SwathXtend

An R package for extending SWATH assay libraries, performing statistical analysis and reliability check for SWATH results

Jemma Wu, Dana Pascovici and Xiaomin Song
APAF, Australia
jwu@proteome.org.au

September 14, 2026

SwathXtend is an R package aiming to facilitate extended assay library generation, stastical data analysis and reliability check for SWATH data. This package contains the major functions decribed in Wu et al. 2016. and Wu et al. 2017. This vignette describes how to use the major funcitons in SwathXtend.

Introduction

The first integrated DIA and quantitative analysis protocol, termed SWATH was shown to offer accurate, reproducible and robust proteomic quantification (Gillet et al 2012). An important concept in DIA analysis is use of a LC-retention time referenced spectral ion library to enable peptide identification from DIA generated multiplexed MS/MS spectra. SwathXtend is an R based software package to facilitate the generation of extended assay libraries for SWATH data extraction. It also contains functions to perform statistical analysis and reliability check for SWATH results.

Package installation

To install the SwathXtend package the following commands can be executed within R.

```
> if (!requireNamespace("BiocManager", quietly=TRUE))
+     install.packages("BiocManager")
> BiocManager::install("SwathXtend")
```

Typically the workspace is cleared and the SwathXtend pakcage is loaded.

```
> rm(list=ls())
> library(SwathXtend)
```

The example data, that is included in the package, consists of four assay libraries and two SWATH result files. The libraries can be loaded using *readLibFile*. Library format can be "PeakView" (AB Sciex 2014) or "OpenSWATH"

(Rost et al. 2014) format which is in a tab-delimitated .txt or comma-delimitated .csv file. The parameter *clean* in function *readLibFile* specifies if the library to be clearned, which will be describe later.

```
> filenames <- c("Lib2.txt", "Lib3.txt")
> libfiles <- paste(system.file("files",package="SwathXtend"),
+         filenames,sep="/")
> Lib2 <- readLibFile(libfiles[1], clean=TRUE)
> Lib3 <- readLibFile(libfiles[2], clean=TRUE)
```

If the file format is "peakview", it requires the following columns:

- Q1: Q1 m/z (precursor m/z)
- Q3: Q3 m/z (fragment m/z)
- RT_detected: retention time
- protein_name: protein name
- isotype: isotype type
- relative_intensity: fragment ion intensity
- stripped_sequence: peptide sequences without modifications
- modification_sequence: peptide sequences with modifications
- prec_z: peptide charge
- frg_type: fragment type (b or y ion)
- frg_z: fragment charge
- frg_nr: ion number
- iRT: calibrated retention time (the values might not be meaningful for experiments with no iRT peptides spiked in)
- uniprot_id: database accession number
- decoy: whether the peptide a decoy or not (TRUE or FALSE)
- confidence: the confidence of the identified peptide (a value between 0 and 1)
- shared: whether the peptide is shared by multiple proteins (TRUE or FALSE)
- N: a ranking number for the protein

Optional columns for PeakView format libaries include:

- score: score for peptide identification
- prec_y: the precursor ion intensity
- rank: ion intensity ranking

- mods: modification
- nterm: N terminal modification
- cterm: C terminal modification

If the file format is “openswath", it must contain the following columns:

- PrecursorMz: precursor m/z
- ProductMz: fragment m/z
- Tr_recalibrated: retention time
- ProteinName: protein name
- GroupLabel: isotype type
- LibraryIntensity: fragment ion intensity
- PeptideSequence: peptide sequences without modifications
- FullUniModPeptideName: peptide sequences with modifications
- UniprotID: database accession number
- decoy: whether the peptide a decoy or not
- PrecursorCharge: precursor charge
- FragmentType: fragment type
- FragmentCharge: fragment charge
- FragmentSeriesNumber: fragment ion number

Building extended assay library

To build an extended library using SwathXtend, one seed library and one add-on library are needed. The seed library is usually a local assay library which was generated with SWATH data using the same instrument and the same chromatography condition. The add-on library can be a local archived assay library or an external library downloaded from public data repositories such as SWATHAtlas(Biology IfS 2014).

Library cleaning

All candidate assay libraries were first subject to a cleaning process which removes low confident peptides and low intensity ions by user-defined thresholds. The default values for these two thresholds are 99% for peptide confidence and 5 for ion intensity. The clearning process can also opt to remove peptides with modifications for miss cleavages. The clearning process can be done separately using function *cleanLib* or as part of the library reading process as shown above.

```
> Lib2 <- cleanLib(Lib2, intensity.cutoff = 5, conf.cutoff = 0.99,
+                          nomod = FALSE, nomc = FALSE)
> Lib3 <- cleanLib(Lib3, intensity.cutoff = 5, conf.cutoff = 0.99,
+                          nomod = FALSE, nomc = FALSE)
```

Matching quality checking

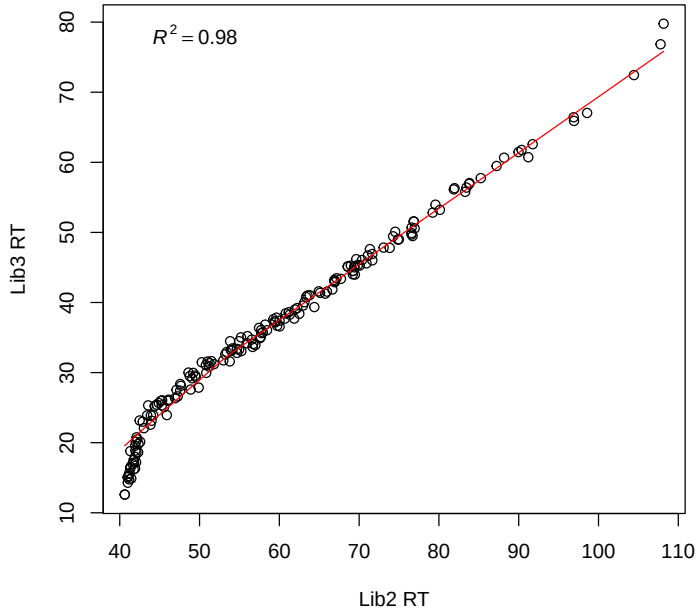
It is very important to check the matching quality between the seed and add-on libraries before building the extended library. Function *checkQuality* can be used to perform the library matching quality check based on the retention time and the relative ion intensity. Three measurements, including the retentiont time correlation, the predicted average error of the RT and the relative ion intensity correlation, will be returned.

```
> checkQuality(Lib2, Lib3)

$RT.corsqr
[1] 0.9778811

$RT.RMSE
[1] 0.9949378

$RII.cormedian
[1] 0.9
```

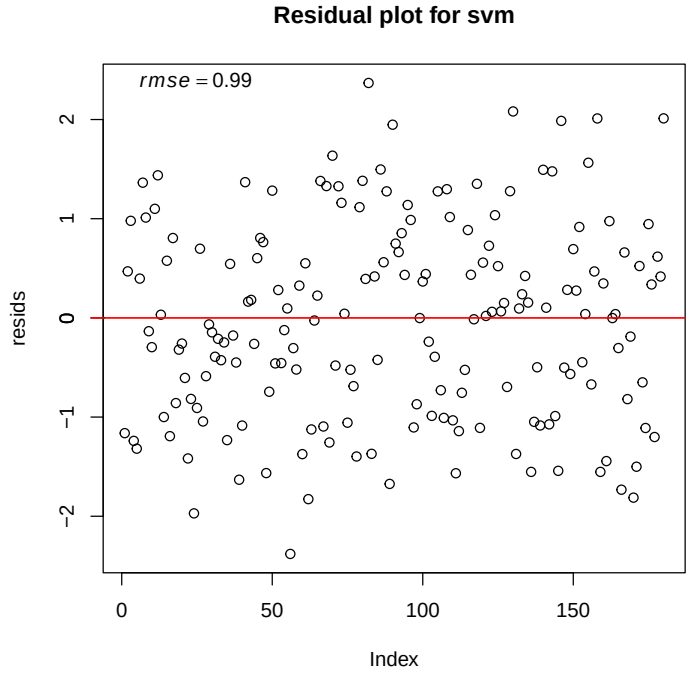
The first two outputs, RT.corsqr and RT.RMSE, represent the R^2 of the retention time correlation between the two libraries and the root of mean squared error of the RT prediction. We recommend if RT.corsqr is greater than 0.8 and the RT.RMSE less than 2 minutes, the retention time matching quality is good. The third output, RII.cormedian, repesents the median spearman correlation of the relative ion intensity (RII) of the common fragment ions. We suggest if it is greater than 0.6, these two libraries have good matching quality. We suggest the integration of libraries should be performed only when the RT and RII matching quality are good.

We can visualise the retention time correlation, prediction residual and relative ion intensity correlation using function *plotRTCor*, *plotRTResd* and *plotRIICor* respectively.

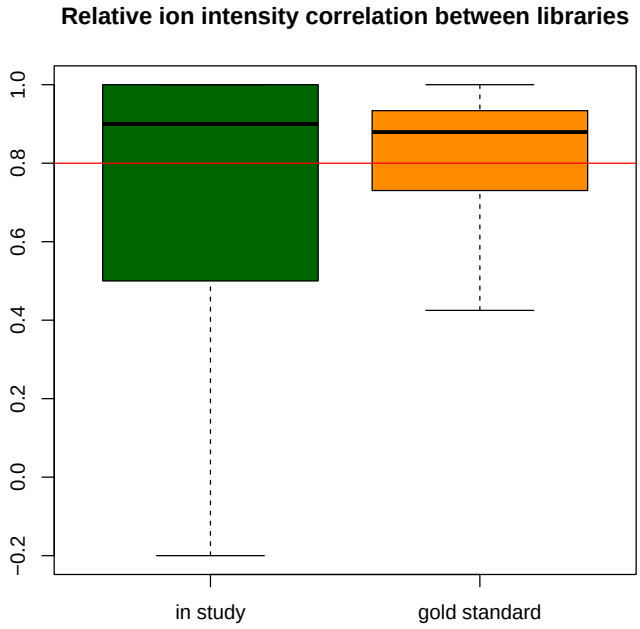
```
> plotRTCor(Lib2, Lib3, "Lib2", "Lib3")
```




```
> plotRTResd(Lib2, Lib3)
```




```
> plotRIICor(Lib2, Lib3)
```



Various statics about the two libraries can be plotted and exported into a multi-tab spreedsheet using *plotAll* function. These include barplots of the number of proteins and peptides of the seed library, add-on library and their relationship (including overlapping proteins, peptides, retention time scatter plots and spearman correlation coefficient boxplots)

```
> plotAll(Lib2, Lib3, file="allplots.xlsx")
```

Build the extended library

If the seed and add-on libraries have good matching quality, we can generate an extended library by integrating them using function *buildSpectraLibPair*.

```
> Lib2_3 <- buildSpectraLibPair(libfiles[1], libfiles[2], clean=T,
+                                nomc=T, nomod=T, plot=F,
+                                outputFormat = "peakview",
+                                outputFile = "Lib2_3.txt")
```

SwathXtend provides two methods of retention time alignment: time-based and hydrophobicity-based. If the retention time correlation between the seed and addon libraries are good (e.g., $R^2 > 0.8$), time-based method is recommended. Otherwise, hydrophobicity-based method can be tried. The hydrophobicity index for peptides can be calculated using SSRCalc(Krokhin 2006). The format of a hydrophobicity index file should include three columns, Sequence, Length and Hydrophobicity. An example of the hydrophobicity index file is included this package. The peptides in this file are all the peptides appearing in the three single assay libraries, i.e., *Lib1.txt*, *Lib2.txt* and *Lib3.txt*.

```
> hydroFile <- paste(system.file("files",package="SwathXtend"),
+                                  "hydroIndex.txt",sep="/")
> hydro <- readLibFile(hydroFile, type="hydro")
> head(hydro)
```

```
                 Sequence Length Hydrophobicity
1                   AGIQLSPK      8          18.66
2               DASAGIQLSPK     11          22.84
3                   DLVEHVAK      8          15.74
4 DPANLPWGSSNVDIAIDSTGVFK     23          47.65
5                 FVMGVNEEK      9           24.4
6       GIEEGLMTTVHSLTATQK     18          34.76
```

To build extended libraries using hydrophobicity-based retention time alignment, we can use the following command. The “method" can also be “hydrosequence" which will the combination of hydrophobicity index and the peptide sequence when building the model.

```
> Lib2_3.hydro <- buildSpectraLibPair(libfiles[1], libfiles[2], hydro,
+                                  clean=T,
+                                  nomc=T, nomod=T, plot=F,
+                                  method="hydro",
+                                  outputFormat = "peakview",
+                                  outputFile = "Lib2_3.txt")
```

Export the library

The output of the library format can be “PeakView" and “OpenSwath".

```
> outputLib(Lib2_3, filename="Lib2_3.txt", format="peakview")
```

Reliability check for SWATH with extended library

The package was extended to include functions for checking SWATH results quality with extended library. Details can be found in Wu et al. 2017. It should be noted that the reliability checking functions are only compatible with SWATH result files extracted by using *PeakView* with SWATH Acquisition MicroApp (AB Sciex 2014).

Library checking

Function *reliabilityCheckLibrary* compares the extended library with the seed library and checks the peptide and protein coverage. The input is the seed library and extended library files, and the output is a table containing the number of peptides and proteins in each library and their coverage percentage.

```
> libfiles <- paste(system.file("files",package="SwathXtend"),
+                  c("Lib2.txt", "Lib2_3.txt"),sep="/")
> res = reliabilityCheckLibrary(libfiles[1], libfiles[2])
```

SWATH results checking

A PeakView Swath result file is an Excel file (.xlsx) with six worksheets: “Area - ions", “Area - peptides", “Area - proteins", “Score", “FDR" and “Observed RT". The SWATH result checking functions require that worksheet “Area - peptides" and “FDR" must exist.

All the reliability checks for SWATH results are wrapped in a general function, *reliabilityCheckSwath*. The first and second parameter is the SWATH result file extracted using the seed and extended library, respectively. The third parameter is the upper threshold value for the number of samples passing the FDR threshold (i.e., <0.01). The fourth parameter is the upper threshold value for the number of peptides that a protein has.

```
> fswaths = paste(system.file("files",package="SwathXtend"),
+         c("Swath_result_seed.xlsx", "Swath_result_ext.xlsx"),
+         sep="/")
> res = reliabilityCheckSwath(fswaths[1], fswaths[2], max.fdrpass=8,
+         max.peptide=2)
```

After the execuation, a number of plots and tables will be generated to show the reliability measurements of the SWATH results under various FDR filtering thresholds. For example, the *fdr bins barplot* (Figure 1) shows the FDR distributions of the seed and extended SWATH results with different FDR filtering thresholds, i.e., the minimum number of samples that have a FDR less than 0.01 for a particular peptide.

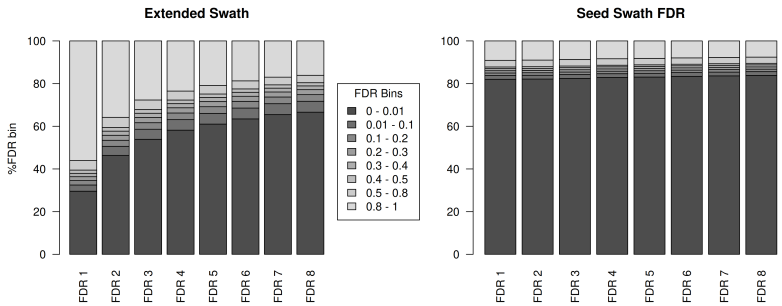


Figure 1: FDR distributions for FDR filtering thresholds

Figure 2 shows the number of proteins and peptides in the SWATH results extracted by the seed and extended libraries as the FDR filtering threshold changing.

Figure 3 shows the quantification consistency between the SWATH results extracted by the seed and extended library measured by Coefficient of Variation (CV). Refer to Wu et al. 2017 for details.

The users can also break down the whole process into individual steps. First, read in the *FDR* tab in the SWATH results.

```
> fdr.seed = readWorkbook(fswaths[1], sheet='FDR')
> fdr.ext = readWorkbook(fswaths[2], sheet='FDR')
```

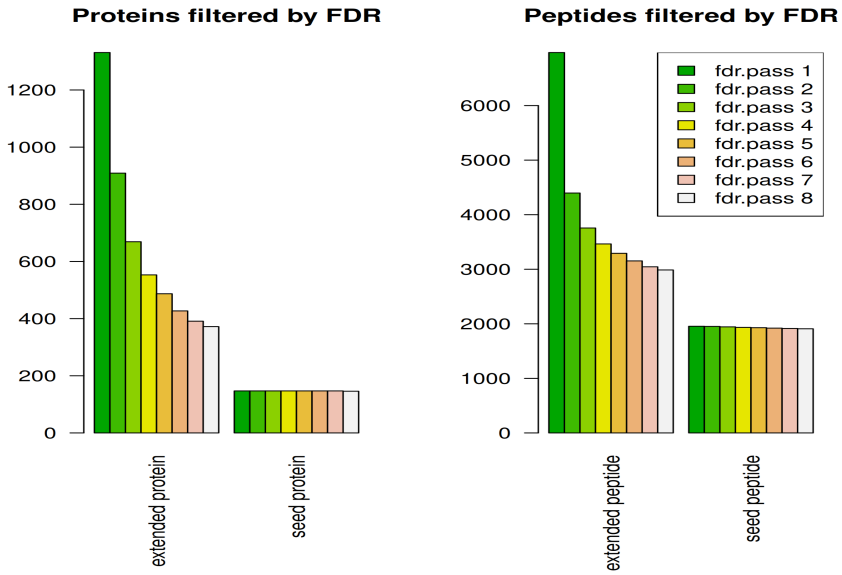


Figure 2: Protein and peptide numbers by FDR filtering thresholds

Function *fdr.crit* adds one column, *nfdr.pass*, to the fdr data frame. This columns represents the number of samples having a fdr less than 0.01.

```
> fdr.seed = fdr.crit(fdr.seed)
> fdr.ext = fdr.crit(fdr.ext)
> head(fdr.ext[,c(1:2,ncol(fdr.ext))])
```

```
                Protein                Peptide nfdr.pass
1  sp|P02768|ALBU_HUMAN             LVNEVTEFAK        43
3  sp|P02768|ALBU_HUMAN         KVPQVSTPTLVEVSR        69
5  sp|P02768|ALBU_HUMAN           LC[CAM]TVATLR        79
7  sp|P02768|ALBU_HUMAN       YIC[CAM]ENQDSISSK        80
9  sp|P02768|ALBU_HUMAN AAFTEC[CAM]C[CAM]QAADK        80
11 sp|P02768|ALBU_HUMAN            AVMDDFAAFVEK        62
```

To get the quantificaion accuracy between the SWATH results extracted using the extended library and seed library, we need to read in the *Peptide* tab in the SWATH result.

```
> swa.seed = readWorkbook(fswaths[1], 2)
> swa.ext = readWorkbook(fswaths[2], 2)
```

Function *quantification.accuracy* can be used to calculate the quantification consistency between the SWATH results extracted by the seed and extended library. The measurement methods can be one of "cv" (Coefficient of Varation), "cor" (Coefficient of Correlation) and "bland.altman"(Bland-Altman).

```
> fdr.seed = fdr.seed[fdr.seed$nfdr.pass > 0,]
> fdr.ext = fdr.ext[fdr.ext$nfdr.pass > 0,]
```

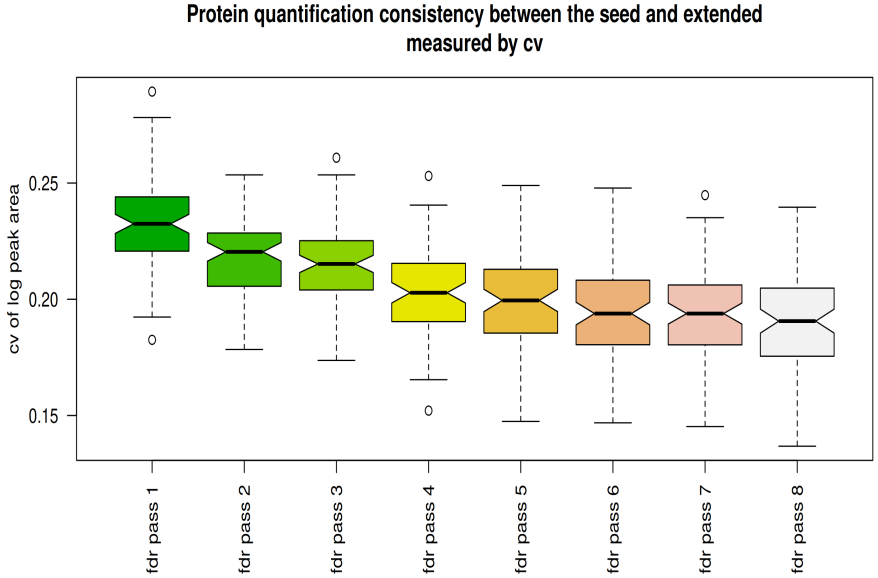


Figure 3: Quantificaton consistency measured by CV with different FDR filtering thresholds

```
> res = quantification.accuracy(swa.seed[fdr.seed$nfdr.pass >= 1,],
+          swa.ext[fdr.ext$nfdr.pass >= 1,], method="cv")[[1]]
```

The returned value is a numeric vector representing the quantification consistency measurment.

References

Biology IfS (2014) SWATHAtlas.
Gillet LC et al. “Targeted data extraction of the MS/MS spectra generated by data-independent acquisition: a new concept for consistent and accurate proteome analysis". Molecular and Cellular Proteomics 11. 2012
Krokhin OV. “Sequence-specific retention calculator. Algorithm for peptide retention prediction in ion-pair RP-HPLC". Analytical chemistry 78:7785-7795. 2006
Wu JX et al. “SWATH mass spectrometry performance using extended peptide MS/MS assay libraries". Molecular and Cellular Proteomics 15.7 (2016): 2501-2514
Wu JX et al. “Improving protein detection confidence using SWATH mass spectrometry with large peptide reference libraries". Proteomics. (Under review 2017)
AB Sciex. “MS/MS with Swath Acquisition MicroApp 2.0 User Guide". 2014
Rost H L., et al. “OpenSWATH enables automated, targeted analysis of data-independent acquisition MS data." Nature biotechnology 32.3 (2014): 219-223.