User manual of JUMPp Batch

Center for Proteomics and Metabolomics

The Peng Laboratory

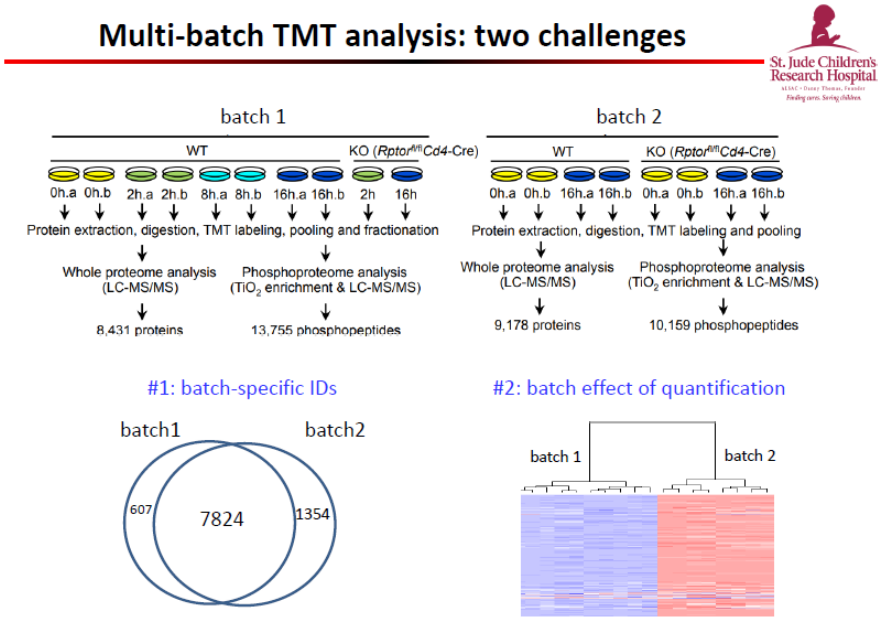
St. Jude Children's Research Hospital

June 2020

Contents

1. Introduction

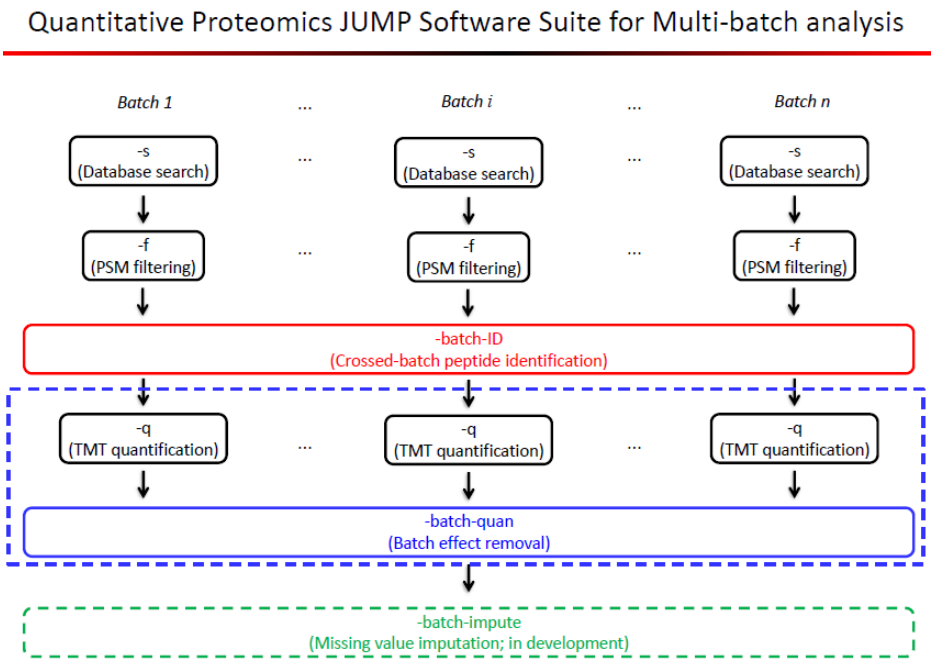
There are two kinds of challenges in multi-batch TMT analysis as below.



For challenge #1, we developed JUMPp -batch-ID by using peptides identified from other batches to rescue peptides in the current batch.

For challenge #2, we developed JUMPp -batch-quan by using internal standard or linear model fitting to normalize crossed-batch signals.

Here is the overall workflow for the multi-batch TMT analysis.



2. How to setup and run JUMPp Batch

2.1 Run JUMPp Batch

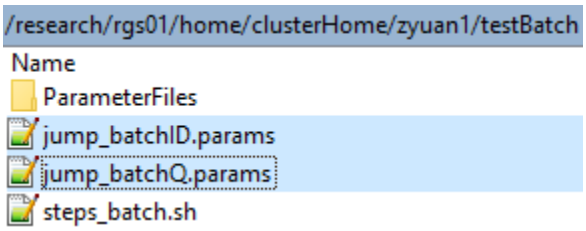
2.1.1 JUMPp Batch setup

Login HPC (PuTTY and WinSCP) and go to a work path (e.g. /home/zyuan1/testBatch).

a. In the work path, load batch params in PuTTY.
   module load jump/1.13.004
   jump -params

```
[zyuan1@splprhpc05 testBatch]$ module load jump/1.13.004
[zyuan1@splprhpc05 testBatch]$ jump -params
```

b. Edit batch params (i.e. jump_batchID.params, jump_batchQ.params) in WinSCP. Skip this step if you want to run the example data.



c. Run batch-ID in PuTTY (by default the results are in folder 'batch_id').
   jump -batch-id jump_batchID.params

```
[zyuan1@splprhpc06 testBatch]$ jump -batch-id jump_batchID.params
```

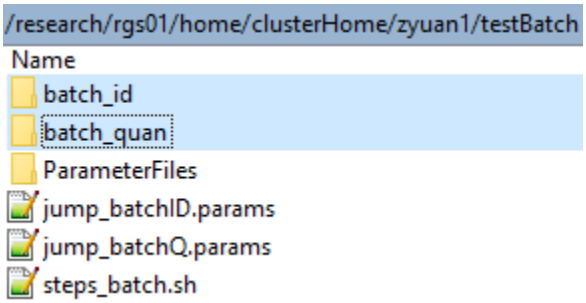
d. Run batch-quan in PuTTY (by default the results are in folder 'batch_quan').
   jump -batch-q jump_batchQ.params

```
[zyuan1@splprhpc06 testBatch]$ jump -batch-q jump_batchQ.params
```

Alternatively, step c & d can be run in one command: bash steps_batch.sh

```
[zyuan1@splprhpc06 testBatch]$ bash steps_batch.sh
```

Here show the results folders.



2.1.2 Edit batch params

jump_batchID.params

```
# Inputs: absolute path of publication tables from JUMP -f results (IDwDecoy.txt)
input_path_batch1 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/batch_test/b1/sum_HH_tmt/IDwDecoy.txt
input_path_batch2 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/batch_test/b2/sum_HH_tmt/IDwDecoy.txt

# thresholds:
min_Jscore = 10          # minimum Jscore cutoff to be considered (10 for JUMP; 1 for Comet)
multiHit_max_dJn = 0.1       # for considering non-top hits for a PSM: PSMs within such dJn range will be considered in a re

enable_group_specific_Jscore = 1    # enable group specific Jscore filtering. each group defined by charge states and pepti
score_cutoff_quantile = 0   # Jscore threshold for each group, defined as the lower quantile (default 5%) of Jscores in eac

# modifications
mods = 0             # Display modified peptides and their unmodified (0:Off, K:Lys, STY: Phosphorylation, ...); same as -f
#mods = STY          # Display modified peptides and their unmodified (0:Off, K:Lys, STY: Phosphorylation, ...); same as -f

# Output:
output_folder = batch_id        # output folder name

# other parameters:
jump_f_path = /research/rgs01/applications/hpcf/authorized_apps/proteomics_apps/pipeline/release/version1.13.003/JUMP/bin/b
# (in JUMP, pit_file = 0; in Comet, set pit_file according to search)
# pit_file = /hpcf/authorized_apps/proteomics_apps/database/20150201/human_ft_mc2_c57_TMT_K229.pit
pit_file = 0
database = 0

# HPC parameters
dispatch = localhost
```

jump_batchID.params: pay attention to the below params, and change others if you need.

(1) input_path_batch1, …, input_path_batchn: absolute path of publication tables from JUMP -f results (IDwDecoy.txt)

(2) min_Jscore: minimum Jscore cutoff to be considered (10 for JUMP; 1 for Comet)

(3) output_folder: output folder name (the default is batch_id). If it is changed here, remember to change it in ‘path_batch_id’ of ‘jump_batchQ.params’.

(4) pit_file: in JUMP, pit_file = 0; in Comet, set pit_file according to search

jump_batchQ.params

```
# JUMP batch correction parameter file
input_mode = 1                  # 1: proteins (for whole proteome); 2: pho site; 3: peptides (from either phosphor- or whole

# output path of -batch-id results
path_batch_id = /research/rgs01/home/clusterHome/zyuan1/testBatch/batch_id

# ATTENTION: jump -q will run automatically for all batches using DEFAULT parameters
# Otherwise, User can edit filtering options within ParameterFiles/TMThh/jump_qj_HH_tmt10_human.params to customize jump -q r

# Specify TMT-plex for each batch (that match to jump -batch-id results)
input_n_batch1 = 10
input_n_batch2 = 10

# Outputs:
output_folder = quan        # output folder suffix name; prefix always 'batch'

# Parameters:
normalization_method = 1            # 0: None (i.e., just combine publication tables); 1: using internal standard; 2: using l
isoform_rescue = 1              # 0: turn off; 1: turn on function. Suppose for a gene, there are two isoforms (say a and b)

# Internal standards for each batch. The most comprehensive batch should be put first
internal_standard_batch1 = sig126           # internal standard for each batch
internal_standard_batch2 = sig126           # internal standard for each batch

# -i parameters
jump_i_path = /research/rgs01/applications/hpcf/authorized_apps/proteomics_apps/pipeline/release/version1.13.003/JUMP/bin/bat

# jump -q parameters (the values here will overwrite the default values copied from the ParameterFiles/ folder above)
ppi_filter = 50                             # precursor peak intensity percentage threshold
impurity_correction = 1                         # 1 = Yes; 0 = No; if only a part of reporters are used, it should be set to
loading_bias_correction = 1                     # 1 = Yes; 0 = No;
interference_removal = 0                        # 1 = Yes; 0 = No;
```

jump_batchQ.params: pay attention to the below params, and change others if you need.

(1) path_batch_id: output path of -batch-id results, the same folder name as 'output_folder' of 'jump_batchID.params'.

(2) jump -q will run automatically for all batches using DEFAULT parameters. Users can edit filtering options within ParameterFiles/TMThh/jump_qj_HH_tmt10_human.params to customize jump -q runs.

(3) input_n_batch1, …, input_n_batchn: specify TMT-plex for each batch (that match to jump -batch-id results).

(4) internal_standard_batch1, …, internal_standard_batchn: internal standards for each batch if normalization_method = 1 (using internal standard).

(5) jump -q parameters (the values here will overwrite the default values copied from the ParameterFiles/ folder above)

Here we show 4 cases to edit parameters.

Case1: phos data, input_mode = 1

jump_batchID.params:

input_path_batch1 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/phos_test/b1/sum_HH_tmt/IDwDecoy.txt

input_path_batch2 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/phos_test/b2/sum_HH_tmt/IDwDecoy.txt

mods = 0

It can be edited: /ParameterFiles/TMThh/jump_qj_HH_tmt10_human.params

jump_batchQ.params:

input_mode = 1 # 1: proteins (for whole proteome)

input_n_batch1 = 11

input_n_batch2 = 11

Case2: phos data, input_mode = 2

jump_batchID.params:

input_path_batch1 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/phos_test/b1/sum_HH_tmt_mod/IDwDecoy_mod.txt

input_path_batch2 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/phos_test/b2/sum_HH_tmt_mod/IDwDecoy_mod.txt

mods = STY

It can be edited: /ParameterFiles/jump_l.params

It can be edited: /ParameterFiles/TMThhpho/jump_qj_HH_pho_tmt10_human.params

jump_batchQ.params:

input_mode = 2 # 2: pho site

input_n_batch1 = 11

input_n_batch2 = 11

Case3: phos data, input_mode = 3

jump_batchID.params:

input_path_batch1 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/phos_test/b1/sum_HH_tmt_mod/IDwDecoy_mod.txt

input_path_batch2 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/phos_test/b2/sum_HH_tmt_mod/IDwDecoy_mod.txt

mods = STY

It can be edited: /ParameterFiles/TMThhpho/jump_qj_HH_pho_tmt10_human.params

jump_batchQ.params:

input_mode = 3 # 3: peptides (from either phosphor- or whole proteome)

input_n_batch1 = 11

input_n_batch2 = 11

Case4: comet data, input_mode = 1

jump_batchID.params:

input_path_batch1 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/comet_test/b1/sum_HH_tmt/IDwDecoy.txt

input_path_batch2 = /hpcf/authorized_apps/proteomics_apps/pipeline/release/SampleData/comet_test/b2/sum_HH_tmt/IDwDecoy.txt

min_Jscore = 1

mods = 0

pit_file = /home/yli4/database/MS_proteomics/2017Feb_human_comprehensiveDB/humanComprehensive_v1_ft_mc2_c57_TMT_K229.pit

It can be edited: /ParameterFiles/TMThh/jump_qj_HH_tmt10_human.params

jump_batchQ.params:

input_mode = 1 # 1: proteins (for whole proteome)

2.2 Troubleshoot

(1) If you cannot use jump as below, run 'module load jump/1.13.004' first.

```
[zyuan1@splprhpc05 testBatch]$ jump -batch-id jump_batchID.params
-bash: jump: command not found
```

(2) If batch numbers of 'input_path_batch' in 'jump_batchID.params', 'input_n_batch' in 'jump_batchQ.params', and 'internal_standard_batch' in 'jump_batchQ.params' are not the same, the workflow will stop and ask you to make them consistent.

(3) If 'path_batch_id' of 'jump_batchQ.params' is not consistent with 'output_folder' of 'jump_batchID.params', it will stop and ask you to make them consistent.

(4) If you make changes in jump_batchID.params and jump_batchQ.params in the work path, jump_qj_HH_tmt10_human.params in /ParameterFiles/TMThh, the changes will be replaced if you rerun 'jump -params'.

(5) Please be patient when applying a compute node with required RAM as sometimes there is long waiting time.