Chapter 3

Differential Enriched Scan 2 DEScan2

Epigenetics the
Epigenetic, as shown in introduction (cite), is a pretty wide and complex field, and the sequencing technology to adopt depends on the biological question under investigation. have

Some studies [1, 2] demonstrated the importance of genome-wide genomewide chromatin accessibility of a broad spectrum of chromatin phenomena activation using sequencing techniques as *Atac-Seq*, *Sono-Seq*, etc. Even if there are some methods for the analysis of these omic data types, there still is lack of them, in particular for an emerging omic as *Atac-Seq*. ATAC-seq non Atac-seq

need
To address this lack, we decided to create a useful instrument for analysing chromatine regions accessibility data (such as *Atac-Seq*, *Sono-Seq*). Very often the biological questions, to be answered, as for [redacted] RNA-Seq, need the comparison of two or more different biological conditions. Starting from a set of already published [1] scripts, we designed Differential Enriched Scan 2 (DEScan2), a software for [redacted] the analysis of chromatin accession sequencing data.

In this chapter we firstly inllustrate the developed methodologies and then,

with a case study, we will show the obtained results as an application [redacted]

3.1 Introduction

[redacted] DEScan2 is an R [3] tool developed for detecting open chromatin regions signal in order to facilitate the differential enrichment of genomic regions between two or more biological conditions.

The package has been implemented using Bioconductor [4] data structures and methods, and it is available through the Bioconductor repository since version 3.7.

The tool is organized in three main steps. A peak caller, which is a standard moving scan window that compares the reads coverage signal within a sliding window [redacted] to the signal in a larger region outside the window. It uses a Maximum Likelihood Estimator of on a Poisson Distribution, providing a final score for each detected peak.

The filtering and alignment steps are aimed to determine if a peak is a "true peak" on the basis of its replicability in other samples. These steps are grouped in a single procedure and are based on a double user-defined threshold, one on the peaks's scores and one on the number of samples.

The third step produces a count [redacted] matrix where each column represents a sample and each row a peak. The value of each cell represents the number of reads for the peak in the sample.

The [redacted] produced counts matrix, as illustrated in [redacted] figure 3.2.1, is useful both for doing differential enrichment between multiple conditions and for integrating the epigenomic data with other -omic data types.

3.2 Methods

The package is organized in three main steps, the peak caller in section 3.2.1, the filtering and alignment of the peaks in section 3.2.2 and the peak counting described in section 3.2.3.

Furthermore, it offers some additional features as described in 3.2.4.

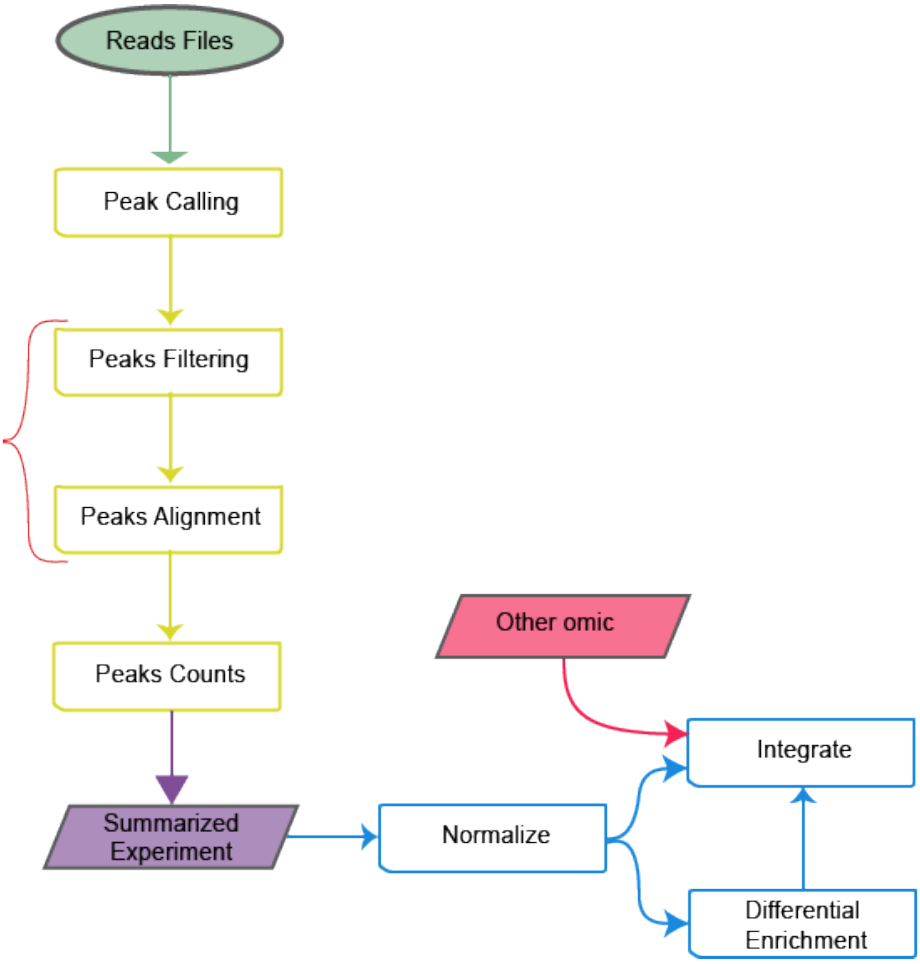


Figure 3.2.1: A differential enrichement flow representation. DEScan2 steps are highlighed in yellow.

3.2.1 Peak Caller

The Peak Caller (defined by the `findPeaks` function) takes as input a set of alignment files (BAM [5] or BED format) with the code identifier of the refer-

ence genome (i.e. *mm10* for Mus Musculus version 10) and several additional parameters, useful for the peak detection setup.

The alignment data are stored as *GenomicRangesList* [6], where each element represents a file. In order to facilitate the parallelization of the computations over the chromosomes, the list is re-arranged as a chromosome list of *GenomicRangesList*, where each element represents the file containing just the *GenomicRanges* of the specific chromosome (see section 3.2.4 for a detailed description of this procedure).

For each element of this data structure the algorithm firstly divides each chromosome in bins of `binSize` parameter length (the default value is 50bp) and then computes the reads coverage of the bins with moving scan windows, spanning from `minWin` to `maxWin` parameters of `binSize` interval.

In order to be able to catch narrow and wide peaks the algorithm computes the coverage also using windows of two different lengths, that can be defined with `minCompWinWidth` and `maxCompWinWidth` (defaults values are 5000bp and 10000bp) parameters, computing a matrix of n bins and p windows.

The coverage matrix is useful to merge contigous regions and to compute a score for each of them, applying a Maximum Likelihood Extimator (MLE), assuming a Poisson distribution of the coverages across the windows.

Formalizing: assuming that each window is distributed as a Poisson random variable, we assume to observe the n coverages as an IID sequence X_n. Thus, the probability mass function is described as:

$$p(x_i) = \frac{\lambda^{x_i}}{x_i!} exp(-\lambda)$$

Where the integer nature of the data support the Poisson distribution as the set of non-negative integer number and where λ is the Poisson parameter to estimate with a MLE, described as the estimator:

$$\lambda_n = \frac{1}{n} \sum_{i=1}^{n} x_i$$

Which corresponds to the sample mean of the n observations in the sample.

Additionally, on user request, the function provides as output, for each alignment file, a Tab Separated Value (tsv) file within the regions coordinates and the score of the detected peaks.

3.2.2 Peak Filtering and Alignment

In order to filter out false positives peaks, we designed a method (defined in the `finalRegions` function) which firstly filters out low score regions and then aligns the resulting regions between samples, using two different thresholds. One on the peaks's score and one on the number of samples.

The filtering step is designed to take as input a list of peaks as *GenomicRangesList*, where each element represents a file. This is the data structure produced by the peak caller, but, we also developed a method to load peaks produced by other software like MACS [7], as described in section 3.2.4.

Firstly, using the threshold on the peaks's score (defined by the `zThreshold` parameter), the method filters out the peaks with a score lower than the user-defined threshold value.

Then, for aligning the peaks between the samples, it extends a 200bp window in both directions of remaining regions, computing the overlaps using the `findOverlapsOfPeaks` method (using the `connectedPeaks` parameter set as `merge`), as defined in *ChIPpeakAnno* [8] R/Bioconductor package.

Based on this idea, the filtering step is developed to filter out those peaks not present in at least a user-defined number of samples, defined by the `minCarriers` parameter. In the light of this, the user can decide the minimum number of samples where each peak has to be detected. our experience, we suggest to set the samples threshold as a mutiple of the number of replicates of the conditions.

3.2.3 Counting Peaks

The counting step (`countFinalRegions` method) is designed to take a *GenomicRanges* data structure as input, where for each peak additional attributes are saved, as well as the score and the number of samples. Moreover, to quantify

the peaks given as input, it requires also the path of the alignment files where the reads are stored.

For each region the method counts the number of reads present in each sample. In so doing, it produces a matrix of counts, where the rows and the columns, respectively, represent the regions and the samples.

In order to keep track of all information associated to the regions, it produces a *SummarizedExperiment* [9] data structure, giving the possibility to retrieve the *GenomicRanges* of associated peaks and the count matrix, respectively, using the `rowRanges` and `assays` methods.

The choice to produce a count matrix is guided by the versatility of this data structure, useful not only for the differential enrichment of the regions between multiple conditions, but also for integrating epigenomic data with other such omic data types, as RNA-Seq.

3.2.4 Additional Features

The package offers some additional features for loading data (i.e. peaks) resulting from other sources, and for manipulating *GenomicRanges* data structure.

To give the possibility to use our pipeline with external peak callers, the function `readFilesAsGRangesList` takes as input a directory containing BAM or BED data, to load in *GenomicRangesList* format. This data structure is useful to store genomic information, as peaks or mapped reads, produced by other software like *MACS2* or *STAR* and, in case of peaks, it is necessary during the DESCan2 filtering/aligning step. Additionally to `fileType` (BAM, BED, BED.zip) parameter specification it requires the genome code to use during the file processing. Moreover, when the input files represent peaks the `arePeaks` flag needs to be set to `TRUE`.

Furthermore, DEScan2 provides several functionalities for GenomicRanges data structure handling. One example is `fromSamplesToChrsGRangesList`, which gives the possibility to split a GenomicRangesList by chromosome. This procedure could be useful for parallelizing computations on the chromosomes, when common operations on them, between multiple samples, are needed. Assigning a single chromosome to a single computing unit. Taken

as input a *GenomicRangesList* organized by samples, this method returns a list of chromosomes, where each element has a *GenomicRangesList* of samples, containing only the regions associated to the single chromosome.

Other useful utilities are `keepRelevantChrs`, that takes a *GenomicRangesList* and a list of chromosomes and return only the interested chromosomes with a cleaned *genomeInfo* assigned; the `saveGRangesAsTsv` function that saves a tab separated value file starting from a GenomicRanges; the `saveGRangesAsBed` that save a standard BED file format starting from a GenomicRanges data structure; and the `setGRangesGenomeInfo` which, starting from a genome code, sets a specific *genomeInfo* to a *GenomicRanges* object.

3.3 Case Study

Atac-Seq is an emerging evoluted technique which enables to investigate the open chromatin regions at whole genome level. [redacted] in the regulation of mouse brain activity under different conditions. Aggiungi referenza!

The capability of this technology has been demonstrated

To illustrate the performances of DEScan2 we chose a dataset [10] that describes in vivo adult mouse dentate granule neurons before and after synchronous neuronal activation using Atac-Seq and RNA-Seq technologies (see sections 1.2.2 and 1.2.1 for a description of these sequencing techniques).

This dataset is organized in 62 samples of Atac-Seq and RNA-Seq, extracted at four different time points (0, 1h, 4h, 24h), with four replicates at each time point. We chose to compare the differences between the first two stages, time 0 (E0) and 1 hour after neuronal induction (E1), in order to show a potential Atac-Sec workflow for Differential Enrichment, and how to integrate this data type with RNA-Seq. A general illustration of this dataset is represented in figure 3.3.1.

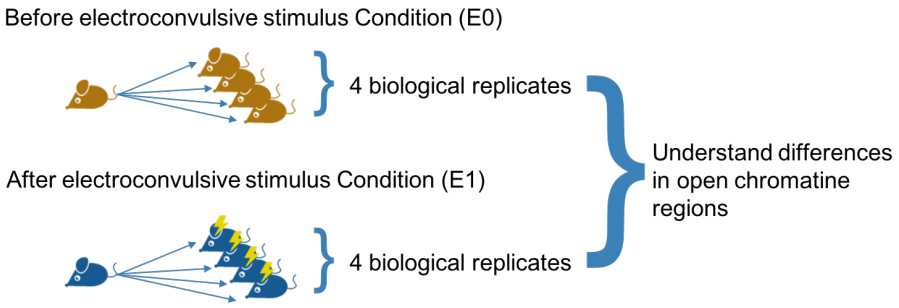


Figure 3.3.1: An illustration of our extraction of the GSE82015[10] dataset.

We downloaded the data from Gene Expression Omnibus (GEO) database [11, 12] with accession number GSE82015[1] and mapped the raw data using *STAR* [13] with default parameter on Mus Musculus Genome ver.10 (mm10).

In order to detect open chromatin regions we run our peak caller, cutting the genome in bins of 50bp and using running windows of minimum 50bp and maximum 1000bp. In this way we are able to detect not just broad peaks, but also smaller peaks.

To be confident with our results we run DEScan2 and *MACS2* on the same samples, and (as shown in figure 3.3.2) looking to the numbers DEScan2 always outperforms *MACS2* peaks.

Servono più dettagli su che cosa intendi per "outperforms": solo il fatto che trovi più picchi non vuol dire che sia meglio, potrebbero essere tutti falsi positivi.

[1] https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE82015

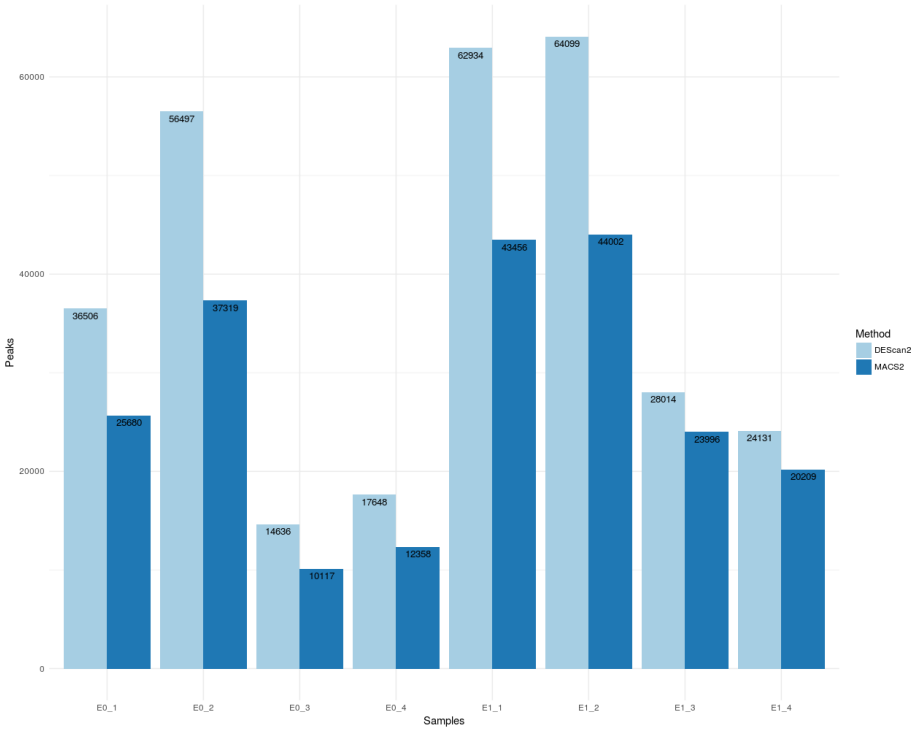


Figure 3.3.2: A comparison of DEScan2 and *MACS2* detected peaks for each sample in the dataset.

To be more robust, we compared DEScan2 detected peaks with the same validated regions (*Arc*[2] and *Gabrr1*[3]) of the original work [10]. The lower part of figure 3.3.3 shows the detected and validated regions (in blue and red) resulting differentially enriched between the E0 (in pink) and E1 (in green) conditions, while the upper part shows DEScan2 filtered and aligned peaks (in blue) between the samples, highlighting a capability to catch not only the same regions of the published ones, but also (gold circles) to be more [redacted] in the smaller peaks detection.

accurate

[2]https://www.genecards.org/cgi bin/carddisp.pl?gene=ARC
[3]https://www.genecards.org/cgi bin/carddisp.pl?gene=GABRR1

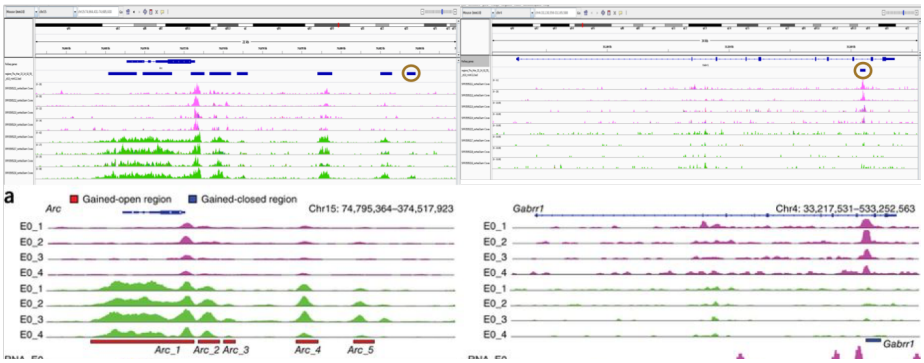
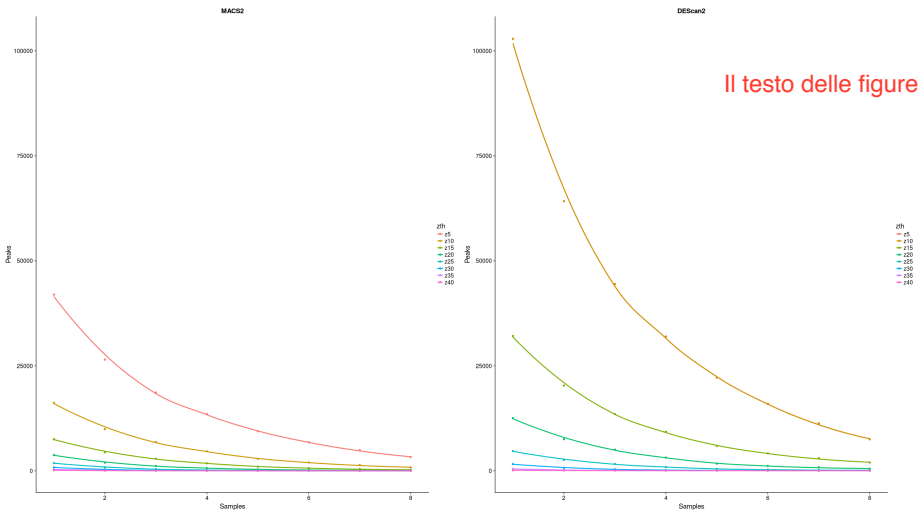


Figure 3.3.3: A comparison of DEScan2 detected peaks with validated peaks in article [10].

Cosa intendi per "reliable regions"?

While it is very important to detect good peaks with a peak caller, it seems to be more relevant to detect reliable regions. Indeed, during the filtering/aligning step, the number of peaks depends not only by the peak score, but also by the number of replicates designed in the experiment. [redacted] figure 3.3.4 puts in relation these two relevant information for both MACS2 and DEScan2. On the x-axis is represented the number of replicates, while on the y-axis is traced the number of peaks, and each curve represents a different threshold on the peaks score, showing that the higher are the thresholds on the scores and the number of replicates, the lower is the number of the detected peaks. Highlighting a [redacted] inverse relationship [redacted] between the number of [redacted] peaks and the combination of the number of samples and the detected regions score.



Il testo delle figure deve essere più grande

Figure 3.3.4: Filtering the detected regions with different thresholds on peak scores between *MACS2* and DEScan2.

Moreover, comparing left and right panels, we notice the high difference in pooling the samples-peaks together with the DEScan2 filtering/aligning step when using [redacted] *MACS2* and [redacted] DEScan2 peaks. Using [redacted] *MACS2* peaks the pooling highly reduces the number of detected peaks, even using a [redacted] threshold as low as 5 on the score, showing that there are many peaks with a score lower than 5. While in the DEScan2 case the curves representing the threshold equal to 5 and the threshold equal to 10 totally overlap, highlighting that the DEScan2 peak caller produces scores higher than 10.

Questo confronto non so quanto sia interessante perché gli score sono su scale diverse. Forse puoi ribadire che gli score non sono direttamente confrontabili

Afterwards, the filtered-in regions can be processed by DEScan2 in order to obtain a count matrix with samples on the columns and peaks on the rows. This type of data structure is very versatile, because it enables to perform several operations, like the Differentially Enriched genomic Regions (DERs) and the integration with other kind of omics, as RNA-Seq.

Parli di filtered-in regions ma non hai detto come fai il filtraggio. Quale threshold sullo score? Quanti campioni?

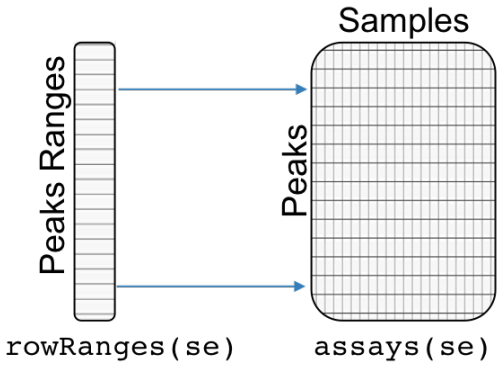


Figure 3.3.5: An illustration of the *SummarizedExperiment* data structure produced by DEScan2.

Questa di per se non [redacted] una ripetizione di quanto detto prima metterei cosi: This is especially needed in neuroscience, where many possible sources of technical and biological noise can confound the analysis. E aggiungerei magari una referenza al mio paper con Lucia su NAR (Peixoto et al. 2015).

In order to preserve the information associated to the peaks, DEScan2 produces as output a *SummarizedExperiment* (figure 3.3.5) data structure, which enables to retrieve the count matrix with `assays` method, and to access the peaks information in *GenomicRanges* format with the `rowRanges` method.

Before [redacted] detecting DERs, it is a good [redacted] practice to normalize the data, [redacted]. The nature of the data, in count format, makes it possible to apply several well known RNA-Seq normalizations techniques, such as *TMM*, *upper-quartile*, *full-quantile*, *RUV-Seq*, etc [14 16]. In this case, we fixed the peaks's score threshold to 10, in order to have as much signal as possible.

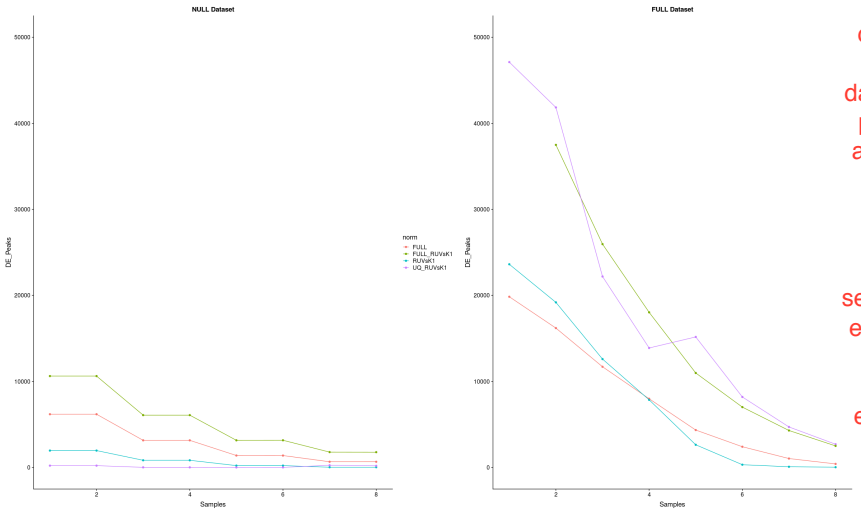
While the *TMM* and *upper-quartile* normalizations modify the data in a way that makes it impossible to detect DERs, other kind of normalizations and combinantions of them give good results.

[redacted] figure 3.3.6 summarizes this concept very well, highlighting a relation between the number of DERs (y-axis) and the minumum number of samples (x-axis) used for filtering the data during the DEScan2 filtering step.

To better compare the normalization effect, we created a *null dataset* of 8 samples, simply doubling the E0 samples and performed a differential analysis between E0 (4 samples) and E0-fake (4 samples).

Non capisco come hai fatto questa analisi. Se hai semplicemente raddoppiato gli stessi campioni, come possono esserci regioni differenziali? Hai aggiunto qualche effetto random?

The right panel of the plot, representing the "full dataset", shows that *upper-quantile*, even if combined with *RUV-Seq* normalization, is not able to linearly detect a good amount of DERs, while *full-quantile*, when combined with *RUV-Seq* seems to affect the data in a way that overdetectsthe number of DERs. When looking at the *full-quantile* and *RUV-Seq* by themself seem to perform better than the other normalizations. The first one has a downhill almost linear, while the second one has a very fast downhill with a regrowth when the number of samples is higher.



Non mi è molto chiaro cosa ci dovremmo aspettare da questi due dataset. In teoria nel null dataset dovrei avere il 5% di falsi positivi. Quindi sarebbe buono aggiungere una riga orizzontale che rappresenta il 5%, e/o mettere l'asse delle y in percentuale.
Inoltre più per fare quest'analisi ti serve un test quindi immagino usi edgeR, che però introduci dopo. Forse vale la pena introdurlo prima (oppure dire qui che usi edgeR riferendoti alla prossima sezione).

Figure 3.3.6: The figure shows the effects of different normalizations on the epigen omic differentially enriched regions.

Even if these normalization methods show good performances with this type of epignomic data, our investigations suggest that more testing is required, and maybe an ad-hoc normalization method for these data has to be developed.

The left panel represent the "null dataset" highlighting that portion of DERs due to randomness/bias. Indeed, any kid of normalization produces almost the same trend, underlying that *full quantile*, even if combined with *RUV-Seq* still not reduces the bias. While *upper quartile* preserves oscillations when using

7/8 samples. The one which seems to well interpret the data, producing a good compromise between bias and signal, is RUV-Seq. Indeed, it preserves a gradually downhill of the DER without totally flatten the signal.

A note to be accounted for is that in the case of the "null dataset" we had to set the RUV-Seq `k` parameter to 1, otherwhise we were no able to obtain any result. Spiega il motivo. E anche come hai scelto k nel full dataset.

To estimate the DERs, any of the RNA-Seq methods can be applied, such as *DESeq2*, *edgeR*, *NOISeq*, etc [17 19].

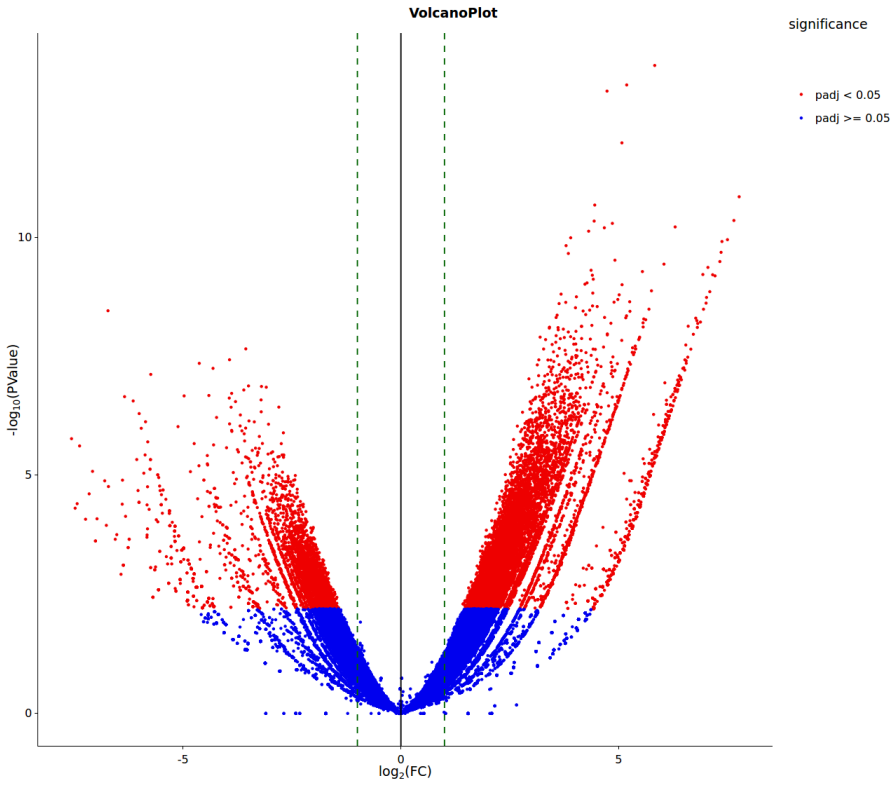


Figure 3.3.7: A volcano plot of Differential Enriched Regions. Blue dots represent the not significant DERs, while the red ones represent the significant DERs.

In this case, we decided to use *edgeR* package, because of its wide range of available statistical approaches and the possibility to better tune the design of the experiment. Indeed, because we used the RUV-Seq normalized counts with `k` parameter set to 4, we modeled the experimental design with the `model.matrix` function, adding to our model not only the experimental conditions, but also the RUV-Seq estimated factors. Then we used the resulted design to estimate the dispersion and fit a Quasi-Likelihood test, as defined in edgeR[17].

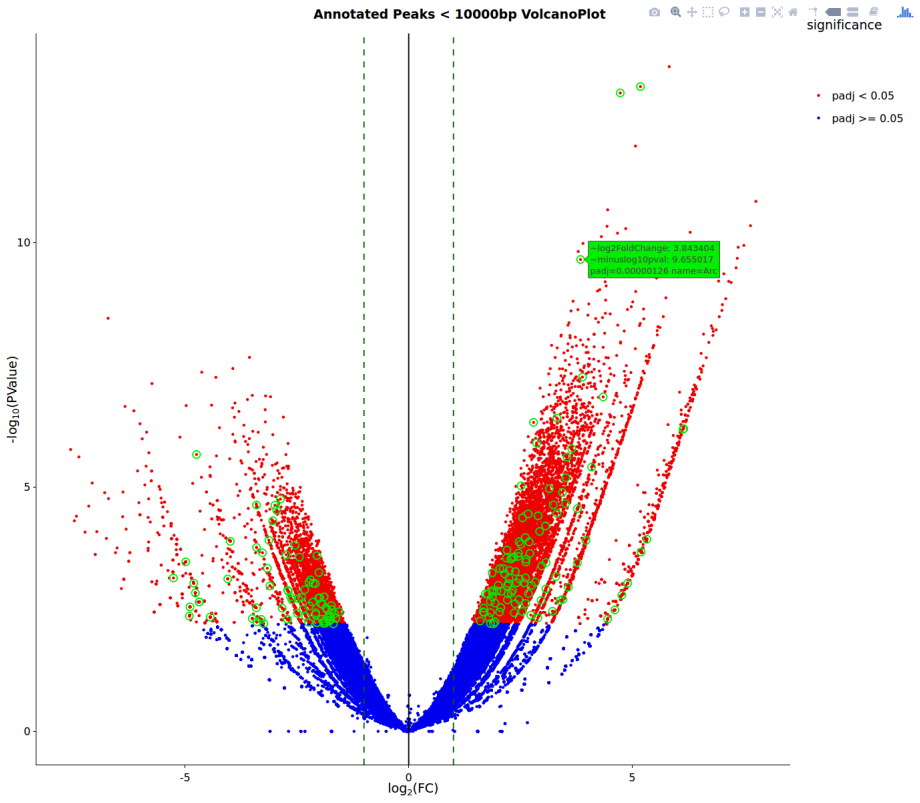


Figure 3.3.8: A volcano plot of DERs. Blue dots represent the not significant DERs, while the red ones represent the significant DERs. Green circles highlights the peaks with a DEG annotated.

figure 3.3.7 shows a volcano plot of DERs between E0 and E1 conditions. Red dots highlight the regions with a False Discovery Rate (FDR)[20] lower than 0.05, while blue dots highlight non significant regions.

Next task is to integrate the obtained results with other omic data types, as RNA-Seq. Because of the low number of samples, the easiest way to integrate the data is to annotate the DERs with DEGs resulting from the analysis of RNA-Seq.

For the differential expression of RNA-Seq data we firstly quantified the signal with the `featureCounts` methods available in the *Rsubread* [21] R/Bioconductor package. Then we filtered lowly expressed genes with the *proportion* test as implemented in *NOISeq* package, and applied the `noiseq` method for differential expression.

We used the resulting significant DEGs (with posterior probability higher than 0.95) to annotate the peaks with `annotatePeakInBatch` method of *ChIPpeakAnno*. Figure 3.3.8 illustrates with green circles the peaks with an annotated gene with distance lower than 10000bp from the gene Transcription Starting Site (TSS), producing a total of 430 annotated peaks. Realizing the plot with *ggplot2* combined with *plotly* library it is possible to enhance the names of the genes with a tooltip.

Then we used the annotated genes to do functional annotation on Gene Ontology (GO) [22, 23] and Reactome pathways, which showed several interesting results for the neuronal regulation.

Insert tables for the functional results, to discuss with Davide/Lucia

3.4 Conclusions and Future Works

In the lack of methodologies for open chromatin region detection and analysis, we developed a novel approach which, compared with very well known tools as *MACS2*, seems to be competitive in the detection of the signal,

We demonstrated to be able to catch not only signal, but also small re-

gions across the samples. And with our filtering/aligning step we demonstrated to be able to keep relevant signl producing data structures as *SummarizedExperiment* which are candidates to become standards in the biological data analysis. With our 3-steps analysis we put our tool at the top of a pipeline for open chromatin regions data analysis, proposing also a possible candidate for a standard analysis of this data type.

In the next future we plan to check if other distributions, as *Negative Binomial*, fit better this of data kind and to improve our filtering/aligning step with additional probabilistic methodology.

Chapter 7

Bibliography

1. Koberstein, J. N. *et al.* Learning-dependent chromatin remodeling highlights noncoding regulatory regions linked to autism. *Science Signaling.* ISSN: 19379145. doi:10.1126/scisignal.aan6500 (2018).
2. Auerbach, R. K. *et al.* Mapping accessible chromatin regions using Sono-Seq. *Proceedings of the National Academy of Sciences.* ISSN: 0027-8424. doi:10.1073/pnas.0905443106 (2009).
3. Ihaka, R. & Gentleman, R. R: A Language for Data Analysis and Graphics. *Journal of Computational and Graphical Statistics.* ISSN: 15372715. doi:10.1080/10618600.1996.10474713. arXiv: arXiv:1011.1669v3 (1996).
4. Gentleman, R. *et al.* Bioconductor: open software development for computational biology and bioinformatics. *Genome Biology.* ISSN: 1465-6914. doi:10.1186/gb-2004-5-10-r80 (2004).
5. Li, H. *et al.* The Sequence Alignment/Map format and SAMtools. *Bioinformatics.* ISSN: 13674803. doi:10.1093/bioinformatics/btp352. arXiv: 1006.1266v2 (2009).

6. Lawrence, M. *et al.* Software for Computing and Annotating Genomic Ranges. *PLoS Computational Biology* **9** (ed Prlic, A.) e1003118. ISSN: 1553-7358 (2013).

7. Zhang, Y. *et al.* Model-based analysis of ChIP-Seq (MACS). *Genome Biology.* ISSN: 14747596. doi:10.1186/gb-2008-9-9-r137 (2008).

8. Zhu, L. J. *et al.* ChIPpeakAnno: A Bioconductor package to annotate ChIP-seq and ChIP-chip data. *BMC Bioinformatics.* ISSN: 14712105. doi:10.1186/1471-2105-11-237 (2010).

9. Morgan M, Obenchain V, Hester J, P. H. SummarizedExperiment: SummarizedExperiment container. doi:https://doi.org/doi:10.18129/B9.bioc.SummarizedExperiment (2018).

10. Su, Y. *et al.* Neuronal activity modifies the chromatin accessibility landscape in the adult brain. *Nature Neuroscience.* ISSN: 15461726. doi:10.1038/nn.4494 (2017).

11. Edgar, R. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research.* ISSN: 13624962. doi:10.1093/nar/30.1.207 (2002).

12. Barrett, T. *et al.* NCBI GEO: Archive for functional genomics data sets - Update. *Nucleic Acids Research.* ISSN: 03051048. doi:10.1093/nar/gks1193 (2013).

13. Dobin, A. *et al.* STAR: Ultrafast universal RNA-seq aligner. *Bioinformatics.* ISSN: 13674803. doi:10.1093/bioinformatics/bts635. arXiv: 1201.0052 (2013).

14. Risso, D., Ngai, J., Speed, T. P. & Dudoit, S. Normalization of RNA-seq data using factor analysis of control genes or samples (RUVSeq). *Nature Biotechnology* **32,** 896 902. ISSN: 1087-0156 (2014).

15. Robinson, M. D. & Oshlack, A. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biology.* ISSN: 14747596. doi:10.1186/gb-2010-11-3-r25. arXiv: PMC2864565 (2010).

16. Dillies, M. A. *et al.* A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Briefings in Bioinformatics.* ISSN: 14675463. doi:10.1093/bib/bbs046 (2013).

17. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics (Oxford, England)* **26,** 139 140. ISSN: <null> (2009).

18. McCarthy, D. J., Chen, Y. & Smyth, G. K. Differential expression analysis of multifactor RNA-Seq experiments with respect to biological variation. *Nucleic Acids Research* **40,** 4288 4297. ISSN: 03051048 (2012).

19. Tarazona, S., García, F., Ferrer, A., Dopazo, J. & Conesa, A. NOIseq: a RNA-seq differential expression method robust for sequencing depth biases. *EMBnet.journal* **17,** 18. ISSN: 2226-6089 (2012).

20. Benjamini, Y. & Hochberg, Y. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological).* ISSN: 00359246. doi:10.2307/2346101. arXiv: 95/57289 [0035-9246] (1995).

21. Liao, Y., Smyth, G. K. & Shi, W. The Subread aligner: Fast, accurate and scalable read mapping by seed-and-vote. *Nucleic Acids Research.* ISSN: 03051048. doi:10.1093/nar/gkt214 (2013).

22. Gene Ontology Consortium. The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Research.* ISSN: 1362-4962. doi:10.1093/nar/gkh036 (2004).

23. Gene Ontology Consortium. Gene Ontology Consortium: going forward. *Nucleic acids research.* ISSN: 1362-4962. doi:10.1093/nar/gku1179 (2015).

Acronyms

DEG Differentially Expressed Gene. 27, 29

DER Differentially Enriched genomic Region. 24 29

DEScan2 Differential Enriched Scan 2. 15 17, 20 26, 45

FDR False Discovery Rate. 27

GEO Gene Expression Omnibus. 21

MLE Maximum Likelihood Extimator. 18

mm10 Mus Musculus Genome ver.10. 22

List of Figures