

ProteomicsAnalysis Documentation

Module name: ProteomicsAnalysis

Description:Runs the proteomics analysis on the set of input spectraAuthor:D. R. Mani (Broad Institute), gp-help@broad.mit.edu

Summary: The entire proteomics analysis is run on the set of input spectra. This consists of the following steps:

- 1. Quality Assessment. Compute quality for each spectrum and retain only spectra that pass the quality.threshold specified.
- 2. Normalization. Normalize the spectra to enable comparison of peaks across different spectra. Implemented normalization schemes include: tan (total area normalization), tin (total intensity normalization), 01 (scale intensities between 0 and 1) and none (no normalization).
- 3. Peak detection.
- 4. Peak matching, where peaks that are the same across different spectra are identified using either a m/z-window based strategy or using a mixture of Gaussians optimized using EM (model-based clustering).

The result of running the module is a table containing the union of peaks (m/z) detected and matched in all spectra, with intensities marked at each m/z for every input spectrum. If a peak is missing in a spectrum, the intensity can be marked missing (NA) if fill.na=FALSE, or the actual intensity at the specific m/z value interpolated from the spectrum (if fill.na=TRUE).

The low.Da and high.Da values should be chosen to include relevant parts of the spectra, and exclude noise, especially matrix signal. If window-based peak matching is used, 2*mz.precision is used to set window size.

For more details on the proteomics analysis see Mani & Gillette, 2005.

Notes

 A file named "integer" is created as a by-product of running the analysis. This file can be ignored.

References:

- D. R. Mani & Michael Gillette. 2005. Proteomic Data Analysis: Pattern Recognition for Medical Diagnosis and Biomarker Discovery. In Mehmed Kantardzic and Jozef Zurada (Eds.) New Generation of Data Mining Applications, IEEE Press.
- Gillette, M.A., Mani, D.R., and Carr. Place of Pattern in Proteomic Biomarker Discovery, S.A. *J. Proteome Res.*, 4, 4, 1143 1154, 2005, 10.1021/pr0500962

Parameters:

zip.filename	Zip file of csv files for each spectrum
output.file	The output file prefix
quality.threshold	discard spectra with quality < quality.threshold
fill.na	whether to fill missing peaks [TRUE FALSE]
normalize	normalization strategy [tan tin 01 none]
peak.detection.method	peak detection method [detect input random]



filter.peaks	filter peaks based on peak intensity / noise [yes no]
filter peaks factor	Retain peaks if peak intensity >= filter.peaks.factor *
	stdev(noise)
peak list filename	file containing one M/Z per line when peaks=input
random seed	random seed when peaks=random
random n peaks	Number of random M/Z's selected when peaks=random
low.Da	Minimum M/Z to include
high.Da	Maximum M/Z to include
percentile	After applying filters, threshold above which peaks are
	located (default=0.65)
smoothing.size	Size of the smoothing filter (default=21)
adaptive.background.correctio	Strength of the adaptive background correction filter
n.strength	(default=0.75)
adaptive.background.correctio	Size of the adaptive background correction filter
n.size	(default=21)
high.pass.filter.strength	Strength of the high pass filter (default=10)
high.pass.filter.factor	Filter factor for the high pass filter (default=5)

Output Files:

- 1. <output.file>.gct file containing the spectra x m/z table. This table contains the union of the detected and matched peaks (m/z) with respective peak intensities (or absence) marked for every input spectrum.
- 2. <output.file>-stats.odf file with statistics (min, max, normalization factor, selection flag, etc.) for every input spectrum.
- 3. <output.file>-mzarray.odf file listing the actual peaks detected for every input spectrum.
- 4. If EM peak matching is used (default)
 - a. <output.file>-empeaks.odf file containing a table listing the mean, variance and mixing probability for the Gaussian mixtures representing the matched peaks
 - b. <output.file>-mzarray-em.odf file that shows the mapping of actual peaks to EM-matched peaks for every spectrum.

Platform dependencies:

Module type: Proteomics

CPU type: any OS: any Language: R