Manual (for build v1.4)

Chih-Chiang Tsou (tsouc@umich.edu), Dmitry Avtonomov (dmitriya@umich.edu),
and Alexey Nesvizhskii (nesvi@umich.edu)

May 31, 2015

Table of Contents

INTRODUCTION **3**

DIA-UMPIRE MODULES **3**

SUGGESTED WORKFLOW **4**

DOWNLOAD INSTRUCTIONS **5**

REQUIREMENTS **5**

DIA-UMPIRE SIGNAL EXTRACTION MODULE (STEP A) **6**

INPUT (DIA-UMPIRE SIGNAL EXTRACTION MODULE) 6
Basic parameters (for parameter file diaumpire.se_params) *6*
Signal extraction parameters 6
DIA isolation window settings 7
Other parameters 8
Output files of DIA-Umpire signal extraction module *9*

UNTARGETED MS/MS DATABASE SEARCH (STEP B) **10**

DIA-UMPIRE QUANTITATION AND TARGETED RE-EXTRACTION MODULE (STEPS C AND D) **11**

INPUT (DIA-UMPIRE QUANTITATION AND TARGETED RE-EXTRACTION MODULE) 11
Basic parameters (for parameter file diaumpire.quant_params) *11*
OUTPUT (DIA-UMPIRE QUANTITATION AND TARGETED RE-EXTRACTION MODULE) 13

STEP BY STEP INSTRUCTIONS AND EXAMPLES **14**

GETTING STARTED, SETTING UP THE ENVIRONMENT 14
IN A NUTSHELL 16
PROCESSING SAMPLE DATASETS 17
Download the sample data *17*
Raw spectral data files conversion to mzXML *18*
Signal extraction (feature finding) using DIA-Umpire *20*
Untargeted MS/MS database search using X!Tandem and TPP *21*
Quantitation and targeted re-extraction analysis using DIA-Umpire *27*

CONTACT INFORMATION **27**

APPENDIX A: ADVANCED PARAMETERS FOR SIGNAL EXTRACTION MODULE **28**

Precursor-fragment grouping parameters *28*
Signal extraction parameters *28*

APPENDIX B: ADVANCED PARAMETERS FOR QUANTITATION MODULE 31

FDR estimation parameters 31
Quantitation parameters 31

Introduction

DIA-Umpire is an open source Java program for computational analysis of data independent acquisition (DIA) mass spectrometry-based proteomics data. It enables untargeted peptide and protein identification and quantitation using DIA data, and also incorporates targeted extraction to reduce the number of cases of missing quantitation. For more details about the algorithms used and performance evaluation, please refer to following DIA-Umpire publication.

- Chih-Chiang Tsou, Dmitry Avtonomov, Brett Larsen, Monika Tucholska, Hyungwon Choi, Anne-Claude Gingras, and Alexey I. Nesvizhskii, "DIA-Umpire: comprehensive computational framework for data independent acquisition proteomics," ***Nature Methods***, 2015.

DIA-Umpire modules

The overview of the complete workflow in DIA-Umpire is presented in Fig 1. The analysis starts with the signal extraction algorithm to detect all possible precursor and fragment ion features in MS1 and MS2 data, which detects monoisotopic masses and elution profile shapes. Precursor and fragment signals are then grouped based on correlation of their elution profiles (Step A). The tool generates "pseudo MS/MS spectra" (from MS1 features grouped with fragments) for untargeted MS/MS database search to identify peptides and proteins (Step B). Please note currently the analysis of MS/MS database search is not provided by DIA-Umpire, we recommend Trans Proteomics Pipeline (TPP) for the purpose. An optional step (Step C) allows addition of targeted identifications by taking confidently identified peptides from untargeted MS/MS search and building an internal spectral library. This library (built using all DIA data in the analyzed experiment) is then used for targeted extraction of protein quantitation information from each DIA run, resulting in improved identification/quantitation coverage across all samples. All IDs from either untargeted MS/MS database search or targeted re-extraction are linked to the corresponding precursor-fragment groups which carry quantitative information in the form of precursor and fragment ion intensities. This quantitative information is stored at different levels (fragment → peptide → protein) and is reported in Step D.

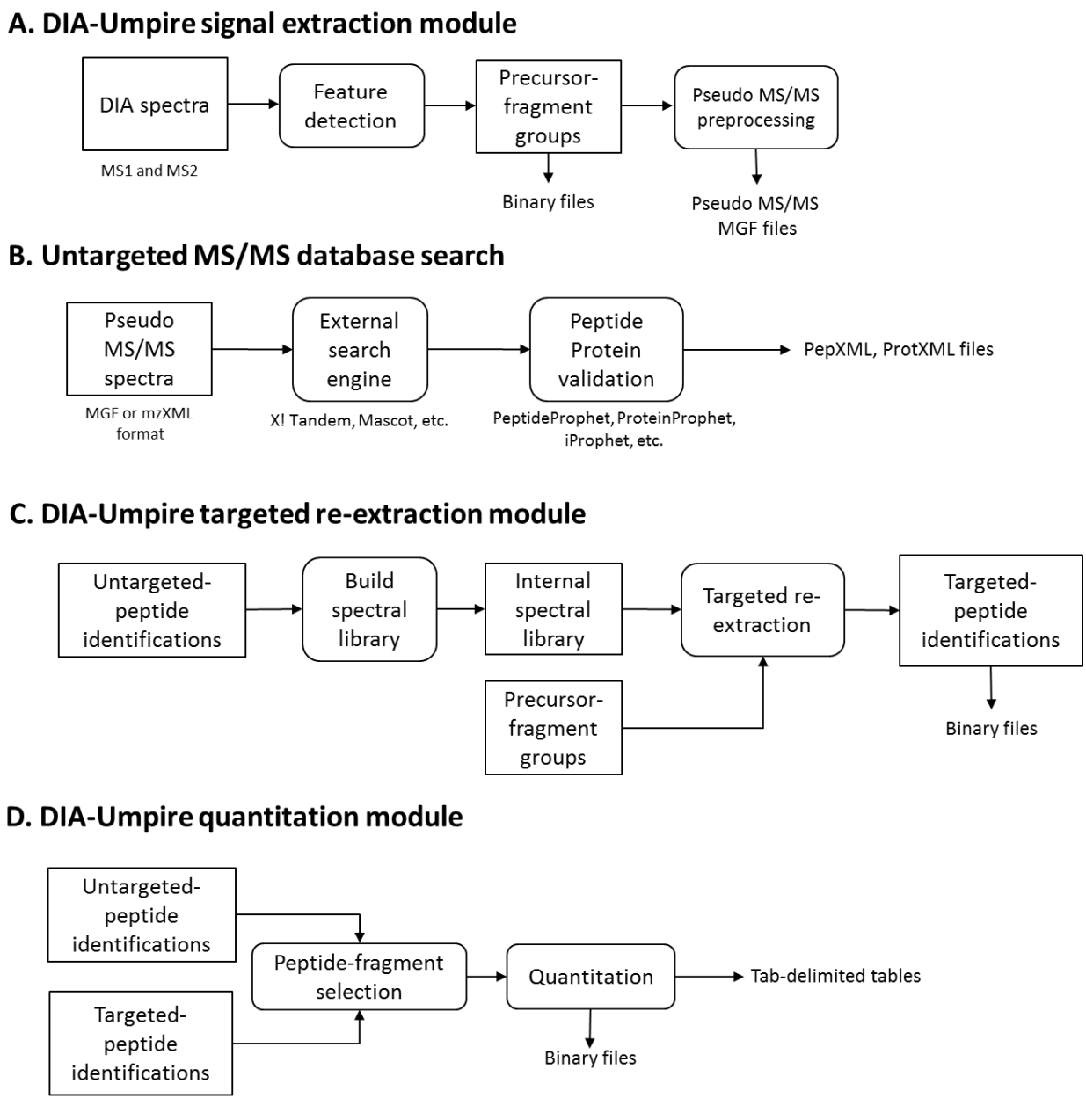


Figure 1. DIA-Umpire workflow.

Suggested workflow

Depending on the scale of applications, we describe here different application scenarios which require different combinations of DIA-Umpire modules.

1. Identification only analysis (Steps **A** → **B**): For users who need protein and peptide identifications and don't need to have quantitation analysis.
2. Small scale identification and quantitation analysis with minimal computational costs (Steps **A**→**B**→**D**): For users who have single or few samples to do identification and

quantitation analysis but do not wish to spend computational time on targeted re-extraction step, you can skip the step C and directly go to the quantitation step.

3. Complete DIA-Umpire identification and quantitation analysis (Steps **A→B→C→D**): For larger scale dataset with multiple replicates or samples, we recommend the complete analysis workflow.

Download instructions

Please visit http://diaumpire.sourceforge.net/ to download DIA-Umpire executable Java program. The zip file contains two executable Java JAR files (DIA_Umpire_SE.jar and DIA_Umpire_Quant.jar). You can also find parameter files at the website.

Requirements

DIA-Umpire is written in Java, which is cross operating system programming language. To execute DIA-Umpire, Java 7 or higher (download link: Java SE Runtime Environment 7) version is required. As a rule of thumb, it is recommended have at least double the amount of RAM as the average size of your *mzXML* files (*mzXML* written in 32-bit format without zlib compression). If *mzXML* is in 64-bit format, then RAM requirements should be approximately the size of the file.

DIA-Umpire signal extraction module (Step A)

DIA_Umpire_SE.jar provides the signal extraction module for DIA data (regular SWATH with fixed isolation window size, variable window SWATH, MSX) which generates pseudo MS/MS spectra to be searched against a protein database using conventional proteomics search engines such as X!Tandem, SEQUEST, MSGF+, OMSSA, etc.

Input (DIA-Umpire signal extraction module)

1. Spectral data in *mzXML* format

   ***Important*:** for AB SCIEX data, use AB SCIEX MS Data Converter (http://goo.gl/wf7KRV):

   Use it for *.wiff* -> *.mzML* conversion, then use MSConvert for *.mzML* -> *.mzXML*. Read "Raw spectral data files conversion to mzXML" section for more details.

2. Parameter file (An example "diaumpire_1.4.se_params" can be downloaded at http://goo.gl/J2f7uI)

Basic parameters (for parameter file diaumpire_1.4.se_params)

Here are basic parameters that the users usually need to modify accordingly based on their mass spectrometry instrument/experiment settings. Some other advanced parameters are described in Appendix A.

Signal extraction parameters

SE.MS1PPM: (Unit: ppm) Maximum mass error for two MS1 peaks in consecutive spectra to be considered signal of the same ion. Used in MS1 signal detection and precursor alignment between samples/runs.

Recommended value: Depends on the instrument. Typical values are 5-10 ppm for Thermo Orbitrap, 20-40pm for AB SCIEX Triple TOF 5600.

SE.MS2PPM: (Unit: ppm) Maximum mass error for two MS2 peaks in consecutive spectra to be considered signal of the same ion.

Recommended value: Depends on the instrument. If fragmentation spectra are measured with the same detector as MS1 spectra, set the same as Para.MS1PPM or a little higher, e.g. if you've set Para.MS1PPM=30 ppm for AB SCIEX Triple TOF 5600, consider setting to 40ppm.

SE.Resolution: Used only if the input spectra are stored in profile mode (i.e. not centroided, e.g. by using "Peak Picking" option in MSConvert when converting raw spectral data to mzXML format).

Profile spectra will be centroided using a sliding window. The window is moved across the entire mass range of a spectrum. Only the most intense peak in the window centered at the peak m/z is kept, others are discarded. The window width is calculated based on this parameter as: *width = mz / para.Resolution*.

Recommended value: Depends on the instrument and acquisition settings. Either check raw data to see the real average resolution of peaks in spectra or consult vendor specifications for the instrument. For AB SCIEX TripleTOF 5600 we use 15000-20000.

SE.StartCharge: The minimum charge state for MS1 precursor ion to be detected during isotopic peak grouping.

SE.EndCharge: The maximum charge state for MS1 precursor ion to be detected during isotopic peak grouping.

Recommended value: it is not recommended to set this parameter higher than 5 for typical proteomic experiments, as it is unlikely to observe peptides of higher charge states.

SE.MS2StartCharge: The minimum charge state for MS2 ***unfragmented precursor*** ion to be detected during isotopic peak grouping.

SE.MS2EndCharge: The maximum charge state for MS2 ***unfragmented precursor*** ion to be detected during isotopic peak grouping.

Recommended value: it is not recommended to set this parameter higher than 5 for typical proteomic experiments, as it is unlikely to observe peptides of higher charge states.

DIA isolation window settings

WindowType: DIA experiment type. DIA is implemented differently by different vendors and current support for data-formats is lacking, so the program needs additional info to properly interpret input spectral data.

Supported values in this version:

- SWATH - fixed window size SWATH, as described in the original SWATH paper. If you're using this option, it's mandatory to specify ***WindowSize*** option as well.
- V_SWATH - variable window size SWATH. If you're using this option, it's mandatory to specify ***Variable SWATH window setting*** (see section below).
- MSX – 2Da isolation window, its position is shuffled randomly until the whole MS1 range is covered, the process is then repeated but coverage of MS1 range by isolation windows will be different because of randomization.
- MSE – as originally implemented in Waters instruments. The full MS1 range is being fragmented at once.

WindowSize: Isolation window size setting for fixed window SWATH. (Please skip this part if the data is from Thermo instrument)

***Note*:** The window size is to be specified including overlapping regions. I.e. if your windows are: 399.5-425.5, 424.5 – 450.5, etc., then the window size should be set to 26.

Note: Was tested only on AB SCIEX TripleTOF 5600 and Thermo Q-Exactive and Fusion data.

Variable SWATH window setting: Isolation settings for variable window size SWATH. (Please skip this part if the data is from Thermo instrument). The format should be a tab-delimited list of m/z low and high values, one window per row. List begins with "==window setting begin" on a separate line and ends at "==window setting end". Example (2 windows: 400-451m/z and 449-600m/z):

```
==window setting begin
400	451
449	600
==window setting end
```

Other parameters

Thread: the maximum number of processing threads to be used.

ExportPrecursorPeak: set to ***true*** if you want detailed information about detected MS1 precursor and MS2 unfragmented precursor signals to be written to plain text file.

ExportFragmentPeak: set to ***true*** if you want detailed information about detected MS2 signals to be written to plain text file.

Output files of DIA-Umpire signal extraction module

1. Three .mgf files - pseudo MS/MS spectra sets for different quality categories of detected precursor signals (see the Online Methods of the publication for details). Example:

   *<filename>*_Q1.mgf
   *<filename>*_Q2.mgf
   *<filename>*_Q3.mgf

   Note: Each file corresponds to a different "quality level" of precursor ions (Q1= More than two isotopic peaks detected in MS1, Q2 = only two isotopic peak detected, Q3 = detected unfragmented precursor in MS2). These spectra are written to separate files, because they must be searched separately against a protein database as a consequence of differences in FDR estimates for these varying quality data.

2. Binary files (.ser) containing contain all necessary information for quantitation procedures (parameter settings, all detected precursor and fragment peaks, precursor-fragment grouping information).

3. If ***ExportPrecursorPeak*** and/or ***ExportFragmentPeak*** options were set to ***true***, text files with detailed information about detected MS1 and/or MS2 features will be generated.

Untargeted MS/MS database search (Step B)

As you can note from the diagram in Fig. 1, Step B does not involve DIA-Umpire directly, but is rather a standard proteomics peptide search and protein inference.

After signal extraction (Step A), generated .mgf files can be searched using conventional database search engines (X!Tandem, SEQUEST, OMSSA, Mascot, etc.) or using spectral library search engines such as SpectraST. We strongly recommend using msconvert.exe to convert .mgf files into mzXML format for the database search. Msconvert.exe is a part of ProteoWizard and Trans Proteomic Pipeline (TPP) packages, more info can be found here:

- ProteoWizard: http://proteowizard.sourceforge.net
- TPP: http://goo.gl/JAhvr5
- A tutorial for running X!Tandem search using TPP: http://goo.gl/gvD7KV

If you would like to use DIA-Umpire quantitation analysis, please do not rename the .mgf files and make sure that pep.xml search result files are named consistently, with the following format:

```
interact-<filename>_Q1.pep.xml
interact-<filename>_Q2.pep.xml
interact-<filename>_Q3.pep.xml
```

If the input files are iProphet results (which contains iProphet probabilities), DIA-Umpire uses iProphet probabilities instead of PeptideProphet probabilities to estimate peptide level FDR. For protein inference result, the quantitation module assumes one ProteinProphet file for the entire dataset. To get that, please perform ProteinProhet analysis in TPP using all pep.xml files from separate single-run searches.

Please refer to ***Step by step instructions and examples*** section of this manual for a detailed example of using X!Tandem and TPP for pseudo MS/MS spectra protein/peptide identification.

DIA-Umpire quantitation and targeted re-extraction module (Steps C and D)

DIA_Umpire_Quant.jar provides quantitation and targeted re-extraction analysis by taking results from Step A signal extraction and Step B untargeted MS/MS database search.

Input (DIA-Umpire quantitation and targeted re-extraction module)

1. Identification results: a .pep.xml result file for each .mgf file and a prot.xml for the entire dataset.
2. Protein sequence database in FASTA format which was used in ***Step B*** (untargeted MS/MS database search).
3. All files, including the binary files (.serFS) and .mgf files generated from the signal extraction module, as well as the mzXML files converted from mgf files.
4. Quantitation parameter file (An example "diaumpire.quant_params" can be downloaded at http://goo.gl/wThAVI)

Basic parameters (for parameter file diaumpire.quant_params)

Here are basic parameters that the users usually need to modify accordingly. Some other advanced parameters are described in Appendix B.

Path: The aims of working directory are two folds, one is that all output files for whole experiment level will be stored here (e.g. Internal spectral library file, output csv files). The second is to provide an easy way to assign all the files included for quantitation analysis. All mzXML files in the working directory which have been processed with the signal extraction module will be included in quantitation analysis (with or without targeted re-extraction).

Value: Absolute path of a folder

Alternatively, you can assign a list of files in parameter file, an example is shown below.

```
==File list begin
D:/DIA-Umpire_Test/UPS/LongSwath_UPS1_1ug_rep1.mzXML
D:/DIA-Umpire_Test/UPS/LongSwath_UPS1_1ug_rep2.mzXML
==File list end
```

TargetedExtraction: Whether to process targeted re-extraction across samples and replicates. Set it as false if you don't want to reprocess data but wish to export quantitation report based on different fragment/peptide selection options

(options described blew: FilterWeight, MinWeight, TopNFrag, TopNPep, and Freq)

Value: boolean (true/false) (default: true)

ExternalLibSearch (new parameter in v1.4): Whether to process targeted extraction across samples and replicates to research unidentified peptide ions from specified external spectral library. Peptide ions in external library will be research if it satisfies the two conditions. (1) unidentified from initial database search, and (2) unidentified or identified but the probability was lower than the specified threswhold described below. (Please note that this feature is still being tested, and contact us if you have any questions)

Fasta: Path to a protein sequence database in FASTA format which was used for untargeted MS/MS database search.

Combined_Prot: Path to the combined ProteinProphet .prot.xml file.

DecoyPrefix: Tag/prefix of decoy protein names that you used for protein database search.

Typical values: if you are unsure what that prefix was, check protein names in the FASTA file. “rev_” and “DECOY_” are common choices.

InternalLibID: Identifier for the internal spectral library.

If you are processing the dataset for the first time, it will be used as the name for the new library, if you are reprocessing data (e.g. using different thresholds/FDR levels, etc.) first a library with that name will be looked up and used if found.

Recommended value: you can use the same name for all analysis; however it is beneficial to provide unique meaningful names, to make the library more easily reusable.

ExternalLibPath (new parameter in v1.4): File path of external spectral library file. Currently only traML and custom binary (.serFS) formats are supported, and a decoy spectrum for each forward peptide ion sequence is required in the library file. (Effective only when ExternalLibSearch is set as true)

ExternalLibDecoyTag (new parameter in v1.4): Decoy tag of decoy spectra. (default: DECOY)

ReSearchProb **(new parameter in v1.4)**: Probability threshold to determine which peptide ion will be re-searched using external spectral library. (default: 0.5)

Output (DIA-Umpire quantitation and targeted re-extraction module)

a) Binary files which include identification and quantitation information, and possibly the internal spectral library.
b) Three summary tables for protein, peptide ion, and fragment level reports (***<filename>*** denotes the name of the raw file in which a peptide was identified):

1. Columns printed in protein summary table (ProtSummary.xls)
   1.1. **Protein Key**: Protein accession number
   1.2. ***<filename>*****_Prob**: Protein identification probability
   1.3. ***<filename>*****_Peptides**: Number of identified peptide ions assigned to a protein
   1.4. ***<filename>*****_PSMs**: Number of identified pseudo MS/MS spectra assigned to a protein
   1.5. ***<filename>*****_MS1_iBAQ**: Protein abundance estimated by MS1 peptide intensities (See manuscript for details) (iBAQ: sum of all identified peptide intensities divided by the number of theoretical tryptic peptides)
   1.6. ***<filename>*****_TopNpep/TopNfra, Freq>freq**: Protein abundance estimated by top scored peptide ions and fragments (See manuscript for details).

2. Columns printed in peptide ion summary table (PeptideSummary.xls)
   2.1. **Peptide Key**: Peptide ion identifier
   2.2. **Sequence**: Peptide sequence
   2.3. **ModSeq**: Peptide sequence with modification information
   2.4. **Proteins**: Parent proteins
   2.5. **mz**: Precursor m/z of peptide ion
   2.6. **Charge**: Charge state of peptide ion
   2.7. **MaxProb**: Maximum identification probability of peptide ion across the whole data-set from untargeted MS/MS database search
   2.8. ***<filename>*****_Spec_Centric_Prob**: Identification probability of a peptide ion from untargeted MS/MS database search
   2.9. ***<filename>*****_Pep_Centric_Prob**: Identification probability of a peptide ion from targeted re-extraction matching
   2.10. ***<filename>*****_PSMs**: The number of identified pseudo MS/MS spectra assigned to a peptide ion
   2.11. ***<filename>*****_RT**: Retention time of a peptide ion

2.12. ***<filename>*_MS1**: Peptide abundance estimated by MS1 precursor intensity
2.13. ***<filename>*_TopNfra**: Peptide abundance estimated by top N fragment ions

3. Columns printed in fragment summary table (FragSummary.xls)
   3.1. **Fragment Key**: Fragment ion identifier
   3.2. **Protein**: Parent protein accession number
   3.3. **Peptide**: Parent peptide ion identifier
   3.4. **Fragment**: Fragment ion type
   3.5. **FragMz**: m/z of fragment ion
   3.6. ***<filename>*_RT**: Retention time of parent peptide ion
   3.7. ***<filename>*_Spec_Centric_Prob**: Identification probability of peptide ion from untargeted MS/MS database search
   3.8. ***<filename>*_Pep_Centric_Prob**: Identification probability of peptide ion from targeted re-extraction matching
   3.9. ***<filename>*_Intensity**: fragment intensity
   3.10. ***<filename>*_Corr**: Elution profile Pearson correlation between fragment ion and precursor peptide ion
   3.11. ***<filename>*_PPM**: Mass error of an observed fragment m/z to the theoretical one

Step by step instructions and examples

Getting started, setting up the environment.

We will only be covering Windows setup in this tutorial. Please note that most of the processes used in the whole pipeline are also compatible with Linux system. (Because of the libraries for accessing raw spectral files from different instruments all are executable on Windows machine only, therefore the conversion of raw spectral files is not available on the machines other than Windows operating system)

1. Install Java 7 runtime environment (JRE) from the following link if you have not done so:
   - Link: http://goo.gl/gPaVR6
2. Open a terminal window.
   - Open Start menu, click “Run…” button. Or just press “Win+R” combination
   - Type cmd.exe, hit enter to start the terminal

3. Check your java installation. Run “java -version” command, output should be similar to the following, but will be different, depending on your version of Java and the JVM (do not type the “>” symbol, it represents the beginning of a command):

```
> java -version
java version "1.7.0_51"
Java(TM) SE Runtime Environment (build 1.7.0_51-b13)
Java HotSpot(TM) 64-Bit Server VM (build 24.51-b03, mixed mode)
```

4. Install ActivePerl v.5.16 (required by TPP), link: http://goo.gl/L4Xz2H
5. Install the latest version of TPP, link: http://goo.gl/ejnfUD
   Installation guide for TPP can be found here: http://goo.gl/Ckl5Bp
6. [**Optional**] Install ProteoWizard from:
   http://proteowizard.sourceforge.net/downloads.shtml

You will be asked to install .NET during the installation.
Having ProteoWizard is optional, because we will only be using the MSConvert program, which also comes with TPP installation.

7. Check that MSConvert command is recognized. Run “msconvert” command to see if you get any output:

```
> msconvert
Usage: msconvert [options] [filemasks]
Convert mass spec data file formats.
Return value: # of failed files.
Options:
  -f [ --filelist ] arg          : specify text file containing filenames
…
```

   a. If you don’t see the output, you will need to add “tpp-bin” folder of TPP installation to the environmental variables. To do that, go to *“Control Panel” -> “System” -> “Advanced system settings” -> “Environment variables”*, in the box “*System variables*” locate variable named PATH (or Path), click “*Edit*” and append the string with the path to “*tpp-bin*” (if you used the default installation path of TPP, it is “*C:\Inetpub\tpp-bin*”) using semicolon as a separator.
   b. Restart the command shell and try the command again.

In a nutshell

A quick summary of DIA-Umpire usage (see Fig.2):

1. Convert raw spectral data to mzXML using ProteoWizard or any other converter
    a. For AB SCIEX instruments first use AB_SCIEX_MS_Converter to convert from *.wiff* to *.mzML* (use "-centroid" option), then convert *.mzML* to *.mzXML*.
2. Run DIA-Umpire signal extraction
    a. Run the signal extraction module (DIA_Umpire_SE.jar) on mzXML files.
    b. The module produces three separate .mgf files for every input mzXML file (corresponding to Q1, Q2, Q3 quality tiers).
    c. Convert all .mgf files to mzXML to prepare them for database search
3. Run the database search:
    a. Identify peptides for each run separately, using a search engine of your choice or multiple search engines capable of producing .pep.xml output (X!Tandem, Mascot, Comet, MS-GF+, etc.).
    b. Validate identifications with PeptideProphet (part of TPP) to get interact-<filename>.pep.xml files
    c. Infer proteins to get a single .prot.xml file from all interact-<filename>.pep.xml files using ProteinProphet. (if the pep.xml files are iProphet results, please specify "IPROPHET" option when you run ProteinProphet)
4. Run DIA-Umpire quantitation module to get quantitation tables for peptides and proteins.

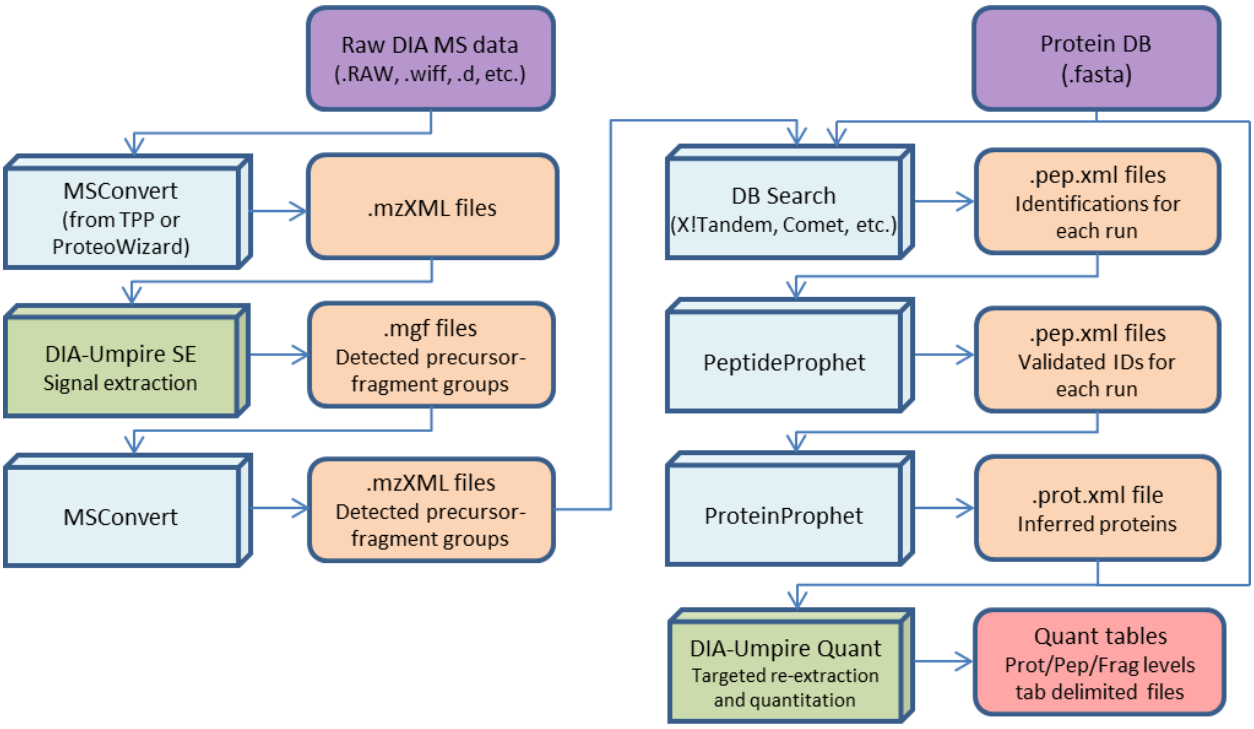


Figure 2. Dataflow chart for the full DIA-Umpire pipeline.

Processing sample datasets

We provide two sample datasets to try out the package that can be downloaded here:

1. UPS1 data (http://nesvilab.org/tsouc/DIA_Umpire_SampleDataUPS.zip)
   Simple protein mixture - UPS1 proteomics standard (48 human proteins in equal concentrations).
2. *E. coli.* data (http://nesvilab.org/tsouc/DIA_Umpire_SampleDataEcoli.zip)
   E. coli. lysate sample.

Each zip file contains spectral data in mzXML format, protein sequence fasta file, X! Tandem parameter file, and Batch command file for Windows. The samples represent low and high complexity datasets, respectively, and each dataset requires a different amount of RAM to be processed (UPS1 data: ≈4Gb, *E. coli* data: ≈8Gb). For each sample, there are two DIA (SWATH) runs from AB SCIEX TripleTOF 5600 instrument. We will now use the UPS1 dataset as an example to demonstrate all the steps of a complete DIA-Umpire analysis.

The DIA-Umpire signal extraction and quantitation modules each requires a parameter file. Here we provide example of parameter files specifically for each example dataset.

1. UPS1 data:
   a. Example parameter file for signal extraction module:
      http://nesvilab.org/tsouc/ups.se_params
   b. Example parameter file for quantitation module:
      http://nesvilab.org/tsouc/ups.quant_params
2. *E. coli.* data:
   a. Example parameter file for signal extraction module:
      http://nesvilab.org/tsouc/ecoli.se_params
   b. Example parameter file for quantitation module:
      http://nesvilab.org/tsouc/ecoli.quant_params

Download the sample data

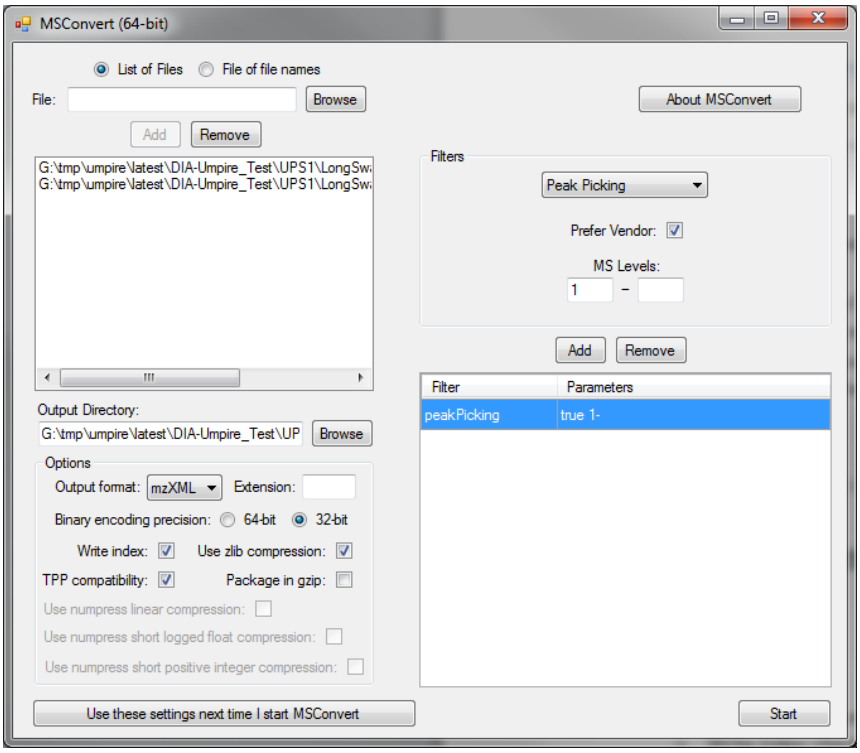
1. Download latest "*DIA-Umpire.zip*" which contains executable DIA-Umpire java JAR files from http://diaumpire.sourceforge.net/.
2. Create a folder named "*DIA-Umpire_Test*" at your local machine (We will be using "*D:/DIA-Umpire_Test/*" throughout the example), unzip *DIA-Umpire.zip* into the folder you have just created.
3. Create a folder "*UPS*" as a subdirectory of "*DIA-Umpire_Test*". Download the UPS1 sample data and unzip the file into "*D:/DIA-Umpire_Test/UPS/*".

Raw spectral data files conversion to mzXML

The sample spectral files provided are already in mzXML format so you can skip this section.

For all instruments, except AB SCIEX ones (see below), the easiest and common way for raw file format conversion is to use MSConvert (msconvert.exe) from ProteoWizard or TPP.

- Run MSConvert
- Click “Browse” and select your raw files (you can select multiple files at once by holding Shift/Ctrl keys)
- Specify “Output Directory”
- In the “Options” section make sure to select “mzXML” as the output format
- To make the size of output files smaller and processing faster, use the following set of options (as shown in the screenshot below):
    - Binary encoding precision: 32-bit
    - Write index: checked
    - TPP Compatibility: checked
    - Use zlib compression: checked
    - Package in gzip: unchecked
- If you acquired data in profile mode, it is recommended to use centroiding. In “Filters” section select Peak Picking, check “Prefer Vendor” checkbox, click “Add” button, to add the filter. This filter tells MSConvert to use vendor-provided centroiding algorithms which are available for all major mass-spec vendors.
- Your screen should look something like this:



- Press “Start”. The process might take quite a while depending on sizes of files and your computer.

Important (AB SCIEX instruments):

For data from AB SCIEX instruments we recommend using their proprietary data converter to convert *wiff* to *mzML* first and then use MSConvert to convert those to *mzXML*. It can be downloaded from here (*AB SCIEX MS Data Converter (Beta Version 1.3)*):

- http://goo.gl/wf7KRV

Documentation for the converter can be found here:

- http://goo.gl/a8LCth

Use “*-centroid*” option during the conversion, e.g.:

```
> AB_SCIEX_MS_Converter WIFF "path/to/input/file.wiff" -centroid MZML
"path/to/output/file.mzML"
```

followed by MSConvert

```
> msconvert --mzXML --32 --zlib -o "path/to/output/directory"
"path/to/input/file"
```

Note: This cumbersome conversion chain is needed, because AB SCIEX converter does something to the intensities of peaks in spectra, so you won't get the same result if you directly use msconvert.exe. This behavior might change in the future.

If you want to automate the process, instead of running via GUI, you can use the command line interface of MSConvert, the equivalent command is:

```
> msconvert --mzXML --32 --zlib --filter "peakPicking true 1-" -o
"path/to/output/directory" "path/to/input/file"
```

Signal extraction (feature finding) using DIA-Umpire

8. Open a terminal window.
   - Open Start menu, click "Run…" button. Or just press "Win+R" combination
   - Type cmd.exe, hit enter to start the terminal
4. In the terminal change directory to "D:/DIA-Umpire_Test/".
   - Enter "D:", hit enter - to change the drive
   - Enter "cd DIA-Umpire_Test", hit enter - to change the working directory

   Run the following command to run the signal extraction module on the first file:

```
> java -Xmx8G -jar D:/DIA-Umpire_Test/DIA_Umpire_SE.jar D:/DIA-
Umpire_Test/UPS/LongSwath_UPS1_1ug_rep1.mzXML D:/DIA-
Umpire_Test/UPS/ups.se_params
```

   - **-Xmx8G**: indicate how much memory will be allowed to use. The memory usage depends on the number of signals found in the data and the minimum signal threshold in feature detection. We recommend specifying at least double the amount of the mzXML file size.

     ***Note*:** Your computer will need at least 8Gb of RAM available to run this specific command. If you have a lower amount, try lowering this setting, e.g. use –Xmx4G for 4Gb.
   - **../DIA_Umpire_SE.jar**: the full path to DIA_Umpire_SE.jar file
   - **../LongSwath_UPS1_1ug_rep1.mzXML**: the full path to DIA mzXML file
   - **../ups.se_params**: input parameter file for the signal extraction module

   Windows users may use "DIA_Umpire_SE_win.bat" file provided with the sample data. You will need to edit the .bat file if you're using a different path than in this example ("*D:/DIA-*

Umpire_Test/"). Save the file when you are done editing. You can now run DIA-Umpire simply by double-clicking the .bat file.

Note: If you have installed Java Runtime Environment (JRE) but "java" is still not recognized as a valid command in your machine, please check if the path of java installation was added to environment variables or try using the full path of java.exe (e.g. on Windows a typical path would be: "C:/Program Files/Java/jre7/bin/java". Use this path instead of "java -jar" in the above command.)

5. Run the following command to process the second DIA file (if you ran the .bat file, you don't need that step, both files have already been processed):

```
> java -Xmx8G -jar D:/DIA-Umpire_Test/DIA_Umpire_SE.jar D:/DIA-
Umpire_Test/UPS/LongSwath_UPS1_1ug_rep2.mzXML D:/DIA-
Umpire_Test/UPS/ups.se_params
```

6. When the signal extraction process is complete, you will find three .mgf files for each DIA file in "*D:/DIA-Umpire_Test/UPS/*" folder. In this example, there will be six .mgf files for two DIA replicates.

```
LongSwath_UPS1_1ug_rep1_Q1.mgf
LongSwath_UPS1_1ug_rep1_Q2.mgf
LongSwath_UPS1_1ug_rep1_Q3.mgf

LongSwath_UPS1_1ug_rep2_Q1.mgf
LongSwath_UPS1_1ug_rep2_Q2.mgf
LongSwath_UPS1_1ug_rep2_Q3.mgf
```

7. Use MSConvert (GUI or command line version) to convert those six .mgf files into mzXML format in "*D:/DIA-Umpire_Test/UPS/*" folder. The command line version is:

```
> msconvert --mzXML D:/DIA-Umpire_Test/UPS/*.mgf
```

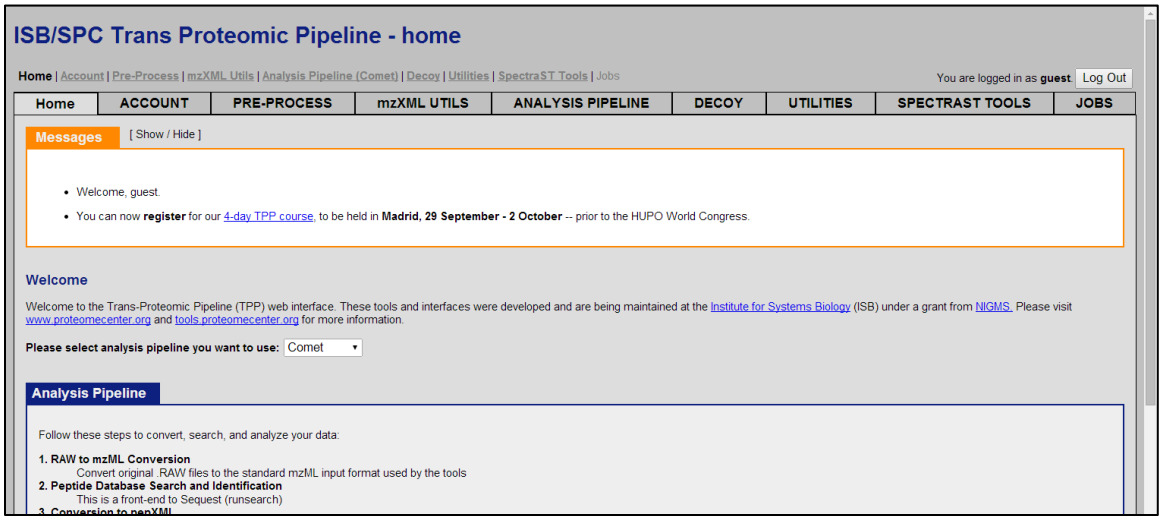
8. After the conversion, the .mzXML files are ready to be searched with a search engine.

Untargeted MS/MS database search using X!Tandem and TPP

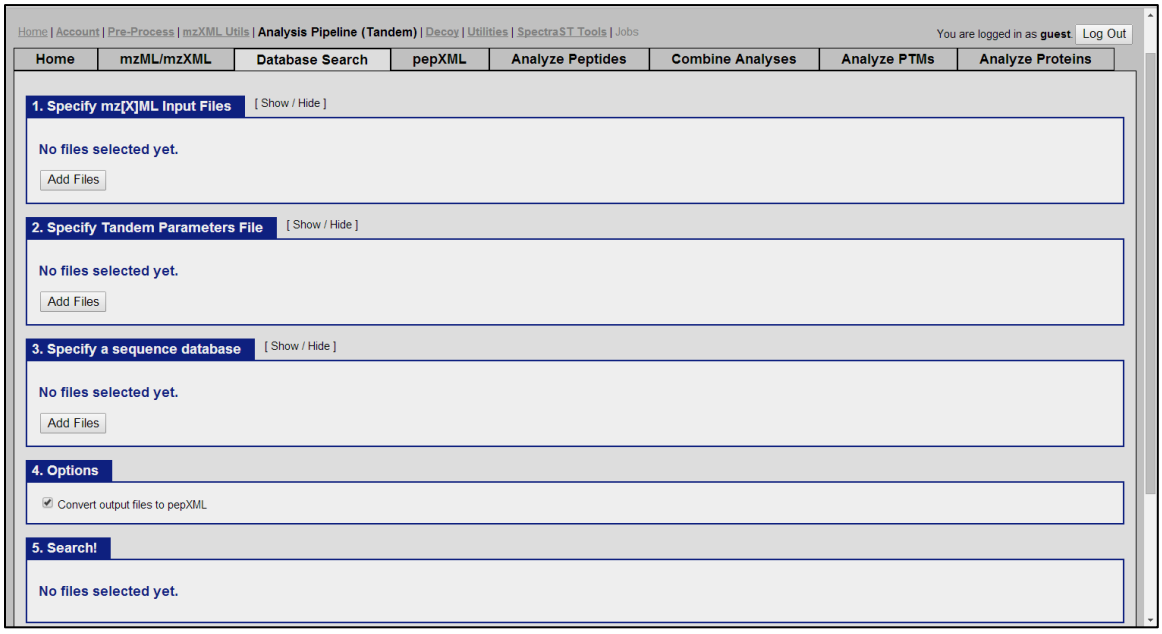
9. Create a folder named "DIA-Umpire_TPP" in the data folder of your TPP server. The default folder of TPP Windows version is C:/Inetpub/wwwroot/ISB/data/. Copy into the newly created folder the following files:
   - six .mzXML files converted from .mgf files

- the FASTA file “UPS_PlusRev.fasta“ – this is the database containing UPS proteins and decoys
- X!Tandem parameter file “tandem.params”

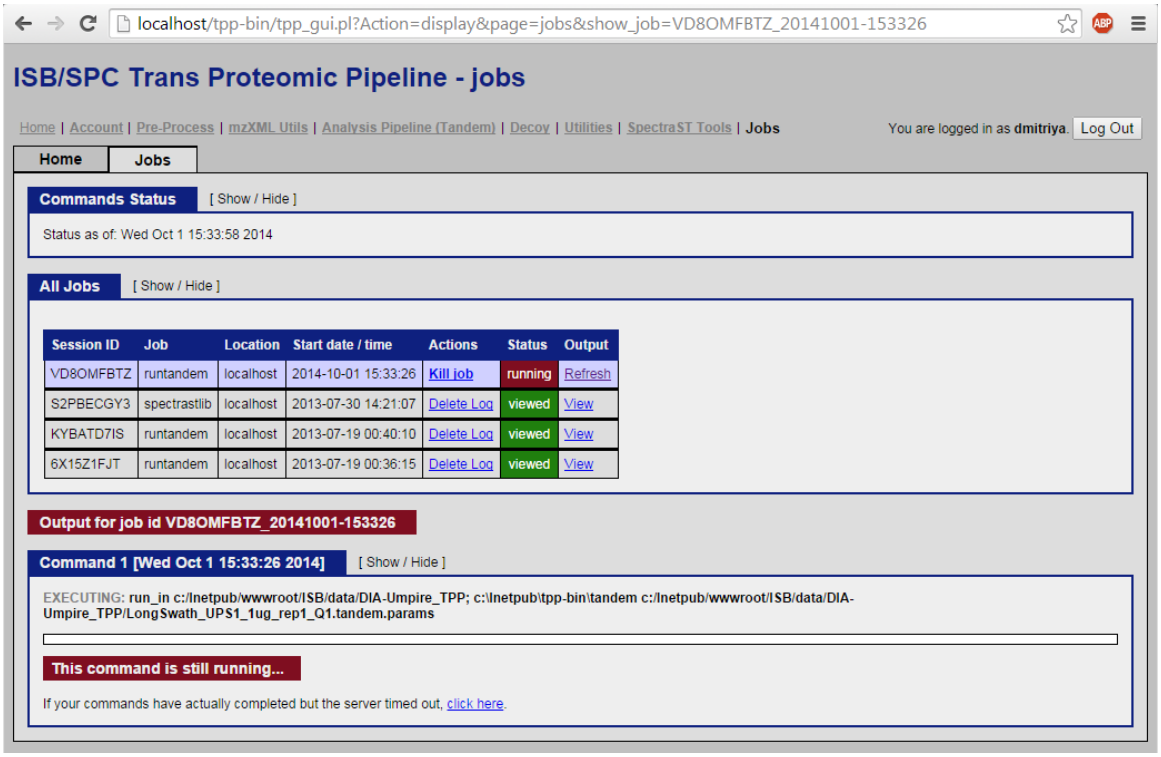
10. Start TPP welcome page (ex: http://localhost/tpp-bin/tpp_gui.pl) using internet browser (we’re using Google Chrome in this example) and log into the server using login guest, password guest.



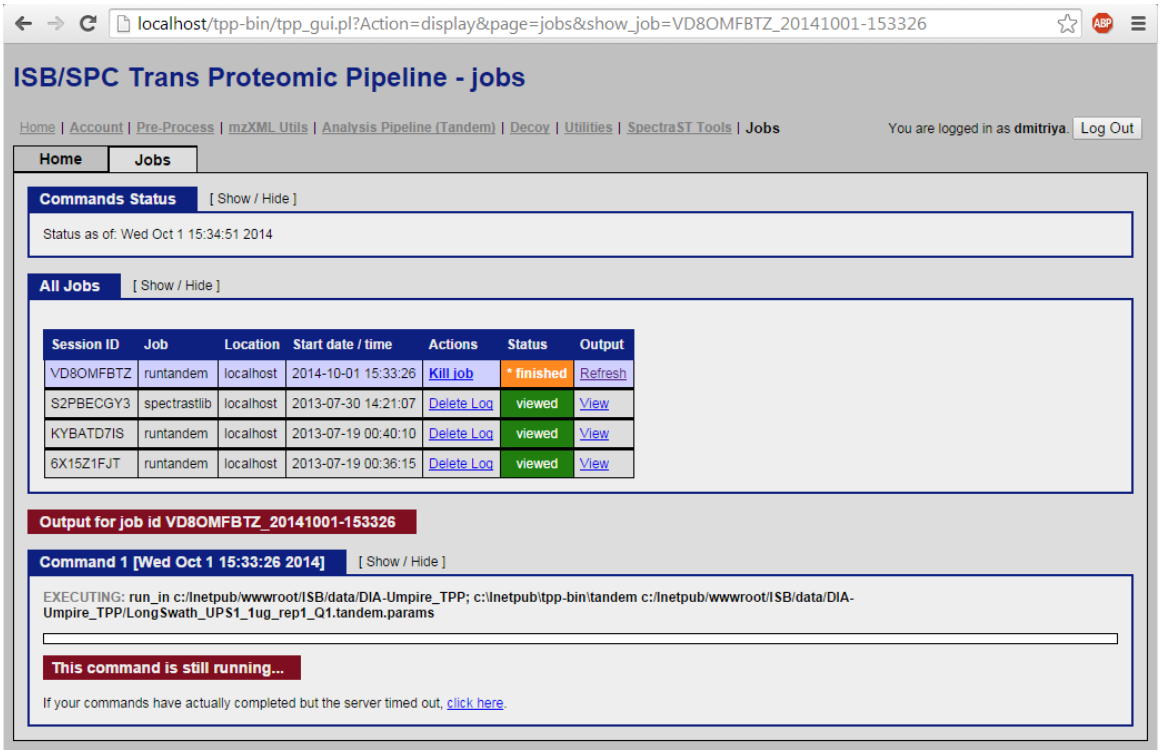
11. Change the search engine to “*Tandem*” using the dropdown menu “*Please select analysis pipeline you want to use*”
12. Click “Analysis Pipeline (Tandem)” at top of the page and then click “Database Search” to go to the page shown below.



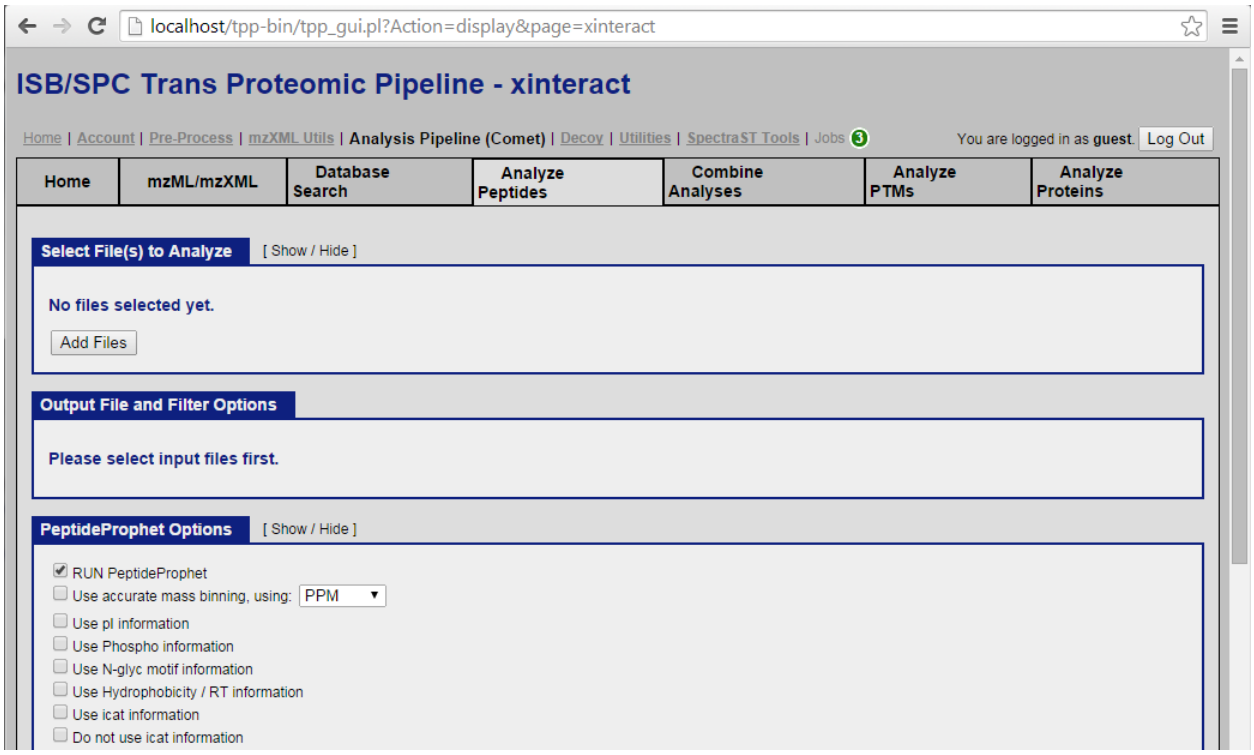
13. Click "*Add Files*" in section "*1. Specify mzXML Input Files*" and add six UPS1 .mzXML files which we've converted from .mgf (Q1, Q2, Q3) result files of DIA-Umpire signal extraction module.
    - On the right panel (Directory Tree) navigate to "DIA-Umpire_TPP" folder by clicking it
    - Check all checkboxes next to mzXML files and click "Select"
14. Click "Add Files" in section "*2. Specify Tandem Parameters Files*" and add *tandem.params*.
15. Click "Add Files" in section "*3. Specify a sequence database*" and add *UPS_PlusRev.fasta*.
16. Check "*Convert output files to pepXML*" if it's not checked.
17. Click "*Run Tandem Search*" to start the process, you will be presented with a job monitor, which should display your newly submitted job as running:



18. The search might take a while. When done, job status will change from “running” (in red) to “*finished” (in orange):

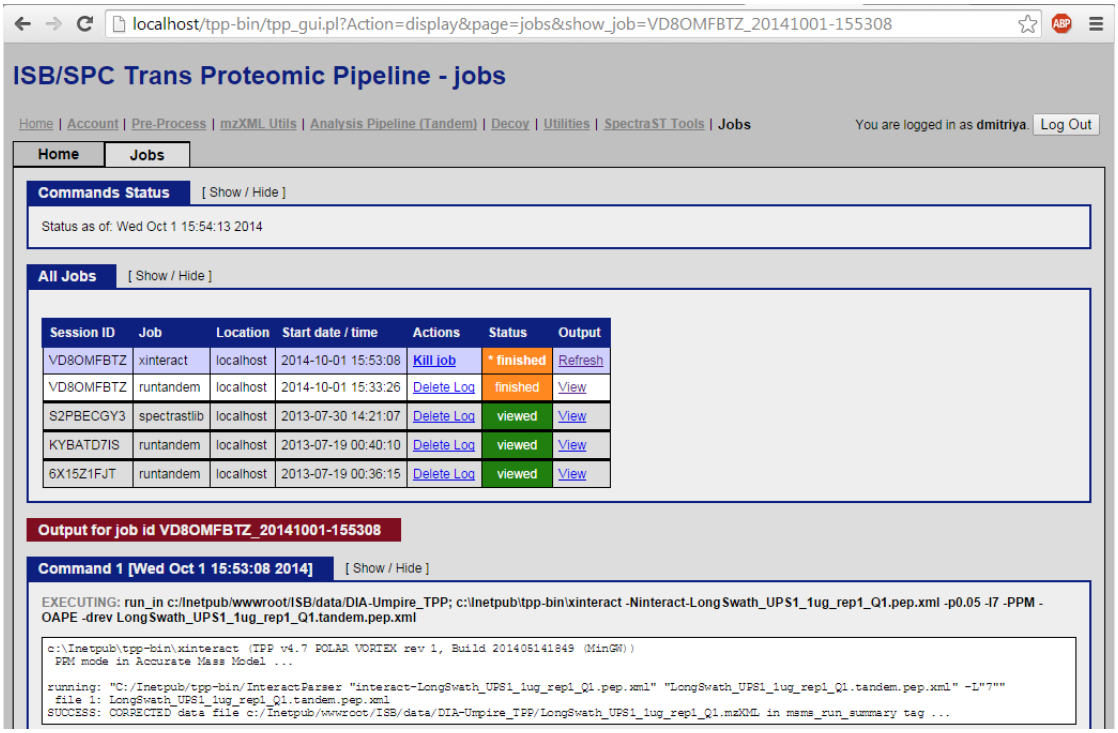


19. Click "*Analysis Pipeline (Tandem)*"->"*Analyze Peptides*" in the menu at the top of the page to go to the following page:



20. Click "*Add Files*" in "*Select File(s) to Analyze*" section and add all .tandem.pep.xml files. Make sure to NOT select the taxonomy.xml file by mistake.
21. In "*Output File and Filter Options*" section, check "*Do not merge into single analysis file (process each input file independently)*" option
22. In "*PeptideProphet Options*" section,
    a. Check "*Use accurate mass binning, using: PPM*" (-OA and -PPM option in command line)
    b. Check "*Only use Expect Score as the discriminant - helpful for data with homologous top hits, e.g. phospho or glyco (Tandem and Comet only)*" option (-OE option in command line)
    c. Check "*Use decoy hits to pin down the negative distribution.*" (-d option in command line)
    d. Type "rev" into the text box after "*Decoy Protein names begin with*" (-drev option in command line). The "rev" prefix is what was used for decoy protein identifiers in the FASTA file, which you've downloaded with the sample data.
    e. Check "*Use Non-parametric model (can only be used with decoy option)*" (-OP option in command line)
    f. Check "*Report decoy hits with a computed probability (based on the model learned).*" (-Od option in command line)
    - Click "*Run XInteract*" to start the process

- Again wait for the status to change to orange "*finished".



23. Go to "*Analysis Pipeline (Tandem)*"->"*Analyze Proteins*" to start ProteinProphet analysis.
24. In section "*Select File(s) to Analyze*" click "*Add Files*" to add all "*interact-<filename>.pep.xml*" files. Make sure to NOT select .pep.xml files whose name doesn't start with *interact-* or *taxonomy.xml* file.

    Note: in this example the names will be like: *interact-longswath_ups1_1ug_rep1_q1.pep.xml*
25. Click "*Run ProteinProphet*" at the bottom of the page to start analysis.
26. When the job finishes, click on "ProtXML" link in "Output Files" section to view protein and peptide identification results as well as annotated spectra.

For more information and tutorials covering TPP applications, please visit http://tools.proteomecenter.org/wiki/index.php?title=TPP_Tutorial.

Quantitation and targeted re-extraction analysis using DIA-Umpire

27. Copy the following .pep.xml and prot.xml files from "*C:\Inetpub\wwwroot\ISB\data\DIA-Umpire_TPP*" back to "*D:/DIA-Umpire_Test/UPS/*".

```
interact-LongSwath_UPS1_1ug_rep1_Q1.pep.xml
interact-LongSwath_UPS1_1ug_rep1_Q2.pep.xml
interact-LongSwath_UPS1_1ug_rep1_Q3.pep.xml

interact-LongSwath_UPS1_1ug_rep2_Q1.pep.xml
interact-LongSwath_UPS1_1ug_rep2_Q2.pep.xml
interact-LongSwath_UPS1_1ug_rep2_Q3.pep.xml

interact.prot.xml
```

28. Using a text editor open "*diaumpire.quant_params*" file, edit the path parameters to match your actual paths:
    - path – set this to the directory where all the UPS1 files are ("*D:/DIA-Umpire_Test/UPS*" in this example)
    - Fasta – set to FASTA file we used for MS/MS database search ("*D:/DIA-Umpire_Test/UPS/UPS_PlusRev.fasta*" in this example)
    - Combined_Prot – ProteinProphet analysis result, this is the *interact.prot.xml* file that we have copied back from the TPP folder ("*D:/DIA-Umpire_Test/UPS/interact.prot.xml"* in this example).
29. Execute the following command to start DIA-Umpire quantitation analysis.

```
> java -Xmx8G -jar D:/DIA-Umpire_Test/DIA_Umpire_Quant.jar D:/DIA-
Umpire_Test/UPS/ups.quant_params
```

30. Once the quantitation analysis is done, you shall find three tab-delimited tables for protein, peptide, and fragment summaries at "*D:/DIA-Umpire_Test/UPS/*".

Contact information

For questions, suggestions and bug reports please go to http://diaumpire.sourceforge.net/.

Appendix A: advanced parameters for signal extraction module

Here we describe the advanced parameters for signal extraction module. Usually you do not need to change these parameters.

Precursor-fragment grouping parameters

RPmax: (integer) Determines how many precursors a single fragment is allowed to be grouped to. Precursors are first sorted by Pearson correlation of elution profiles; this option specifies the rank of a precursor in this sorted list. Lowering the value for this parameter increases the stringency of precursor-fragments grouping. (Default: 25)

RFmax: (integer) Determines how many fragments a single precursor is allowed to have. Fragments are first sorted by Pearson correlation of elution profiles; this option specifies the rank of a fragment in this sorted list. The lower - the more stringent. (Default: 300)

CorrThreshold: (0.0~1.0) Minimum Pearson correlation between a precursor and a fragment to be considered, the higher, the more stringent. (Default: 0.2)

DeltaApex: (Unit: minute) Maximum retention time difference of LC profile apexes between precursor and fragment (the lower, the more stringent). (Default: 0.6)

BoostComplementaryIon: (true or false) set to ***true*** if you want to boost complementary ions' intensity. The process of complementary ion boosting will also deisotope fragment peaks into singly charged m/z position. (Default: true)

AdjustFragIntensity: (true or false) set to ***true*** if you want to adjust fragment intensity by the Pearson correlation between a precursor and a fragment. (Default: true)

Note: if you want to keep fragment intensity unchanged and without being deisotoped, please set both BoostComplementaryIon and AdjustFragIntensity as false.

Signal extraction parameters

SE.MinMSIntensity: Minimum signal intensity for a peak in an MS1 spectrum to be considered as a valid signal. Any MS1 peak having intensity lower than this threshold will be ignored. It is the main parameter controlling how many peaks and isotopic envelopes will be detected.

Recommended value: Depends on the data. Check raw data for average noise-levels. E.g. TOF data often have thousands of random small intensity peaks.
Warning: Setting this parameter too low (or zero) in such a case will significantly increase processing time and memory requirements.

SE.MinMSMSIntensity: Same as *para.MinMSIntensity*, but for MS2 signals.

SE.MaxCurveRTRange: (Unit: minute) The maximum allowed retention time (RT) range for elution profile of a single ion. If a detected elution profile exceeds that time span, it will be trimmed around the apex to fit into this range. Used to avoid having lots of ions which elute during the whole LC/MS run or over a very long period of time, as this greatly complicates grouping of precursors to fragments. Such long-eluting ions are likely to be contaminants, lock-mass ions, calibrants, etc.

Recommended value: The expected maximum peak chromatographic time. E.g. set to several percent of the whole run time, if the run was 100 min long, set to 5 min.

SE.SN: Minimum signal-to-noise threshold for MS1 precursor signal detection. It is not the real S/N value, but rather a multiplier for *para.MinMSIntensity*, if a detected elution profile is less intense in the apex than (*para.SN* x *para.MinMSIntensity*) it will be discarded.

Recommended value: Typical values depend on the *para.MinMSIntensity* setting. If you've set *para.MinMSIntensity* to a very low value, consider setting this one to some small number in range 1.0 – 5.0.

SE.MS2SN: Same as *para.SN*, but for possible unfragmented precursors in MS2 data (i.e. for selecting precursors to generate Q3 tier pseudo spectra).

SE.NoMissedScan: Maximum number of consecutive "gaps" allowed during extraction of elution profile (scans, in which the precursor mass being traced was not detected). E.g. if set to 1 and a particular mass can be found at every second scan, the algorithm will trace such a peak unless it can't find the peak in 2 scans in a row.

SE.MinFrag: Minimum number of fragments for a precursor. Precursors which have less than the set number of fragments will be removed from pseudo MS/MS spectra.

SE.EstimateBG: (true or false) set to ***true*** if you want to perform the background detection algorithm to determine minimum intensities for MS1 and MS2 spectra. Note that the settings of SE.MinMSMSIntensity and SE.MinMSIntensity will be ignored.

SE.MinNoPeakCluster (new parameter in v1.4): Minimum number of isotope peaks for a precursor feature. When it is set as 1, the algorithm will group fragments even for peaks without any isotope signal being found. For these cases, the assumed charged states will be from the parameter SE.StartCharge to SE.EndCharge.

SE.MaxNoPeakCluster (new parameter in v1.4): Maximum number of isotope peaks for a precursor feature.

Appendix B: advanced parameters for quantitation module

Here we describe the advanced parameters for quantitation module. Usually you do not need to change these parameters.

FDR estimation parameters

PeptideFDR: Target peptide level FDR.

DIA-Umpire estimates peptide level FDR by target-decoy approach according to peptide ion's maximum PeptideProphet probability. (default: 0.01)

Recommended value: 0.01 or 0.05 are the standard thresholds used in proteomics studies, corresponding to 1% and 5% FDR.

ProteinFDR: Target protein level FDR.

DIA-Umpire fist removes protein identifications with low protein group probability (<0.5) and estimates protein level FDR of the remaining list by target-decoy approach according to the maximum peptide ion's probability. (default: 0.01)

Recommended value: 0.01 or 0.05.

ProbThreshold: (0.0~0.99) Probability threshold for peptide-centric targeted extraction.

This probability is calculated by DIA-Umpire based on LDA analysis of true and decoy targeted identifications. (default: 0.99)

Recommended value: 0.99 corresponds to 99% confidence in an ID. Which means FDR should be less than 1% in that case.

Quantitation parameters

FilterWeight: (GW or PepW) Choice of using peptide group weight or peptide weight (computed by ProteinProphet) to remove shared peptides for protein quantitation. (default: GW)

MinWeight: (0.0~0.99) Minimum weight (peptide group weight or peptide weight chosen from the previous option) threshold of peptides to be considered for protein quantitation. Higher weight (closer to 1) of a peptide for a protein is more likely to be a unique peptide for the protein. (default: 0.9)

Recommended value: 0.9

TopNFrag: Top *N* fragments in terms of fragment score (Pearson correlation × fragment intensity) used for determining peptide ion intensity (default:6).

Recommended value: 3~6

TopNPep: Top *N* peptide ions in terms of peptide ion intensity (determined by top fragments) used for determining protein intensity (default:6)

Recommended value: 3~6

Freq: Minimum frequency of a peptide ion or fragment across all samples/replicates to be considered for Top *N* ranking. (default:0.5)

Recommended value: 0.5 or more