**Brief description of what can be done** with s\_adhoc\_data\_processing\_master.m

This script contains a modular mass spectrometry imaging (MSI) data analysis pipeline.

By Teresa Murta, June 2021

This script can be used to perform the following steps:

# DESI, MALSI, SIMS or REIMS data pre-processing (using SpectralAnalysis functions)

# Peak detection on total spectrum (using SpectralAnalysis functions)

# Assignment of peaks detected in the total spectrum, using HMDB as well as a group of lists of molecules of interest defined by the user

# Creating and saving a datacube (SpectralAnalysis DataRepresentation struct) for each imzml of interest (using SpectralAnalysis functions)

# Creating and saving a data matrix for each imzml and normalisation algorithm of interest

# Multivariate Analysis (e.g. PCA, NMF, k-means clustering, t-SNE, fast density clustering) using:

## N most intense peaks detected in the representative spectrum

## Percentile P of all peaks detected in the representative spectrum

## One or more lists of molecules of interest defined by the user

## One or more superclass, class, or subclass of molecules (as defined by HMDB)

## A list of m/z values

## Running MVAs

## Saving MVAs results

# Saving single ion images for:

## One or more lists of molecules of interest defined by the user

## One or more superclass, class, or subclass of molecules (as defined by HMDB)

## A lists of m/z values

# Creating masks for regions of interest (SpectralAnalysis RegionsOfInterest struct) by combining (adding and/or multiplying) the results of k-means clustering, t-SNE, or fast density clustering, with any areas of the entire image, which are interactively chosen by the user.

# Defining a “new” dataset by combining a group of (often masked) imzmls files.

## Specifying a group of imzmls files that need to be combined

## Specifying one or more masks to be used for each imzml (defining in 8), which reduce each image (imzml) to one or more smaller group of pixels of interest

## The position to be occupied by each sample/tissue (i.e. small mask or group of pixels of interest) in a 2D grid of images that will represent the new “dataset”

# Saving k-means or t-SNE space clustering maps as regions of interest (SpectralAnalysis RegionsOfInterest struct) when the clustering was done for the entire “new” data set. These regions of interest can be used in any subsequent analyses (note: the have to be moved and rename according to what makes sense for the study in particular).

# Saving a table of mean and median ion intensities per region of interest or mask.

# Running and saving the results of univariate analyses (e.g.: t-test, ranksum test, ROC analysis), which compare mean, median, and the whole samples of ion intensities, of two user defined groups of regions of interest (masks). These tests are done mass by mass.

# Running and saving the results of an ANOVA using N user defined groups of regions of interest (masks), which should relate to “conditions” or “effects” of interest for the study (e.g. data acquisition date, glass slide number, sample ID, tissue type, etc).

# Discarding groups of ions found using the ANOVA (13) before running the MVAs (6).

# Saving the data from an original imzml or a new “dataset” (defining in 9) in a csv file. Each row represents a pixel. This file contains ion intensity for all mass values in the datacube, pixel coordinates in the related imzml space, the name of the imzml file, the name of the main mask, and the name of the small mask.

**Requirements** to run s\_adhoc\_data\_processing\_master.m:

# SpectralAnalysis v1.4.0, released in Aug 2020, available at <https://github.com/AlanRace/SpectralAnalysis/releases> added to the Matlab path.

* The most recent version of “adhoc-data-processing-pipeline” available at <https://github.com/NICE-MSI/adhoc-data-processing-pipeline> added to the Matlab path.
* The location of (i.e. the path to) the SpectralAnalysis pre-processing file (extension “.sap”) to be used. An example can be found in “required-files” within the git repository “adhoc-data-processing-pipeline” specified above. The parameters of the pre-processing need to be adequate to the data. The pre-processing file can be edited in Matlab.
* The location of (i.e. the path to) the imzML and ibd data files, which have to be saved in modality and polarity specific folders.
* An excel file named “inputs\_file” saved in the folder that contains the imzml and ibd data files. An example can be found in “required-files” within the git repository “adhoc-data-processing-pipeline” specified above. This file needs to be adjusted to the specific requirements of the analysis, dataset, study goals, etc.

**Tree of functions** called by s\_adhoc\_data\_processing\_master.m

By Teresa Murta, June 2021

# Data pre-processing (uses SpectralAnalysis functions)

f\_saving\_spectra\_details

f\_reading\_inputs

# Peak detection (uses SpectralAnalysis functions)

For an imzml file at a time:

f\_saving\_peaks\_details (for an imzml at a time) or f\_saving\_peaks\_details\_ca (for a combination of imzmls)

f\_reading\_inputs

# Peak assignments

## HMDB

f\_saving\_hmdb\_assignments (for an imzml at a time) or f\_saving\_hmdb\_assignments\_ca (for a combination of imzmls)

f\_reading\_inputs

f\_makeAdductMassList

f\_stringToFormula

## Lists of molecules of interest defined by the user

f\_saving\_relevant\_lists\_assignments (for an imzml at a time) or f\_saving\_relevant\_lists\_assignments\_ca (for a combination of imzmls)

f\_reading\_inputs

f\_molecules\_list\_mat

f\_makeAdductMassList

f\_stringToFormula

# Datacube

## Datacube specific peak details

f\_saving\_datacube\_peaks\_details (for an imzml at a time) or f\_saving\_datacube\_peaks\_details\_ca (for a combination of imzmls)

f\_reading\_inputs

f\_peakdetails4datacube

## Datacube itself (i.e. SpectralAnalysis DataRepresentation struct) (uses SpectralAnalysis functions)

f\_saving\_datacube

f\_reading\_inputs

# Data matrix for each normalisation

f\_saving\_normalised\_data

f\_reading\_inputs

f\_norm\_datacube

f\_crukNormalise

# Multivariate Analysis

## Running MVAs

f\_running\_mva (for an imzml at a time) or f\_running\_mva\_ca (for a combination of imzmls)

f\_reading\_inputs

f\_datacube\_mzvalues\_vector

f\_black\_peaks\_list\_removal

f\_datacube\_mzvalues\_lists

f\_datacube\_mzvalues\_highest\_peaks

f\_datacube\_mzvalues\_highest\_peaks\_percentile

f\_datacube\_mzvalues\_ampl\_ratio\_highest\_peaks

f\_datacube\_mzvalues\_ampl\_ratio\_highest\_peaks\_percentile

f\_datacube\_mzvalues\_classes

f\_running\_mva\_auxiliar

f\_kmeans

f\_select\_k\_kmeans

f\_tsne

f\_tsne\_space\_clustering

f\_kmeans

f\_select\_k\_kmeans

f\_densityParam

f\_densityClust

f\_decisionGraph

## Saving MVAs results

f\_saving\_mva\_outputs (for an imzml at a time) or f\_saving\_mva\_outputs\_ca (for a combination of imzmls)

f\_reading\_inputs

f\_saving\_mva\_auxiliar (for an imzml at a time) or f\_saving\_mva\_auxiliar\_ca (for a combination of imzmls)

f\_40colourscheme

f\_mva\_output\_collage (for a combination of imzmls only)

f\_mva\_output\_table (for a combination of imzmls only)

f\_saving\_sii\_files (for an imzml at a time) or f\_saving\_sii\_files\_ca (for a combination of imzmls)

f\_saving\_curated\_top\_loadings\_info (for a combination of imzmls only)

# Saving single ion images

f\_saving\_sii (for an imzml at a time) or f\_saving\_sii\_ca (for a combination of imzmls)

f\_reading\_inputs

f\_saving\_sii\_sample\_info (for an imzml at a time) or f\_saving\_sii\_sample\_info\_ca (for a combination of imzmls)

f\_reading\_inputs

f\_unique\_extensive\_filesToProcess (for a combination of imzmls only)

f\_saving\_sii\_files (for an imzml at a time) or f\_saving\_sii\_files\_ca (for a combination of imzmls)

# Creating regions of interest / masks

f\_mask\_creation

f\_reading\_inputs

# Defining a “new” dataset

f\_*ReplaceByStudyName*\_samples\_scheme\_info

f\_check\_datacubes\_mass\_axis

f\_reading\_inputs

f\_unique\_extensive\_filesToProcess

# Saving k-means or t-SNE space clustering maps as regions of interest (SpectralAnalysis RegionsOfInterest struct) when the clustering was done for the entire “new” data set.

# f\_saving\_mva\_rois\_ca

# f\_reading\_inputs

# Saving a table of mean and median ion intensities per region of interest or mask.

f\_ion\_intensities\_table

f\_reading\_inputs

f\_saving\_curated\_hmdb\_info

# Univariate Analyses

f\_univariate\_analyses

f\_unique\_extensive\_filesToProcess

f\_reading\_inputs

f\_saving\_curated\_hmdb\_info

f\_saving\_sii\_sample\_info (for an imzml at a time) or f\_saving\_sii\_sample\_info\_ca (for a combination of imzmls)

f\_reading\_inputs

f\_unique\_extensive\_filesToProcess (for a combination of imzmls only)

f\_saving\_sii\_files (for an imzml at a time) or f\_saving\_sii\_files\_ca (for a combination of imzmls)

# ANOVA

f\_anova

f\_unique\_extensive\_filesToProcess

f\_reading\_inputs

f\_saving\_curated\_hmdb\_info

# Discarding groups of ions found using the ANOVA before running the MVAs.

f\_anova\_based\_unwanted\_mzs

f\_unique\_extensive\_filesToProcess

f\_reading\_inputs

*All function in 6*

# Saving the data from an original imzml or a new “dataset” (defining in 9) in a csv file.

f\_saving\_labelled\_data\_ca

f\_reading\_inputs

The outputs of running this script are:

# A spectral details folder containing:

## totalSpectrum\_intensities - total spectrum counts (per imzml and main mask)

## totalSpectrum\_mzvalues - total spectrum mass channels (per imzml and main mask)

## pixels\_num - number of pixels of interest (per imzml and main mask)