Command-line ChIP-seq tutorial

19th of September, 2014

Endre Barta & Gergely Nagy

Basic UNIX commands

- *man:* show manual of a command
- *cd:* change directory
- *pwd:* print working directory
- *ls (-l) target_directory:* list the content of the target directory (default: present working directory)
- *mkdir (-p):* make directory (make parent directory(-ies) also)
- *cp:* copy (and rename) file(s)
- *mv:* move (and rename) file(s)
- *ln -s:* make symbolic link of a file
- *rm (-rf):* remove file(s) (in a force recursive manner, for directories)
- *cat file.txt; less file.txt:* show the content of file.txt
- *basename filename.ext (.extension):* get filename without path (and extension)
- *vi or nano file.txt:* edit the content of file.txt
- *wc -l (file.txt):* count the number of lines (of file.txt)
- *sort (-kn,n):* sort lines (by column n)
- *awk '{print}' (file.txt):* a data processing and reporting language
- *sh script.sh:* run a Shell script
- *for i in list; do commands; done* – for loop

Pipeline

1. Raw data: fastq (e.g. from the SRA database)
2. BWA / samtools: create aligned sequence data (bam)
3. MACS2: peak prediction (bed)

   or
4. Homer (uses own tag directory format)
   - Coverage file (bedgraph => tdf)
   - Peak prediction (bed)
   - Peak (and motif) annotation (tsv)

    - Read distribution plots (tsv => histogram, heatmap)
    - *de novo* motif enrichment (html, png, motif)

Downstream analysis

- BedTools:
    - Filtering based on overlaps
    - Creating merges => Venn diagrams
- DiffBind: differential binding analysis
    - Comparing peak size
    - Clustering

Login to *taito-shell.csc.fi* server by Putty setting X11!
Load modules for analysis:

```
module load cs-course
```

The precompiled course files are in the */wrk/trng24/coursefiles* directory.

1. Running basic analyses on ChIP-seq samples in command line by *ChIP-seq_anal-v1_9_CSC.sh*

1.1. Preparing directories, raw files and a sample list file

1.1.1. Search for the MCF7 H3K4me1 ChIP-seq sample submitted by Postech in the NCBI SRA database (in internet browser)! (*SRX115153*) Path: *ftp://ftp-trace.ncbi.nlm.nih.gov/sra/sra-instant/reads/ByExp/sra/SRX/SRX???/SRX??????*

How many reads were sequenced for this sample?

1.1.2. Create the following directory structure in your working directory (already on the server): (*mkdir (-p) directory_name*)

```
/wrk/trng??/ChIP-seq/list
/wrk/trng??/ChIP-seq/bigfiles/fastq
/wrk/trng??/ChIP-seq/analysis/hs_MCF7_CTCF/bam
/wrk/trng??/ChIP-seq/logs
/wrk/trng??/ChIP-seq/scripts
```

1.1.3. Copy *fastq.gz* files (or create symbolic link) to (the own) *bigfiles/fastq* directory! (*SRX100531, SRX476480*)

```
cd /wrk/trng??/ChIP-seq/bigfiles/fastq

ln -s /wrk/trng24/coursefiles/hs_HEPG2_CTCF.fastq.gz hs_HEPG2_CTCF.fastq.gz

ln -s /wrk/trng24/coursefiles/hs_THP1_CTCF.fastq.gz hs_THP1_CTCF.fastq.gz
```

1.1.4. Copy the *bam* file (or create symbolic link) to *analysis/hs_MCF7_CTCF/bam* directory!

```
cd ../../analysis/hs_MCF7_CTCF/bam
```

```
ln -s /wrk/trng24/coursefiles/hs_MCF7_CTCF.bam hs_MCF7_CTCF.bam
```

```
cd ../../.. (to /wrk/trng??/ChIP-seq)
```

1.1.5. Create *list/CTCF.lst* (*tsv*) file by ***nano*** or by any other text editor with the following order!

1. column: Sample_name (hs_cellline_CTCF/H3K4me1)
2. column: ftp_site (if needed)
3. column: antibody_type (factor/histone)

1.2. Sending the job

1.2.1. Copy job scheduler script to your own *scripts* directory!

```
cp /wrk/trng24/ChIP-seq/scripts/run_script.qsub scripts/run_ChIP-seq_anal.qsub
```

1.2.2. Type your email address in your copy of script!

1.2.3. Set the path of your working directory in the script! Type the following:

```
cd /wrk/trng??/ChIP-seq
```

1.2.4. Type the following command into the job scheduler script: (*nano*)

```
/wrk/trng24/coursefiles/ChIP-seq_anal-v1_9_CSC.sh `pwd`/list/CTCF.lst `pwd`/analysis `pwd`/bigfiles > logs/ChIP-seq_anal-v1_9_CSC.log 2> logs/ChIP-seq_anal-v1_9_CSC.err
```

```
((sh) script.sh list.lst analysis_directory bigfiles_directory)
```

1.2.5. Send the job to the SLURM scheduler!

```
sbatch scripts/run_ChIP-seq_anal.qsub
```

1.2.6. Check the status of the run!

```
squeue job_id
```

If the job was not properly started, you can delete it:

```
scancel job_id
```

2. Visualizing (the pre-ran) coverage and position data by IGV

All files must be downloaded to your computer!

2.1. Creating (*bai*) index file for a *bam* (in terminal)

2.1.1. Change directory to *analysis/hs_MCF7_CTCF/bam*! (*cd*)

2.1.2. Sort the *bam* file by the following command:

```
samtools sort -m 1000000000 hs_MCF7_CTCF.bam hs_MCF7_CTCF_sorted
2> sort.err &
```

2.1.3. Create index file by the following command:

```
samtools index hs_MCF7_CTCF_sorted.bam
```

2.1.4. Change back the directory! (*cd ../..*)

2.2. Loading bam files into IGV genome browser

2.2.1. To download files login to https://sui.csc.fi (in internet browser) and choose *My files*!

2.2.2. Open Integrative Genomics Viewer (IGV) and set hg19 genome assembly!

2.2.3. Download *hs_MCF7_CTCF_sorted.bam* and the belonging *bai* file and open the *bam* file!

2.2.4. Search for potential rSNPs! (*non-gray columns in the coverage part, e.g. chr9:107,505,887*)

2.3. Download and open the 4 *tdf* files (originated from bedgraph.gz by igvtools; *coursefiles/bedgraph_index*)!

2.3.1. Format the tracks in the left panel (right click):
Set data range e.g. to 0-100 and change track colors to make the samples more comparable!

2.3.2. Compare coverages of the *bam* and *bedgraph* (*tdf*) files!
What can cause the difference between them?

2.4. Comparison of Homer and MACS2 peak prediction

2.4.1. Download and open genomic coordinate (*bed*) files showing read enrichments predicted by Homer and MACS2! (*analysis/bed*)
*_CTCF_macs_peaks.bed
*_CTCF_macs_summits.bed
*_CTCF-homerpeaks.bed

2.4.2. Compare the exact positions determined by the two methods!

2.4.3. Download and open *coursefiles/CTCF_motifs.bed* containing all relevant CTCF binding sites throughout the human genome!

2.4.4. Compare these sites with summit predictions!

How can you explain the findings?

3. Creating proportional Venn diagram of the predicted CTCF peaks of MCF7, HEPG2, and THP1 cells

3.1. Getting human blacklist file

3.1.1. Find and download ENCODE ChIP-seq blacklist for human!

3.1.2. Rename to *hg19-blacklist.bed* and load in IGV!

3.1.3. Compare blacklist with the coverage files in IGV browser!

3.2. Filtering blacklist regions out

3.2.1. Create a *Venn* directory! (*mkdir*)

3.2.2. Copy blacklist file to server (https://sui.csc.fi) and subtract blacklist regions from MACS2 peak files (one by one or with a loop)!

```
intersectBed -a bed/hs_*_CTCF_macs_peaks.bed -b hg19-blacklist.bed -v > Venn/hs_*_CTCF_cleaned.bed
```

3.3. Running the loop producing the (VennMaster input) list file

3.3.1. Create a merge of the filtered files!

```
cd Venn
cat *_cleaned.bed | sort -k1,1 -k2,2g | mergeBed > CTCF_merge.bed
```

3.3.2. Count peak numbers (in *../bed* directory also)! (*wc (-l) *.bed*)

How many false matches were found by MACS2 compared to the blacklist in the 3 samples?

Which algorithm did find more peaks?

3.3.3. Create the list file for VennMaster!

```
for i in *_cleaned.bed; do b=`basename $i _cleaned.bed`; intersectBed -a CTCF_merge.bed -b $i -u | awk -v b=$b -F"\t" '{OFS="\t"; print $1":"$2"-"$3,b}'; done > CTCF_Venn.lst
```

3.4. Creating figure

3.4.1. Open Xming on your PC and VennMaster on the server (*vennmaster*) and load list file!

3.4.2. Set number of edges to maximum in Options menu!

3.4.3. Optimize the overlaps until the best approximation!

3.4.4. Change set colors!

3.4.5. Save picture, download and edit/label e.g. in Microsoft PowerPoint!

4. Creating read distribution heatmap (RD plot)

4.1. Creating *bed* file from the common CTCF binding sites

4.1.1. Create an RDplot directory!

```
cd ..
mkdir RDplot
```

4.1.2. Create the common peak set!

```
intersectBed -a Venn/hs_MCF7_CTCF_cleaned.bed -b
Venn/hs_HEPG2_CTCF_cleaned.bed -u | intersectBed -a stdin -b
Venn/hs_THP1_CTCF_cleaned.bed -u > RDplot/CTCF_common.bed
```

4.2. Run Homer for getting “coverage table”!

4.2.1. For the MCF7 H3K4me1:

```
annotatePeaks.pl RDplot/CTCF_common.bed hg19 -size 1500 -hist 15 -ghist
-noann -d hs_MCF7_H3K4me1/homer/hs_MCF7_H3K4me1 >
RDplot/MCF7_H3K4me1_heatmap.txt 2>
RDplot/MCF7_H3K4me1_heatmap.err &
```

```
(annotatePeaks.pl benchmark.bed genome -size whole_window -hist binsize -
ghist -noann -d tag_directory > table.txt 2> error.err)
```

4.2.2. For the MCF7 CTCF:

```
annotatePeaks.pl RDplot/CTCF_common.bed hg19 -size 1500 -hist 15 -ghist
-noann -d hs_MCF7_CTCF/homer/hs_MCF7_CTCF >
RDplot/MCF7_CTCF_heatmap.txt 2> RDplot/MCF7_CTCF_heatmap.err &
```

4.3. Creating (*cdt*) input files for TreeView

4.3.1. Login to *taito.csc.fi* server by Putty setting X11!

4.3.2. Open Gene Cluster! (*cluster*)

4.3.3. Open *txt* files and save them as *cdt* files!

4.4. Visualize data by Java TreeView!

4.4.1. Open TreeView (*treeview*) and then the *cdt* files!

4.4.2. Set pixel settings:
Fill the full area
Change to log scale
Change colors

4.4.3. Export to image without tree drawing (clear selection on the left by Ctrl-click)!

4.4.4. “Merge” images e.g. in MS PowerPoint!

4.4.5. Mark the size of the peak set common in the three samples! (*wc*)

4.5. Repeat 4.1-4. with CTCF binding sites sorted by the MCF7 peak scores

4.5.1. Sort the common CTCF peak set!

```
sort -k5,5gr RDplot/CTCF_common.bed > RDplot/CTCF_common_sorted.bed
```

4.5.2. Run Homer for the MCF7 H3K4me1 and CTCF tag directories! (similarly to 4.2.)

4.5.3. Create figure as described in 4.3-4.!

4.6. Creating average read distribution curve

4.6.1. Run the following command for both tag directories together!

```
annotatePeaks.pl RDplot/CTCF_common.bed hg19 -size 1500 -hist 15 -noann -d hs_MCF7_H3K4me1/homer/hs_MCF7_H3K4me1 hs_MCF7_CTCF/homer/hs_MCF7_CTCF > RDplot/MCF7_both_curve.txt 2> RDplot/MCF7_both_curve.err &
```

4.6.2. Download *MCF7_both_curve.txt* file and open it in MS Excel!
Use the second and fifth column to create the curves (smooth lined scatter)! (CTCF peaks shows a much higher coverage thus values should be devided by e.g. 10)
Set the axes and titles!

5. Checking Homer motif enrichment results

5.1. Download and open motif enrichment report of Homer!
hs_MCF7_CTCF/homer/hs_MCF7_CTCF_homermotifs_10_12_14_16

5.1.1. Open *homerResults.html* and check the p-value, enrichment and similar motifs of the top two matches!

5.2. Download and open the other CTCF motif enrichment results also!

5.2.1. Compare the results of the three CTCF samples!

Is there any significant difference between the motif enrichments?

5.2.2. Check the following files!

bed/hs_*CTCF-homer_top.bed

What is the overlap between the top1000 CTCF peaks of the three samples? (*intersectBed, wc*)

5.3. Searching for CTCF binding sites in the genome

5.3.1. Create an *MCF7_CTCF* directory for the annotation files!

5.3.2. Run the following command:

```
annotatePeaks.pl bed/hs_MCF7_CTCF-homerpeaks.bed hg19 -mbed MCF7_CTCF/MCF7_CTCF_motifs.bed -m hs_MCF7_CTCF/homer/hs_MCF7_CTCF_homermotifs_10_12_14_16/homerResults/motif1.motif -size 200 > MCF7_CTCF/MCF7_CTCF_motifs.txt 2> MCF7_CTCF/MCF7_CTCF_motifs.err &
```

```
(annotatePeaks.pl peakset.bed genome -mbed motif.bed -m matrix.motif -size of_search > annotation.txt 2> log.err)
```

```
cd ..
```

5.3.3. Download *MCF7_CTCF_motifs.bed* file and open it in IGV!

5.3.4. Compare CTCF motif files!

6. Checking statistical results of the run

6.1. Run the following command!

```
sh /wrk/trng24/coursefiles/get_tag_statistics-CSC.sh list/CTCF.lst analysis > logs/stats.log &
```

```
(sh script.sh list.lst analysis_directory)
```

6.2. Download and open *statistics_all.txt* file in MS Excel!

6.3. Set tab as column separator and compare the results!