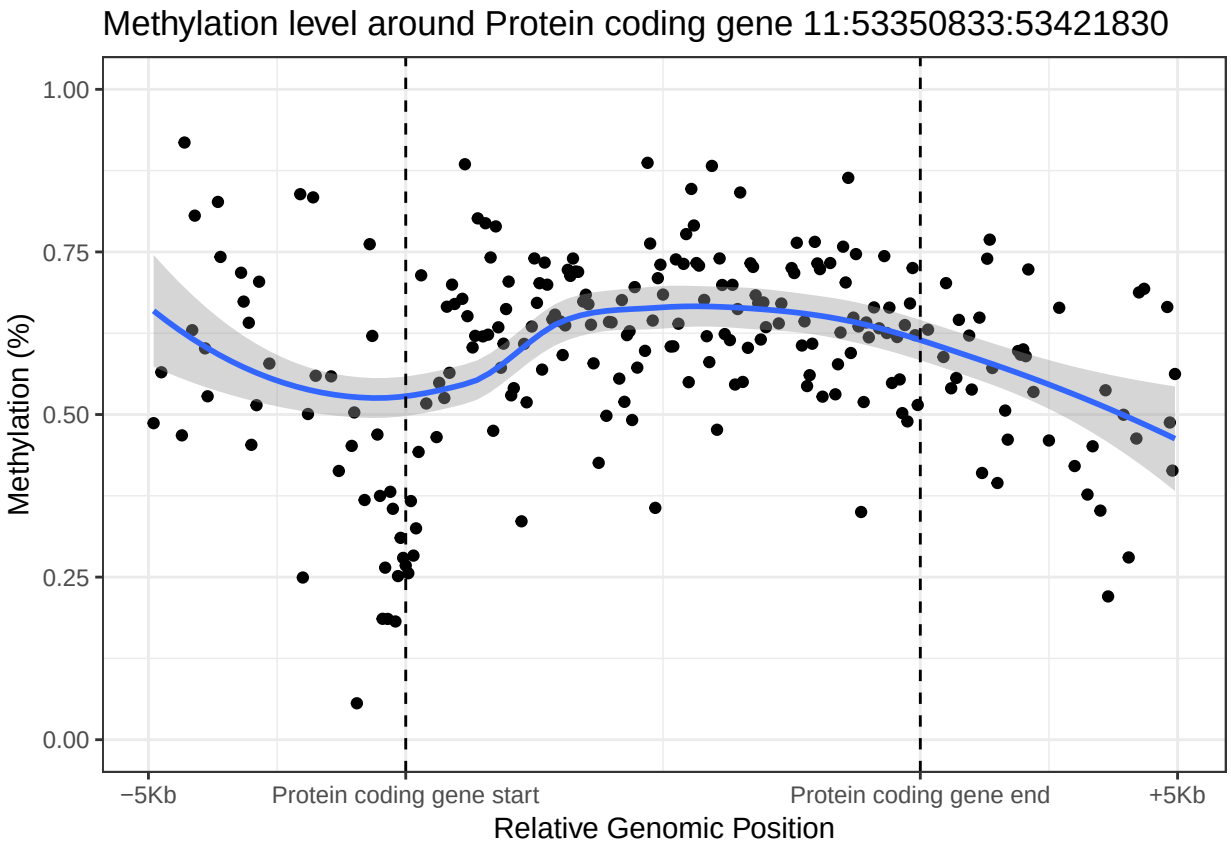
```
plot_single_meth(overlap_protein, 1, feature_name="Protein coding gene")
```




```
plot_single_meth(overlap_protein, 10, feature_name="Protein coding gene")
```

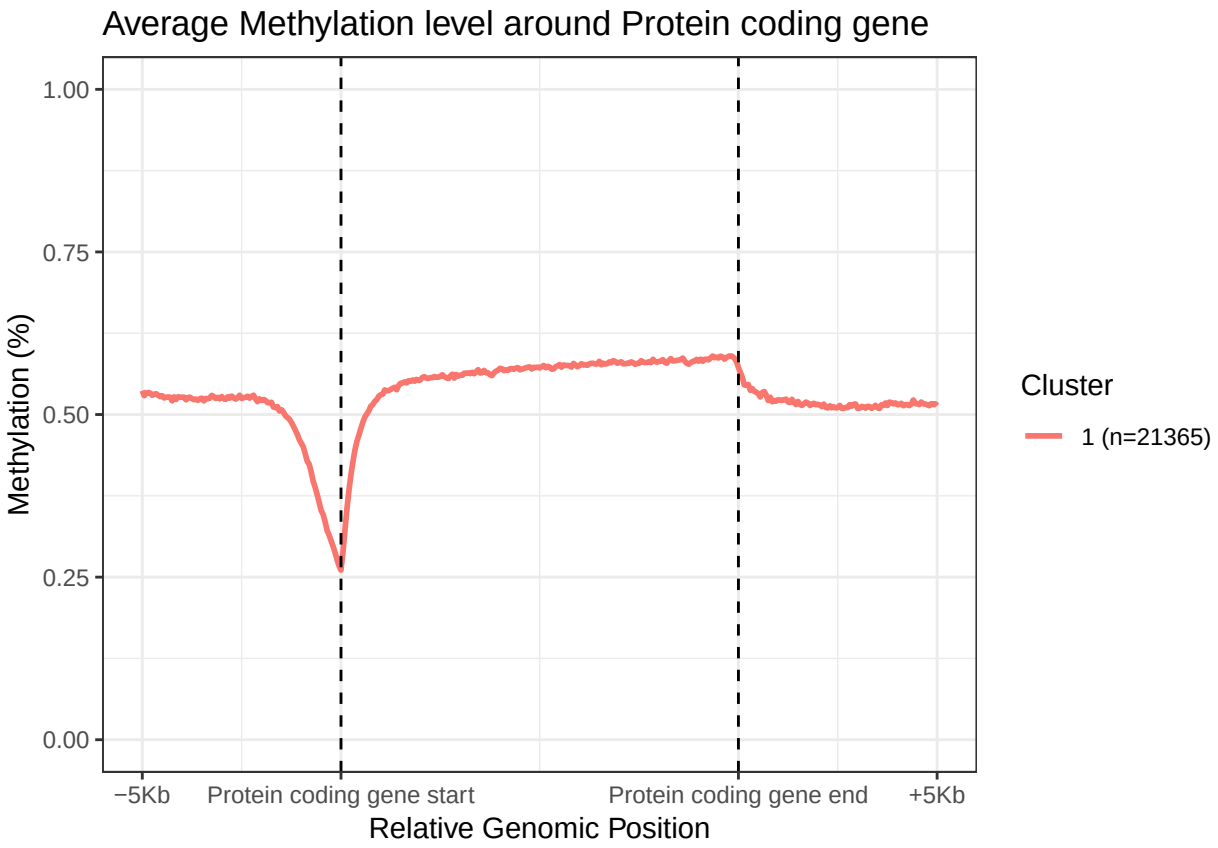



```
plot_single_meth(overlap_protein, 1000, feature_name="Protein coding gene")
```

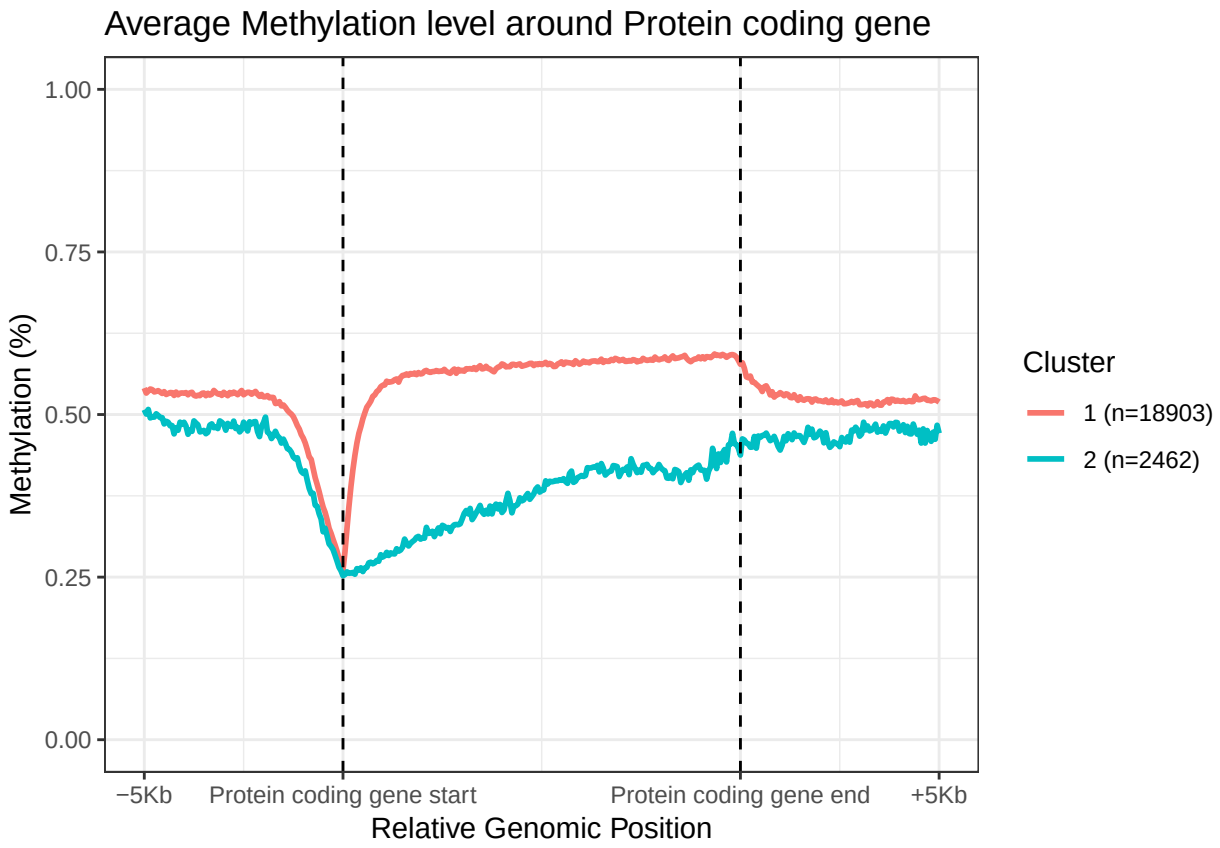


Plot average protein-coding gene methylation level

```
plot_meth(overlap_protein, 1, feature_name="Protein coding gene")
```

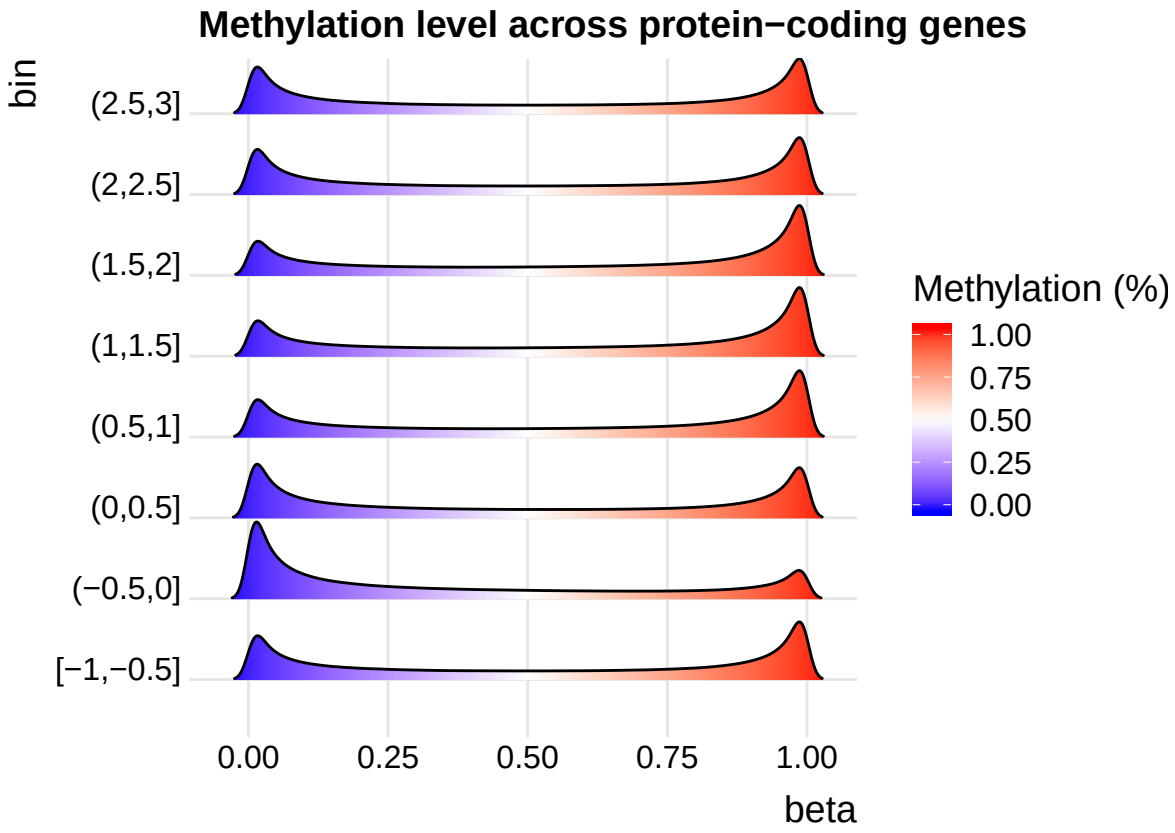



```
plot_meth(overlap_protein, 2, feature_name="Protein coding gene")
```




```
overlap_protein <- overlap_protein %>%
  mutate(bin=cut_interval(pos, length=0.5))

ggplot(overlap_protein, aes(x=beta, y=bin, fill=..x..)) +
  geom_density_ridges_gradient(scale=0.95, rel_min_height=0.01) +
  scale_fill_gradient2(name="Methylation (%)",
                       space="Lab", low="blue", mid="white", high="red",
                       midpoint=0.5) +
  labs(title="Methylation level across protein-coding genes") +
  theme_ridges()
```

```
## Picking joint bandwidth of 0.0119
```

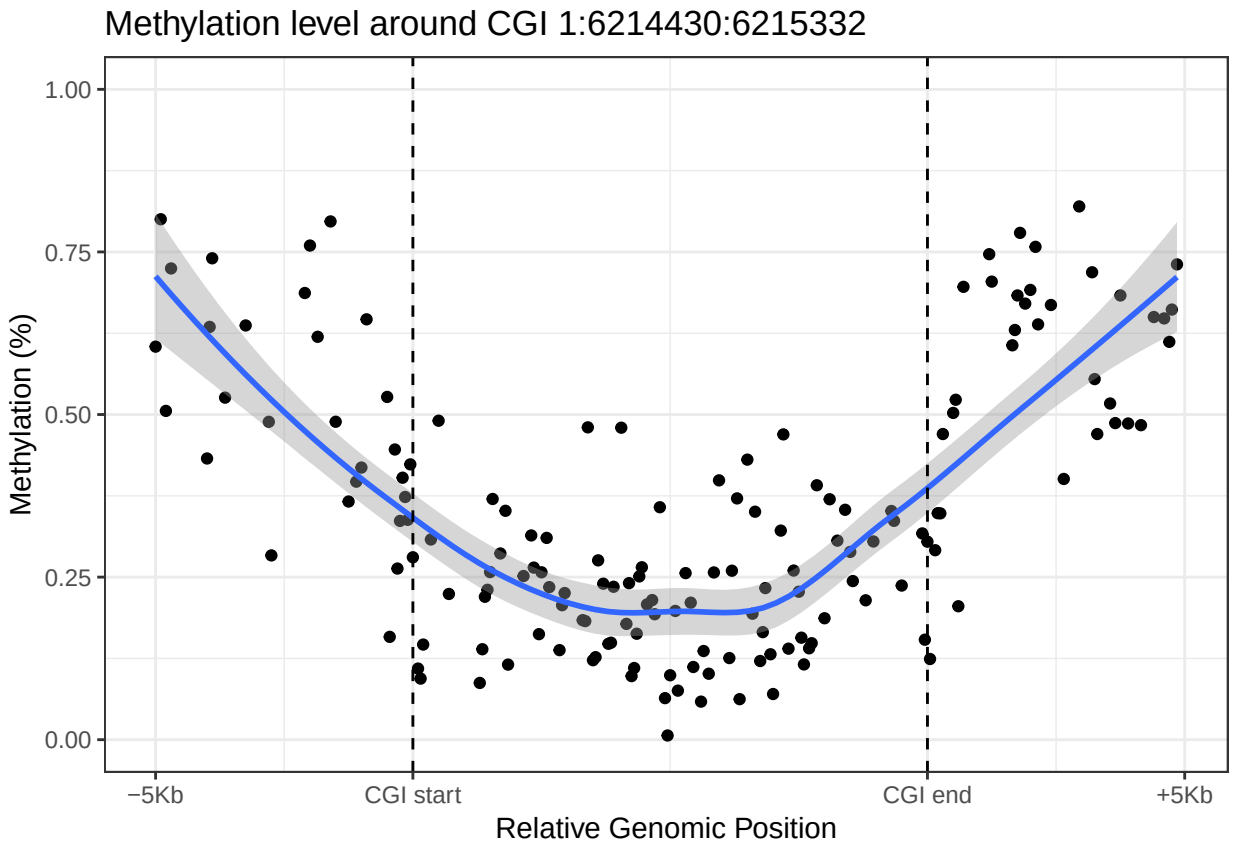


Find overlaps between methylation data and CGI

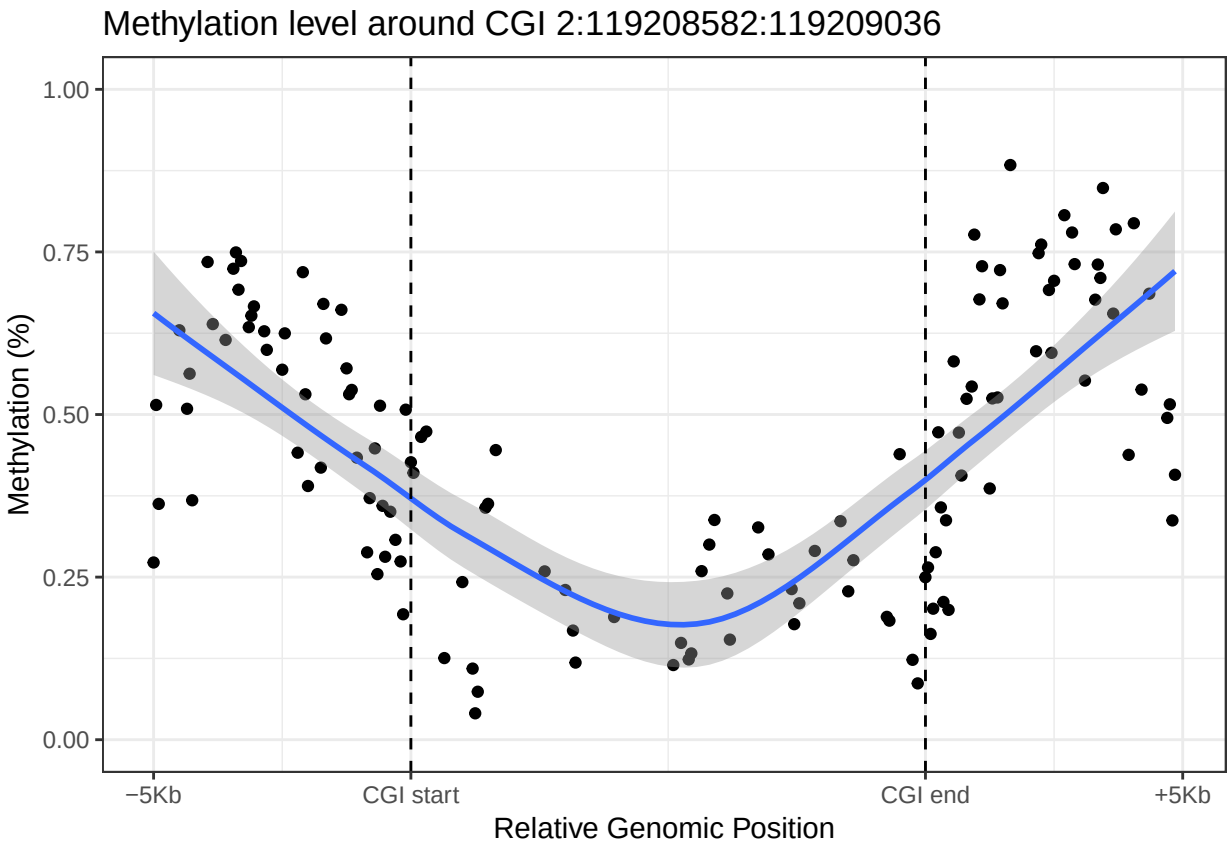
```
overlap_ggf <- find_overlaps(cgi_ggf, mouse_per_read_stats, overhang=overhang, feature_width=feature_wi
```

Plot a single methylation level

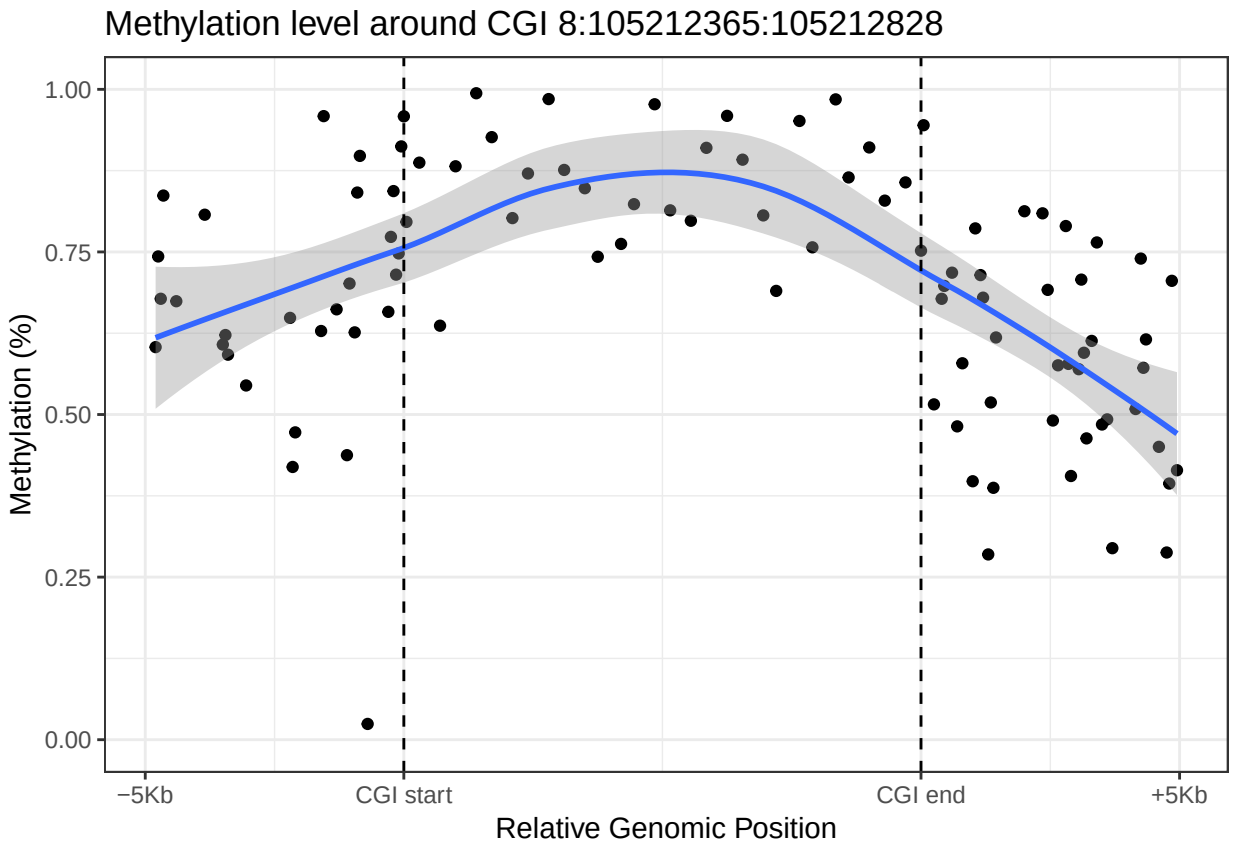
```
plot_single_meth(overlap_ggf, 20)
```



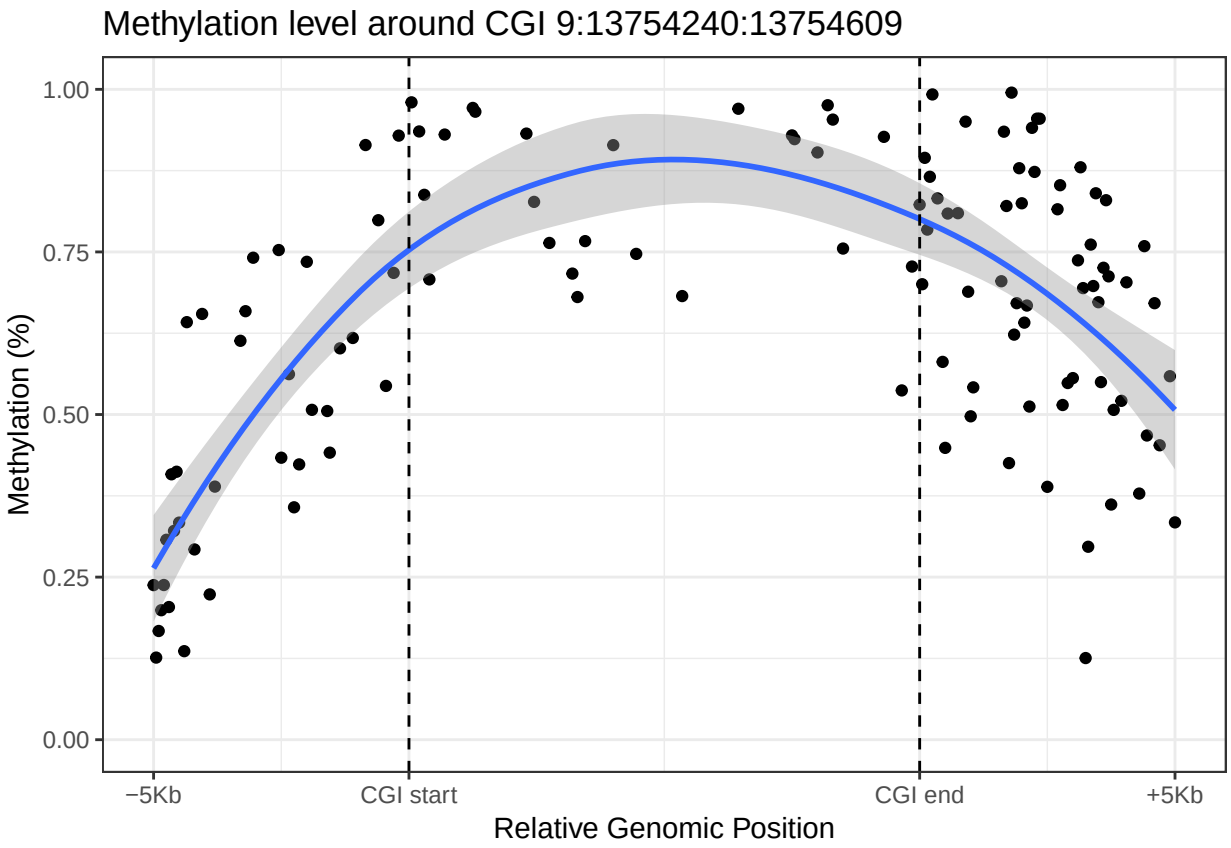

```
plot_single_meth(overlap_ggf, 2500)
```




```
plot_single_meth(overlap_ggf, 11199)
```

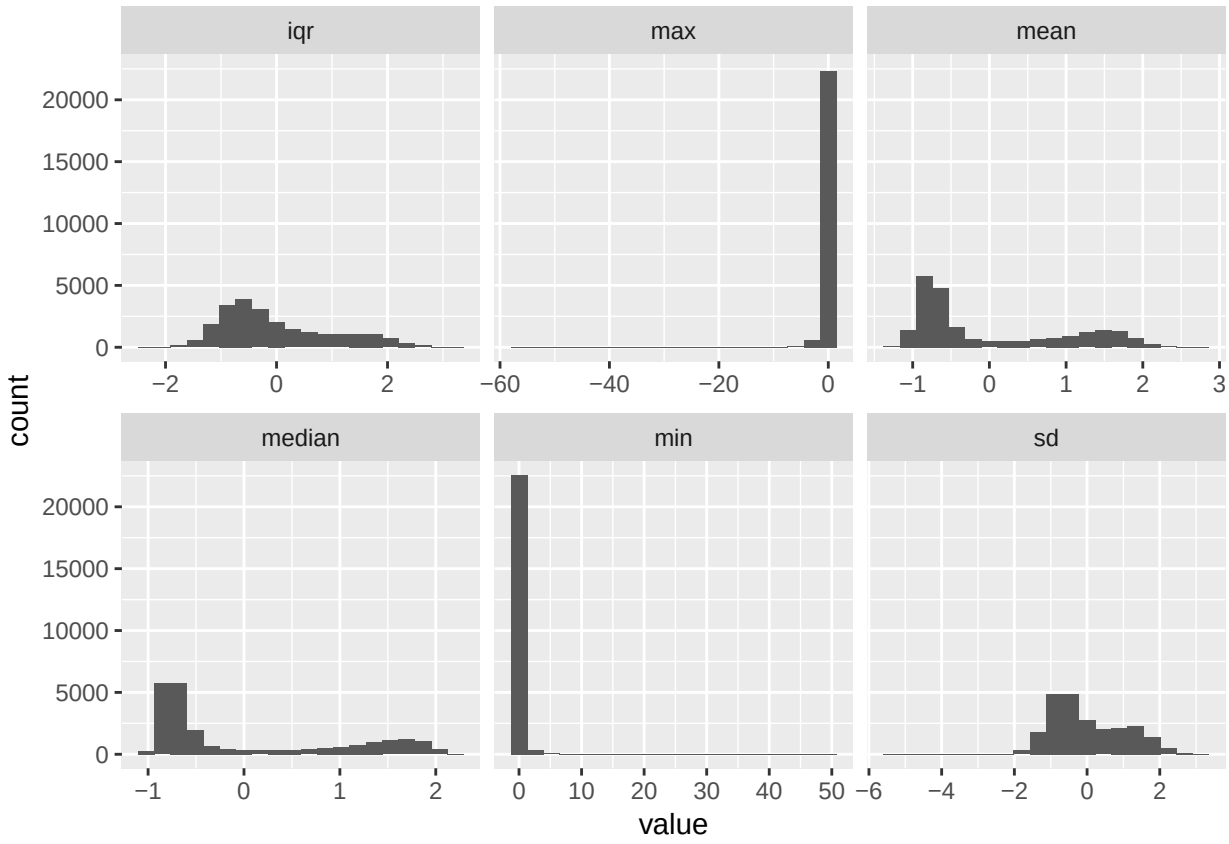



```
plot_single_meth(overlap_ggf, 11651)
```

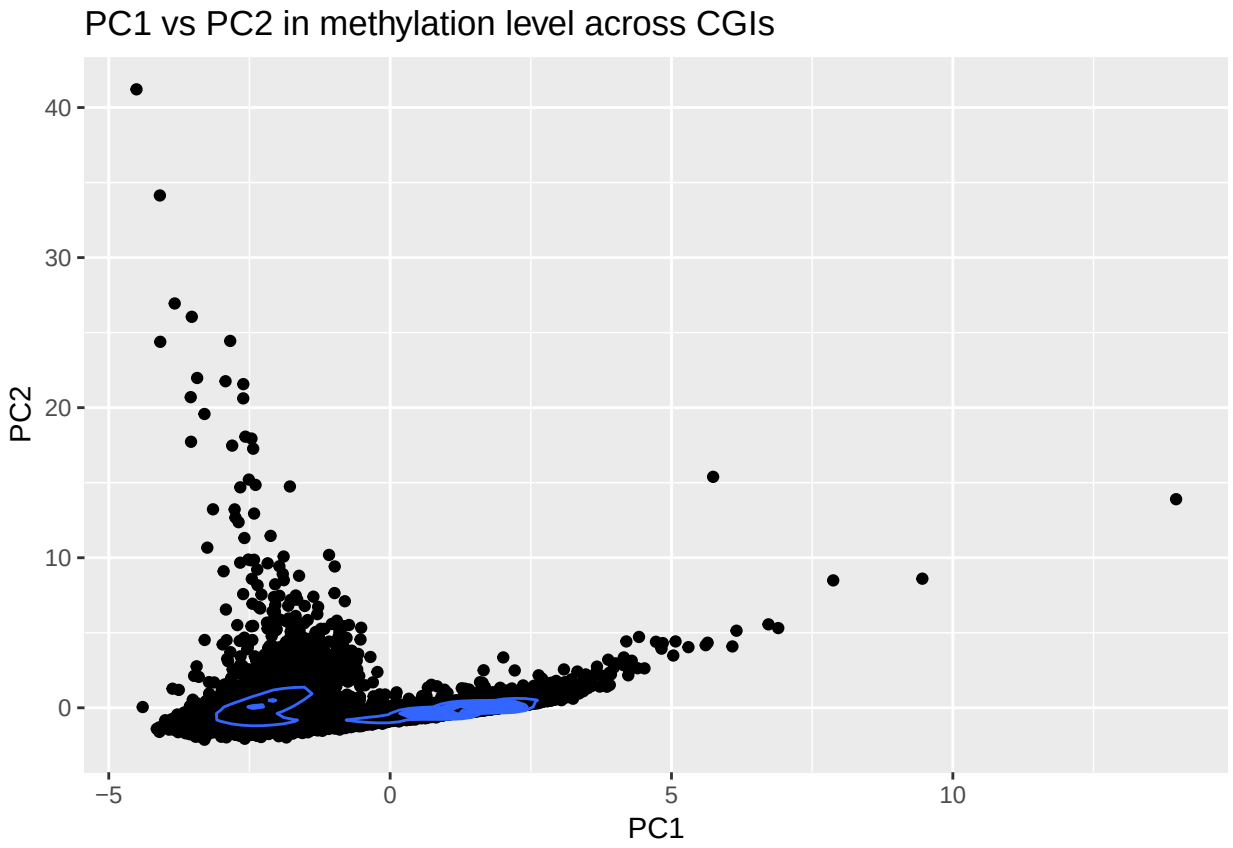


Check methylation summaries used for clustering

```
meth_stats <- overlap_ggf %>%
  group_by(id) %>%
  filter(pos >= 0, pos <= feature_width) %>%
  summarise(mean=mean(beta),
            median=median(beta),
            max=max(beta),
            min=min(beta),
            iqr=IQR(beta),
            sd=sd(beta)) %>%
  ungroup() %>%
  select(-id) %>%
  na.omit() %>%
  as.matrix() %>%
  scale()

ggplot(gather(data.frame(meth_stats)), aes(value)) +
  geom_histogram(bins=20) +
  facet_wrap(~key, scale="free_x")
```

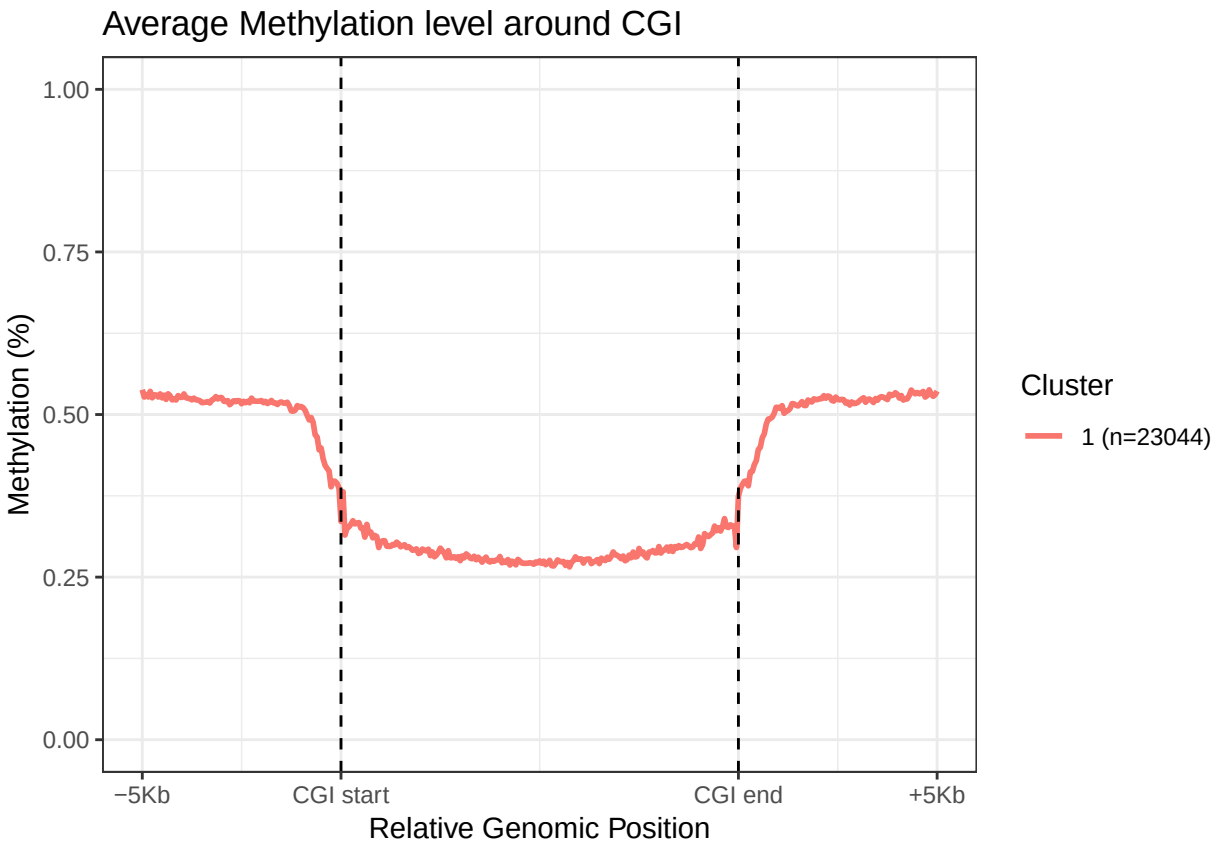



```
meth_pca <- prcomp(meth_stats, center=TRUE, scale=TRUE)

ggplot(data.frame(meth_pca$x), aes(x=PC1, y=PC2)) +
  geom_point() +
  geom_density_2d(binwidth=0.05) +
  labs(title="PC1 vs PC2 in methylation level across CGIs")
```

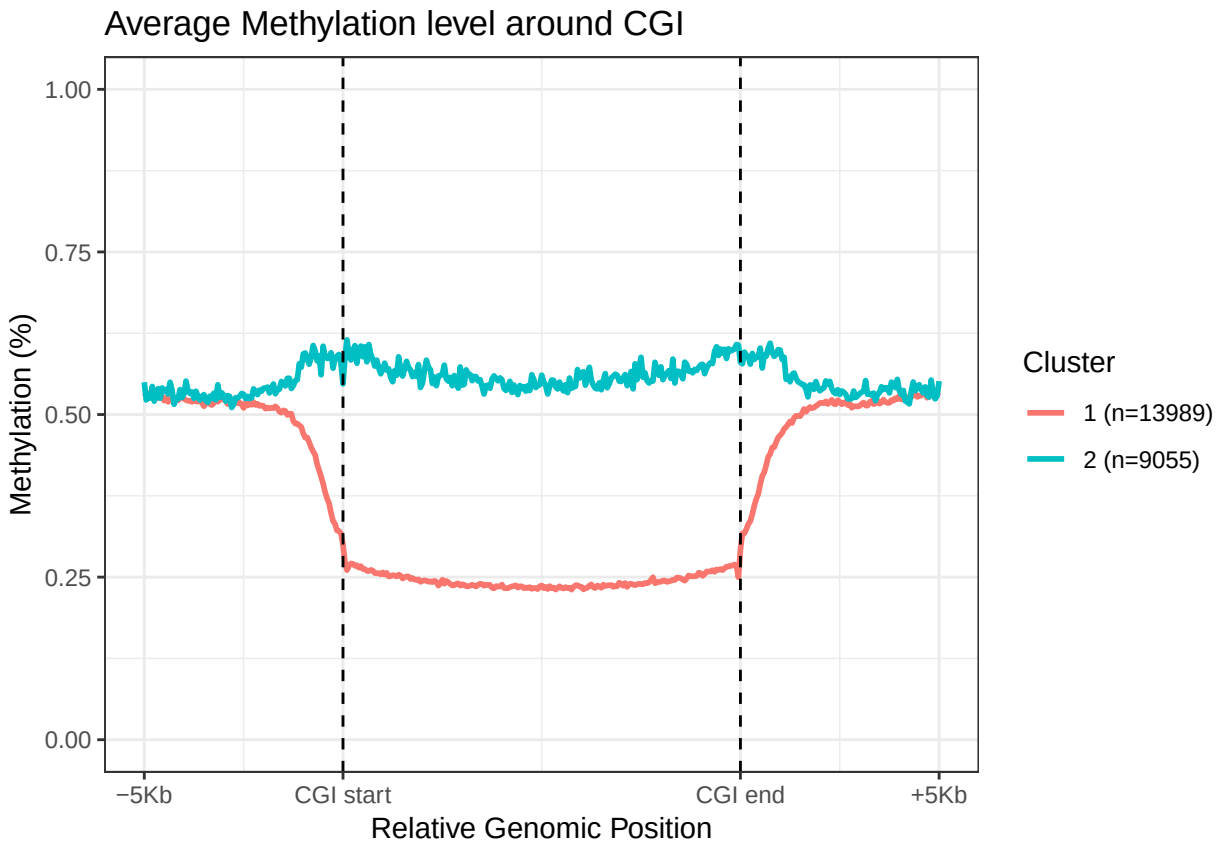


Plot average methylation level across all CGIs

```
plot_meth(overlap_ggf, 1)
```

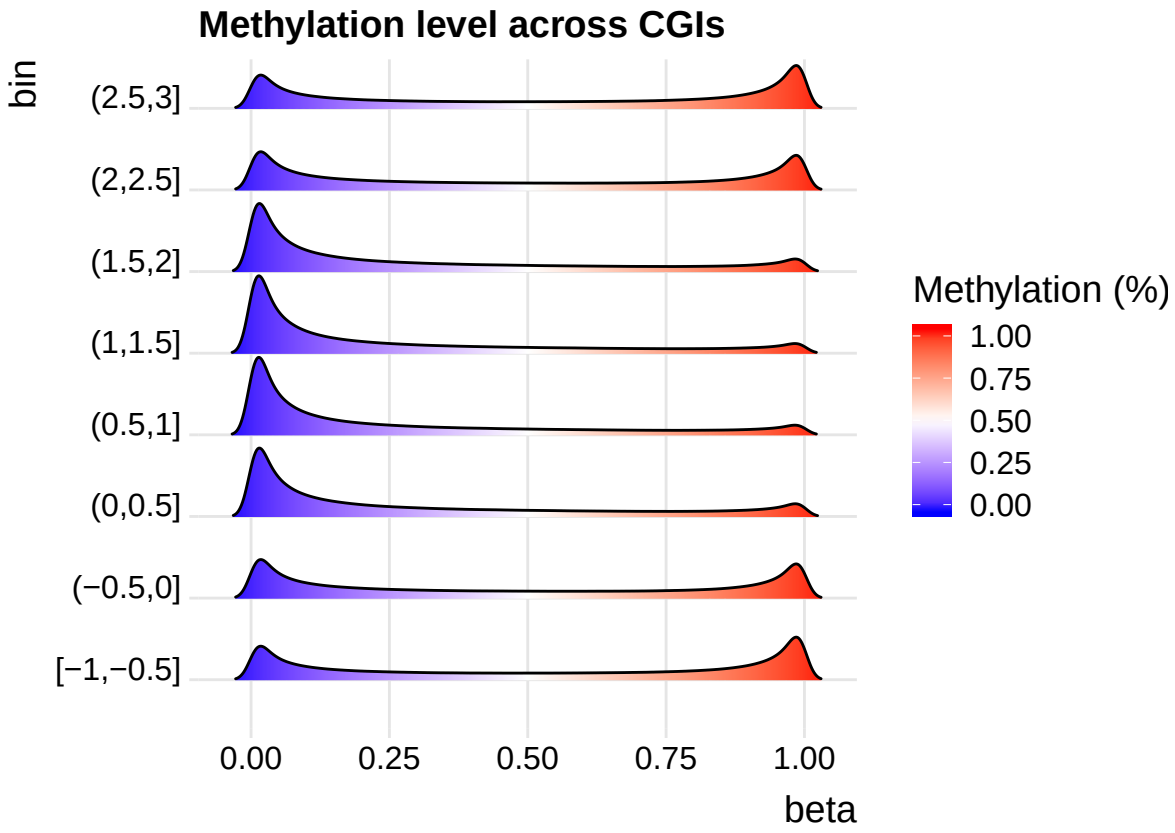



```
plot_meth(overlap_ggf, 2)
```




```
overlap_ggf <- overlap_ggf %>%
  mutate(bin=cut_interval(pos, length=0.5))

ggplot(overlap_ggf, aes(x=beta, y=bin, fill=..x..)) +
  geom_density_ridges_gradient(scale=0.95, rel_min_height=0.01) +
  scale_fill_gradient2(name="Methylation (%)",
                       space="Lab", low="blue", mid="white", high="red",
                       midpoint=0.5) +
  labs(title="Methylation level across CGIs") +
  theme_ridges()
```

```
## Picking joint bandwidth of 0.0136
```


Methylation level across CGIs
bin
(2.5,3]
(2,2.5]
(1.5,2]
(1,1.5]
(0.5,1]
(0,0.5]
(−0.5,0]
[−1,−0.5]
0.00
0.25
0.50
0.75
1.00
beta
Methylation (%)
1.00
0.75
0.50
0.25
0.00