PePr v1.1.10

Introduction

PePr is a ChIP-Seq Peak-calling and Prioritization pipeline that uses a sliding window approach and models read counts across replicates and between groups with a negative binomial distribution. PePr empirically estimates the optimal shift/fragment size and sliding window width, and estimates dispersion from the local genomic area. Regions with less variability across replicates are ranked more favorably than regions with greater variability. Optional post-processing steps are also made available to filter out peaks not exhibiting the expected shift size and/or to narrow the width of peaks.

Installation

1. Make sure your python version is higher than 2.6. Version 3.X may not be fully supported yet.
2. Install **pip** in your system if you don't have it.
3. `pip install PePr` or `pip install PePr --user`(if you don't have administrator privilege). Optionally, you can download tarball (PePr-[version].tar.gz) from github and install using `pip install PePr-[version].tar.gz`
4. If installation is successful, you could directly invoke the script by typing `PePr`. A help message will show up.

Supported File Formats

- Single-end: BED, BAM, SAM.
- Paired-end: BAM, SAM. The files must be sorted by the read names. Users can use `samtools sort -n sample.bam sample.sorted_by_name` to sort the file.

Basic Usage Examples

Warning: These are working examples with minimal required parameters. For the best performance (or to avoid bad fitting) on your data, please read this manual carefully and choose the right parameters.

- For peak-calling, run:

```
PePr -c chip_rep1.bam,chip_rep2.bam -i input_rep1.bam,input_rep2.bam -f bam
```

- For differential binding analysis with input samples, run:

```
PePr -c chip1_rep1.bam,chip1_rep2.bam -i input1_rep1.bam,input1_rep2.bam --chip2
chip2_rep1.bam,chip2_rep2.bam --input2 input2_rep2.bam,input2_rep2.bam -f bam
--diff
```

- For differential binding analysis without input samples, run:

```
PePr -c chip1_rep1.bam,chip1_rep2.bam --chip2 chip2_rep1.bam,chip2_rep2.bam -f bam
```

```
--diff
```

- To use a parameter file, run: `PePr -p parameter_file.txt`. For how to write a parameter file, see the section `Parameter File` below.

Parameters

Parameter	Description
-p/--parameter-file	Use parameter file instead of command line options. Using a parameter file will ignore all other command line options. See the next section for parameter file configuration.
-i/--input1	Group 1 input files. Multiple file names are separated by comma, e.g. input1.bam,input2.bam. you can also specify relative path to the file names, like folder1/input1.bam,folder2/input2.bam,folder3/input3.bam
-c/--chip1	Group 1 ChIP files.
--input2	Group 2 input files. Use in differential binding analysis.
--chip2	Group 2 ChIP files. Use in differential binding analysis.
-n/--name	Experiment name. It will be prefix to all output files from PePr. Default: "NA"
-f/--file-format	Read file format. Currently support bed, sam, bam, sampe (sam paired-end), bampe (bam paired-end)
-s/--shiftsize	Half the fragment size. The number of bases to shift forward and reverse strand reads toward each other. If not specified by user, PePr will empirically estimate this number from the data for each ChIP sample.
-w/--windowsize	Sliding window size. If not specified by user, PePr will estimate this by calculating the average width of potential peaks. The lower and upper bound for PePr estimate is 100bp and 1000bp. User provided window size is not constrained, but we recommend to stay in this range (100-1000bp).
--diff	Tell PePr to perform differential binding analysis.
--threshold	p-value cutoff. Default:1e-5.
--peaktype	sharp or broad. Default is broad. PePr treats broad peaks (like H3k27me3) and sharp peaks(like most transcriptions factors) slightly different. Specify this option if you know the feature of the peaks.
--normalization	inter-group, intra-group, scale, or no. Default is intra-group for peak-calling and inter-group for differential binding analysis. PePr is using a modified TMM method to normalize for the difference in IP efficiencies between samples (see the supplementary methods of the paper). It is making an implicit assumption that there is substantial overlap of peaks in every sample. However, it is sometimes not true between groups (for example, between TF ChIP-seq and TF knockout). So for differential binding analysis, switch to intra-group normalization. *scale* is simply scaling the reads so the total library sizes are the same. *no* normalization will not do normalization.
--keep-max-dup	maximum number of duplicated reads at *each single position* to keep. If not specified, will not remove any duplicate.
--num-processors	Number of CPUs to run in parallel.
--input-directory	where the data files are. The path specified here will be a prefix added to each of the files. The best practice is to always use absolute path in here.
--output-directory	where you want the output files to be. PePr will add this path as a prefix to the output files. It is recommended to use the absolute path.
--version	Will show the version number and exit.

Parameter File

The parameter file is an easier way of running PePr by including the running parameters in one file. It is effectively the same as running from the command line. A basic example is provided below:

```
#filetype       filename
chip1   chip_rep1.bed
chip1   chip_rep2.bed
input1  input_rep1.bed
input1  input_rep2.bed
file-format     bed
peaktype     broad
difftest     FALSE
keep-max-dup 2
threshold     1e-5
name    test
```

PePr will also output a complete parameter file for you to keep a record of your running parameters and produce the same results.

Output Files

- **NAME__PePr_peaks.bed**: A tab-delimited file containing chromosomal position of the peak, name, signal value, fold change, p-value and Benjamini-Hochberg FDR. Peak format is same as the ENCODE BroadPeak format.
- **NAME__PePr_[chip1/2]_peaks.bed**: this is the same as above, but only available when you run in differential binding mode. "chip1_peaks" are enriched in chip1, "chip2_peaks* are enriched in chip2.
- **NAME__PePr_parameters.txt**: A file containing the parameters to reproduce the results.
- **NAME-Date-Time-SessionID-debug.log**: This file contains the detailed information about the running status. Useful debugging information contains: Chromosomes analyzed, shift size and window size estimation, number of candidate windows, etc.

Links

- https://github.com/shawnzhangyx/PePr/ # Source code
- https://ones.ccmb.med.umich.edu/tags/PePr/ # PePr FAQ
- https://pypi.python.org/pypi/pepr #PyPI package index

Questions?

Questions are preferred to be posted on https://ones.ccmb.med.umich.edu/, you'll very likely find similar problems in there too (https://ones.ccmb.med.umich.edu/tags/PePr/?tab=Summary). You're also welcome to shoot me an e-mail at yanxiazh@umich.edu, I'll try replying to you as soon as possible.

Cite PePr

Zhang Y, Lin YH, Johnson TD, Rozek LS, Sartor MA. PePr: A peak-calling prioritization pipeline to identify consistent or differential peaks from replicated ChIP-Seq data. Bioinformatics. 2014.