Plotting metagenes with MetaPlotR

Summary

MetaPlotR is a Perl/R pipeline for creating metagene plots. A metagene is a density plot or histogram of sites of interest (e.g. protein binding sites or RNA modifications) along a simplified transcript model containing a 5'UTR, coding sequence and 3'UTR.

Requirements

1. Unix/Linux based operating system (tested with Debian 7.8 and OS X 10.10.5)
2. Perl (tested with version 5.22.2)
3. R (tested with version 3.2.2), and "scales" package
4. Bedtools (tested with version 2.22.1)

Prepare primary data

Create query bed file

A six-column bed file (i.e. BED6) is required (see here for specifications). This tutorial uses a bed file of N6-methyladenosine (m6A) sites generated from Linder et al. Nat. Methods, 2015 (miclip_cims.sorted.bed). The sample bed file is located in the Github repository along with the MetaPlotR scripts. This file was sorted using the Unix sort command:

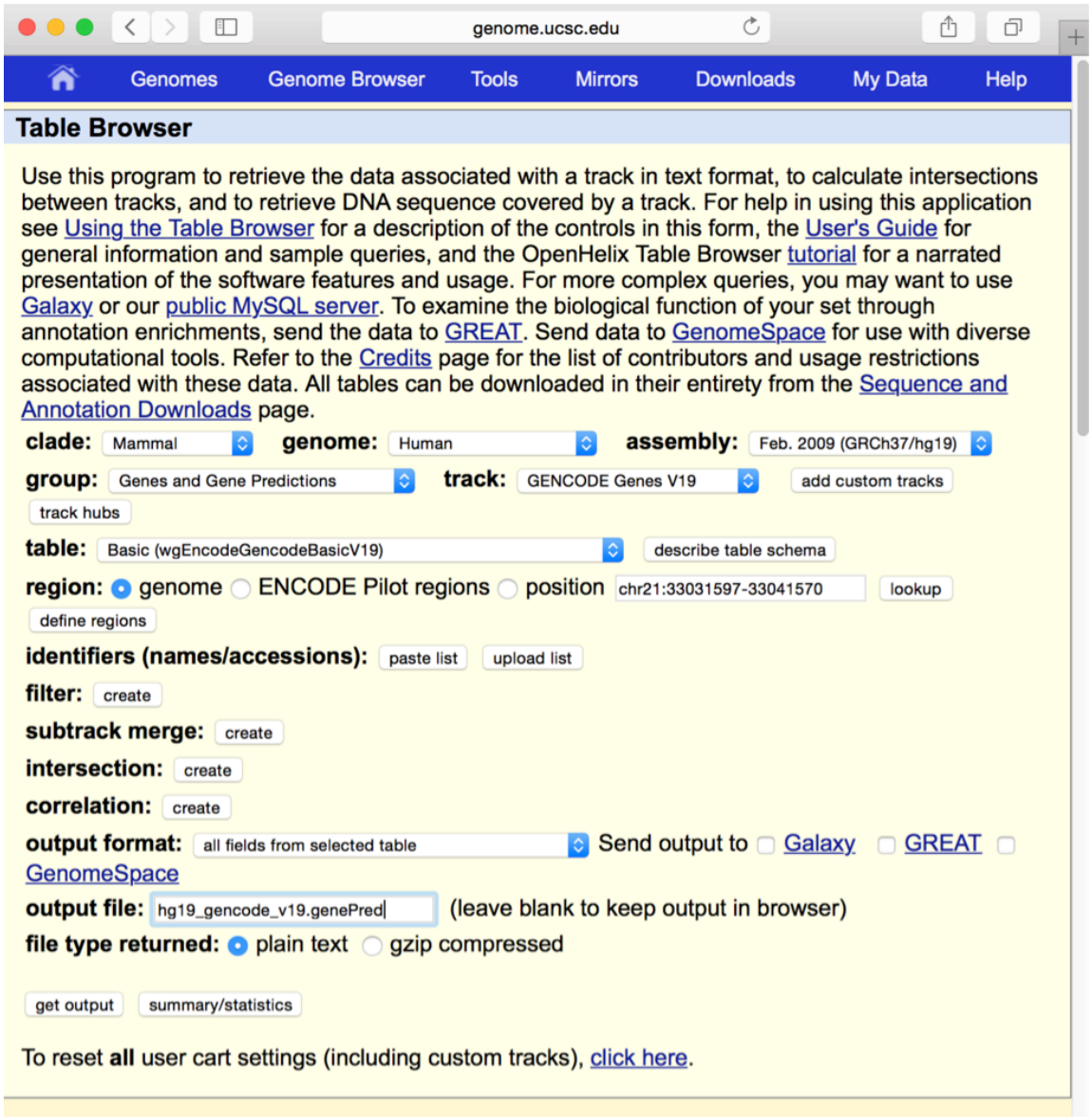
```
sort –k1,1 –k2,2n miclip_cims.bed > miclip_cims.sorted.bed
```

NOTE: MetaPlotR expects a bed file with 0-based single nucleotide coordinates.

Download genome and annotation file

Download genome of interest from the UCSC genome browser download page (http://hgdownload.soe.ucsc.edu/downloads.html). Here we use the hg19 human genome located here (http://hgdownload.soe.ucsc.edu/goldenPath/hg19/bigZips/chromFa.tar.gz).

Next, download the extended gene prediction tables from the UCSC Table Browser (http://genome.ucsc.edu/cgi-bin/hgTables). The figure below shows the necessary drop-down options to download the gencode gene annotations for the hg19 human genome.



Pre-process data

1. **make_annot_bed.pl** creates a master annotation file (bed format) of every nucleotide in the transcriptome. The script is supplied with the locations of the genome directory (chroms/) and the gene prediction table (hg19_gencode.genePred): `perl make_annot_bed.pl --genomeDir chroms/ --genePred hg19_gencode.genePred > hg19_annot.bed`

- Sort the master annotation file using the unix sort command:
  `sort -k1,1 -k2,2n hg19_annot.bed > hg19_annot.sorted.bed`

2. **size_of_cds_utrs.pl** creates a file cataloging the transcriptomic coordinates of the start and end sites of the transcript regions (i.e. 5'UTR, CDS and 3'UTR). It takes the sorted master annotation file as input (*hg19_annot.sorted.bed*) and outputs a region annotation file. The region annotation file is necessary for determining the distance of queried sites from the transcriptomic features (i.e.

transcriptional start site, start codon, stop codon and transcript end).
`perl size_of_cds_utrs.pl --annot hg19_annot.sorted.bed > region_sizes.txt`

3. **annotate_bed_file.pl** annotates the user supplied bed file (*miclip_cims.sorted.bed*) containing single nucleotide genomic coordinates of sites of interest. It serves as a wrapper for Bedtools Intersect and essentially labels every line in the user supplied bed file with the matching line (i.e. same coordinates) in the master annotation file (*hg19_annot.sorted.bed*). The outputted file is called the annotated query file.
`perl annotate_bed_file.pl --bed miclip_cims.sorted.bed --bed2 hg19_annot.sorted.bed > annot_miclip.cims.bed`

- Alternatively, Bedtools intersect can be evoked directly using the command: `intersectBed -a miclip_cims.sorted.bed -b hg19_annot.sorted.bed -sorted -wo –s > annot_miclip.cims.bed`

4. **rel_and_abs_dist_calc.pl** identifies the region of the transcript in which the user supplied sites fall and converts the transcriptomic coordinates to metagene coordinates. Namely, sites that occur in the 5'UTR have a value from 0 to 1, where 0 and 1 represent the 5' and 3' ends of the 5'UTR, respectively. Similarly, sites in the CDS have a value from 1 to 2 and the 3'UTR 2 to 3. The script takes as input the annotated query file *annot_miclip.cims.bed* and the region annotation file *utr_cds_ends.txt*. The outputted distance measure file contains all the values needed to plot the metagenes. `perl rel_and_abs_dist_calc_v2.pl --bed annot_miclip.cims.bed --regions utr_cds_ends.txt > dist.measures.txt`

Understanding the distance measure file

All proceeding code are in R (https://www.r-project.org/). We recommend working with R using RStudio (https://www.rstudio.com/).
The input for this section is the metagene coordinates file outputted from **rel_and_abs_dist_calc.pl**

Read in file

```
dist <- read.delim ("dist.measures.txt", header = T)
```

View the number of rows and columns in the dataset

```
dim(dist)

## [1] 20903    14
```

View the first few lines

```
head(dist)

##    chr  coord gene_name          refseqID rel_location utr5_st
utr5_end
## 1 chr1 878151    SAMD11 ENST00000342066.3     1.624145    1359
```

```
1277
## 2 chr1 879955     NOC2L ENST00000327044.6     2.242857    2418
2369
## 3 chr1 934375      HES4 ENST00000304952.6     2.673684     867
730
## 4 chr1 934375      HES4 ENST00000428771.2     2.659794    1006
808
## 5 chr1 934375      HES4 ENST00000484667.2     2.719101     641
634
## 6 chr1 934423      HES4 ENST00000304952.6     2.168421     819
682
##   cds_st cds_end utr3_st utr3_end utr5_size cds_size utr3_size
## 1   1276    -769    -770    -1191        82     2045       421
## 2   2368     119     118     -371        49     2249       489
## 3    729      64      63      -31       137      665        94
## 4    807      64      63      -33       198      743        96
## 5    633      64      63      -25         7      569        88
## 6    681      16      15      -79       137      665        94
```

This input file contains 20903 rows and 14 columns. Each row represents a single site (in this example an m6A site). The column headers for the first four columns are self explanatory. The fifth column "rel_location" (for relative location) contains the calculated metagene coordinates. In its simplest form (i.e. non-normalized), the metagene coordinates from 0 to 1 represent the 5'UTR with 0 being closer to the beginnig of the 5'UTR and 1 closer to the end. Similarly, 1 to 2 represents the CDS and 2 to 3 the 3'UTR. A histogram/density plot of the "rel_location" value gives the standard metagene.

In addition to the standard metagene which is based on the relative location of sites in transcripts, this next six columns (utr5_st, utr5_end, cds_st, cds_end, utr3_st, utr3_end) contain information for plotting the absolute distance of sites from several points of interest. For example, in this dataset the third row has a value of +63 under column header "utr3_st". That means the site is 63 nucleotides upstream of the 3'UTR start site.

The last three columns contain the lengths of the 5'UTRs, coding sequences and 3'UTRs.

Selecting gene isoforms for metagene analysis

The dataset is redundant -- a given site is represented by multiple transcript isoforms. The choice of which isoforms to choose should be informed by the underlying biology. For example, if a gene expression dataset is available, one option may be to pick the highest expressed isoform. Another option is to pick the longest isoform, which is likely to capture more sites. Below is sample code for picking the largest isoforms

```
trx_len <- dist$utr5_size + dist$cds_size + dist$utr3_size
dist <- dist[order(dist$gene_name, trx_len),] # sort by gene name, then
```

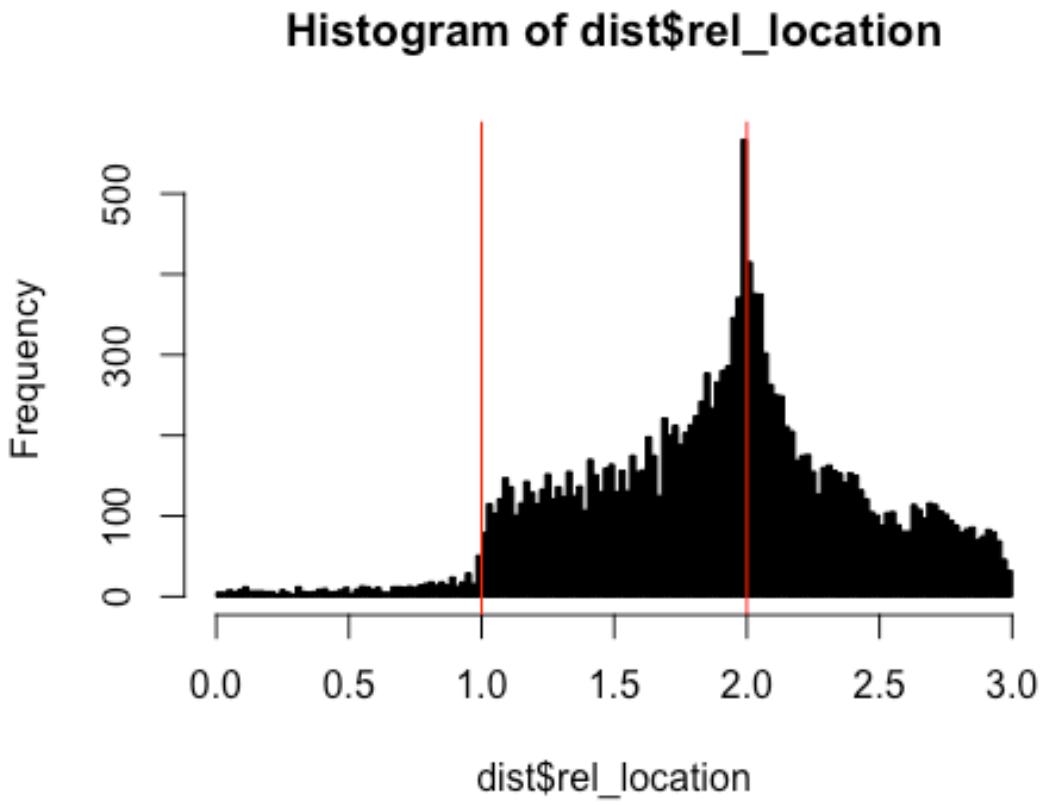
```
transcript length
dist <- dist[duplicated(dist$gene_name),] # select the longest isoform
#dist <- dist[duplicated(dist$gene_name),] # select the shortest
isoform
dim(dist)

## [1] 16721    14
```

Visualizing the metagene

A simple histogram

```
hist(dist$rel_location, breaks = 200, col = "black")
abline (v = 1, lty  = 1, col = "red")
abline (v = 2, lty  = 1, col = "red")
```



In this plot, the range 0 to 1 represents the 5'UTR, 1 to 2 the CDS, and 2 to 3 the 3'UTR (as delineated by the red vertical lines). From this figure, one may conclude that the events (in this case m6A sites) occur througout the gene body with a peak around the stop codon and a precipitous transition from the 5'UTR to the CDS. However, one caveat is that the three regions of interest are drawn with equal widths. On average, this is not the case. We can view the average lenghts in this dataset:

```
summary(data.frame(dist$utr5_size, dist$cds_size, dist$utr3_size))
```

```
##  dist.utr5_size    dist.cds_size   dist.utr3_size
##  Min.   :    0.0   Min.   :   38   Min.   :    0
##  1st Qu.:  103.0   1st Qu.:  824   1st Qu.:  332
##  Median :  206.0   Median : 1442   Median :  768
##  Mean   :  289.8   Mean   : 1924   Mean   : 1205
##  3rd Qu.:  362.0   3rd Qu.: 2402   3rd Qu.: 1586
##  Max.   :14959.0   Max.   :26393   Max.   :17320
```

The median lengths are 206, 1442, and 768 for the 5'UTR, CDS and 3'UTR, respectively.

To account for these length differences in the metagene, we can re-scale the widths of the 5'UTR and 3'UTR relative to the CDS (which is set constant to a width of 1 unit). So first we calculate a simple scale factor (SF):

```
utr5.SF <- median(dist$utr5_size, na.rm = T)/median(dist$cds_size,
na.rm = T)
utr3.SF <- median(dist$utr3_size, na.rm = T)/median(dist$cds_size,
na.rm = T)
```

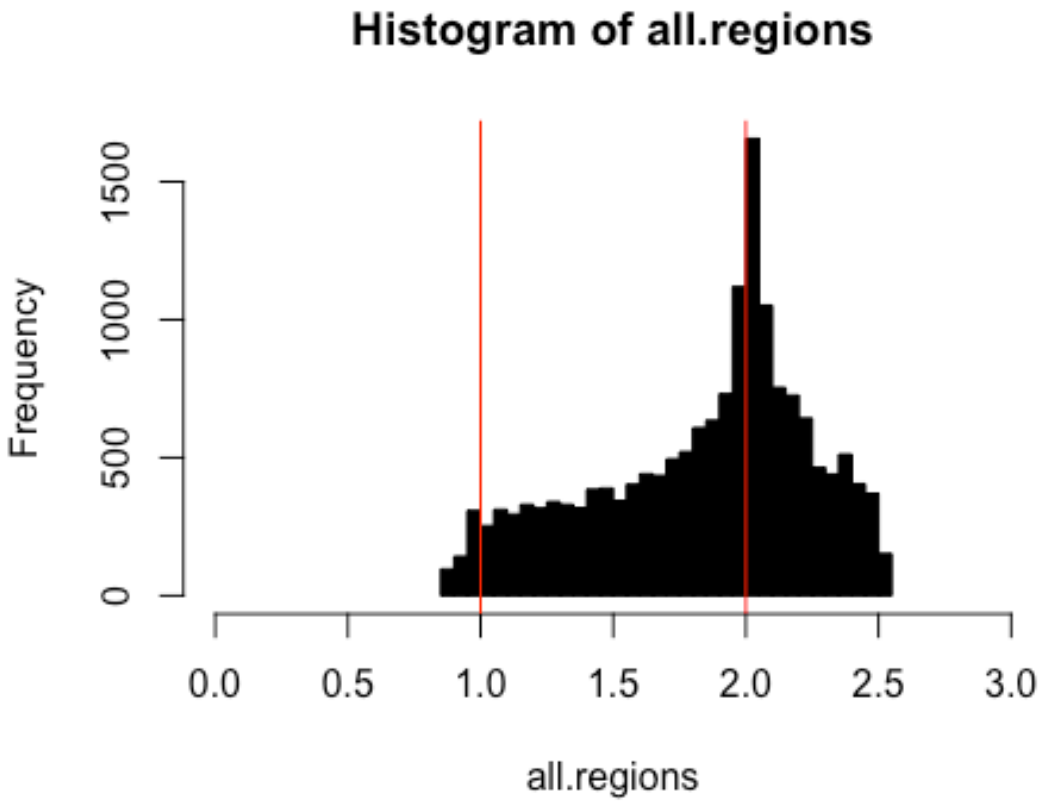
The SF for the 5'UTR is 0.14 and for the 3'UTR is 0.53. The followign code rescales these regions accordingly:

```
# assign the regions to new dataframes
utr5.dist <- dist[dist$rel_location < 1, ]
cds.dist <- dist [dist$rel_location < 2 & dist$rel_location >= 1, ]
utr3.dist <- dist[dist$rel_location >= 2, ]

# rescale 5'UTR and 3'UTR
library("scales")
utr5.dist$rel_location <- rescale(utr5.dist$rel_location, to = c(1-
utr5.SF, 1), from = c(0,1))
utr3.dist$rel_location <- rescale(utr3.dist$rel_location, to = c(2,
2+utr3.SF), from = c(2,3))
```

Finally, plot the metagene with the rescaled UTRs

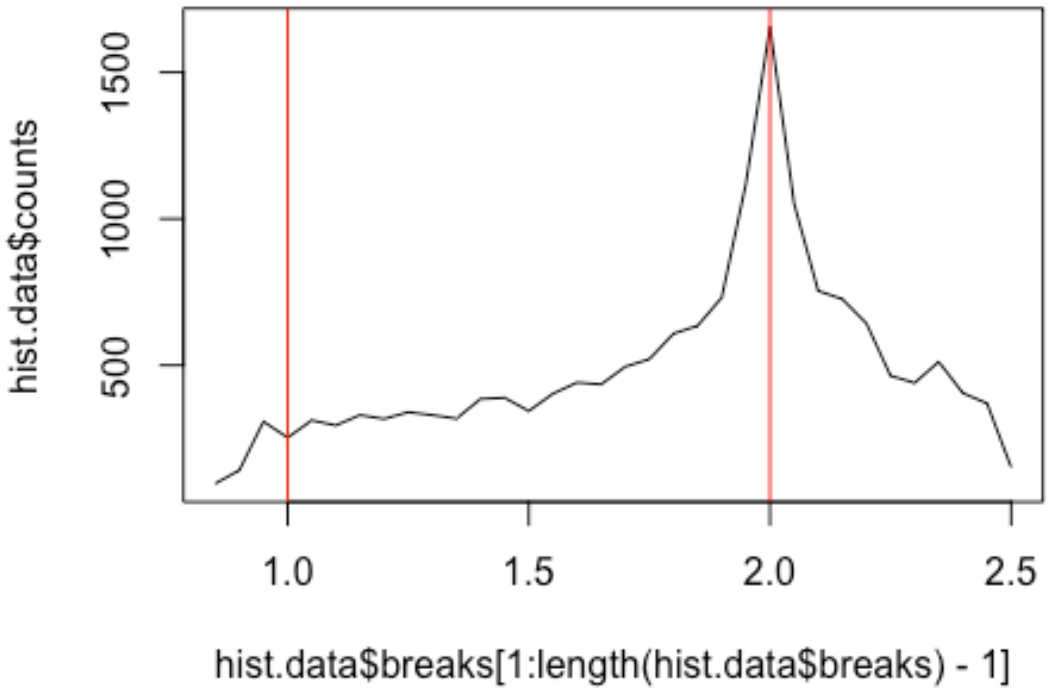
```
# Combine and plot
## Histogram
all.regions <- c(utr5.dist$rel_location, cds.dist$rel_location,
utr3.dist$rel_location)
hist.data <- hist(all.regions, breaks = 50, col = "black", xlim =
c(0,3)) # plot and save to variable
abline (v = 1, lty  = 1, col = "red")
abline (v = 2, lty  = 1, col = "red")
```



Alternate representations of the metagene

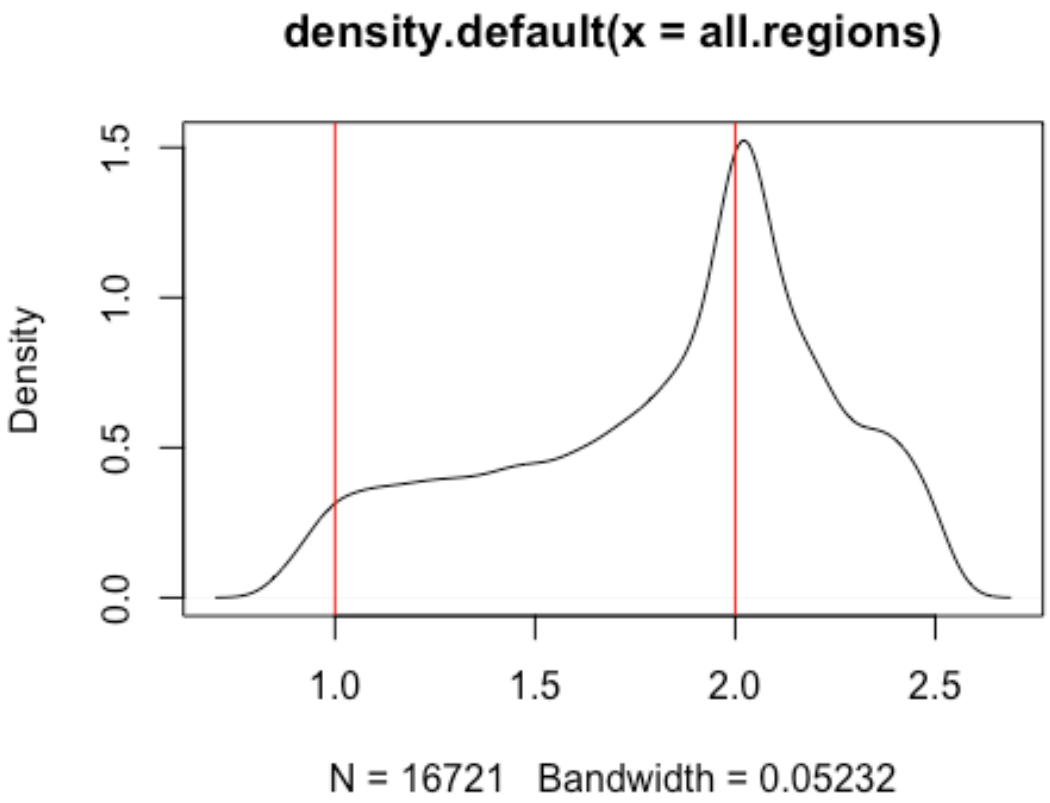
A line plot

```
plot(hist.data$breaks[1:length(hist.data$breaks)-1], hist.data$counts,
type = 'l')
abline (v = 1, lty  = 1, col = "red")
abline (v = 2, lty  = 1, col = "red")
```



A smooth density plot

```
plot(density(all.regions))
abline (v = 1, lty  = 1, col = "red")
abline (v = 2, lty  = 1, col = "red")
```



Mapping the absolute distance of sites from fixed features

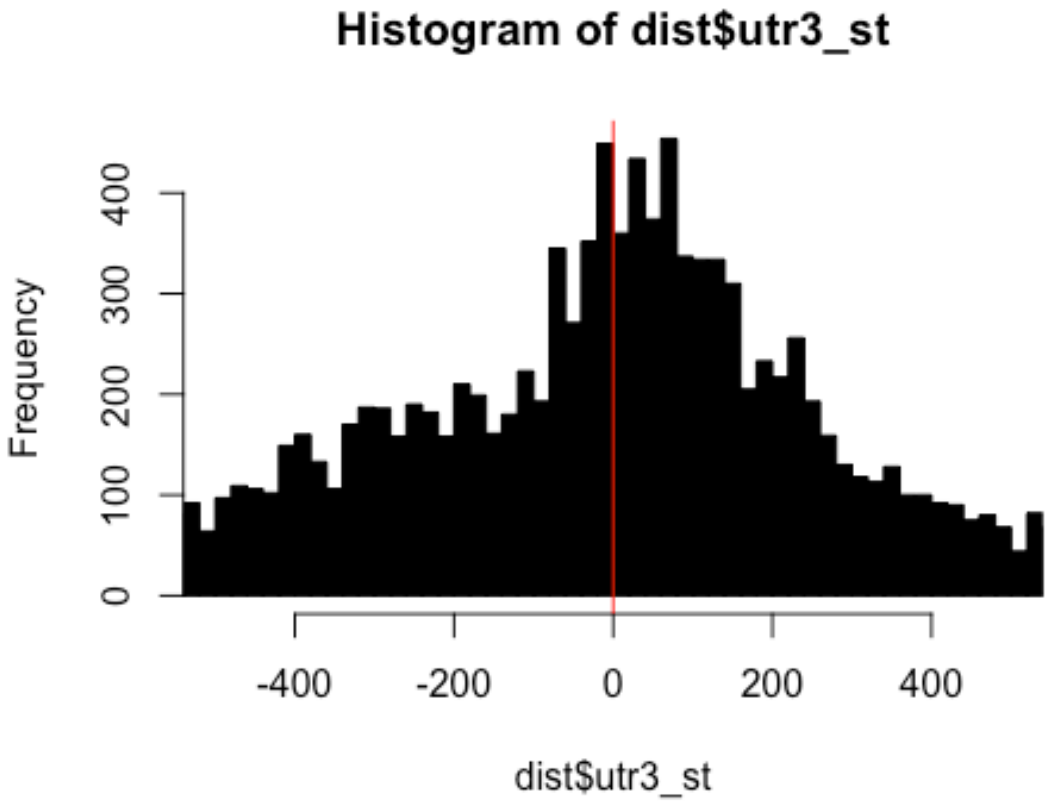
An alternative to the metagene plot is a feature distance plot which shows the absolute distance of sites from a given transcriptomic feature (e.g. stop codon, transcription start site, etc). As discussed earlier columns 6-11 of the dataset contains the absolute distance data.

```
head(dist[,6:11])
```

```
##       utr5_st utr5_end cds_st cds_end utr3_st utr3_end
## 4713     1057      927    926    -615    -616     -645
## 4712     1191     1026   1025    -615    -616     -648
## 13159     448      356    355    -841    -842    -1017
## 13154     190      104    103   -1051   -1052    -2239
## 13157     442      356    355    -799    -800    -1987
## 13160    2270     2184   2183    1029    1028     -159
```

For example, we can view the distribution of sites within 500 nucleotides of the stop codon:

```
hist(dist$utr3_st, xlim = c(-500,500), breaks = 1000, col = "black")
abline(v=0, col = "red")
```



Final remarks

R is a powerful language and there are many customizations that can be made to all the plots shown above. This tutorial was meant to serve as a starting point for creating metagenes and exploring the underlying data using **MetaPlotR**.