Sinle-cell ChIP-seq Data Mining

Institut Curie - DEpiC

March 20, 2019

Contents

1 Overview

The Shiny App described in this manual aims to perform Data Mining (basic statistics, unsupervised classification and differential analysis) on single-cell ChIP-seq data, in order to cross check data, cluster similar samples and identify potential loci from the genome presenting an overall over enrichment in protein H3K27me3 and other histone marks among some samples.
The application represents an analysis workflow with different steps that depend on each other (see figure 1). Thus, please make sure to perform each step in the intended order as intermediary results will be reloaded in downstream analysis. The application permits you to work on several data sets at the same time by saving all results obtained so far in a local directory of your computer. This way, you can continue the analysis on any saved data set at the point you stopped at the last time.

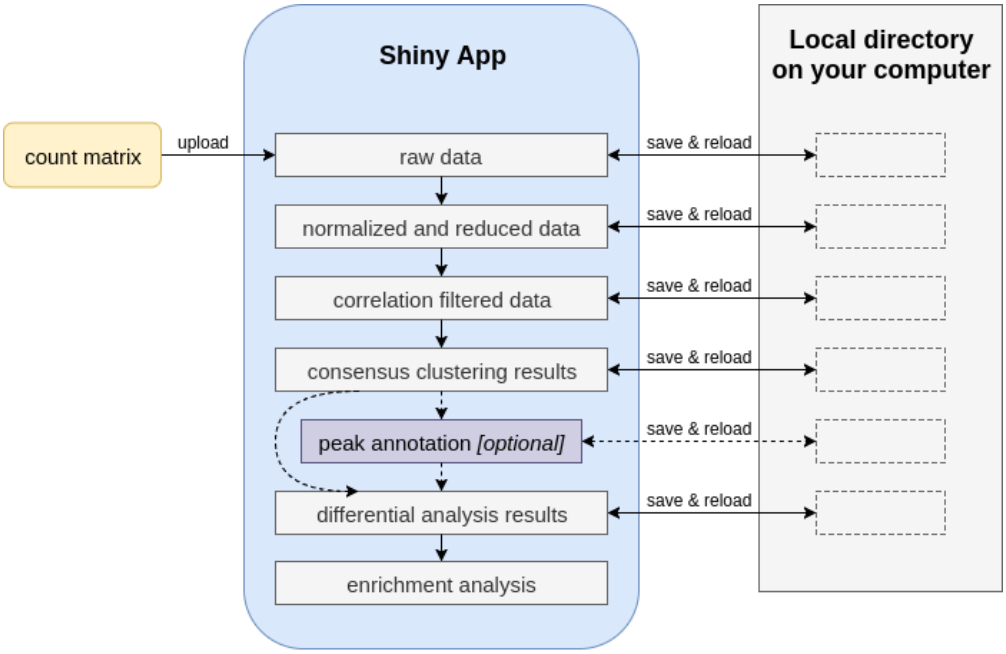

Figure 1: Data upload and saving scheme. After uploading the count matrix once, all intermediary results will be saved in a directory on your computer and will automatically be reloaded the next time you start the application.

2 Select or upload a new data set

On the front page, you can select the data set that you want to work on. It will be loaded on each of the subsequent pages (see figure 2).

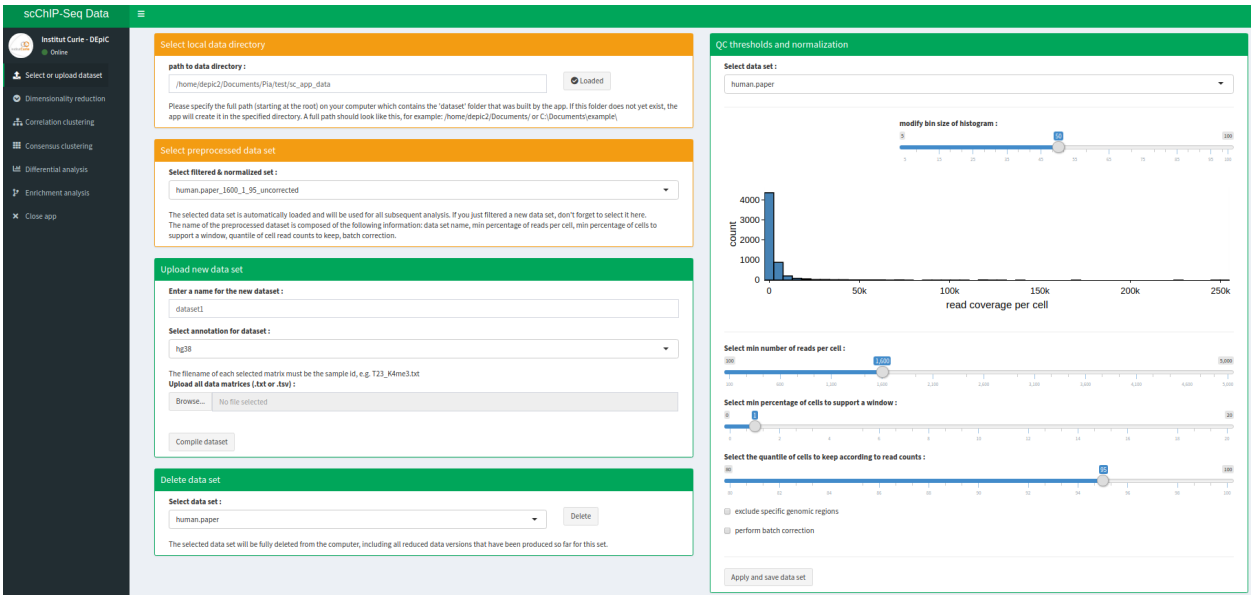

Figure 2: Front page of the application. This page is used to upload raw scChIP-seq data, perform normalization or select a previously saved data set for subsequent analysis.

2.1 Select local data directory

If this is the first time you are using the app, choose a directory on your computer where you want to save all the internal files produced by the application. Copy the whole path (starting from the root directory) in the input box in the top left corner of the page and click the loading button. The application will create a folder called 'datasets' in your specified directory and use it for storing all preprocessed data.
If you already worked with the app and want to continue working on a data set or upload new data, specify the path you chose the last time (the directory containing the *datasets* folder, not the *datasets* folder itself). Please make sure again to specify the complete path starting from the root.

2.2 Select a saved and normalized data set

Once a data set has been uploaded successfully, it will stay saved within the app and can be selected at any time. If it has already been filtered and normalized, it will appear in the box with the title *Select preprocessed data set*. Please note that this is the box where you select the data set to work on at all subsequent pages. This means that if you decide to upload and normalize a new data set, you still need to select it here after the normalization is finished so that it will be used for downstream analysis.

2.3 Upload a new data set

Uploading a new data set requires the following data files and information:

- **data set name:** Choose a descriptive name for the project that does not yet exist. Please use only letters, numbers, underscores and points. Avoid whitespaces.

- **annotation:** Choose the correct annotation (hg38 or mm10) that was used to generate the data.
- **Count matrices:** Upload one or multiple scChIP-seq count matrices.
  File specifications:
  - The file name must be the sample name, with the file extension .txt or .tsv (note that the sample name can only contain letters, numbers, underscores and points and should be as short as possible as it will appear in plot legends)
  - Columns correspond to cells (column names are cell barcodes), rows correspond to genomic regions (rownames are genomic regions)
  - Columns are separated by whitespace (blank)
  - Note that if you upload several matrices, the row names should match as only those rows will be kept which exist in all matrices. Row names that appear not in all matrices will be discarded.
  - Example:

	BC100011	BC100122	BC100125	BC100919
chr1:0-50000	0	3	7	2
chr1:50000-100000	0	5	2	3
chr1:100000-150000	0	0	1	0

After the upload is finished, the data set will appear in the selection in the box on the right side called *QC thresholds and normalization*. You can now select it there and specify how to filter and normalize it.

2.4 Filtering and normalization

Please consider the box on the right side of the page for this step. At the top, you can select a data set that was previously uploaded. Upon selection, the read coverage per cell of this data set will be visualized to give you a first impression of the data quality. The plot permits you also to zoom in and adjust the size of the histogram bins. This can be useful for deciding how to filter the data: you are able to set thresholds for minimum number of reads per cell, the minimum percentage of cells to support a genomic window and the quantile of cells to keep according to read counts (e.g. by selecting the 95 percent quantile you remove the top 5% of highest counts, which may represent errors).
In addition, in case that you uploaded more than one sample, you can specify whether batch correction should be performed. This is an important step if you know that the data generation is affected by differing conditions, because a lack of correction in this case would lead to the identification of differences that are not due to biological reasons but due to for example library preparation. However, it must be treated with caution as it could also hide biological differences when you declare samples from different biological conditions without batch effects as different batches. Once you decided which samples to group to batches,

specify the total number of batches in the field that appears after clicking the button for batch correction. You can now give a name to each batch and select one or more samples for each batch. Please make sure that a sample is not assigned to several batches.
In case you want to exclude certain genomic regions from the analysis (e.g. sites known to have copy number variations), you can click the checkbox *exclude specific genomic regions* and upload a .bed file containing the regions you don't want to include. The file should not have any row- or column names, a small example is shown here:

chr1	5528	5690
chr1	820305	841558
chr2	48901	50237
chrX	7823	7868

Please note that the start position (second column) must always be smaller or equal than the stop position (third column).
You can perform normalization several times for the same data set with different thresholds. This is useful if you are not sure which filtering works best, or if you want to compare the effects of different filtering on downstream analysis. All versions of the normalized data set will be saved within the app and can be selected in the box *Select preprocessed data set*.

2.5 Delete a data set

In case something went wrong while uploading or processing a data set or you don't need it anymore, you can select it here for deletion. This will delete the raw data set as well as all normalized versions of this set and downstream analysis results saved within the app.

3 Dimensionality reduction

The following page (see figure 3) in the application shows the data after dimensionality reduction by principle component analysis (PCA). In addition, it shows the data set after applying the tSNE algorithm. On top of each of the two plots, you can select according to which annotation you want the data points to be colored and add an annotation for each cell. For the PCA plot, you can also select two principle components that should be plotted against each other. As for many plots in this app, you can also zoom in by clicking in the plot and then dragging it to select a window you want to enlarge. Hover with the mouse over the plots to see further options, for example the possibility to download the image.

At the bottom of the page, you can manually change the colors used in the plot. You can either type the name of a specific color that is supported by R (e.g. *steelblue*), give a RGB hex color code (e.g. *#88E342*) or just click in the color sprectrum field that will pop up. By clicking the button *Save colors and apply to all* the colors you just selected will be saved in the app and reloaded everytime you select this data set. In addition, it will also be

Figure 3: Page for visualizing the data after dimensionality reduction by PCA and tSNE

used for the other plots, including the visualization of PCA, tSNE and the annotation for clustering heatmaps on the following pages.

The PCA and tSNE plots can be a good way to check if the data forms clusters in a way that you expect. For example, if your data contains batch effects and you have not corrected for it, you might observe very distinct clusters that are only defined by the samples, with even samples from the same condition being far apart from each other. On the contrary, if you have applied batch correction even though it was not appropriate, you might see that all structure in the data got lost. If you are not sure which preprocessing to apply, you can generate different filtered versions of the same data set (e.g. also with and without batch correction) on the start page of the application and then compare the plots on this page.

4 Correlation clustering

Using the dimensionality-reduced data as input, hierarchical clustering is performed on the cells. As it is beneficial for downstream analysis to keep only cells which show at least some minimum amount of correlation to other cells in the data set, another filtering step is applied here. On the right side of the page, the distribution of cell to cell correlation scores is plotted. For comparison, the distribution of the randomized data is also shown, as well as a correlation threshold quantile that you can set below. Select the minimum percentage of correlation to other cells in the data set or set it to zero in case you don't want to filter the cells, and click the button to filter the data and save it for further analysis. Note that saving the data is also necessary when the correlation threshold is set to zero. Upon filtering, the updated clustering heatmap will be shown below (see figure 7).

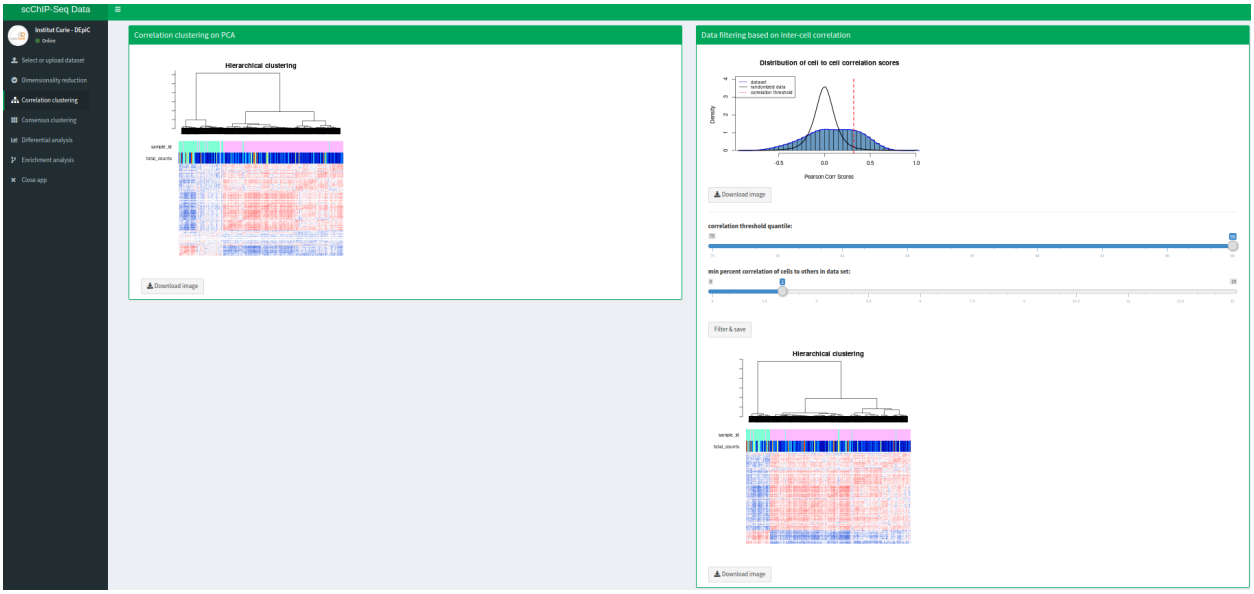

Figure 4: Page for visualizing the correlation clustering and filtering the data based on correlated cells.

5 Consensus clustering

Consensus clustering is applied in order to find a single clustering which is the best fit by considering a number of different (input) clusterings. In the plot, dark blue areas signify that these cells were always clustered together, whereas white areas mean that these cells are never in the same cluster and light blue areas mean that the cells are clustered together only in some cases.

5.1 Choosing the number of clusters

This step uses the data saved on the previous page. All filtered and saved versions will be available for selection in the box at the top left (see figure 5). After choosing one of these versions, click the button *Perform clustering* in the box below. The clustering is performed using the *ConsensusClusterPlus* function and will take some time. It is also saved locally so that it is not necessary to rerun it the next time. As soon as the clustering is finished, a PDF will be displayed that shows the plots for different numbers of clusters and some statistics which may help you decide for the optimal number of clusters. The Cumulative Distribution Function (CDF) of the consensus matrix for each k, where k is the number of clusters, is estimated by a histogram of 100 bins (see figure 6a). It allows to determine at what k the CDF reaches an approximate maximum, thus achieving a maximum consensus and cluster confidence. Figure 6b shows an example for the plot delta area plot. It represents the relative change in area under the CDF curve comparing k and k-1. As there is no k-1 for k=2, the total area under the curve is plotted for this case. This visualization allows you to determine the relative increase in consensus and pick k at which there is no appreciable increase. Finally, figure 6c shows the cluster-consensus value of clusters at each k, which is

the mean of all pairwise consensus values between a cluster's members. High values indicate a cluster has high stability and low values indicate a cluster has low stability.

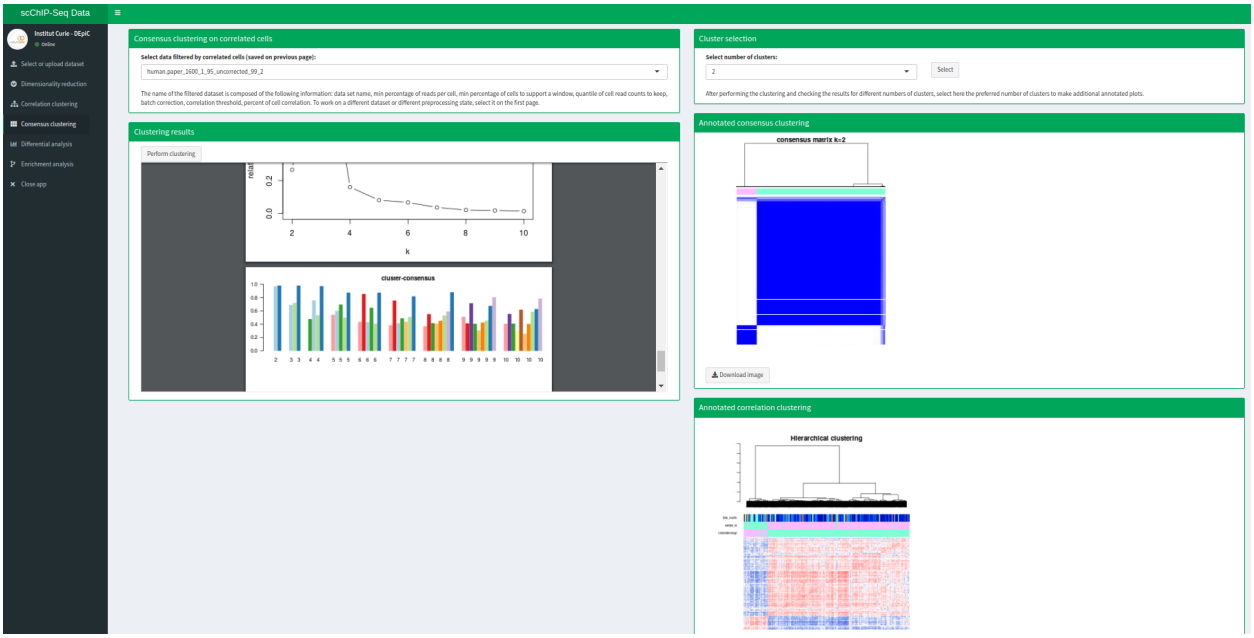

Figure 5: Page for performing consensus clustering on filtered data and choosing a clustering.

5.2 Cluster membership plots

After choosing the number of clusters, select it in the box on the right in order to generate plots annotated by the cell cluster identity (see figure 5). This includes the consensus clustering plot for the chosen k, the hierarchical clustering heatmap from the previous page, with an extra annotation row for the cluster identity, a table showing the number of cells of a sample per cluster and the tSNE plot from the second page of the application which can also be colored by cluster identity.

6 Peak calling

This step of the analysis is optional, but recommended in order to obtain more precise results in enrichment analysis. To be able to run this module, some additional command line tools are required (see section 10). You will also need the BAM files for the samples (one separate BAM file per sample). Note that the default interval size for genomic regions is 50kb, so we only included this window size for now, which means that your input data needs to have the same window size. This module can be executed once a clustering has been performed on the previous page. After providing the paths to the .bam files on your computer for each sample, select the number of clusters (only those numbers are shown which have been computed on the previous page). You can set the significance threshold for peak detection and decide if this threshold should be applied to the p-vaue or the q-value. Finally, press the

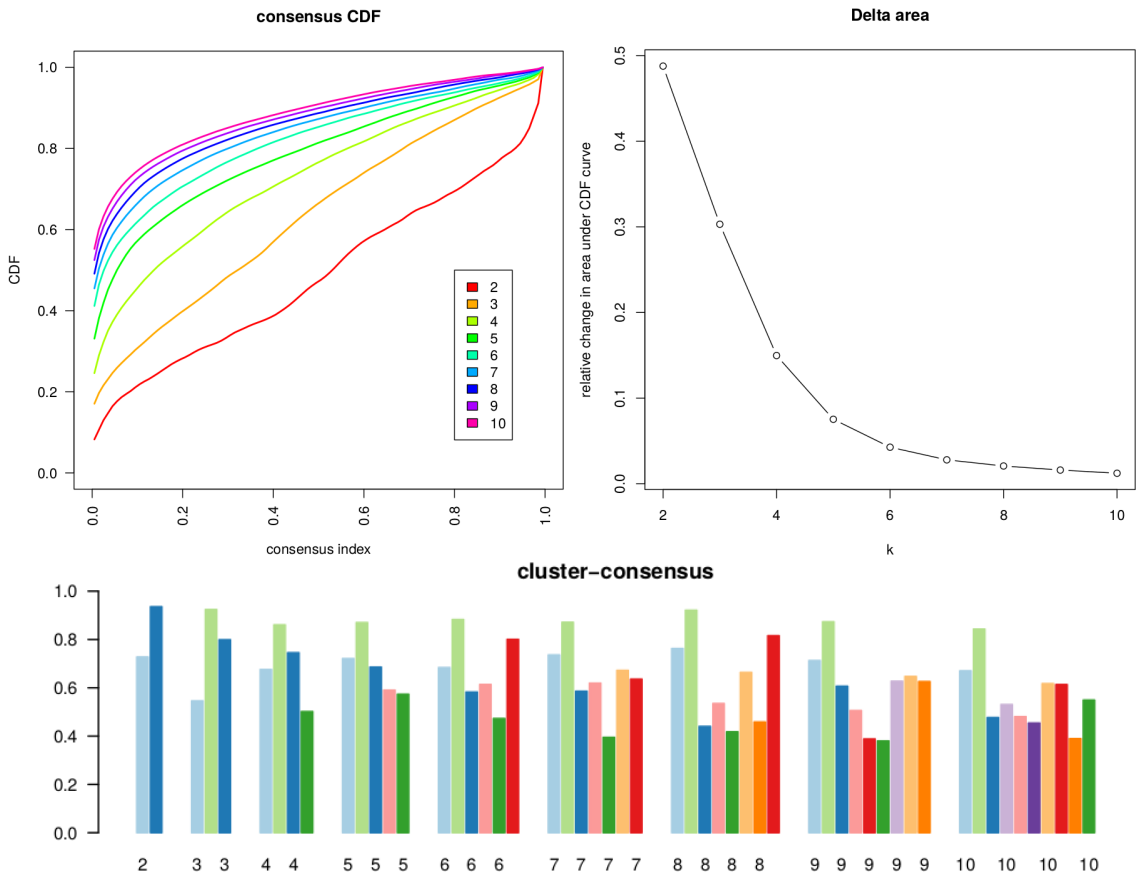

Figure 6: Consensus clustering statistics for choosing a clustering. (a) Cumulative distribution function (CDF) of the consensus matrix for each k. (b) Delta area (c)

start button. This will start the peak calling, which is done for each cluster separately by merging the corresponding cells per cluster. Peaks which are in close proximity (<1000bp) to a gene TSS will then be annotated with the corresponding gene name, which can be used later on in enrichment analysis to refine the mapping of read counts in a genomic interval to genes.

7 Differential analysis

After the unsupervised analysis on all the previous pages, we now start with the supervised part. Differential analysis will be performed using the cluster assignment obtained on the previous page. To use a different number of clusters, you need to go to the previous page and first perform the clustering, then select the preferred number of clusters in the box on the right in order to display and save the data. It will then appear here for selection in the box on the top left of the differential analysis page (see figure 8).

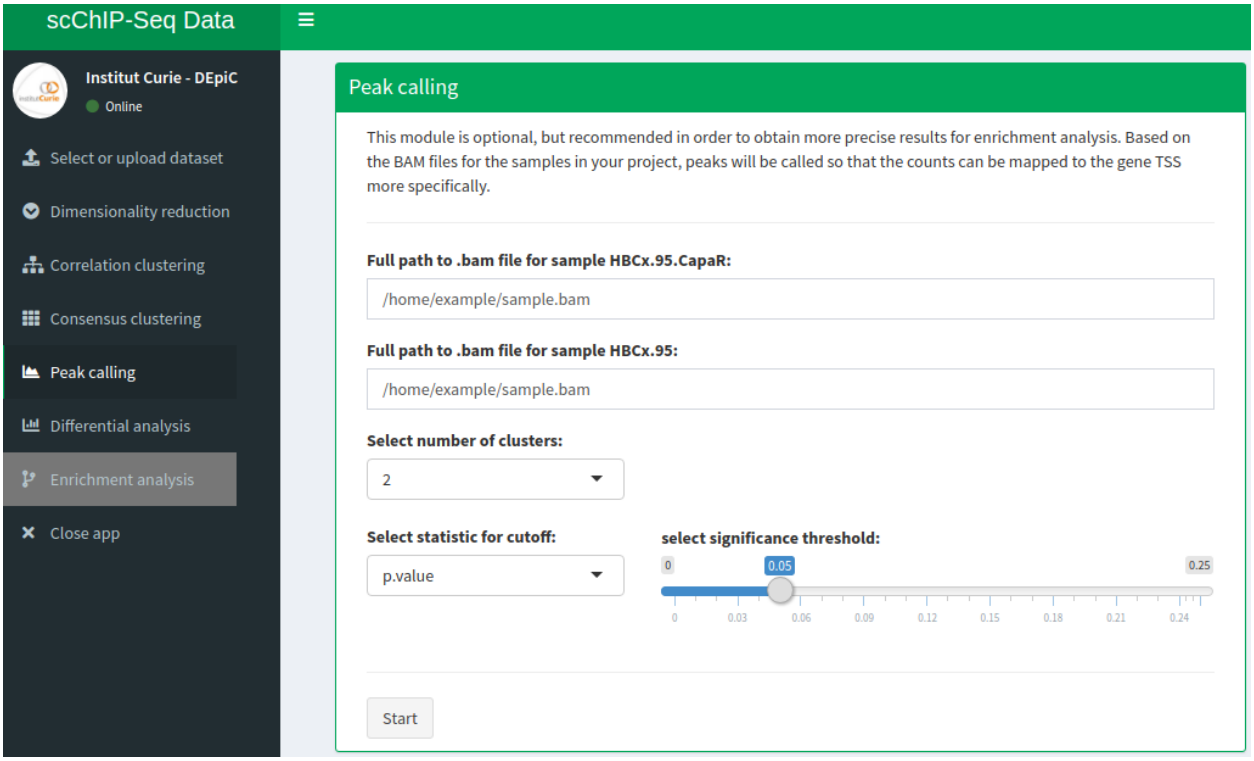

Figure 7: Page for performing peak calling and annotation to refine enrichment analysis later on.

7.1 Parameter selection

Other than for bulk sequencing data, we cannot use the *limma* method here to test for differential bound regions, as the counts per region in each cell are very low and almost have a binary character. This means that the counts are not normal distributed, which is required by *limma*. Instead, we chose to use the Wilcoxon rank test, which is a non-parametric statistical hypothesis test used to compare two related samples to assess whether their population mean ranks differ. It does not make any assumptions about the distribution of the data. Another issue that comes from the nearly binary looking read counts in our data is that taking ratios for computing the log fold change is very risky and does not give reasonable results for a lot of regions. To bypass this problem, we compute instead the mean difference of counts.

After having selected the desired number of clusters in the drop down menu, you can choose between two ways of differential analysis: first, the *one vs. rest* approach, which performs differential analysis for each cluster against the union of all other clusters, i.e. exactly one test per cluster. Alternatively, you can choose the *pairwise* approach, which means that each cluster is compared to each other cluster separately. As this results in multiple tests and thus p-values per cluster, we apply the Simes algorithm to combine the p-values into one p-value per cluster and take the mean of the count difference.

Below in the box, you can choose the desired minimum mean difference of counts and a q-value (BH-adjusted p-value) threshold to search for significant differential bound regions.

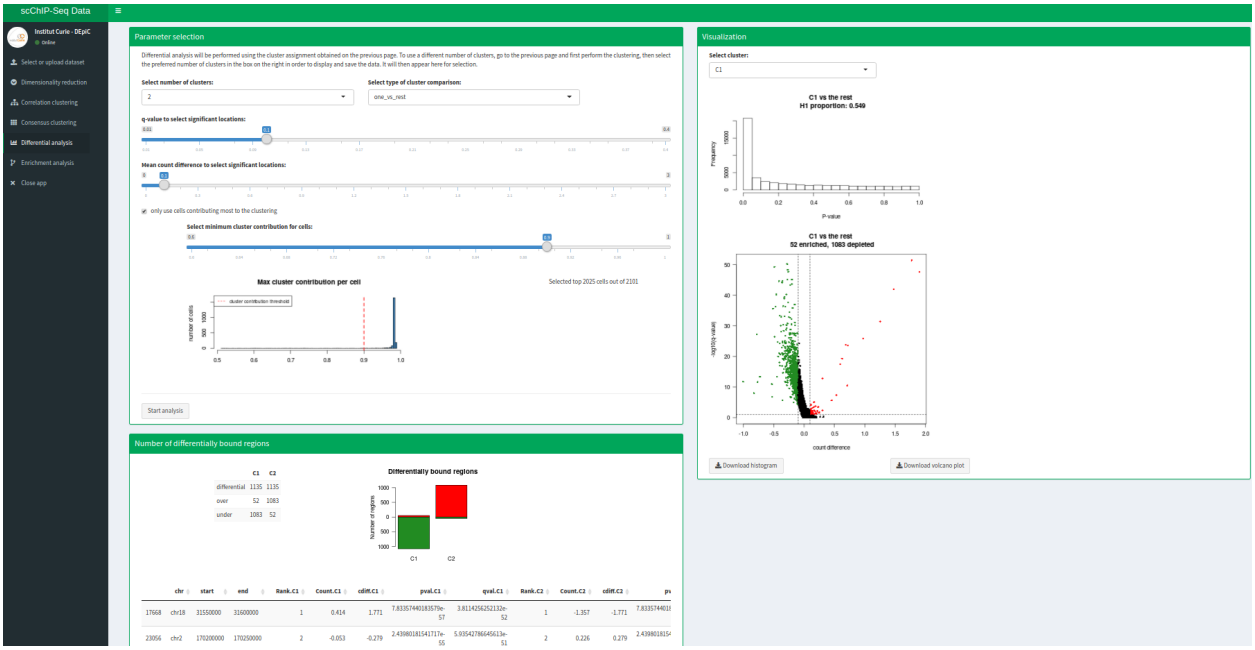

Figure 8: Page for performing differential analysis on genomic regions and visualizing the results.

An additional option is to use only those cells for differential analysis that contribute most to the clustering, i.e. they have a high contribution to one cluster instead of contributing a small part to multiple clusters. Just tick the option, and a selection for the threshold of cluster contribution will appear. Below the selection, you will see a histogram of the maximal cluster contribution of each cell which can help you choose a threshold. On the right side of the plot, the number of selected cells is shown.
To start the analysis, just click the button at the bottom of the parameter box. This will run the differential analysis, display the result table and some statistics and also save the results in your local directory so that it can be reloaded during your next session.

7.2 Result table and figures

Once the analysis is finished, two additional boxes will be displayed on the page (see figure 8). A small summary table shows the number of significantly differential bound regions per cluster according to the thresholds you selected, with an additional bar plot on the left visualizing this information. Below, you will find a table containing the count difference, p-value and q-value for each gene and each cluster. The box on the right shows information about the individual clusters. Just select a cluster at the top to update the plots. A histogram shows the proportion of the alternative hypothesis (H1) in contrast to the null hypothesis (H0). Below, you see a modification of the volcano plot, with the mean count difference on the x-axis and the $\log_{10}$ p-value on the y-axis.

8 Enrichment analysis

The significantly differential genomic regions determined in the previous step will be reloaded here and used as an input. Thus, please make sure that you have run the differential analysis before running the enrichment analysis on this page. If you executed the peak calling module earlier, a check box will appear to decide whether the peak annotation should be used for enrichment analysis. We recommend using this option, as it refines the mapping of read counts to gene TSS and thus gives generally more reliable results.

8.1 Annotation of genomic regions

In order to perform enrichment analysis on genes, it is necessary to select those genes that are possibly associated with the significant genomic intervals. This is done by annotating to each region the gene that is has the closest transcription start site (TSS), but only if the distance is less than 1000 bp. In case that there are several genes with the same minimum distance to the region (i.e. if the TSS of both overlap the region and thus have a distance of 0 bp), they are all annotated to this region. All genes associated with at least one significant region are then used as input for enrichment analysis using the gene sets from the *MSig* database.

8.2 Result table and figures

After clicking the button to start the analysis, a result table will be displayed below (see figure 9). You can select a cluster to see all significant gene sets, the number of genes in this set and the genes that were associated to differential genomic regions and made this gene set appear. On the right side, a tSNE plot is shown that is based on the normalized counts for all cells that were used for differential analysis. Above the plot, you can select a gene in the drop down menu. It contains all genes that are part of at least one enriched gene set. Next, you can select a genomic region associated to this gene. The plot will then automatically be colored according to the normalized count for this exact region. Please note that the tSNE will not be reperformed and is based on all regions, not just the selected one.

9 Close application

Although you can just close your browser window and shut down the application in the console or RStudio, it is recommended to close it using the *close* button on the last page of the app (see figure 10). This will remove temporary data files from the app, thus keeping it clean.

10 Installation Requirements

The application was built and tested using R version 3.5.1. We thus recommend to use this version in order to make sure that all required packages are available and can be loaded without issues. The following R packages are used for the application:

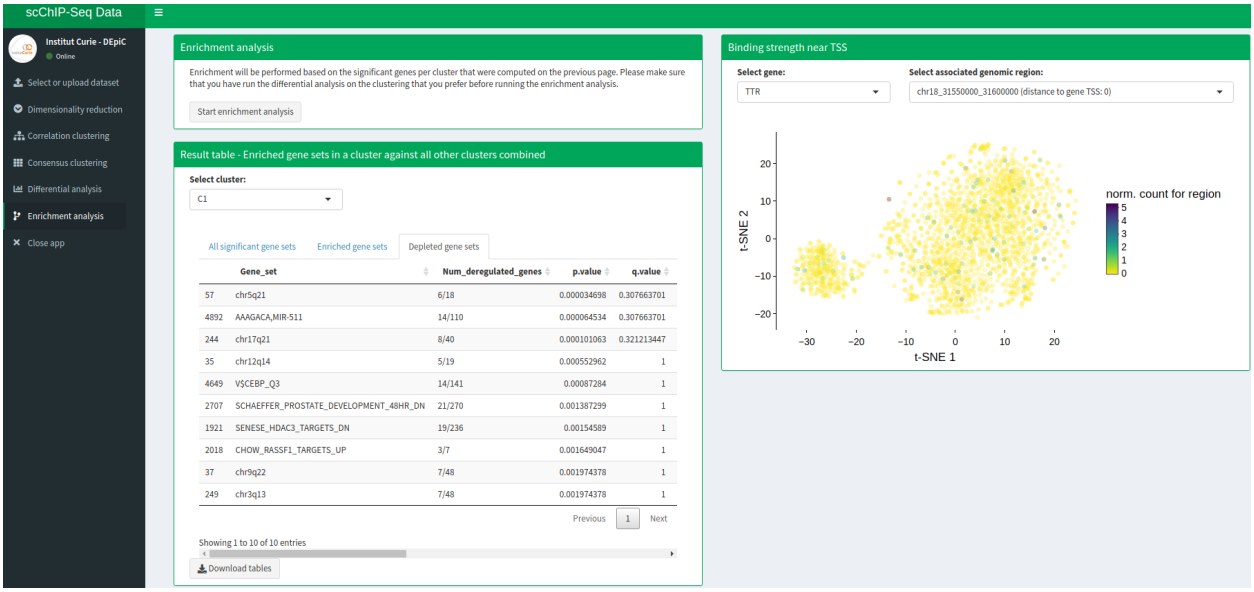

Figure 9: Page for running an enrichment analysis based on the differential bound regions identified on the previous page.

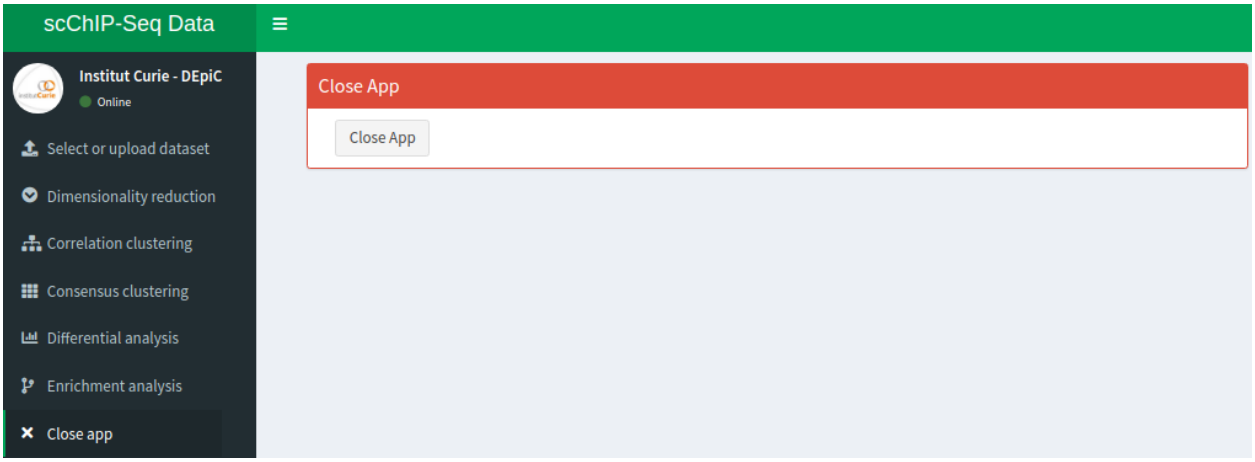

Figure 10: Close button to shut down the application and remove all temporary data.

- *scater*
- *scran*
- *edgeR*
- *ConsensusClusterPlus*
- *GenomicRanges*
- *IRanges*
- *tibble*

- *dplyr*
- *stringr*
- *irlba*
- *reshape2*
- *Rtsne*
- *DT*
- *tidyr*
- *splitstackshape*
- *rlist*
- *shiny*
- *shinydashboard*
- *shinyjs*
- *plotly*
- *RColorBrewer*
- *colorRamps*
- *colourpicker*
- *kableExtra*
- *knitr*
- *viridis*
- *ggplot2*
- *gplots*
- *png*
- *gridExtra*
- *wleepang/shiny-directory-input:* This needs to be installed from github as follows: *devtools::install_github('wleepang/shiny-directory-input')*
- *geco.utils:* This and the following three packages can be obtained from us and need to be installed from source as *install.packages("geco.utils.tar.gz")*
- *geco.visu*

- *geco.unsupervised*
- *geco.supervised*

In addition, some command line tools must be installed on your computer:

- **bedtools**: install and add it to your PATH variable so that it can be called by 'bedtools' from the command line without specifying the installation directory. This tool is required by the app and not optional.
- **deeptools**: *Optional: you only need to install this tool if you wish to apply the peak-calling module in the app.* Install it with pip (as it is a python module) and add it to your PATH variable.
- **samtools**: *Optional: you only need to install this tool if you wish to apply the peak-calling module in the app.*
- **macs2**: *Optional: you only need to install this tool if you wish to apply the peak-calling module in the app.*