June 15th, 2021

Alignment of Prox-seq sequencing data

Hoang Van Phan

Tay Lab, The University of Chicago

Alignment

For the first step, we have to extract the antibody identities, cell barcodes, and UMIs from the sequencing reads. There are currently two options, depending on whether the Drop-seq or plate-based version of Prox-seq was used.

Drop-seq:

```
java -jar PLA_alignment.jar ReadAlignmentDropSeq \
R1=read1.fastq.gz \
R2=read2.fastq.gz \
ABfile=barcode_cocktail.csv \
O=ReadAlignment_out.txt.gz \
SUMMARY=ReadAlignment_summary.txt \
HEADER=TRUE
```

Arguments:

R1 and R2: paths to the read 1 and read 2 fastq.gz files. Read 1 contains the cell barcodes and the UMIs, just like Drop-seq's read 1 files.

ABfile: path to the comma-separated file containing the protein target names (first column) and their barcode sequences (second column).

Example: CD3,AGTCGACA

O: path to store the alignment results

SUMMARY (optional): path to store the summary file. The default is the current working directory.

HEADER (optional): indicates if ABfile contains a header row to be discarded during processing. The default is FALSE.

Plate-based:

```
java -jar PLA_alignment.jar ReadAlignmentSmartSeq \
R1List=R1List.csv \
ABfile=/directory/path/NFKB_barcode_alignment.csv \
O=ReadAlignment_out.txt.gz \
SUMMARY=ReadAlignment_summary.txt \
HEADER=TRUE
```

Arguments:

R1List: path to the csv file containing the paths to single cell read 1 files, and their cell id. The csv file must NOT contain a header row.

Example: /path/to/cell_1_read1.fastq.gz,cell_1

ABfile, O, SUMMARY, and HEADER options: identical to ReadAlignmentDropSeq.

Cell barcode correction

This step is only required for Drop-seq data. The user either supplies a list of “correct” reference cell barcodes (CellBarcodeCorrection), or the program will merge all cell barcodes that are at most 1 Hamming distance apart (CellBarcodeMerging).

CellBarcodeCorrection

```
java -jar PLA_alignment.jar CellBarcodeCorrection \
I=ReadAlignment_out.txt.gz \
O=cellbarcode_out.txt.gz \
CELL_BC_LIST=referencebarcodes.txt.gz \
READCOUNT_CUTOFF=100 \
SUMMARY=cellbarcode_summary.txt \
HEADER=TRUE
```

Arguments:

I: path to alignment results (output of ReadAlignmentDropSeq).
O: path to store output file with cell barcode corrected.
CELL_BC_LIST: path to a comma-separated list of cell barcodes (second column) and their read counts (first column). This is the file produced by Drop-seq alignment tools (output of BAMTagHistogram function).
READCOUNT_CUTOFF (optional): the minimum read count for a reference cell barcode to be used. The default is 1000.
SUMMARY (optional): path to store summary file. The default is the current working directory.
HEADER (optional): indicates if CELL_BC_LIST has a header row to be discarded. The default is FALSE.

CellBarcodeMerging**need to double-check***

```
java -jar PLA_alignment.jar CellBarcodeMerging \
I=ReadAlignment_out.txt.gz \
O=cellbarcode_out.txt.gz \
SUMMARY=cellbarcode_summary.txt
```

Arguments:

I: path to alignment results (output of ReadAlignmentDropSeq).
O: path to store output file with cell barcode corrected.
SUMMARY (optional): path to store summary file. The default is the current working directory.

This function first builds a list of reference cell barcodes by merging all cell barcodes that are 1 Hamming distance apart, and keep the cell barcode with the highest read count. If a cell barcode matches with more than 2 cell barcodes at 1 Hamming distance, then this barcode is discarded. Then, the function proceeds as CellBarcodeCorrection.

Knee plot

This step is only optionally used for Drop-seq data. Here, we plot a knee plot (similar to Drop-seq alignment tools) to identify the true cell barcodes from PLA product sequencing data.

```
java -jar PLA_alignment.jar ReadcountHistogram \
I=cellbarcode_out.txt.gz \
O=cellbarcode_readcounts.txt.gz
```

Arguments:

- I: path to the cell barcode corrected file.
- O: path to export the tab-separated cell barcodes (second column) and their corresponding read counts (first column).

UMI merging

Now, we have to merge PLA products coming from the same single cell with similar UMI sequences together. Here, similar UMIs are defined as those with Hamming distance less than or equal to 1. This step is performed for both Drop-seq and plate-based versions.

Drop-seq:

```
java -jar PLA_alignment.jar UMIMerging \
I=cellbarcode_out.txt.gz \
O=umi_out.txt.gz \
SUMMARY=umi_summary.txt
```

Plate-based:

```
java -jar PLA_alignment.jar UMIMerging \
I= ReadAlignment_out.txt.gz \
O=umi_out.txt.gz \
SUMMARY=umi_summary.txt
```

Arguments:

- I: path to the input file.
- O: path to export the UMI merging results.
- SUMMARY (optional): path to store summary file. The default is the current working directory.

Export to digital count matrix

Finally, we export a count matrix containing the UMI counts of all single-cell PLA products. The count matrix format is PLA product by single cells.

Drop-seq:

```
java -jar PLA_alignment.jar DigitalCount \
I=umi_out.txt.gz \
O=count_matrix.txt.gz \
CELL_BC_LIST=knee_barcodes.txt \
SUMMARY=count_summary.txt \
DUPLICATE_EXPORT=duplicate_export.txt.gz \
REMOVE_DUPLICATE=TRUE
```

Plate-based:

```
java -jar PLA_alignment.jar DigitalCount \
I=umi_out.txt.gz \
O=count_matrix.txt.gz \
CELL_BC_LIST=NONE \
SUMMARY=count_summary.txt \
DUPLICATE_EXPORT=duplicate_export.txt.gz \
REMOVE_DUPLICATE=TRUE
```

Arguments:

I: path to the UMI merging results.

O: path to store the digital count matrix.

CELL_BC_LIST (optional): a list of chosen cell barcodes (from knee plot) (txt format). If NONE (default), export all available cell barcodes.

HEADER (optional): whether the CELL_BC_LIST has a header row, which will be discarded. The default is FALSE.

SUMMARY (optional): path to store the summary file. The default is the current working directory.

DUPLICATE_EXPORT: path to store the list of duplicated PLA products across single cells. These are the PLA products that have the same UMI in more than one cell. The default is the current working directory.

REMOVE_DUPLICATE: whether to remove duplicated PLA products across single cells. The default is FALSE.