ChIP-seq hands-on tutorial using Chipster

AllBio workshop 17.9.2014

Eija Korpelainen
chipster@csc.fi

Section 1: Check the quality of reads and align them to genome

In this section we check the quality of raw reads, try some trimming and filtering approaches, and align the reads to the reference genome.

1. Start Chipster and open a session containing a fastq file

Go to **http://chipster.csc.fi/**, and **Launch Chipster**. Log in (username chip, password training). Select **Open example session** and **course_rawReads.zip**. The session contains reads from GATA1 ChIP-seq of MCF7 cells. For the interest of time only a subset of reads is used.

2. Quality control with FastQC

Select file **reads.fastq.gz**, the tool **Quality control / Read quality with FastQC** and click **Run**. Inspect the results.

-how many reads are there and how long are they (fastqc_data.txt)? What quality encoding is used?
-is the base quality good all along the reads (per_base_quality.png)?

3. Try trimming and filtering approaches to remove low quality bases

Select file **reads.fastq.gz** and run the tool **Preprocessing / Filter reads for several criteria with PRINSEQ** so that you set the **minimum mean quality = 10.** Run also the tool **Preprocessing / Trim reads with Trimmomatic** with the following parameters:

Minumum quality to keep a trailing base = 5

Minimum length of reads to keep = 40

-Inspect the log files to see how many reads were filtered out / trimmed away.
-Run the FastQC tool on the preprocessed fastq files to see if the quality looks better now.

3. Align reads to reference genome using Bowtie2

Select **accepted.fastq.gz** and run the tool **Alignment / Bowtie2 for single end reads**.
-What was the overall alignment rate and how many reads have multiple alignments?

Section 2: ChIP-seq data analysis

In this section we call peaks, visualize them in genome browser, and filter them based on several criteria. We then retrieve the genes which are nearest to the peaks and perform pathway analysis for them. Finally, we also search for sequence motifs in the peaks and match the motifs to a database of known motifs.

Open example session course_STAT1.zip. The session contains two BAM files (aligned using hg18), one for the treatment sample (STAT1 IP) and one for the control sample (input DNA). For the interest of time only reads from chromosome 1 are included. Therefore we need to use the length of chromosome 1 in the following MACS runs.

-Open the treatment BAM file and check the length of chr1 in the header (2.47e8).

1. Detect binding locations for STAT1 with MACS2

Select the **treatment BAM** file and run the tool **ChIP-seq / Find peaks using MACS2, treatment only** using the following parameters:
Mappable genome size = **user-specified**
User specified mappable genome size = 2.47e8

-How many peaks do you get (check the top of the file macs2-peaks.tsv)?

-How many reads were there and how long are they? Were any of them duplicates? To what length were the reads extended to the 3' direction (see macs2-log.txt)?

-Is the model plot nice and smooth (see macs2_model.pdf)?

2. Detect binding locations for STAT1 with MACS2 using both treatment and control sample

Select both **BAM** files and run the tool **ChIP-seq / Find peaks using MACS2, treatment vs. control** so that you set the **Mappable genome size** as before and check at the bottom of the parameter panel that the files have been assigned correctly.

-How many peaks do you get now? How does the use of control sample affect the peak calling?

You can now delete the result files from exercise 1.

3. Visualize the peaks in the Chipster genome browser

-Open **macs2-peaks.bed** as a spreadsheet, click **Detach** and put the new window aside.
-Select **both BAM files**, **macs2-peaks.bed** and **macs2-summits.bed** and the visualization method Genome browser. **Maximize** the visualization panel, select **genome=Homo sapiens hg18**, and click **Go**.
-In the Settings tab, set the **coverage type = strand-specific** and **coverage scale = 250**.
-Sort the previously detached BED file by clicking on **Column4**. Click on the start positions of the peaks to navigate from one peak to another. Do the peaks have the bimodal shape expected?
-Zoom in and out with a mouse wheel.

4. Get the most significant peaks by filtering based on q-value

Select the file **macs2-peaks.tsv** and run the tool **Utilities / Filter table by column value** using the following parameter settings:

Column to filter by = neglog10qvalue
Cutoff = 10
Filtering criteria = larger than

-How many peaks pass the filter? **Rename** the result file to qfiltered.tsv

5. Filter out long peaks

Select the file **qfiltered.tsv** and run the tool **Utilities / Filter table by column value** using the following parameter settings:

Column to filter by = length
Cutoff = 1000
Filtering criteria = smaller than

-How many peaks do you have now? **Rename** the result file to length-filtered.tsv

6. Keep only high peaks

Select the file **length-filtered.tsv** and run the tool **Utilities / Filter table by column value** using the following parameter settings:

Column to filter by = pileup
Cutoff = 100
Filtering criteria = larger than

-How many peaks do you have now? **Rename** the result file to summit-filtered.tsv

7. Retrieve genes which are located closest to the peaks

Select the **summit-filtered.tsv** file and run the tool **ChIP-seq / Find the nearest genes for regions** so that you set the **genome** = **hg18**.

-Are all the peaks located upstream of genes (check the location column)?

8. Retrieve annotation for the nearby genes

Select **nearest-genes.tsv** and run the tool **ChIP-seq / Find unique and annotated genes**.

-How many unique and annotated genes did the list contain?

9. Pathway enrichment analysis using GO categories and ConsensusPathDB

Select **unique-genes.tsv** and run the tool **ChIP-seq / GO enrichment for list of genes**.

-What is the second most significantly enriched GO category in this list of genes?

Select **unique-genes.tsv** and run the tool **RNA-seq / Hypergeometric test for ConsensusPathDB.** Check if any pathways involving STAT were found: Select the result file **cpdb-pathways.tsv** and run the tool **Utilities / Filter table by column term** using the following parameter settings:

Column to filter by = Pathway
Term to match = STAT

10. Find sequence motifs which are common in the detected peaks

Select **summit-filtered.tsv** and run the tool **ChIP-seq / Find motifs with GADEM and match to JASPAR** so that you set **Genome = hg18**.

-Did the peaks have the STAT1 binding motif (check logo-plot-1.pdf)?

11. Save session, get analysis history file, save a workflow

Save session: Select **File / Save local session**. Give a name to your session and save it.
Get a textual report: Select **hypergeo-go.tsv** and click on the history link in the visualization panel.
Save an automatic workflow: Select the file **macs2-peaks.tsv** and **Workflow / Save starting from selected**.

Section 3: DNase-seq data analysis

In this section we call peaks in DNase-seq data using F-seq in different ways and compare the results. We visualize the peaks using the Chipster genome browser and the UCSC genome browser.

Open example session **course_DNase.zip**. The session contains one DNase-seq sample prepared from GM12878 cells using the UW double-hit method. The BAM file was aligned using hg19. For the interest of time only reads from chromosome 22 are included.

1. Detect DNase hypersensitive sites with F-seq

Open example session course_DNase-seq.zip. The session contains one DNase-seq sample from Gm12878 cells prepared using the double-hit protocol. Select the **BAM** file and run the tool **DNase-seq / Find broad peaks using F-seq** so that you set the **Fragment size** = **0**.

-How many peaks do you get (check the top of the file fseq-peaks.tsv)?

2. Detect DNase hypersensitive sites with F-seq using mappability data

Repeat the previous run so that you set also **Mappability data for background model = Human Hg19 35 bp**. **Rename** the result file to fseq-peaks-bff.bed.

-Does the usage of mappability data improve peak detection (you can answer this question also after exercises 3 and 4)?

3. Compare two BED files

Check how many peaks the previous result files have in common: Select both BED files and run the tool **Matching sets of genomic regions / Intersect BED** so that you set the parameters like this:

Write the original A entry once if any overlaps found in B = yes
Minimum overlap required as a fraction of A = 0.1

-How many peaks do the lists have in common?

4. Visualize F-seq peaks in Chipster genome browser

-Detach the file **intersectbed.bed** for navigation aid**.**

-Select **DnaseGm12878_chr22.bam**, **fseq-peaks-bff.bed, fseq-peaks.bed** and the visualization method Genome browser. Set **genome=Homo sapiens hg19**, and click **Go**.

-Navigate using the detached BED file. How do the F-seq peaks look like? How does the usage of mappability data in peak calling change the results?

5. Visualize F-seq peaks in the UCSC genome browser

-Select the file **fseq-peaks-bff.bed,** right-click on it, select **Export**, and save the file on desktop.

-Go to **https://genome.ucsc.edu/.** Select **Genome browser** on the left panel. Select **Add custom tracks**, browse to your file and click **Submit**. Click on **Add custom tracks** and copy-paste the url from your session (url.txt). Click **Go to genome browser**.

-Zoom out 100x and move around. Locate your custom tracks on the top and check how well your peaks match with the ENCODE data you imported.