False Discovery Controlled Metabolite Annotation of Imaging Mass Spectrometry Data

CURRENT STATUS: POSTED

Andrew Palmer
Alexandrov Team, EMBL

research.palmer@gmail.com*Corresponding Author*

Theodore Alexandrov
Alexandrov Team, EMBL

DOI:
10.1038/protex.2017.003

SUBJECT AREAS
Computational biology and bioinformatics *Mathematics and computing*

KEYWORDS
false discovery rate, imaging mass spectrometry, metabolomics

Abstract

In this protocol we explain how to perform false discovery rate controlled metabolite annotation of data acquired with high-resolving-power imaging mass spectrometry. This translates the spectral data into biological annotations for further analysis.

Introduction

Metabolomics is a crucial scientific domain, promising important advances in cell biology, physiology, and medicine. Metabolomics complements genomics, transcriptomics, and proteomics by analyzing the instantaneous state of biochemical processes and revealing the contributions of non-genetic factors.

The introduction of high-mass-resolution (HR) MS that discriminates compounds differing by merely a few mDa promises to achieve unprecedented reliability of metabolite annotation in MS. Imaging MS combines this analytical specificity with spatially resolved measurements where mass spectra are acquired over a grid of pixels directly from a section of tissue, cell culture, or agar to produces images showing directly the distribution of metabolites across a sample. However, due to the lack of bioinformatics for automated molecular annotation, HR imaging MS is not routinely used for untargeted metabolomics and has been restricted mainly to targeted imaging of a few metabolites only. An additional obstacle is the lack of a metabolomics-compatible approach for estimating False Discovery Rate (FDR)

Our framework takes as input: 1) an HR imaging MS dataset in the imzML format, 2) a database of metabolite sum formulas in a CSV format, e.g., exported from HMDB 17, 3) an adduct of interest (e.g., +H, +Na, or +K for positive ionization mode). For a specified FDR level (e.g., 0.1), the framework provides molecular annotations: a list of metabolites from the database detected as present in the sample.

Equipment

The computation required for this protocol has been implmented in Python 2.7 and is available from:

https://github.com/alexandrovteam/pySM

Example data is provided along with the code.

Procedure

The following procedure details how to use the pySM library (https://github.com/alexandrovteam/pySM) to perform FDR controlled annotation.

"See figure in Figures section.":http://www.nature.com/protocolexchange/system/uploads/4803/original/thefigure_workflow_detailed.png?1474008322

The pipeline has two core parts: Calculation of Metabolite Signal Match (MSM) scores for every molecular formula in a metabolite database. Reporting of molecular formula at a specified FDR

Installation

1) Obtaining the code

a. Create a convenient directory, for example spatial_metabolomics and clone the repository into there:

b. mkdir spatial_metabolomics

c. cd spatial_metabolomics

d. git clone https://github.com/alexandrovteam/pySM

2) We recommend installing pySM and its dependencies inside a virtual environment as follows.

a. Next, if you have Anaconda installation of Python, follow the installation instructions Setting up a virtual environment using conda. Otherwise, follow the instructions Setting up a virtual environment using virtualenv.

b. Setting up a virtual environment using conda

i. Initialize and activate an 'pySM' environment with all the dependencies:

ii. cd pySM

iii. conda env create

iv. source activate pySM

v. Install pySM package with pip:

vi. pip install . -r requirements.txt

c. Setting up a virtual environment using virtualenv

i. Setup and activate a new virtual environment:

ii. pip install virtualenv

iii. virtualenv venv

iv. source venv/bin/activate

v. Install pySM and dependencies with pip:

vi. cd pySM

vii. pip install . -r requirements.txt

Annotating a dataset

1) Inputs

a. To process a dataset three things are needed: 1. a high-resolution imaging MS dataset; 2. a metabolite database 3. a configuration file

2) Dataset

Data should be in the .imzML format. The pipeline is designed for and was tested on centroided data.

3) Database

The database is a CSV with columns for id, name, exact_mass, formula

4) Configuration file

A complete example configuration can be found here (https://github.com/alexandrovteam/pySM/blob/master/pySM/example/example_config.json). The following parameters should be set individually for every dataset, other parameters can generally be left at their default values

"name":"dataset_short_name",

"image_generation":{

"ppm":

},

"file_inputs":{

"data_file":"/path/to/imaging_ms_dataset.imzML",

"database_load_folder":"/path/to/tmp_folder_for_storing_isotope_patterns",

"results_folder":"/path/to/folder_for_storing_results",

"database_file":"/path/to/database.csv"

},

"fdr":{

"pl_adducts":[

{"adduct":"+H"},

{"adduct":"+Na"},

{"adduct":"+K"}

],

},

"isotope_generation":{

"charge":[

{"polarity":"+", "n_charges":1}

],

"isocalc_sig":0.01,

a. name: a short name for the dataset, if "name":"" the imzml filename will be used

b. ppm: the m/z window for ion images

c. file_inputs: path for loading data/storing results

d. fdr: false discovery rate properties

e. pl_adducts: real adducts to search for

f. isotope_generation:

g. charge: polarity and charge state to search for (the pipeline currently only supports one charge state at a time). e.g. for negative mode singly charged use "charge":[{"polarity":"-", "n_charges":1}

],

h. isocalc_sig: peaks are predicted with a gaussian shape. This parameter is the sigma parameter. sigma = FWHM/2.3548.

i. isocalc_resoultion is not mass spectral resolution, it is the digitisation rate of the isotope patterns

5) Calculating MSM Scores

a. The spatial_metabolomics module runs the pipeline for calculating MSM scores. To calculate MSM scores for a whole dataset and database simply pass the configuration file to the run_pipeline function:

```
from pySM import spatial_metabolomics
json_filename = '/path/to/config.json'
spatial_metabolomics.run_pipeline(json_filename)
```

6) This will then write the MSM score for every combination of molecular formula and target adduct found in the metabolite database to a text file in the "results_folder" specified in the config file. Additionally a randomly selected set of decoy adducts will be chosen for , and their MSM scores calculated. (The number of decoy adducts is controlled by the config parameter fdr\n_im).

7) Reporting FDR

a. The main use of FDR control is to report which molecular formulas are annotated at a fixed FDR. This uses the results file generated by spatial_metabolomics.run_pipeline and the target and decoy adducts specified in the configuration file.

```
from pySM import spatial_metabolomics, fdr_measures
json_filename = '/path/to/config.json'
results_fname =
spatial_metabolomics.generate_output_filename(spatial_metabolomics.get_variables(json_filename),
[],'spatial_all_adducts')
target_adducts,decoy_adducts = fdr_measures.get_adducts(json_filename)
fdr = fdr_measures.decoy_adducts(results_fname,target_adducts,decoy_adducts)
```

b. To print a list of molecular-formula for each target adduct that have an MSM score which results in an FDR of less than fdr_target.

```
fdr_target=0.1
print fdr.decoy_adducts_get_pass_list(fdr_target,n_reps=20,col='msm')
```

Timing

5) the majority of the time is taken calculating the MSM scores, this depends on the number of pixels

in the dataset and the number of peaks per spectra but is usually approximately one hour per dataset.

The first time a particular combination of isotope generation parameters is used, isotope patterns must be generated for each molecule in the database. This can take several hours.

Figures

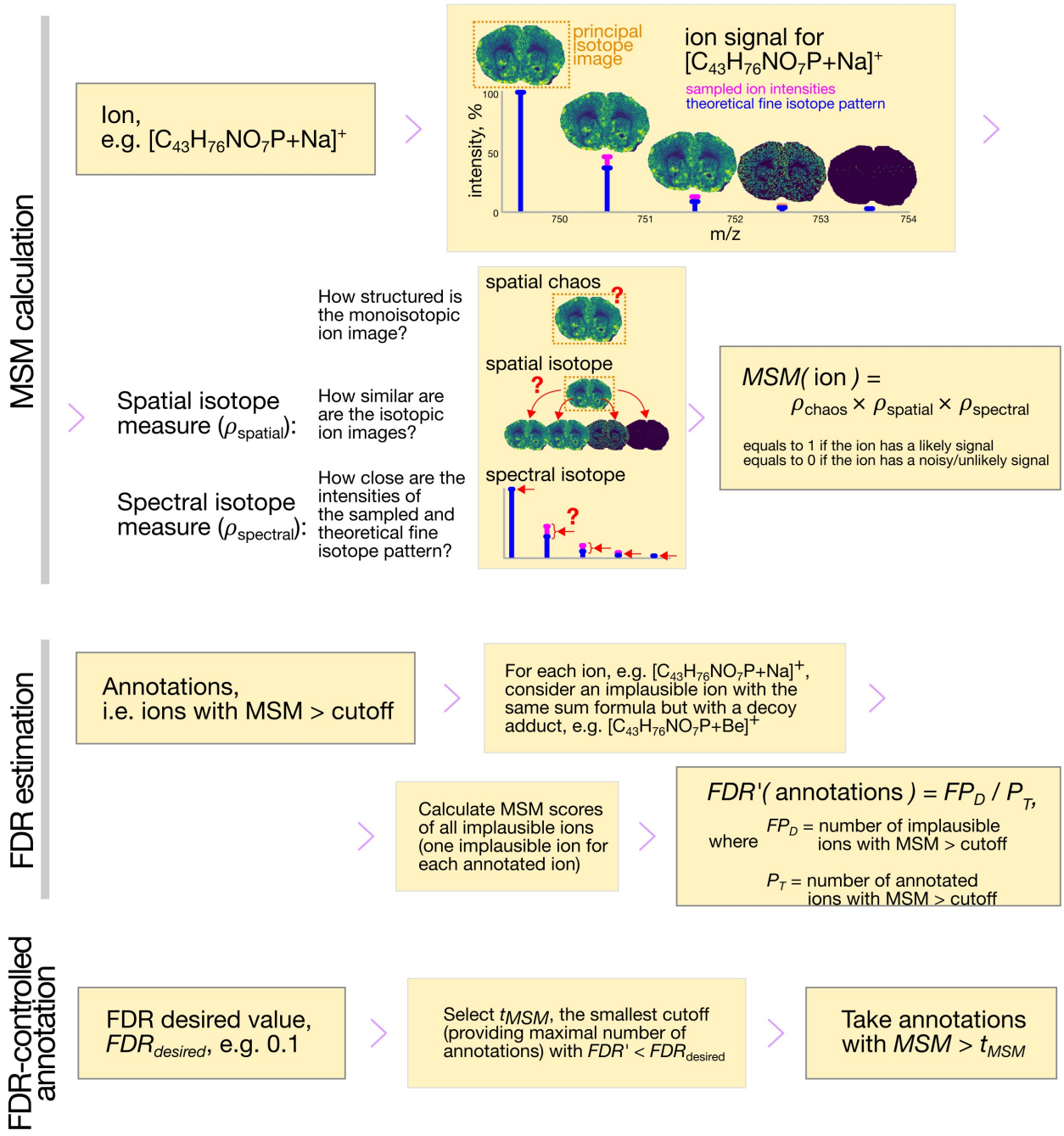

Figure 1

FDR controlled annotation Workflow for FDR controlled annotation of imaging mass spectrometry data

FDR-controlled metabolite annotation for high-resolution imaging mass spectrometry

by Andrew Palmer, Prasad Phapale, Ilya Chernyavsky, +9
Nature Methods (11 January, 2017)