**Demo for:**

**Interference Modeling**

**in Multiplex Proteomics**

## Overview

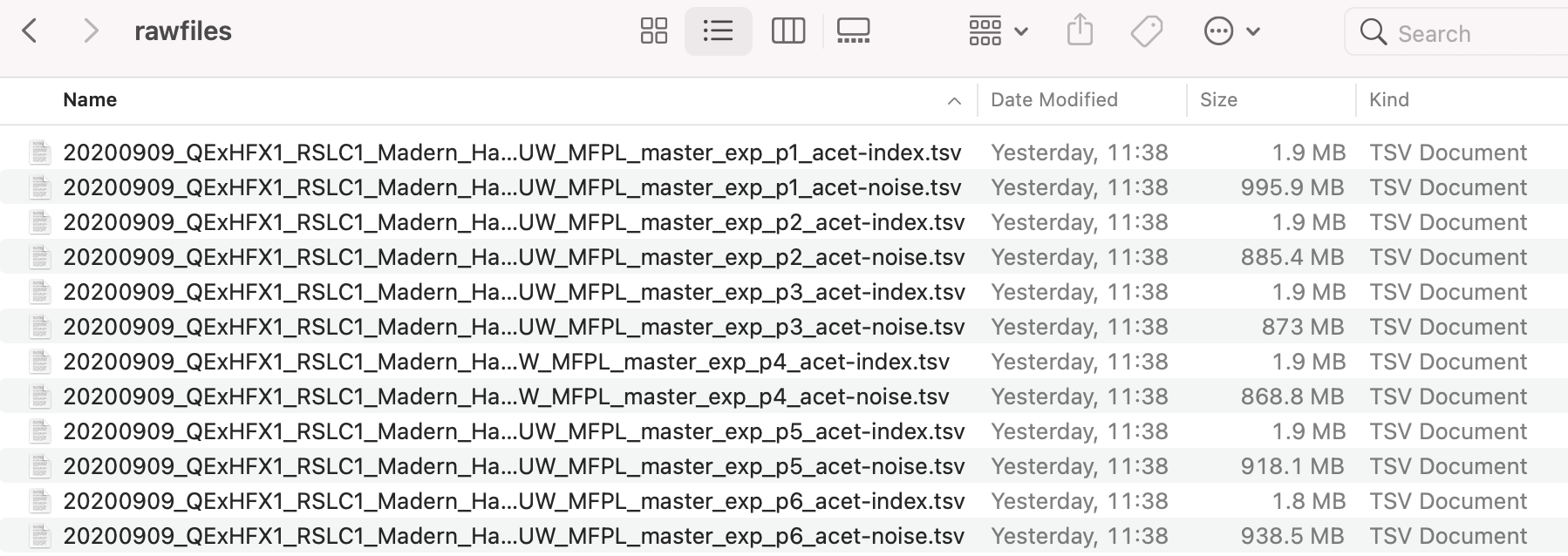
This document serves as a guide to the demo for the interference modeling workflow on GitHub. The demo comes with its own dataset based on which the intended workflow can be experienced from start to finish. I recommend running the demo first before trying the workflow on your own data – this allows you to get familiar with the required data input, the script’s many parameters as well as the data output.

## Required Setup

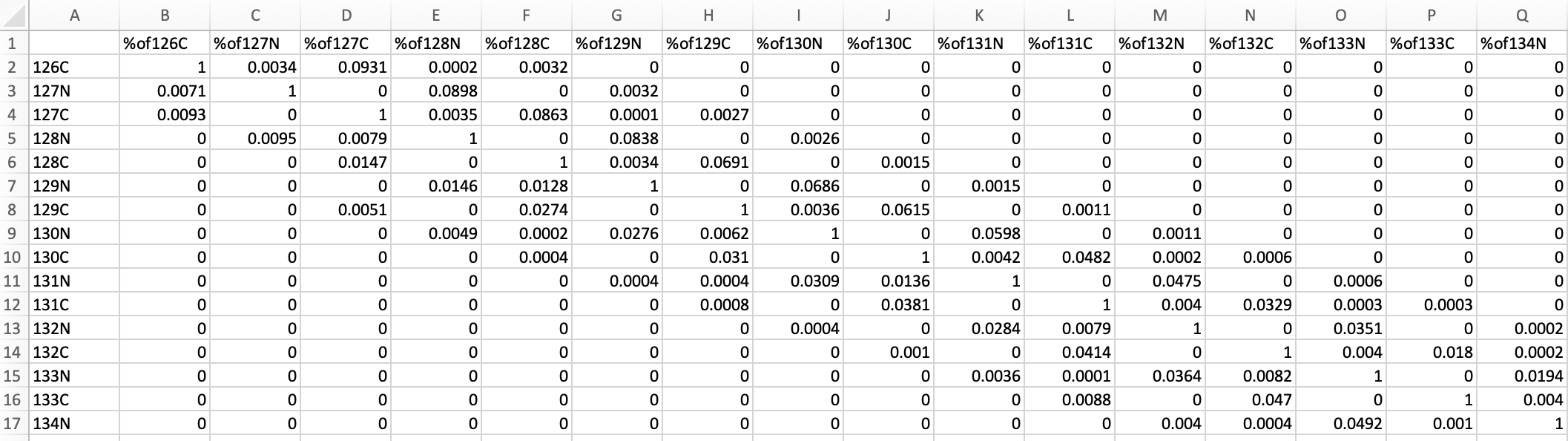
# Every time you want to use this workflow, I recommend creating a new folder (here: “Demo”) in which to put all the necessary software (i.e. scripts and tools) as well as the required data input to run the script. For this demo, the required setup looks like this:

# 

# Let’s go over the individual files:

* **msms.txt** is a MaxQuant’s PSM table output. You can find this file on GitHub in the Demo folder. In this table, each row corresponds to a single PSM. Since the database search usually comprises data of multiple measurements, MaxQuant’s msms.txt usually contains PSMs from any number of raw files. For this demo, which focuses on measurements of acetyl-peptide enriched samples, msms.txt has already been filtered accordingly to minimize the file’s overall size. The unfiltered version of this msms.txt (which produces the same results) can be downloaded on on PRIDE:
* **rawfiles** is a subfolder that should be created. It needs to contain the Thermo raw files (\*raw) containing the spectra of all PSMs of the PSM table input to be used in the interference modeling workflow. For this demo, the raw files folder needs to contain the following six raw files that pertain to acetyl-peptide enriched measurements:  
    
  20200909\_QExHFX1\_RSLC1\_Madern\_Hartl\_UW\_MFPL\_master\_exp\_p1\_acet.raw  
  20200909\_QExHFX1\_RSLC1\_Madern\_Hartl\_UW\_MFPL\_master\_exp\_p2\_acet.raw  
  20200909\_QExHFX1\_RSLC1\_Madern\_Hartl\_UW\_MFPL\_master\_exp\_p3\_acet.raw  
  20200909\_QExHFX1\_RSLC1\_Madern\_Hartl\_UW\_MFPL\_master\_exp\_p4\_acet.raw  
  20200909\_QExHFX1\_RSLC1\_Madern\_Hartl\_UW\_MFPL\_master\_exp\_p5\_acet.raw  
  20200909\_QExHFX1\_RSLC1\_Madern\_Hartl\_UW\_MFPL\_master\_exp\_p6\_acet.raw  
    
  The raw files are available on PRIDE:
* **rawStallion** is a Windows command-line application that reads relevant information (e.g. noise values, intensity values, etc.) from Thermo raw files and writes them to two tsv files per raw file. You can download rawStallion here: <https://github.com/fstanek/rawStallion>. If you don’t work with a Windows operating system, you can find the corresponding tsv files needed for the demo on PRIDE. Download them and put them into the raw files folder:  
    
  

This lets you skip the section in the program requiring rawStallion.

* **impurity\_matrix\_tmtpro.csv** is a csv file that contains an isotopic impurity matrix specific to the labeling reagents used in the experiment:  
    
  

Rows reflect relative contribution of individual reagents to reporter ion channels ordered along the columns.

* **functions\_IM.R** contains functions to be sourced by the main script IM.Rmd. This file is located in the main folder of the repository on GitHub.
* **IM.Rmd** is the R Markdown script to perform the entire workflow. This file is located in the main folder of the repository on GitHub.

## Running the Program

Open the script IM.Rmd in R studio and make sure your working directory is set to the folder (here “Demo”) that contains the necessary software and data described above. We can then proceed to go through the script.

The first code section loads multiple required packages:

Text

Description automatically generated

Make sure these packages are installed prior to running the script. Regular R packages can be installed within R-studio. To install Bioconductor packages, visit the respective Bioconductor website (e.g. <https://bioconductor.org/packages/release/bioc/html/MSnbase.html>) and follow the instructions in the “Installation” section.

The second code section is where the user is required to specify all required input parameters to successfully run the workflow. These parameters aim configure the program to the specific data input. Everything outside of this code section does not need to be changed by the user. In its current form, the script’s parameters are configured to make the demo work:

Text

Description automatically generated

Make sure to read and understand each parameter by reading the comments above each line of code. If specified incorrectly, the program will produce errors down the line. If a parameter is described as “Optional”, the parameter is not required for successfully running the program, as some steps in the workflow can skipped. Set optional parameters to their default value to skip these sections. The default value of optional parameters are described in the comments.

Once the parameters are specified correctly, the rest of the script can be executed. Each subsequent code section performs a specific task and often produces intermediate output (visual and/or textual) of interest. This output should hopefully be insightful once you have read the corresponding paper (“A causal model of ion interference enables assessment and correction of ratio compression in multiplex proteomics”). Additionally, the comments in the code should provide the necessary understanding of what is happening. Code sections described as “Optional” can be skipped, since they are not required for successfully running the program. Note that some code sections will take some time to run, especially if there are many raw files to be processed. The program is currently not optimized to run as fast as possible.

**Output**

The script produces an output table called “modified\_PSM.txt” that is stored in a folder called Results (“Demo/Results/modified\_PSM.txt”). The output contains multiple additional columns that are generated in the course of the script. Notable columns for further analysis are: Normalized reporter intensity columns (suffix “\_norm”); normalized interference-corrected reporter intensity columns (suffix “\_norm\_\_interference\_corrected”); and the columns EIL (Estimated Interference Level) and PPF (Precursor Purity Fraction).

Note that this output table serves as input to the demo for site-to-protein normalization in multiplex proteomics (also on GitHub).