

# ChromENVEE: Chromatin ENVironment and Enhancer Expression

Coulée Manon

2022-09-29

## Contents

<b>Abstract</b>	<b>1</b>
<b>Citation</b>	<b>1</b>
<b>Introduction</b>	<b>1</b>
<b>Initialization of data</b>	<b>2</b>
<b>Distribution of chromatin state in the genome</b>	<b>2</b>
<b>Annotation of enhancer</b>	<b>3</b>
Annotated enhancer binding to enhancer position . . . . .	4
Associated gene expression to enhancer . . . . .	5
Profile of enhancer annotation . . . . .	6
<b>Enhancer annotation comparison</b>	<b>8</b>
<b>Gene environment</b>	<b>10</b>
Coverage of chromatin state in environment binding to TSS regions . . . . .	11
Predominant state in environment binding to TSS regions . . . . .	11
<b>Session Information</b>	<b>13</b>
<b>References</b>	<b>14</b>

## Abstract

## Citation

## Introduction

```
# Loading package
library(ChromENVEE)
```

## Initialization of data

Created initial vector contains informations about chromatin state number `stateNumber`, chromatin state name `stateName` and chromatin state color `colorValue` (used for plot generation).

```
stateNumber = c("U1", "U2", "U3", "U4", "U5", "U6", "U7", "U8", "U9", "U10", "U11", "U12", "U13", "U14", "U15", "U16",
stateName = c("TSSA", "TSSFLnk", "TSSFLnkD", "Tx", "TxWk", "EnhG", "EnhG", "EnhA", "EnhWk", "ZNF/Rpts", "Het", "Ts
colorValue = c("#B71C1C", "#E65100", "#E65100", "#43A047", "#1B5E20", "#99FF66", "#99FF66", "#F5B041", "#FFEB3B
```

Download the reference genome as a `.bed` file, in this study we used mouse genome (mm10).

```
data(genomeFile)
```

```
#>     chr      start      end strand score          gene_ENS
#> 1  chr1    3073253  3074322      +   . ENSMUSG00000102693.1
#> 2  chr1    3102016  3102125      +   . ENSMUSG00000064842.1
#> 3  chr1    3205901  3671498      -   . ENSMUSG00000051951.5
#> 4  chr1    3252757  3253236      +   . ENSMUSG00000102851.1
#> 5  chr1    3365731  3368549      -   . ENSMUSG00000103377.1
#> 6  chr1    3375556  3377788      -   . ENSMUSG00000104017.1
```

Download the chromatin state file (ex. output of ChromHMM). It's a `.bed` file which contains informations like genomic position and information about chromatin state.

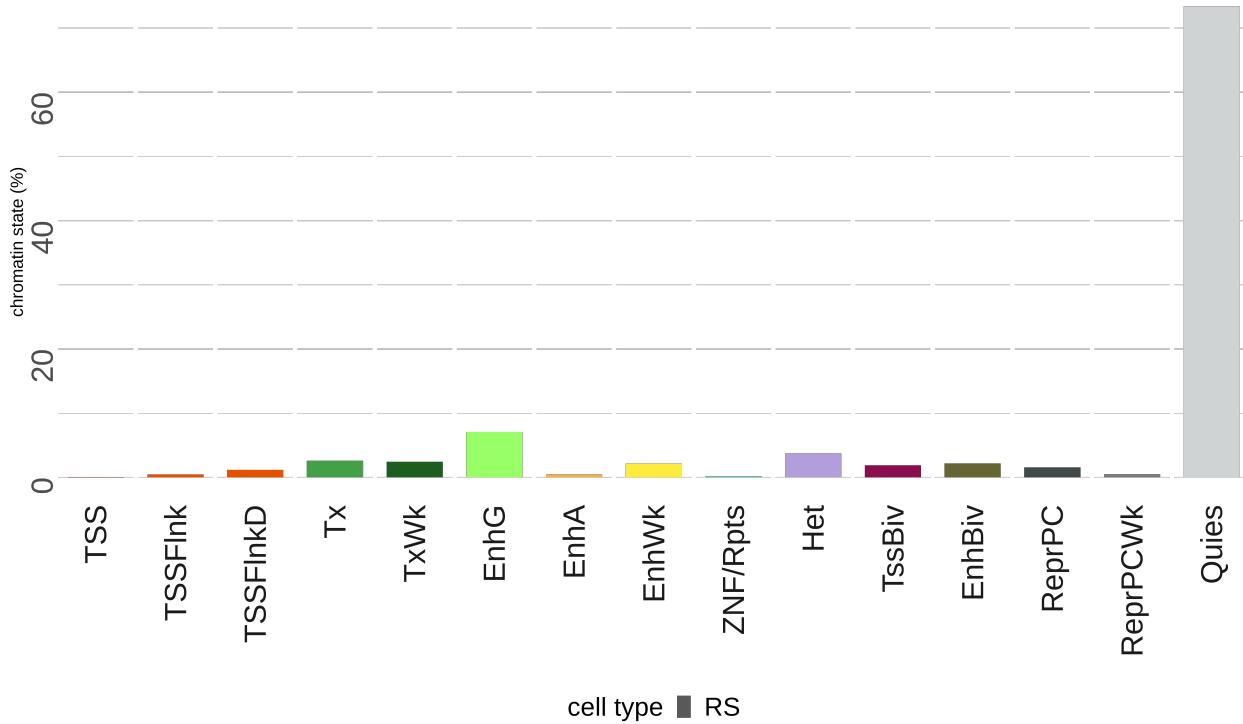
```
data(chromatinState)
```

```
#>     chr      start      end state name state_name
#> 1  chr10      0  3100000    U16  RS     Quies
#> 2  chr10  3100000  3109200    U11  RS     Het
#> 3  chr10  3109200  3110600    U12  RS     TssBiv
#> 4  chr10  3110600  3111000    U14  RS     ReprPC
#> 5  chr10  3111000  3111200    U13  RS     EnhBiv
#> 6  chr10  3111200  3117200    U12  RS     TssBiv
```

## Distribution of chromatin state in the genome

We are interested to know the distribution of chromatin state in the genome. `plotChromatinState` calculates the percentage of each chromatin state in function the length of the genome used. We obtains a data frame with the percentage of coverage for each chromatin state and it's possible to plot the result in `.png` file with the argument `plot = TRUE`.

```
summary_chromatin_state = plotChromatinState(chromatinState, stateName = stateName, stateNumber = stateNumb
head(summary_chromatin_state)
#>           state coverage sample_name
#> TSSA        TSSA 0.08519426      RS
#> TSSFLnk    TSSFLnk 0.45530134      RS
#> TSSFLnkD  TSSFLnkD 1.18900667      RS
#> Tx          Tx 2.60257103      RS
#> TxWk        TxWk 2.44911129      RS
#> EnhG        EnhG 7.10081351      RS
```



## Annotation of enhancer

We are interested to associate at each enhancer, genes regulated by the enhancer. We focused on enhancer chromatin state (in this study, we have 4 type of enhancer : bivalent enhancer (EnhBiv), genic enhancer (EnhG), active enhancer (EnhA) and weak enhancer (EnhWk)). The function uses a `GRanges` object (produced by `GenomicRanges` package).

```
data(listTableEnhancer)

#> GRanges object with 1979 ranges and 1 metadata column:
#>           seqnames      ranges strand | chromatin_state
#>           <Rle>      <IRanges> <Rle> |      <character>
#> [1]     chr10    9164400-9164800   * |          U13
#> [2]     chr10    9342200-9344000   * |          U13
#> [3]     chr10   10476400-10476600   * |          U13
#> [4]     chr10   20520200-20521000   * |          U13
#> [5]     chr10   20952400-20952600   * |          U13
#> ...
#> [1975]   chrX  144286800-144287000   * |          U13
#> [1976]   chrX  155128400-155129200   * |          U13
#> [1977]   chrX  170010800-170013800   * |          U13
#> [1978]   chrY    198400-198800   * |          U13
#> [1979]   chrY  90786000-90788000   * |          U13
#> -----
#> seqinfo: 21 sequences from an unspecified genome; no seqlengths
```

## Annotated enhancer binding to enhancer position

To estimated which gene is regulated by enhancer, we estimated that genes associated enhancer, all TSS genes in interval around enhancer. `enhancerAnnotation()` take few minutes to process in function the length of your enhancer table. It's possible to multithread the job with the `nCore` parameter. For each enhancer position, we obtains two informations, the distance between gene and enhancer (in bp) and the gene.

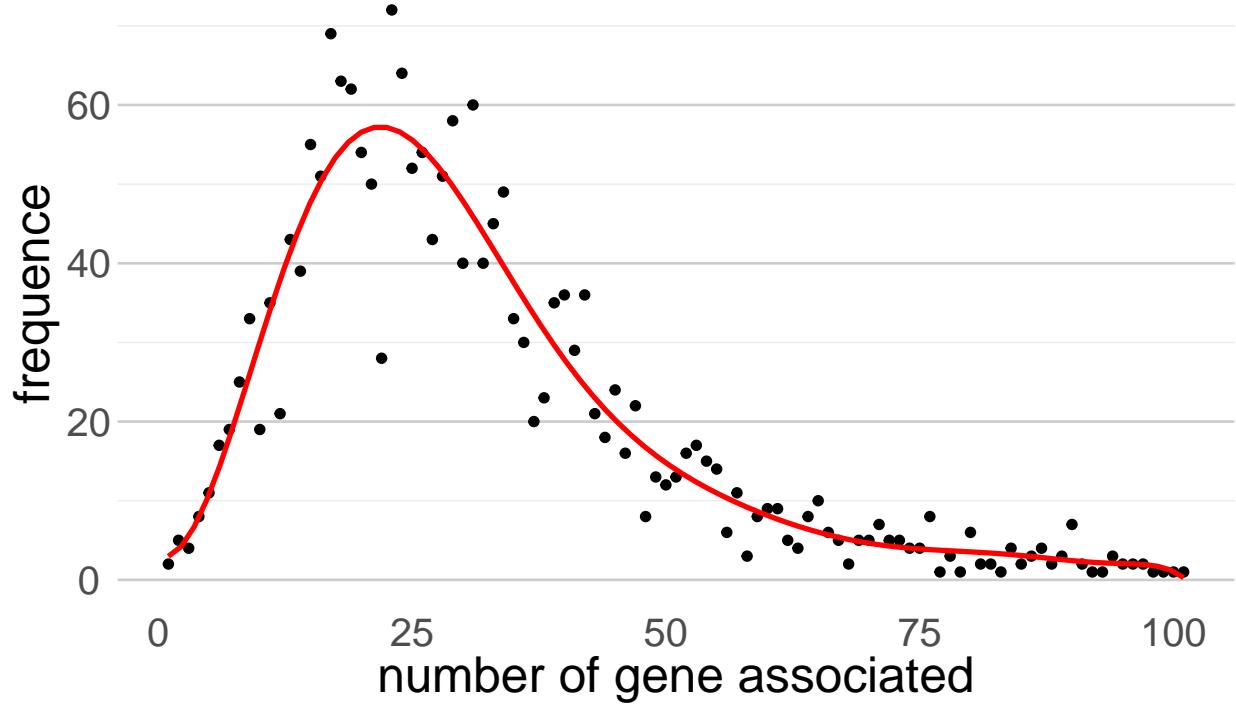
```
table_enhancer_gene = enhancerAnnotation(listTableEnhancer[[1]], genome = genomeFile, interval = 500000, nCore = 4)

#> GRanges object with 1979 ranges and 6 metadata columns:
#>   seqnames      ranges strand | chromatin_state start_500kb end_500kb
#>   <Rle>        <IRanges>  <Rle> |       <character>    <numeric> <numeric>
#> 1 chr10 9164400-9164800 * |           U13     8664400 9664800
#> 1 chr10 9342200-9344000 * |           U13     8842200 9844000
#> 1 chr10 10476400-10476600 * |           U13     9976400 10976600
#> 1 chr10 20520200-20521000 * |           U13     20020200 21021000
#> 1 chr10 20952400-20952600 * |           U13     20452400 21452600
#> . ... ...
#> 1 chrX 144286800-144287000 * |           U13     143786800 144787000
#> 1 chrX 155128400-155129200 * |           U13     154628400 155629200
#> 1 chrX 170010800-170013800 * |           U13     169510800 170513800
#> 1 chrY 198400-198800 * |           U13     -301600 698800
#> 1 chrY 90786000-90788000 * |           U13     90286000 91288000
#>   gene_association      distance      gene_list
#>   <integer>      <character>      <character>
#> 1 19 451159;278330;340253.. ENSMUSG00000111215.1..
#> 1 21 456130;480757;457563.. ENSMUSG00000015305.6..
#> 1 20 499773;435480;392457.. ENSMUSG00000101621.2..
#> 1 16 371729;318362;311710.. ENSMUSG0000019996.1..
#> 1 21 227322;432632;326765.. ENSMUSG00000019990.1..
#> .
#> 1 17 459386;459456;353489.. ENSMUSG00000067276.5..
#> 1 29 492849;417142;353947.. ENSMUSG00000084367.3..
#> 1 7 325601;314901;231165.. ENSMUSG00000035299.1..
#> 1 3 7351;124495;496977 ENSMUSG00000101796.1..
#> 1 9 384752;254355;180136.. ENSMUSG00000096178.7..
#> -----
#> seqinfo: 21 sequences from an unspecified genome; no seqlengths
```

## Number of gene associate at the enhancer

We want to know the distribution of number of genes associated at each enhancer.

```
plotGeneAssociation(table_enhancer_gene, all = FALSE)
```



## Associated gene expression to enhancer

We associated the level of gene expression at each gene associated to enhancer.

```
data(geneExpression)

#>      gene_ENS chr start end strand score gene_expression
#> 1 ENSMUSG00000000001.4 chr3 108107280 108146146 - . 27.7106904
#> 2 ENSMUSG00000000028.15 chr16 18780447 18811987 - . 23.5842993
#> 3 ENSMUSG00000000031.16 chr7 142575529 142578143 - . 0.9386427
#> 4 ENSMUSG00000000037.16 chrX 161117193 161258213 + . 14.4548991
#> 5 ENSMUSG00000000049.11 chr11 108343354 108414396 + . 36.6169129
#> 6 ENSMUSG00000000056.7 chr11 121237253 121255856 + . 5.2791187

table_enhancer_gene_expression = enhancerExpression(table_enhancer_gene, geneExpressionTable = geneExpr

#> GRanges object with 1979 ranges and 8 metadata columns:
#>   seqnames      ranges strand | chromatin_state start_500kb end_500kb
#>   <Rle>      <IRanges>  <Rle> |      <character>    <numeric> <numeric>
#> 1 chr10 9164400-9164800 * |          U13     8664400 9664800
#> 1 chr10 9342200-9344000 * |          U13     8842200 9844000
#> 1 chr10 10476400-10476600 * |          U13     9976400 10976600
#> 1 chr10 20520200-20521000 * |          U13    20020200 21021000
#> 1 chr10 20952400-20952600 * |          U13    20452400 21452600
#> .
#>   ... ...
#> 1 chrX 144286800-144287000 * |          U13    143786800 144787000
#> 1 chrX 155128400-155129200 * |          U13    154628400 155629200
#> 1 chrX 170010800-170013800 * |          U13    169510800 170513800
#> 1 chrY 198400-198800 * |          U13    -301600 698800
#> 1 chrY 90786000-90788000 * |          U13    90286000 91288000
```

```

#>      gene_association           distance          gene_list sample_name
#>      <integer>             <character>        <character> <character>
#> 1       19 451159;278330;340253.. ENSMUSG00000111215.1.. RS
#> 1       21 456130;480757;457563.. ENSMUSG0000015305.6.. RS
#> 1       20 499773;435480;392457.. ENSMUSG00000101621.2.. RS
#> 1       16 371729;318362;311710.. ENSMUSG0000019996.1.. RS
#> 1       21 227322;432632;326765.. ENSMUSG0000019990.1.. RS
#> .
#>       ...   ...
#> 1       17 459386;459456;353489.. ENSMUSG0000067276.5.. RS
#> 1       29 492849;417142;353947.. ENSMUSG0000084367.3.. RS
#> 1       7 325601;314901;231165.. ENSMUSG0000035299.1.. RS
#> 1       3 7351;124495;496977 ENSMUSG00000101796.1.. RS
#> 1       9 384752;254355;180136.. ENSMUSG0000096178.7.. RS
#>      gene_expression
#>      <character>
#> 1 NA;12.8456863815602;...
#> 1 12.8456863815602;2.0..
#> 1 NA;NA;NA;NA;NA;NA;NA..
#> 1 102.374504394998;2.0..
#> 1 0.571438637996035;3...
#> .
#>       ...
#> 1 0.255147894054393;NA..
#> 1 NA;NA;NA;12.00904498..
#> 1 0.161867853849147;NA..
#> 1           NA;NA;NA
#> 1 2.25309950916964;0.7..
#> -----
#> seqinfo: 21 sequences from an unspecified genome; no seqlengths

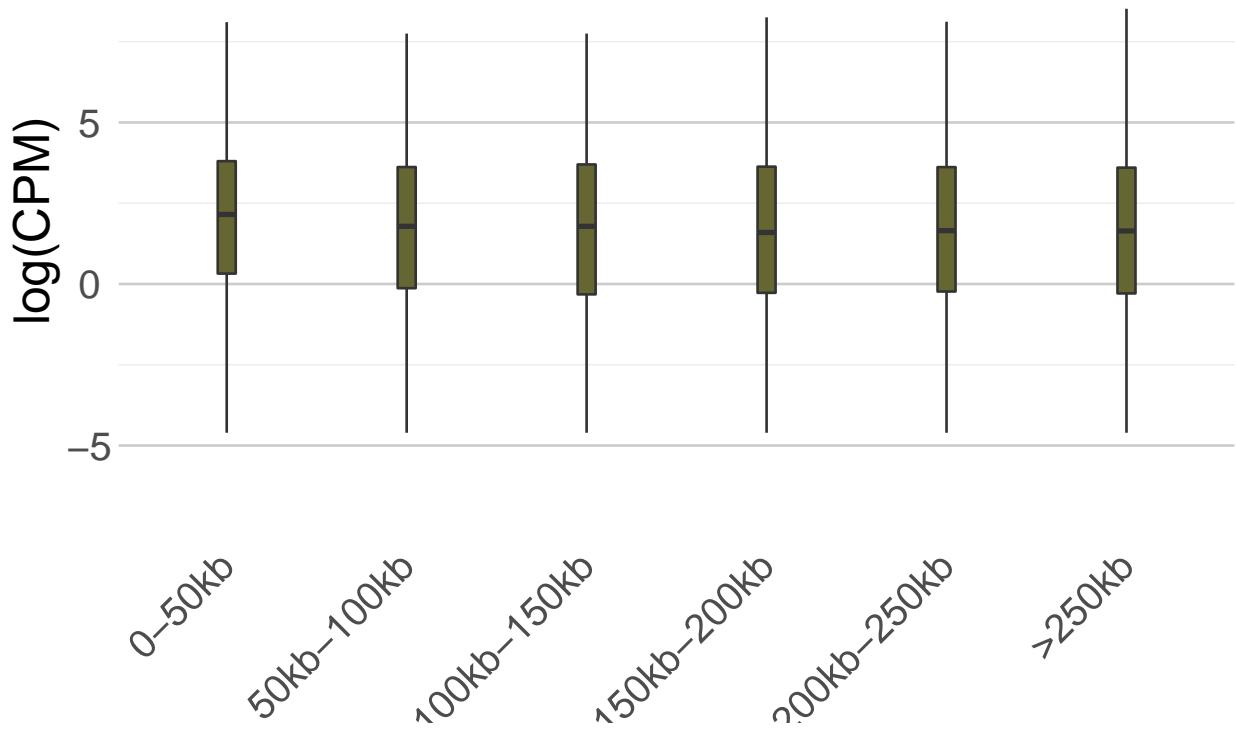
```

## Profile of enhancer annotation

### Distance gene-enhancer according to their expression

We generated plot to estimated the level of gene expression according to the distance between gene and enhancer.

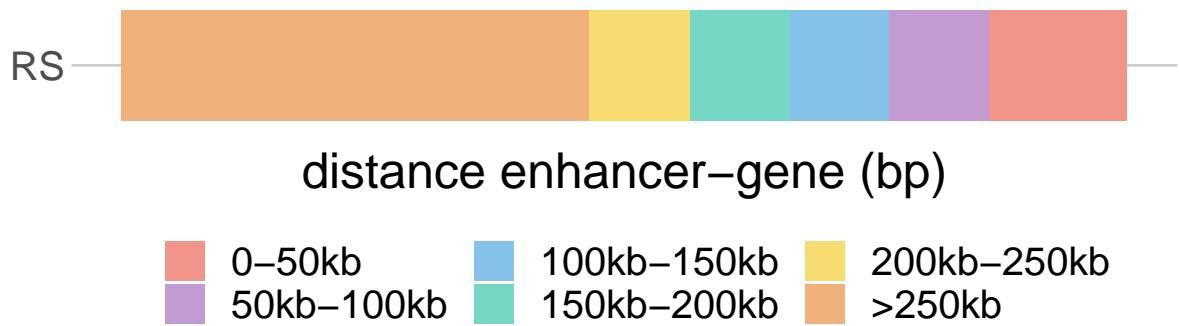
```
plotDistanceExpression(table_enhancer_gene_expression, color = colorValue, stateName = stateName, state
```



### Distance gene-enhancer

We generated plot to estimated the distribution of gene according to the distance between gene and enhancer.

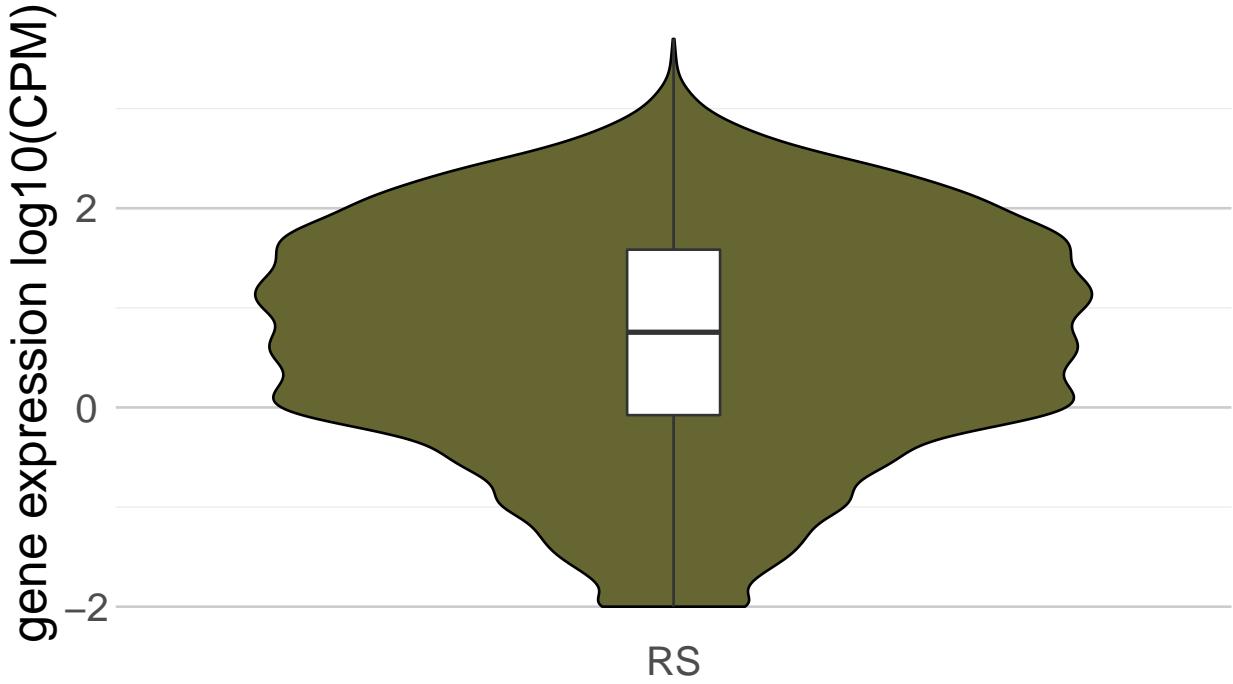
```
plotGeneDistance(table_enhancer_gene_expression)
```



### Enhancer expression

We generated plot with the distribution of gene expression associated at enhancer region.

```
plotEnhancerExpression(table_enhancer_gene_expression, scale = "log10", color = colorValue, stateName =
```



## Enhancer annotation comparison

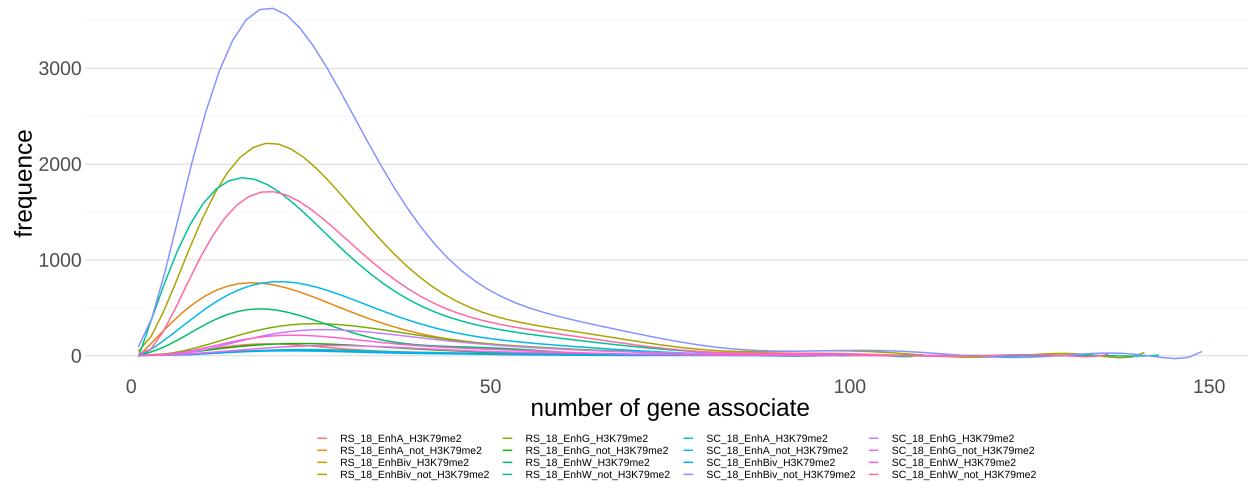
It's possible to compare different categories of enhancer. For that it's necessary to used a list of GRanges object each contains data like `list_table_enhancer` file.

The first step is associated gene to each enhancer using `enhancerAnnotation()` on the list of enhancer. After the gene association, we associated the gene expression using `enhancerExpression()`.

```
# list_table_enhancer_gene = lapply(listTableEnhancer, enhancerAnnotation, genome = genomeFile, intervalSize = 1000)
# list_table_enhancer_gene_expression = lapply(list_table_enhancer_gene, enhancerExpression, gene_expressions = TRUE)
```

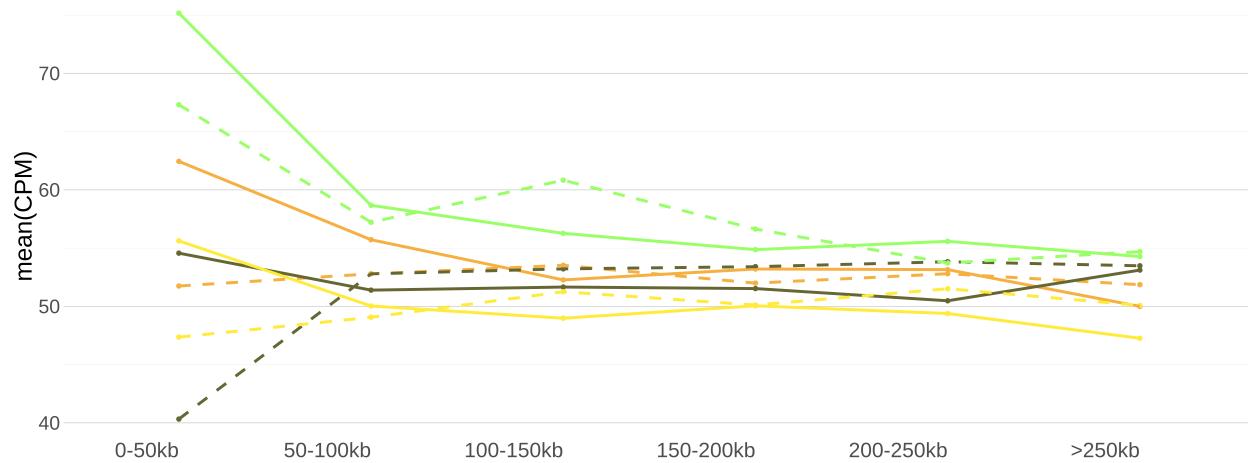
### Number of gene associate at the enhancer

```
# plotGeneAssociation(list_table_enhancer_gene_sample, all = T)
```



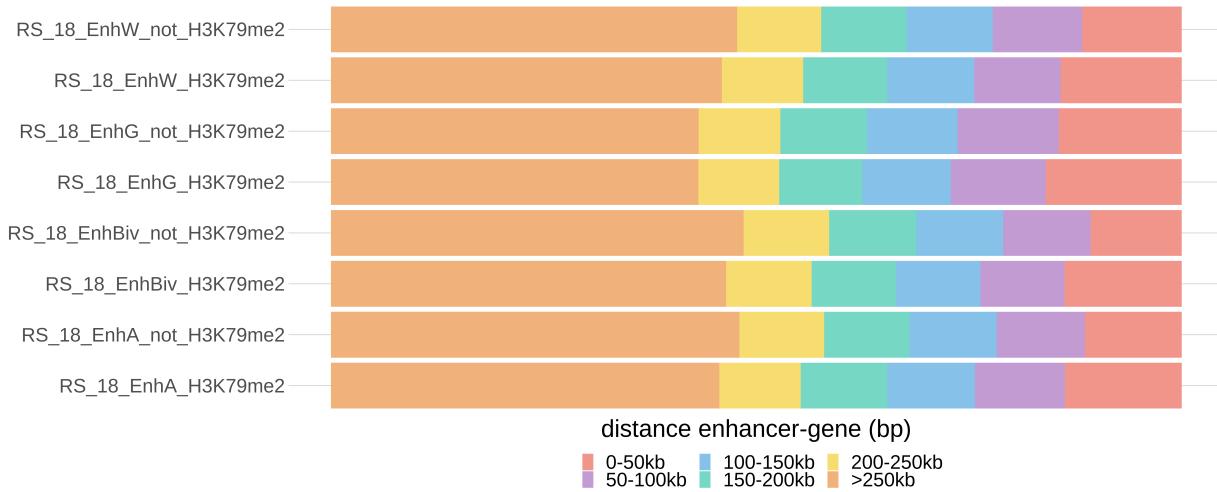
### Distance gene-enhancer in function their expression

```
# plotDistanceExpression(list_table_enhancer_gene_expression, color = colorValue, stateName = stateName)
```



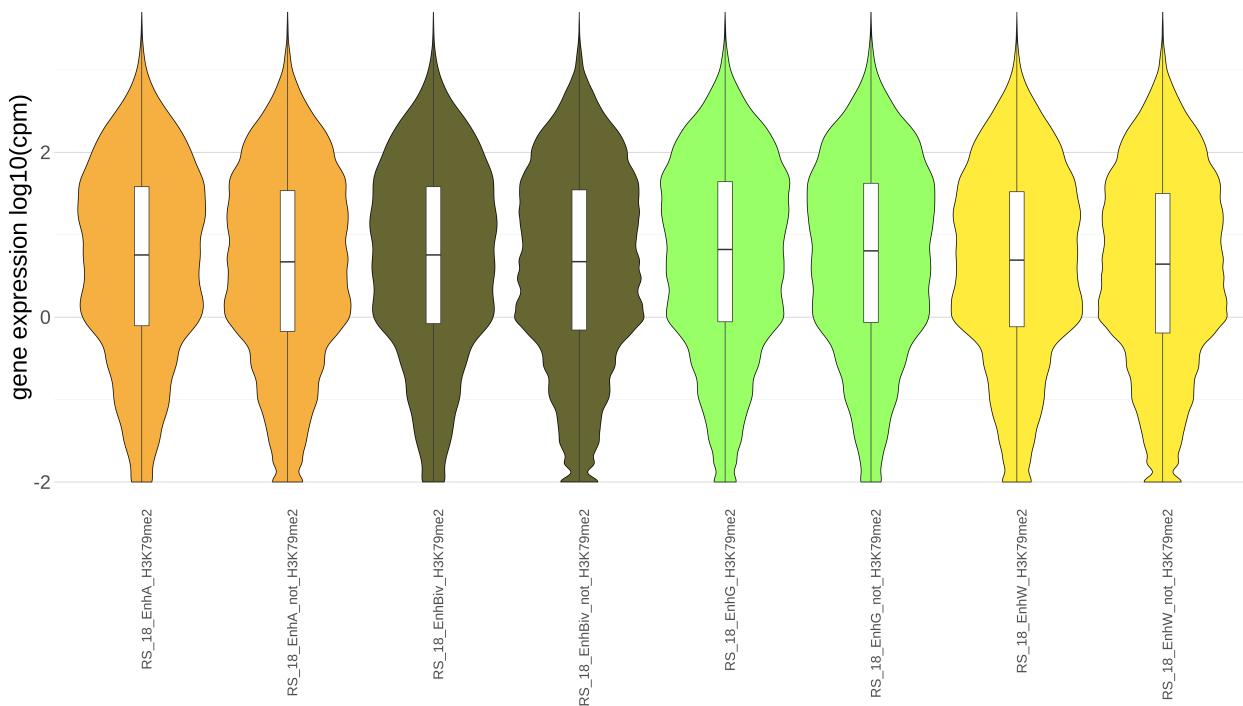
### Distance gene-enhancer

```
# plotGeneDistance(list_table_enhancer_gene_expression)
```



## Enhancer expression

```
# plotEnhancerExpression(list_table_enhancer_gene_expression, scale = "log10", color = colorValue, stat
```



## Gene environment

We are interested to studying the chromatin environment of gene.

```
stateOrderReduce = c("TSSA", "TSSFLnk", "TSSFLnk", "Tx", "Tx", "EnhG", "EnhG", "EnhA", "EnhWk", "ZNF.Rpts", "Het"
```

```
data(geneExpression)
data(chromatinState)
```

## Coverage of chromatin state in environment binding to TSS regions

To studying the environment of genes, we estimated as environment the interval (interval argument) around the TSS. `geneEnvironment()` may take few minutes in function the number of genes analyzed. For each gene, we obtains informations about the coverage of each chromatin state in the environment.

```
table_overlapping = geneEnvironment(geneExpression, chromatinState, stateOrder = unique(stateOrderReduce))
rownames(table_overlapping) = table_overlapping$gene_ENS
```

```
#>                                     gene_ENS    chr      start      end strand
#> ENSMUSG000000000001.4   ENSMUSG000000000001.4  chr3  108107280 108146146   -
#> ENSMUSG000000000028.15 ENSMUSG000000000028.15  chr16 18780447 18811987   -
#> ENSMUSG000000000031.16 ENSMUSG000000000031.16  chr7   142575529 142578143   -
#> ENSMUSG000000000037.16 ENSMUSG000000000037.16  chrX   161117193 161258213   +
#> ENSMUSG000000000049.11 ENSMUSG000000000049.11  chr11 108343354 108414396   +
#> ENSMUSG000000000056.7  ENSMUSG000000000056.7  chr11 121237253 121255856   +
#>                               score gene_expression      TSS TSS_moins_3kb
#> ENSMUSG000000000001.4     .    27.7106904 108146146 108143146
#> ENSMUSG000000000028.15   .    23.5842993 18811987 18808987
#> ENSMUSG000000000031.16   .    0.9386427 142578143 142575143
#> ENSMUSG000000000037.16   .    14.4548991 161117193 161114193
#> ENSMUSG000000000049.11   .    36.6169129 108343354 108340354
#> ENSMUSG000000000056.7   .    5.2791187 121237253 121234253
#>                               TSS_plus_3kb      TSSA      TSSFlnk Tx      EnhG EnhA
#> ENSMUSG000000000001.4    108149146 0.00000000 0.00000000 0 0.7423333 0
#> ENSMUSG000000000028.15   18814987 0.00000000 0.06666667 0 0.6333333 0
#> ENSMUSG000000000031.16   142581143 0.00000000 0.00000000 0 0.0000000 0
#> ENSMUSG000000000037.16   161120193 0.03333333 0.40000000 0 0.0000000 0
#> ENSMUSG000000000049.11   108346354 0.00000000 0.00000000 0 0.0000000 0
#> ENSMUSG000000000056.7   121240253 0.00000000 0.06666667 0 0.6000000 0
#>                               EnhWk ZNF.Rpts Het      TssBiv EnhBiv ReprPC Quies
#> ENSMUSG000000000001.4    0.0000    0 0.2576667 0.0 0.0000 0.0000000
#> ENSMUSG000000000028.15   0.0000    0 0.3000000 0.0 0.0000 0.0000000
#> ENSMUSG000000000031.16   0.0000    0 0.0000000 0.4 0.3095 0.2905000
#> ENSMUSG000000000037.16   0.1655    0 0.0000000 0.0 0.0000 0.4011667
#> ENSMUSG000000000049.11   0.0000    0 0.3000000 0.3 0.3410 0.0590000
#> ENSMUSG000000000056.7   0.0000    0 0.3333333 0.0 0.0000 0.0000000
```

## Predominant state in environment binding to TSS regions

We estimated as predominant chromatin state in the environment, the chromatin state with the higher coverage in the environment. For each gene, we use `umap` package to estimated the cluster dimension.

```
result_umap = predominantState(table_overlapping, state = unique(stateOrderReduce),
  header = unique(stateOrderReduce), neighbors = 32, metric = "euclidean", dist = 0.5)
#>
#> ==> It will be take few minutes to process
```

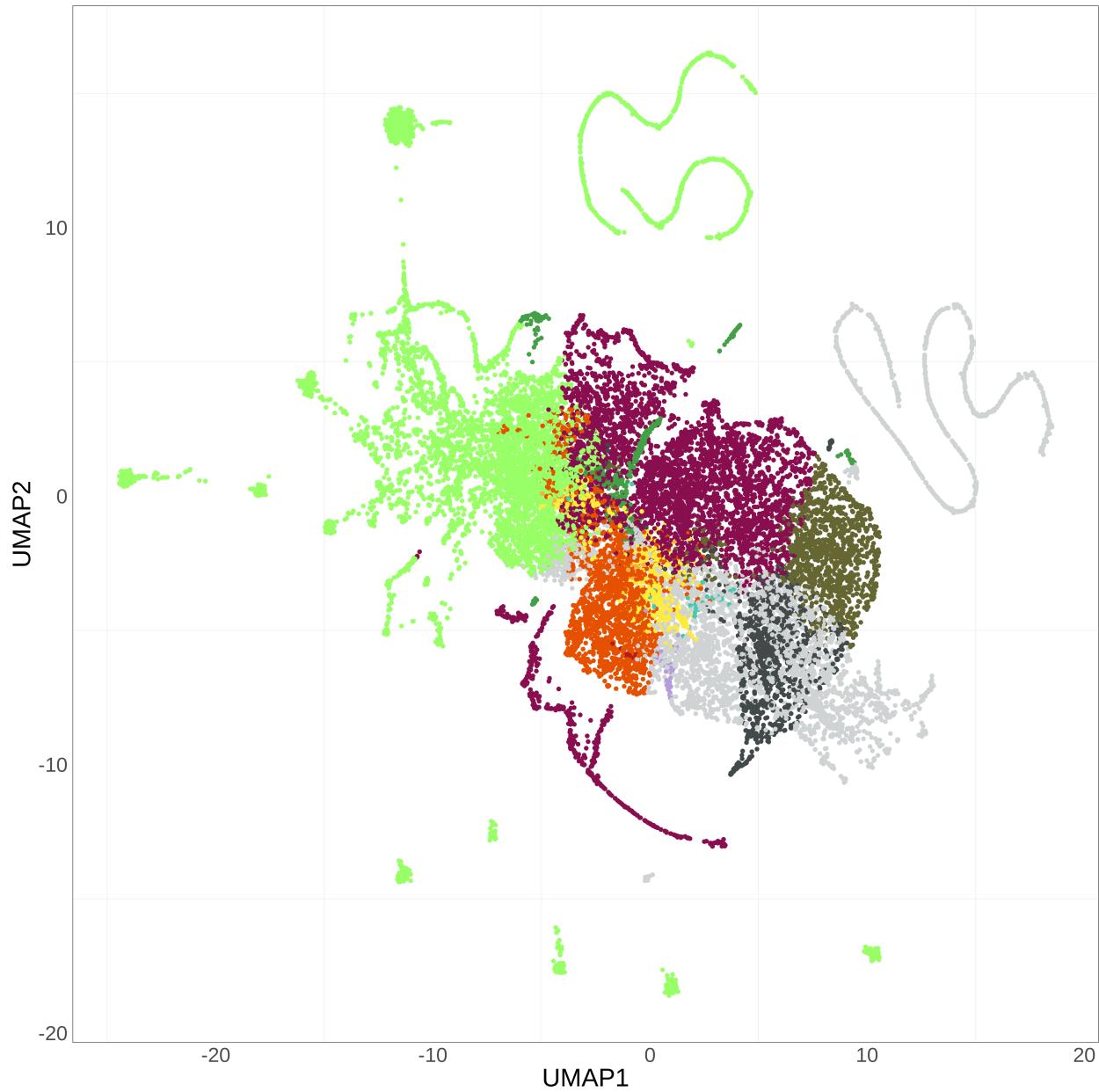
```
#>          TSSA      TSSFlnk Tx      EnhG EnhA EnhWk ZNF.Rpts
#> ENSMUSG000000000001.4 0.00000000 0.00000000 0 0.7423333 0 0.0000 0
```

```

#> ENSMUSG000000000028.15 0.0000000 0.06666667 0 0.6333333 0 0.0000 0
#> ENSMUSG000000000031.16 0.0000000 0.0000000 0 0.0000000 0 0.0000 0
#> ENSMUSG000000000037.16 0.03333333 0.40000000 0 0.0000000 0 0.1655 0
#> ENSMUSG000000000049.11 0.0000000 0.0000000 0 0.0000000 0 0.0000 0
#> ENSMUSG000000000056.7 0.0000000 0.06666667 0 0.6000000 0 0.0000 0
#>           Het   TssBiv EnhBiv ReprPC    Quies      UMAP1
#> ENSMUSG000000000001.4 0 0.2576667 0.0 0.0000 0.0000000 15.3444005
#> ENSMUSG000000000028.15 0 0.3000000 0.0 0.0000 0.0000000 11.2461522
#> ENSMUSG000000000031.16 0 0.0000000 0.4 0.3095 0.2905000 1.3368335
#> ENSMUSG000000000037.16 0 0.0000000 0.0 0.0000 0.4011667 0.9035514
#> ENSMUSG000000000049.11 0 0.3000000 0.3 0.3410 0.0590000 -2.2967628
#> ENSMUSG000000000056.7 0 0.3333333 0.0 0.0000 0.0000000 11.2293023
#>           UMAP2 state
#> ENSMUSG000000000001.4 -9.4548692 EnhG
#> ENSMUSG000000000028.15 -0.3666459 EnhG
#> ENSMUSG000000000031.16 9.5740863 EnhBiv
#> ENSMUSG000000000037.16 3.6425823 Quies
#> ENSMUSG000000000049.11 8.4852121 ReprPC
#> ENSMUSG000000000056.7 -1.6012913 EnhG

```

It's an example of UMAP representation to visualize the predominant chromatin state in each gene.



## Session Information

Here is the output of `sessionInfo()` on the system on which this document was compiled:

```
#> R version 4.1.3 (2022-03-10)
#> Platform: x86_64-conda-linux-gnu (64-bit)
#> Running under: Ubuntu 18.04.6 LTS
#>
#> Matrix products: default
#> BLAS/LAPACK: /home/mcoulee/anaconda3/envs/R_package_3/lib/libopenblas-r0.3.20.so
#>
#> locale:
#> [1] LC_CTYPE=fr_FR.UTF-8          LC_NUMERIC=C
```

```

#> [3] LC_TIME=fr_FR.UTF-8           LC_COLLATE=fr_FR.UTF-8
#> [5] LC_MONETARY=fr_FR.UTF-8      LC_MESSAGES=fr_FR.UTF-8
#> [7] LC_PAPER=fr_FR.UTF-8        LC_NAME=C
#> [9] LC_ADDRESS=C                LC_TELEPHONE=C
#> [11] LC_MEASUREMENT=fr_FR.UTF-8 LC_IDENTIFICATION=C
#>
#> attached base packages:
#> [1] stats      graphics   grDevices utils     datasets  methods   base
#>
#> other attached packages:
#> [1] ChromENVEE_1.1.7
#>
#> loaded via a namespace (and not attached):
#> [1] Rcpp_1.0.9                  lattice_0.20-45      png_0.1-7
#> [4] prettyunits_1.1.1          ps_1.7.1              assertthat_0.2.1
#> [7] rprojroot_2.0.3            digest_0.6.29       utf8_1.2.2
#> [10] RSpectra_0.16-1          R6_2.5.1              GenomeInfoDb_1.30.1
#> [13] stats4_4.1.3             evaluate_0.15      highr_0.9
#> [16] ggplot2_3.3.6            pillar_1.8.1        zlibbioc_1.40.0
#> [19] rlang_1.0.5              callr_3.7.1        S4Vectors_0.32.4
#> [22] Matrix_1.4-1            reticulate_1.26    rmarkdown_2.14
#> [25] labeling_0.4.2          splines_4.1.3      devtools_2.4.3
#> [28] stringr_1.4.1           RCurl_1.98-1.8    munsell_0.5.0
#> [31] umap_0.2.9.0            compiler_4.1.3    xfun_0.31
#> [34] askpass_1.1              pkgconfig_2.0.3    BiocGenerics_0.40.0
#> [37] pkgbuild_1.3.1          mgcv_1.8-40       htmltools_0.5.3
#> [40] openssl_2.0.3           tidyselect_1.1.2  tibble_3.1.8
#> [43] GenomeInfoDbData_1.2.7 IRanges_2.28.0    fansi_1.0.3
#> [46] crayon_1.5.1            dplyr_1.0.9       withr_2.5.0
#> [49] bitops_1.0-7            grid_4.1.3        nlme_3.1-158
#> [52] jsonlite_1.8.0          gtable_0.3.1      lifecycle_1.0.2
#> [55] DBI_1.1.3              magrittr_2.0.3    scales_1.2.1
#> [58] cli_3.4.0               stringi_1.7.8    cachem_1.0.6
#> [61] farver_2.1.1            XVector_0.34.0   fs_1.5.2
#> [64] remotes_2.4.2          ellipsis_0.3.2   vctrs_0.4.1
#> [67] generics_0.1.3          tools_4.1.3      glue_1.6.2
#> [70] purrrr_0.3.4            processx_3.7.0   pkgload_1.3.0
#> [73] parallel_4.1.3          fastmap_1.1.0    yaml_2.3.5
#> [76] colorspace_2.0-3        GenomicRanges_1.46.1 sessioninfo_1.2.2
#> [79] memoise_2.0.1           knitr_1.39       usethis_2.1.6

```

## References

Ernst J, Kellis M. ChromHMM: automating chromatin-state discovery and characterization. *Nature Methods*, 9:215-216, 2012

Papier scientifique associé

McInnes, Leland, and John Healy. “UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction.” arXiv:1802.03426.

Lawrence M, Huber W, Pagès H, Aboyoun P, Carlson M, Gentleman R, Morgan M, Carey V (2013). “Software for Computing and Annotating Genomic Ranges.” *PLoS Computational Biology*, 9. doi: 10.1371/journal.pcbi.1003118, <http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.1003118>.