ChIPP: A pipeline for short read ChIP-seq data analysis

1. Background

Chromatin immunoprecipitation followed by sequencing (ChIP-seq) enables the identification of *in vivo* protein-DNA interactions under a given condition or in a particular cell type. The important application involves transcription factor, chromatin remodelers, RNA polymerases binding site detection and histone modification detection.

A typical ChIP-seq experiment include four steps: 1) Crosslinking DNA binding protein to DNA *in vivo* by treating cells with formaldehyde. 2) Shear the chromatin with sonication into small fragments. 3) An antibody specific to the protein of interest is used to immunoprecipitated the DNA-protein complex. 4) Release the DNA by decrosslinking. The DNA fragments bound by the proteins are sequenced at finally.

2. Workflow

Sequenced DNA fragments information are stored in FASTQ file. ChIP-seq data analysis take these FASTQ file as input. This pipeline include mapping, qualification, deduplication, and peak calling(Figure 1).

2.1 Environment setup

To run this pipeline, a Linux operate system is required. Also Python3 is required for MACS3.

2.2 Mapping reads

DNA fragments sequencing information is recorded in fastq file as reads(Figure 2A). The first step of this pipeline is to map these reads to reference genome. Read is a sequence of DNA fragment from genome, by alignment it can be mapped to certain region of the genome. Each read will have a record called alignment in the alignment file: sam file. While mapping reads to reference genome, there might be mismatches, deletions or insertions which cause low mapping quality(Figure 2B). The mapping quality corresponding to the identity between the reads and the reference genome spot. Low quality reads may lead to false positive enrichment, they will been removed according to custom set quality value. Also, while mapping to reference genome, read can be mapped to several spots of genome, like reads from repetitive region, they are called non-uniquely mapped reads. As in this pipeline, only one of this alignments will be retained in the sam file. A chimeric alignment is represented as a set of linear alignments that do not have large overlaps. Typically, one of the linear alignments in a chimeric alignment is considered the "representative" alignment, and the others are called "supplementary" and are distinguished by the supplementary alignment flag. This step takes fastq file as input, and output alignment file: sam file which store the mapping information. Mapping is achieved by BWA.

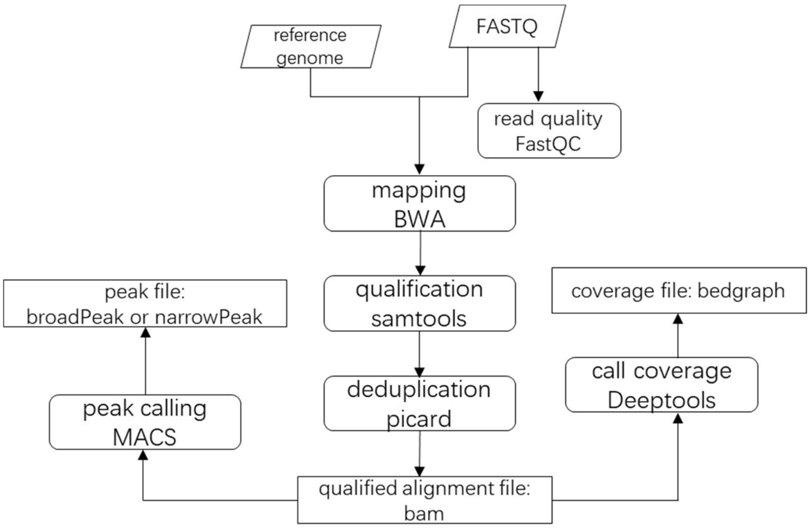

Figure 1. Workflow of ChIPP. ChIPP use BWA, samtools, picard, deeptools and MACS do the mapping, qualification, deduplication and call coverage and peak respectively.

2.3 Qualification

To improve the reliability of enrichment regions of interested protein, low quality alignments are removed at this step. DNA fragments number is the real enrichments situation of interested protein. Since representative alignments and supplementary alignments are from same fragment of chipped DNA, this pipeline only retain the representative alignments. This step takes sam file as input and output bam file, a binary format of sam file.

Qualification is remove records from alignment sam file using samtools.

2.3 Deduplication

While preparing the sample for sequencing, multiple copies of one fragment are produced. These reads are mapped to same genomic positions, which start and end at sample position, they are regarded as duplicates(Figure 2B). Also, while sequencing a single amplification cluster, incorrectly detected as multiple clusters by the optical sensor of the sequencing instrument. Although duplicates can possibly are different fragments, the chances are very low. So, duplicate will be treated as redundant alignments and therefore are removed to one alignment one position(Figure 2D). This step takes qualification bam file as input and output deduplication bam file, which is the final alignment file used for analysis. Duplicates are marked with picard and removed by samtools.

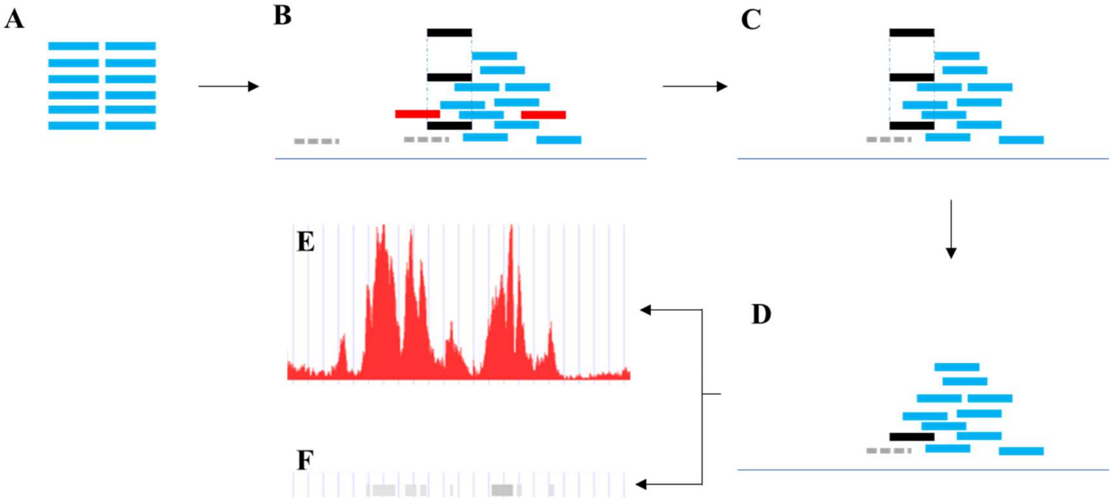

Figure 2. Process scheme. A. Raw reads in fastq file, they are the input and are mapped to reference genome. **B.** Mapped reads (alignments): line at the bottom represent the genome, red means low quality alignments, blue means good alignments, horizonal dotted lines represent non-unique map, only one of them is presented in the sam file; black means duplicated alignments. **C.** Qualification. **D.** Qualification. **E.** Coverage. **F.** Peaks. Peaks are significantly enriched regions, the level of gray is determined by score.

2.4 Call coverage

The coverage is calculated as the number of reads per bin, where bins are short consecutive counting windows of a defined size (Figure 3). Coverage is called from the final deduplicated bam file, shows the relative enrichment of fragments chipped by interested protein (Figure 2E). This step takes deduplicated bam file as input, using deeptools to call the coverage, generate a bedgraph file.

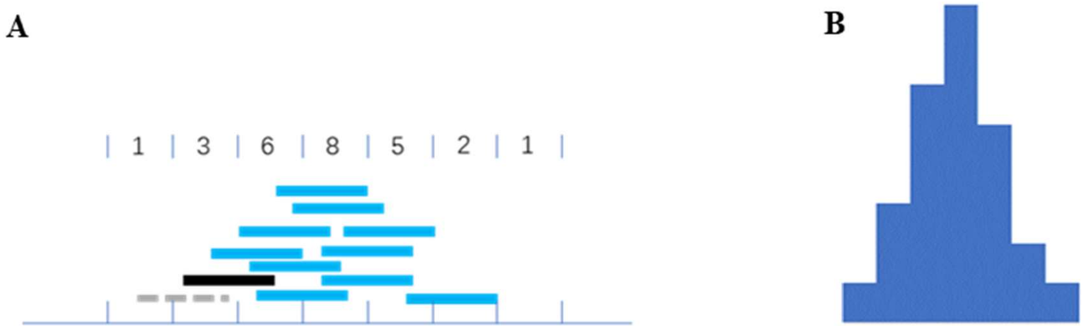

Figure 3. Scheme of coverage. A. Genome is separated into many bins, coverage is the number of reads fall into the bins. **B.** Coverage is visualized via histogram.

2.5 Peak calling

The peak-calling step identifies significantly enriched regions in the genome(Figure 2F) which called peaks. Peaks are represented by a start position and an end position with a score.

There are three types of enriched regions: sharp, broad and mixed. Sharp peaks are generally found for protein-DNA binding or histone modifications at regulatory elements, whereas broad regions are often associated with histone modifications that mark domains and there are no clear peak summits and sequence specificity, for example, transcribed or repressed regions. This pipeline provides narrow peak and broad peak calling.

2.6 Data visualization

Pipeline produced data include bam for alignment, bedgraph for coverage, R scripts for data statistic and bed file for peaks. Bam and bedgraph can be visualized on IGV. R scripts can be converted to graph.