Carmen M. Livi

ChIP-seq quality Control

ChIC

A short introduction

Contents

1 Introduction **2**

2 Input **3**

3 ENCODE Metrics (EM) **4**
3.1 Reading BAM files 4
3.2 Calculate QC-metrics from CrossCorrelation analysis 5
3.3 Remove anomalies in the read distrbution 7
3.4 Calculate QC-metrics from peak calls 7
3.5 Profile smoothing 8

4 Global enrichment profile Metrics (GM) and Fingerprint-plot **9**

5 Local enrichment profile metrics (LM) and metagene profiles **11**
5.1 Plotting an unscaled single-point metagene profile 11
5.2 Plotting a scaled whole gene metagene profile 11

6 Quality assessment using the compendium of QC-metrics as reference **14**
6.1 Available chromatin marks and factors 14
6.2 List the sample IDs of the compendium 15
6.3 Comparing local enrichment profiles 15
6.4 Comparing QC-metrics to the reference values of the compendium 15
6.5 Assessing data quality with machine learning 15

References **19**

7 Appendix **20**

1 Introduction

ChIP-seq quality **C**ontrol package (ChIC) provides functions and data structures to assess the quality of ChIP-seq data. The tool computes three different categories of quality control (QC) metrics: QC-metrics designed for narrow-peak profiles and general metrics, QC-metrics based on the global read distribution and QC-metrics from local signal enrichment around annotated genes. The user-friendly functions allow performing the analysis with a single command, whereas step by step functions are also available for more experienced users.
The package comes with a large reference compendium of precomputed QC-metrics from public ChIP-seq samples. Key features are the calculation, visualisation and creation of summary plots for QC-metrics, tools for the comparison of metagene profiles against reference profiles, tools for the comparison of single QC-metrics against the compendium values and finally a random forest model to compute a single score summarising quality control.
ChIC provides three wrapper functions that are used to calculate a comprehensive set of metrics: qualityScores_EM(), qualityScores_GM() and qualityScores_LM().

2 Input

To run ChIC the user has to provide two bam files: one for ChIP and one for the input. In this tutorial we illustrate ChIC functions using a H3K36me3 ChIP-seq dataset from ENCODE (ID: ENCFF000BFX) and its input (ID: ENCFF000BDQ). The data can can be downloaded from:
`https://www.encodeproject.org/files/ENCFF000BFX/`
`https://www.encodeproject.org/files/ENCFF000BDQ/`

```
> system("wget
+ https://www.encodeproject.org/files/ENCFF000BFX/
+     @@download/ENCFF000BFX.bam")
> system("wget
+ https://www.encodeproject.org/files/ENCFF000BDQ/
+     @@download/ENCFF000BDQ.bam")
> chipName <- "ENCFF000BFX"
> inputName <- "ENCFF000BDQ"
```

PLEASE NOTE: For timing reasons the tutorial will use toy-bam files with a reduced number of chromosomes. Input and chip data are therefore loaded from our datapackage "ChIC.data":

```
> library(ChIC)
> #load tag-list with reads aligned to a subset of chromosomes
> data("chipSubset", package = "ChIC.data",
+     envir = environment())
> chipBam <- chipSubset
> data("inputSubset", package = "ChIC.data",
+     envir = environment())
> inputBam <- inputSubset
```

3 ENCODE Metrics (EM)

qualityScores_EM() is a wrapper function that reads the provided bam files and calculates a number of QC-metrics from cross-correlation analysis and peak-calling. We will refer to the output measures as ENCODE Metrics EM, as originally proposed by ENCODE consortium [1].

```
> ##calculating EM
>
> mc=4 ##for parallelisation
> CC_Result <- qualityScores_EM(chipName=chipName,
+     inputName=inputName,
+     annotationID="hg19",
+     read_length=36,
+     mc=mc)
> finalTagShift <- CC_Result$QCscores_ChIP$tag.shift
```

The function expects two bam files: one for the immunoprecipitation (ChIP) and one for the control (Input). The read length (read_length parameter) can be taken from the bam file itself. The 'mc' parameter is set to 1 by default, when changed it triggers the parallelisation and speeds up the calculations of a few analysis steps that allow using multiple computing cores. 'annotationID' refers to the genome annotation used. ChIC currently supports hg19 and mm9. qualityScores_EM() produces the Cross-correlation plot (see Figure 1) and returns a number of QC-metrics (see: 7 Appendix). Amongst others the tag.shift value, which represents an input parameter for further steps (i.e. peak-calling and metagene calculation). qualityScores_EM() executes the following single steps:

1. Reading BAM files (readBamFile)
2. Calculatig QC-metrics from CrossCorrelation analysis (getCrossCorrelationScores)
3. Removing anomalies in the read distrbution (removeLocalTagAnomalies)
4. Calculating QC-metrics from peak calls (getPeakCallingScores)
5. Profile smoothing for further analysis steps (tagDensity)

PLEASE NOTE: The following code chunks are automatically executed within the qualityScores_EM() wrapper function. Nevertheless each listed function is available for single use if the user starts from the bam file and wants to perform all the steps individually.

3.1 Reading BAM files

The first step in the qualityScores_EM() function reads ChIP-seq data in .bam file format. The function expects the filename, that can also contain the pathname.

```
> chipBam <- readBamFile(chipName)
> inputBam <- readBamFile(inputName)
```

3.2 Calculate QC-metrics from CrossCorrelation analysis

getCrossCorrelationScores() calculates QC-metrics from the cross-correlation analysis and other general metrics, e.g. the non-redundant fractions of mapped reads. An important parameter required by getCrossCorrelationScores() is the binding-characteristics, calculated using spp::get.binding.characteristics() function. The binding-characteristics structure provides information about the peak separation distance and the cross-correlation profile (for more details see [2]).

```
> cluster <- parallel::makeCluster( mc )
> ## calculate binding characteristics
>
> chip_binding.characteristics <- get.binding.characteristics(
+     chipBam,
+     srange=c(0,500),
+     bin = 5,
+     accept.all.tags = TRUE,
+     cluster = cluster)
> input_binding.characteristics <- get.binding.characteristics(
+     inputBam,
+     srange=c(0,500),
+     bin = 5,
+     accept.all.tags = TRUE,
+     cluster = cluster)
> parallel::stopCluster( cluster )
```

```
> ## calculate cross correlation QC-metrics
> crossvalues_Chip <- getCrossCorrelationScores( chipBam ,
+     chip_binding.characteristics,
+     read_length = 36,
+     annotationID="hg19",
+     savePlotPath = filepath,
+     mc = mc)
```

"savePlotPath" sets the path in which the Cross-Correlation plot (as pdf) should be saved. If nothing is provided the plot will be forwarded to default DISPLAY. An example of a Cross-Correlation plot is shown in Figure 1. Along with the visual output, getCrossCorrelationScores() returns a list with EM (see 7 Appendix), from which we have to save tag.shift value for further steps:

```
> finalTagShift <- crossvalues_Chip$tag.shift
```

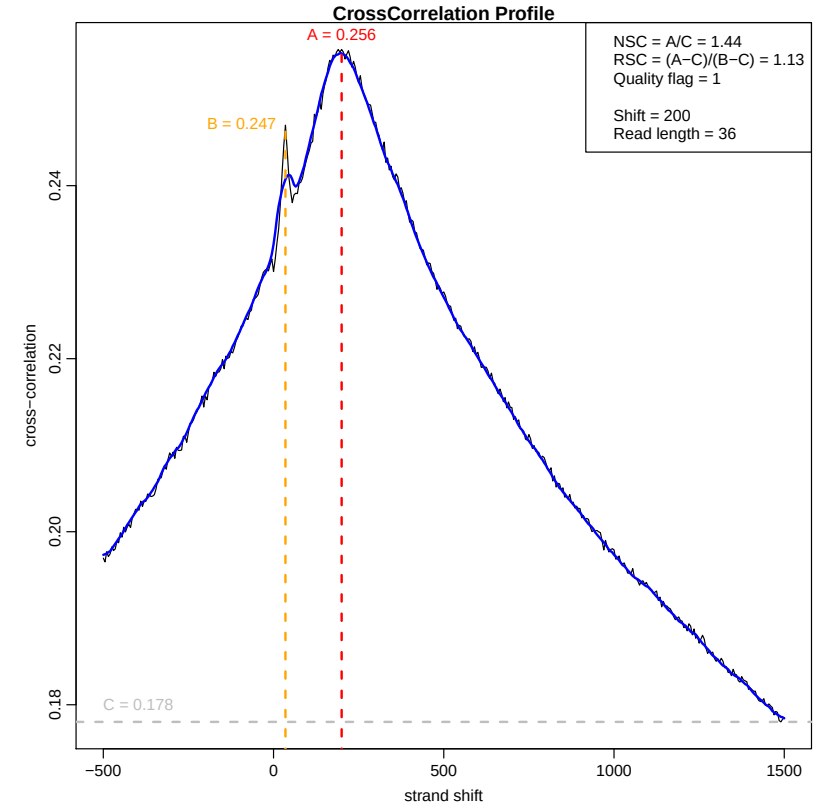

Figure 1: Cross-correlation plot of the ChIP.

3.3 Remove anomalies in the read distrbution

The data is processed further by using removeLocalTagAnomalies() that removes local read anomalies like regions with extremely high read counts compared to the neighborhood (for more details see [2]).

```
> ##get chromosome information and order chip and input by it
> chrl_final <- intersect(names(chipBam$tags),
+     names(inputBam$tags))
> chipBam$tags <- chipBam$tags[chrl_final]
> chipBam$quality <- chipBam$quality[chrl_final]
> inputBam$tags <- inputBam$tags[chrl_final]
> inputBam$quality <- inputBam$quality[chrl_final]
```

```
> ##remove positions with extremely high read counts with
> ##respect to the neighbourhood
> selectedTags <- removeLocalTagAnomalies(chipBam,
+     inputBam,
+     chip_binding.characteristics,
+     input_binding.characteristics)
> inputBamSelected <- selectedTags$input.dataSelected
> chipBamSelected <- selectedTags$chip.dataSelected
```

3.4 Calculate QC-metrics from peak calls

The last set of QC-metrics belonging to the category of EMs are based on the number of called peaks using getPeakCallingScores().

```
> bindingScores <- getPeakCallingScores(chip = chipBam,
+     input = inputBam,
+     chip.dataSelected = chipBamSelected,
+     input.dataSelected = inputBamSelected,
+     annotationID="hg19",
+     tag.shift = finalTagShift,
+     mc = mc)
```

```
finding background exclusion regions ... done
determining peaks on provided 1 control datasets:
using reversed signal for FDR calculations
bg.weight= 1.440726  excluding systematic background anomalies ... done
determining peaks on real data:
bg.weight= 0.6940943  excluding systematic background anomalies ... done
calculating statistical thresholds
FDR 0.01 threshold= 3.165151
finding background exclusion regions ... done
determining peaks on provided 1 control datasets:
```

```
using reversed signal for FDR calculations
bg.weight= 1.440726  excluding systematic background anomalies ... done
determining peaks on real data:
bg.weight= 0.6940943  excluding systematic background anomalies ... done
calculating statistical thresholds
E-value 10 threshold= 5.545838
```

3.5 Profile smoothing

The last step executed in qualityScores_EM() is the smoothing (using a Gaussian kernel) of the read profile to obtain the tag density profile (for more details see [2]).

```
> smoothedChip <- tagDensity(chipBamSelected,
+     annotationID = "hg19",
+     tag.shift = finalTagShift, mc = mc)
```

```
.
3 -858 ...
```

```
> smoothedInput <- tagDensity(inputBamSelected,
+     annotationID = "hg19",
+     tag.shift = finalTagShift, mc = mc)
```

The read density profile is needed to calculate the remaining two categories of QC-metrics: the Global enrichment profile Metrics (GM) and the local enrichment profile metrics (LM) (see 7 Appendix).

4 Global enrichment profile Metrics (GM) and Fingerprint-plot

This category of QC-metrics is based on the global read distribution along the genome for ChIP and Input [3]. The wrapper qualityScores_GM() reproduces the so-called Fingerprint plot (Figure 2), i.e. the cumulative read distribution plot, from which 9 quantitative QC-metrics are derived. Examples of these metrics are the (a) fraction of bins without reads for ChIP and input, (b) the point of maximum distance between the ChIP and input (x-coordinate, y-coordinate for ChIP and input, the distance calculated as absolute difference between the two y-coordinates, the sign of the difference), (c) the fraction of reads in the top 1 percent of bins with highest coverage for ChIP and input.

```
> Ch_Results <- qualityScores_GM(densityChip = smoothedChip,
+     densityInput = smoothedInput,
+     savePlotPath = filepath)
> str(Ch_Results)
```

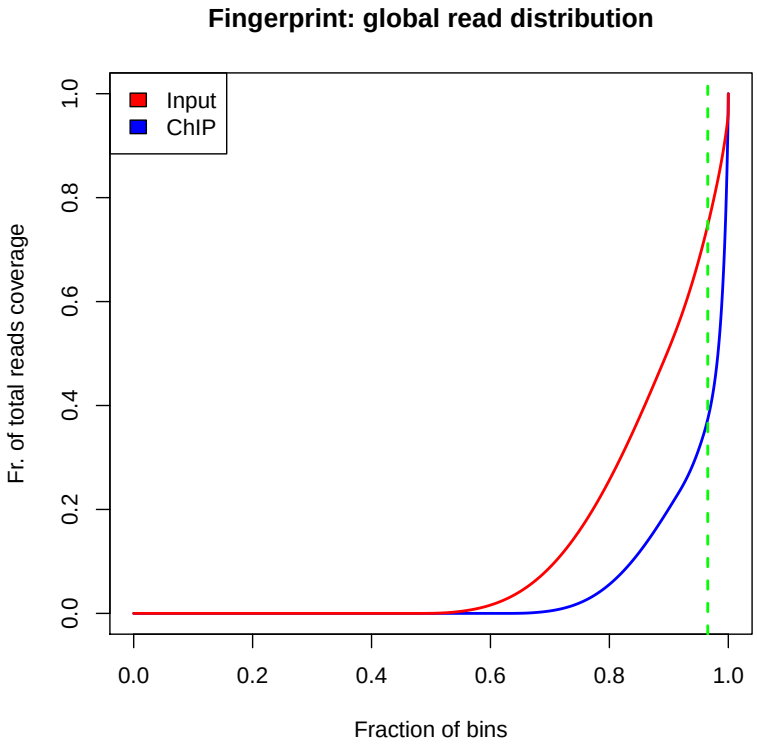

Figure 2: Fingerprint plot of sample ENCFF000BFX and its input ENCFF000BDQ.

```
List of 9
 $ Ch_X.axis                       : num 0.965
 $ Ch_Y.Input                      : num 0.747
 $ Ch_Y.Chip                       : num 0.374
 $ Ch_sign_chipVSinput             : num 1
 $ Ch_FractionReadsTopbins_chip    : num 0.385
 $ Ch_FractionReadsTopbins_input   : num 0.12
 $ Ch_Fractions_without_reads_chip : num 0.632
 $ Ch_Fractions_without_reads_input: num 0.485
 $ Ch_DistanceInputChip            : num 0.373
```

5 Local enrichment profile metrics (LM) and metagene profiles

createMetageneProfile() creates the unscaled single-point and a scaled whole gene metagene profile and returns a list with three items: "TSS", "TES" and "geneBody". Each item is again a list with the metagene profiles for ChIP and input.

```
> Meta_Result <- createMetageneProfile(
+     smoothed.densityChip = smoothedChip,
+     smoothed.densityInput = smoothedInput,
+     annotationID="hg19",
+     tag.shift = finalTagShift,
+     mc = mc)
```

The objects in "Meta_Result" are needed to create the metagene plots and to extract the LMs for the different profiles.
Metagene profiles show the signal enrichment around a region of interest like the transcription start site (TSS) or over the gene body. ChIC creates two types of metagene profiles: an unscaled single-point profile for the TSS and transcription end site, and a scaled whole gene metagene profile including the gene body like in Figure 4. For the metagene profiles the tag density of the immunoprecipitation is taken over all RefSeq annotated human genes, averaged and log2 transformed. The same is done for the input. The normalised profile (Figure 5) is calculated as the signal enrichment (immunoprecipitation over the input) and plotted on the y-axis, whereas the genomic coordinates of the genes like the TSS or regions up- and downstream are shown on the x-axis.

5.1 Plotting an unscaled single-point metagene profile

The "TSS" or "TES" object and the qualityScores_LMgenebody() function are used to plot the unscaled profile (see Figure 3) and return the respective LM values.

```
> TES_Scores=qualityScores_LM(data=Meta_Result$TES,
+     tag="TES")
```

5.2 Plotting a scaled whole gene metagene profile

The "geneBody" object and the qualityScores_LMgenebody() function are used to plot the scaled profile (see Figure 4) and return the respective LM values:

```
> TSS_Scores=qualityScores_LM(data=Meta_Result$TSS,
+     tag="TSS")
```

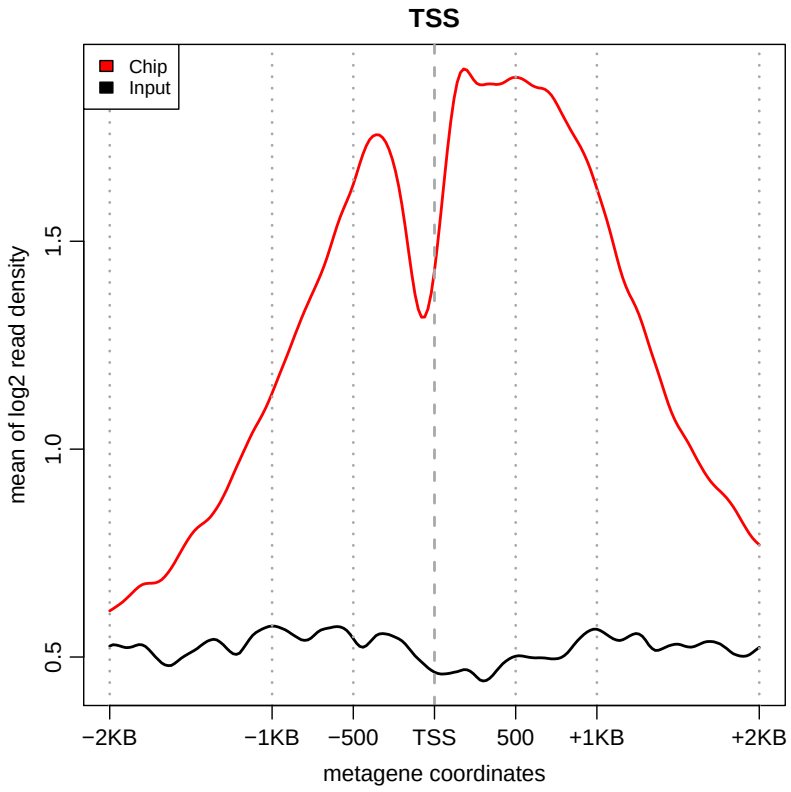

Figure 3: Unscaled single-point metagene profile: Signal enrichment for ChIP and Input at the TSS.

```
> geneBody_Scores <- qualityScores_LMgenebody(Meta_Result$geneBody)
```

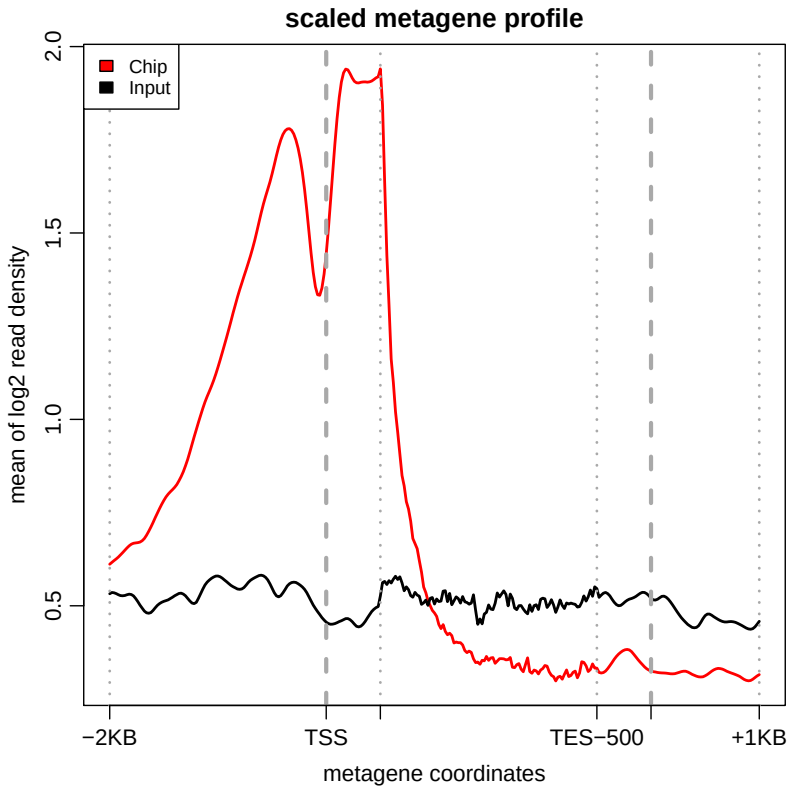

Figure 4: Scaled whole gene metagene profile: Signal enrichment for ChIP and Input along the gene body.

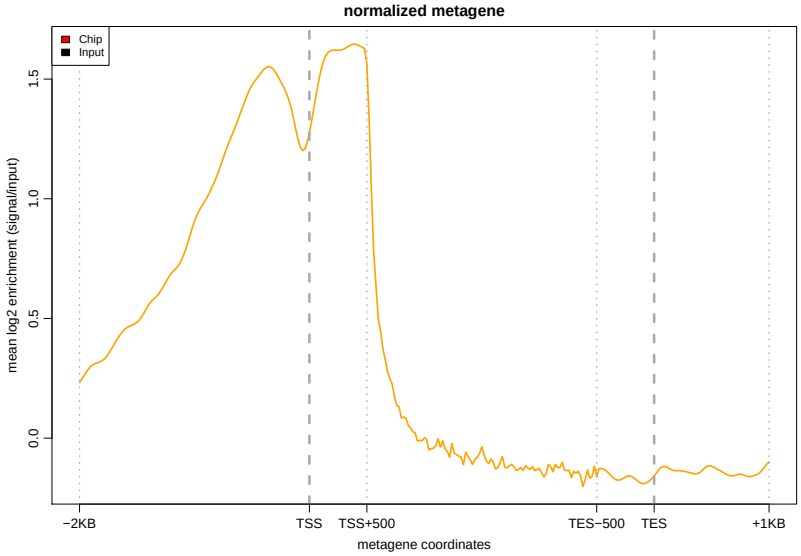

Figure 5: Normalized profile: signal enrichment for ChIP over Input along the gene body for a scaled whole gene profile.

6 Quality assessment using the compendium of QC-metrics as reference

The comprehensive set of QC-metrics, computed over a large set of ChIP-seq samples, constitutes in itself a valuable compendium that can be used as a reference for comparison with new samples. ChIC provides three functions for that:

- *metagenePlotsForComparison* to compare the metagene plots with the compendium
- *plotReferenceDistribution* to compare a single QC-metric with the compendium values
- *predictionScore* to obtain a single quality score from the random forest models trained on the previously computed QC-metrics

6.1 Available chromatin marks and factors

This useful function lists all chromatin marks and transcription factors that are available for the comparison analysis. With the keywords ”mark” and ”TF” the respective lists with the available elements are listed.

```
> listAvailableElements(target="H3K36me3")
```

6.2 List the sample IDs of the compendium

Shows the IDs of all analysed ChIP-seq samples from ENCODE and Roadmap that have been included in the compendium by providing the keyword "ENCODE" or "Roadmap".

```
> head(listDatasets(dataset="ENCODE"))
```

```
[1] "ENCFF001GCN" "ENCFF000RWQ" "ENCFF000VJB" "ENCFF000RWN"
[5] "ENCFF001GCY" "ENCFF000YXP"
```

6.3 Comparing local enrichment profiles

The metagenePlotsForComparison() function is used to compare the local enrichment profile to the reference compendium by plotting the metagene profile against the expected metagene for the same type of chromatin mark. The expected metagene profile is provided by the compendium mean (black line) and standard error (blue shadow). Examples are shown in Figures 6 and 7.

6.4 Comparing QC-metrics to the reference values of the compendium

Plotting a single QC-metric against the reference values from a large number of already published data, adds an extra level of information that can be easily used by less experienced users. An example is shown in Figure 8.

6.5 Assessing data quality with machine learning

The compendium of metrics has been used to train a random forest model that summarizes the sample quality in one single QC-score.

```
> te <- predictionScore(target = "H3K4me3",
+     features_cc = CC_Result,
+     features_global = Ch_Results,
+     features_TSS = TSS_Scores,
+     features_TES = TES_Scores,
+     features_scaled = geneBody_Scores)
> print(te)
```

```
[1] 0.498
```

```
> metagenePlotsForComparison(data = Meta_Result$geneBody,
+     target = "H3K4me3",
+     tag = "geneBody")
>
```

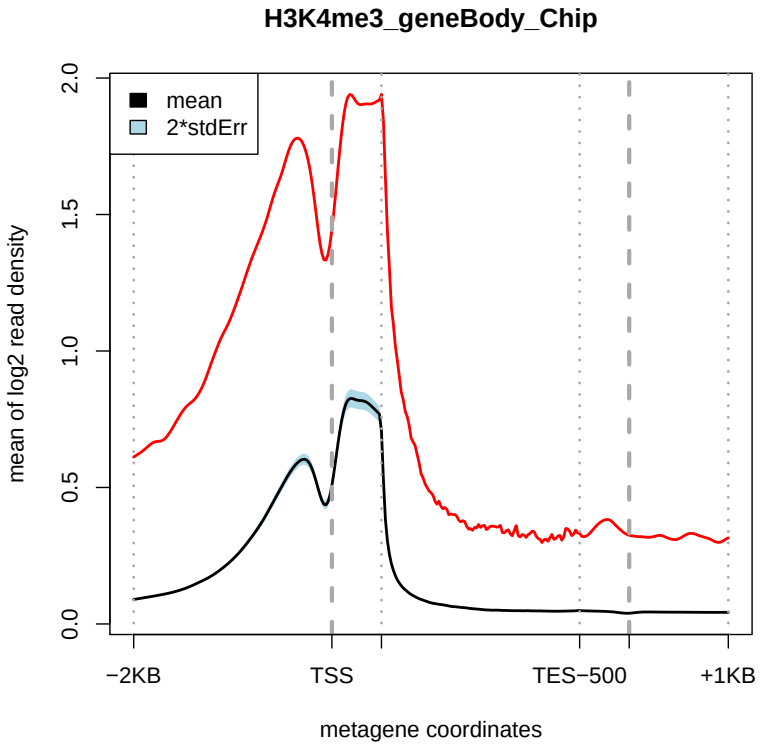

Figure 6: Enrichment profile plotted against the pre-computed profiles of the compendium. The metagene profile shows the sample signal (red line) for the ChIP compared to the compendium mean signal (black line) and the 2x standard error (blue shadow).

```
> metagenePlotsForComparison(data = Meta_Result$TSS,
+     target = "H3K4me3",
+     tag = "TSS")
```

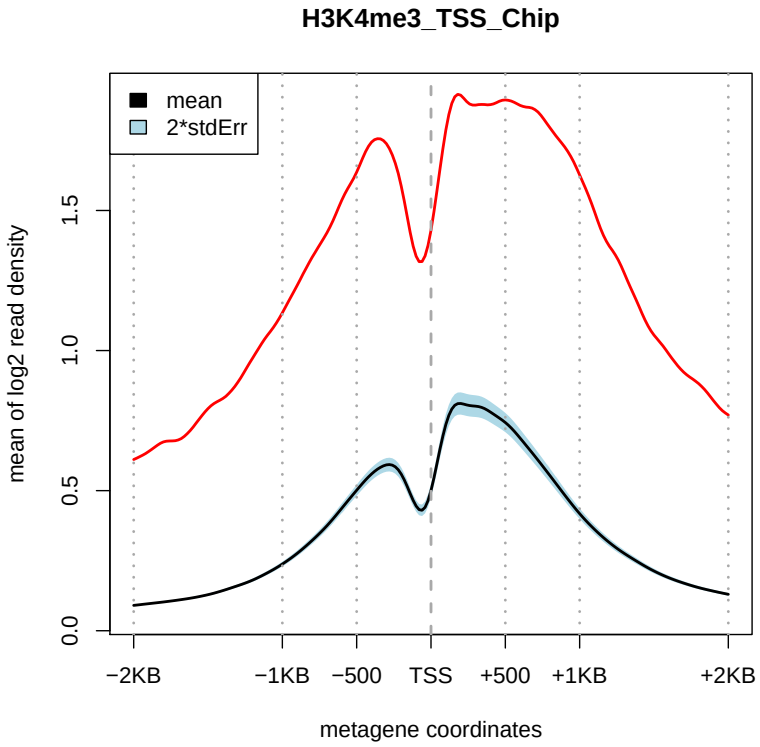

Figure 7: The metagene profile shows the sample signal for the ChIP (red line) compared to the compendium mean signal (black line) and the 2x standard error (blue shadow).

```
> plotReferenceDistribution(target = "H3K4me3",
+     metricToBePlotted = "RSC",
+     currentValue = crossvalues_Chip$CC_RSC )
```

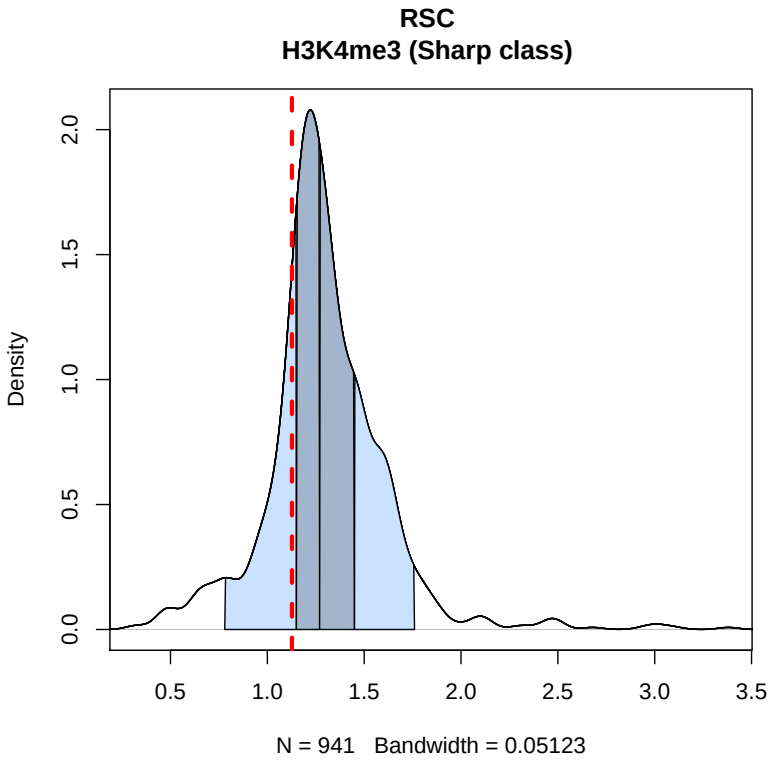

Figure 8: The QC-metric of a newly analysed ChIP-seq sample can be compared to the reference values of the compendium. The density plot shows the QC-metric RSC (red dashed line) of the sample versus the distribution of the same metric in the the compendium for the respective binding profile or TF.

References

[1] Landt, Stephen G. ChIP-seq guidelines and practices of the ENCODE and modENCODE consortia. Genome Research 2012

[2] Kharchenko, Peter V and Tolstorukov, Michael Y and Park, Peter J. Design and analysis of ChIP-seq experiments for DNA-binding proteins. Nat. Biotechnol. 2008,

[3] Diaz, Aaron and Nellore, Abhinav and Song, Jun. CHANCE: comprehensive software for quality control and validation of ChIP-seq data. Genome Biol. 2012

7 Appendix

Complete List of Local enrichment profile Metrics LM

TSS unscaled single-point metaprofiles (identical list for TES unscaled single-point)

chip_hotSpots_TSS [-2000|-1000|-500|0|500|1000|2000]: ChIP-signal at position with relative TSS distance as specified (-2Kb|-1Kb|-500bp|0|+500bp|+1Kb|+2Kb).
chip_localMax_TSS_1_[x|y]: metagene (x|y) coordinate of maximum ChIP -signal in interval [-2Kb, -1Kb]
chip_localMax_TSS_2_[x|y]: metagene (x|y) coordinate of maximum ChIP -signal in interval [-1Kb -500bp]
chip_localMax_TSS_3_[x|y]: metagene (x|y) coordinate of maximum ChIP -signal in interval [-500b, TSS]
chip_localMax_TSS_4_[x|y]: metagene (x|y) coordinate of maximum ChIP -signal in interval [TSS, 500bp]
chip_localMax_TSS_5_[x|y]: metagene (x|y) coordinate of maximum ChIP -signal in interval [500bp, 1Kb]
chip_localMax_TSS_6_[x|y]: metagene (x|y) coordinate of maximum ChIP -signal in interval [1Kb, 2Kb]
chip_auc_TSS_1: ChIP -signal AUC in interval [-2Kb, -1Kb]
chip_auc_TSS_2: ChIP -signal AUC in interval [-1Kb -500bp]
chip_auc_TSS_3: ChIP -signal AUC in interval [-500bp, TSS]
chip_auc_TSS_4: ChIP -signal AUC in interval [TSS, 500bp]
chip_auc_TSS_5: ChIP -signal AUC in interval [500bp, 1Kb]
chip_auc_TSS_6: ChIP -signal AUC in interval [1Kb, 2Kb]
chip_dispersion_TSS_-[500|1000|2000]_variance: variance in the ChIP metagene profile values within the interval TSS +/- (500bp|1Kb|2Kb), as specified
chip_dispersion_TSS_-[500|1000|2000]_sd: standard deviation in the ChIP metagene profile values within the interval TSS +/- (500bp|1Kb|2Kb), as specified
chip_dispersion_TSS_-[500|1000|2000]_[0|25|50|75]%: percentiles as specified (0|25|50|75%) of ChIP metagene profile values within the interval TSS +/- (500bp|1Kb|2Kb), as specified
norm_hotSpots_TSS_[-2000|-1000|-500|0|500|1000|2000]: normalized (ChIP over input enrichment) at position with relative TSS distance as specified (-2Kb|-1Kb|-500bp|0|+500bp|+1Kb|+2Kb).
norm_localMax_TSS_1_[x|y]: metagene (x|y) coordinate of max ChIP enrichment in interval [-2Kb, -1Kb]
norm_localMax_TSS_2_[x|y]: metagene (x|y) coordinate of max ChIP enrichment in interval [-1Kb -500bp]
norm_localMax_TSS_3_[x|y]: metagene (x|y) coordinate of max ChIP enrichment in interval [-500bp, TSS]
norm_localMax_TSS_4_[x|y]: metagene (x|y) coordinate of max ChIP enrichment in interval [TSS, 500bp]
norm_localMax_TSS_5_[x|y]: metagene (x|y) coordinate of max ChIP enrichment in interval [500bp, 1Kb]
norm_localMax_TSS_6_[x|y]: metagene (x|y) coordinate of max ChIP enrichment in interval [1Kb, 2Kb]
norm_auc_TSS_1: normalized (ChIP over input enrichment) AUC in interval [-2Kb, -1Kb]
norm_auc_TSS_2: normalized (ChIP over input enrichment) AUC in interval [-1Kb, -500bp]
norm_auc_TSS_3 normalized (ChIP over input enrichment) AUC in interval [-500bp, TSS]
norm_auc_TSS_4: normalized (ChIP over input enrichment) AUC in interval [TSS, 500bp]
norm_auc_TSS_5: normalized (ChIP over input enrichment) AUC in interval [500bp, 1Kb]
norm_auc_TSS_6: normalized (ChIP over input enrichment) AUC in interval [1Kb, -2Kb]
norm_dispersion_TSS_-[500|1000|2000]_variance: variance in the normalized ChIP over input enrichment metagene profile values within the interval TSS +/- (500bp|1Kb|2Kb), as specified
norm_dispersion_TSS_-[500|1000|2000]_sd: standard deviation in the normalized ChIP over input enrichment metagene profile values within the interval TSS +/- (500bp|1Kb|2Kb), as specified

.

norm _dispersion_TSS_-[500|1000|2000]_[0|25|50|75]%: percentiles as specified (0|25|50|75%) of normalized ChIP over input enrichment metagene profile values within the interval TSS +/- (500bp|1Kb|2Kb), as specified

Scaled whole gene metaprofiles

chip_hotSpots_twopoints_[-2000|0|500|2500|3000|4000]: ChIP-signal at the specified coordinate position in the scaled whole gene metagene: i.e. respectively (TSS-2Kb | TSS | TSS+500bp | TES-500bp | TES | TES+1Kb).
chip_localMax_twopoint_1_[x|y]: metagene (x|y) coordinate of maximum ChIP-signal in interval [-2Kb, TSS]
chip_localMax_twopoint_2_[x|y]: metagene (x|y) coordinate of maximum ChIP-signal in interval [TSS, TSS+500bp]
chip_localMax_twopoint_3_[x|y]: metagene (x|y) coordinate of maximum ChIP-signal in interval [TSS+500bp, TES-500bp]
chip_localMax_twopoint_4_[x|y]: metagene (x|y) coordinate of maximum ChIP-signal in interval [TES-500bp, TES]
chip_localMax_twopoint_5_[x|y]: metagene (x|y) coordinate of maximum ChIP-signal in interval [TES, TES+1Kb]
chip_auc_twopoint_1: ChIP -signal AUC in interval [-2Kb, TSS]
chip_auc_twopoint_2: ChIP -signal AUC in interval TSS, TSS+500bp]
chip_auc_twopoint_3: ChIP -signal AUC in interval [TSS+500bp, TES-500bp]
chip_auc_twopoint_4: ChIP -signal AUC in interval [TES-500bp, TES]
chip_auc_twopoint_5: ChIP -signal AUC in interval [TES, TES+1Kb]
norm_hotSpots_twopoints_[-2000|0|500|2500|3000|4000]: nomralized ChIP over input enrichment at the specified coordinate position in the scaled whole gene metagene: i.e. respectively (TSS-2Kb | TSS | TSS+500bp | TES-500bp | TES | TES+1Kb).
norm_localMax_twopoint_1_[x|y]: metagene (x|y) coordinate of maximum ChIP enrichment in interval [-2Kb, TSS]
norm_localMax_twopoint_2_[x|y]: metagene (x|y) coordinate of maximum ChIP enrichment in interval [TSS, TSS+500bp]
norm_localMax_twopoint_3_[x|y]: metagene (x|y) coordinate of maximum ChIP enrichment in interval [TSS+500bp, TES-500bp]
norm_localMax_twopoint_4_[x|y]: metagene (x|y) coordinate of maximum ChIP enrichment in interval [TES-500bp, TES]
norm_localMax_twopoint_5_[x|y]: metagene (x|y) coordinate of maximum ChIP enrichment in interval [TES, TES+1Kb]
norm_auc_twopoint_1: ChIP enrichment AUC in interval [-2Kb, TSS]
norm_auc_twopoint_2: ChIP enrichment AUC in interval TSS, TSS+500bp]
norm_auc_twopoint_3: ChIP enrichment AUC in interval [TSS+500bp, TES-500bp]
norm_auc_twopoint_4: ChIP enrichment AUC in interval [TES-500bp, TES]
norm_auc_twopoint_5: ChIP enrichment AUC in interval [TES, TES+1Kb]

.

Complete list of ENCODE metrics EM

StrandShift: cross-correlation peak coordinate is fragment-length strand shift value on x-axis
PBC: number of genomic locations to which exactly one uniquely mapping read is located / the number of genomic locations to which at least one uniquely mapping read is located, i.e. the number of non-redundant uniquely mapping reads
readLength: length of the read
A: cross-correlation peak coordinate, y-axis
B: phantom-peak in cross-correlation profile, y-axis
C: baseline of cross-correlation coefficient values at extreme strand-shifts (height of line C on the y-axis)
NSC: NSC=A/C
RSC: RSC=(A-C)/(B-C)
QualityFlag: quality control tag
ALL_TAGS: number of mapped reads
UNIQUE_TAGS: number of uniquely mapped reads
UNIQUE_TAGS_LibSizeadjusted: adjusted by library size
UNIQUE_TAGS_nostrand: ignoring the strand direction
NRF: UNIQUE_TAGS/ALL_TAGS
NRF_nostrand: NRF ignoring the strand direction
NRF_LibSizeadjusted: NRF adjusted by library size
FDRpeaks: number of called peaks using FDR threshold
evalpeaks: number of called peaks using e-value threshold
FRiP_broadPeak: Fraction of reads under broad peaks
FRiP_sharpPeak: Fraction of reads under the sharp peaks
outcountsBroadPeak: number of broad peaks called
outcountsSharpPeak: number of sharp peaks called

Complete list of Global enrichment profile Metrics GM

X.axis: point of maximum distance between ChIP and Input, x-coordinate in the CHANCE plot
Y.Input: point of maximum distance between ChIP and Input, y-coordinate of Input in the CHANCE plot
Y.Chip: point of maximum distance between ChIP and Input, y-coordinate of ChiP in the CHANCE plot
DistanceInputChip: maximum distance between ChIP and Input
sign_chipVSinput: sign of the maximum distance
FractionReadsTopbins_chip: fraction of reads in the top 1% of bins with highest coverage for ChIP
FractionReadsTopbins_input: fraction of reads in the top 1% of bins with highest coverage for Input
Fractions_without_reads_chip: the fraction of bins without reads for ChIP
Fractions_without_reads_input: the fraction of bins without reads for Input

.