User manual: EndoGenius (Updated 8/2/2023)

About EndoGenius:

EndoGenius is an open-source Python-based algorithm designed for identification of neuropeptides from mass spectral data.

License:

EndoGenius is freely available for download from GitHub (https://github.com/lingjunli-research/EndoGenius) and has an included user interface for increased accessibility.

Computational requirements:

EndoGenius is only supported for use with Windows OS, though limited functionality may be available with MacOS and Linux/UNIX. EndoGenius is capable of running on a standard personal computer. For design purposes, the program was validated on a Windows Desktop computer with the following specifications:

Processor: Intel(R) Xeon(R) CPU E5-1607 v3 @ 3.10GHz

Installed RAM: 128 GB

System Type: 64-bit OS

System: Windows 10

Setup:

There are two packages of EndoGenius. The recommended version involves zero prior command line knowledge, and can be accessed via the "releases" portion of our GitHub repository (https://github.com/lingjunli-research/EndoGenius/releases/tag/v1.0.0). This version includes an installation wizard to walk through the entire process.

An advanced version is available in the "AdvancedSetup" folder of the GitHub repository (https://github.com/lingjunli-research/EndoGenius/tree/main/AdvancedSetup). To run this, command line knowledge or familiarity with a python IDE is required. While we recommend running in a virtual environment in Anaconda, any command line should be operational. Simply navigate to the directory and run "`python GUI.py`", and the program with the GUI will deploy. For successful operation, the following packages are required, some of which come standard with Python:

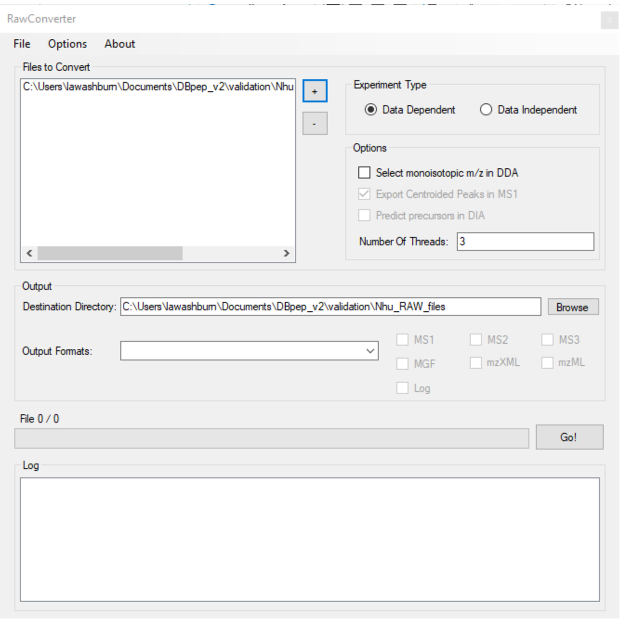
- Pathlib
- Tkinter
- Bio
- Pandas
- Csv
- Numpy

- Re
- OS
- Smtplib
- Itertools
- Random
- Collections
- Time
- Scipy
- Datetime
- Pyopenms
- Pyteomics
- Statistics

Spectral requirements:

EndoGenius is designed for functionality with .mzML and MS2 spectra files for each RAW file. .mzML files must be stored in the same directory as the .MS2 file, and have the same name. .mzML files can be obtained through MSconvert (https://proteowizard.sourceforge.io/download.html). MS2 files must be generated through RawConverter (http://fields.scripps.edu/rawconv/).

When converting spectra in RawConverter, select output format as “MS1,MS2,MS3”, and select only the box for “MS2”. See screenshot for details.



Files to Convert: C:\Users\lawashburn\Documents\DBpep_v2\validation\Nhu

Experiment Type: ● Data Dependent ○ Data Independent

Options:
□ Select monoisotopic m/z in DDA
☑ Export Centroided Peaks in MS1
□ Predict precursors in DIA
Number Of Threads: 3

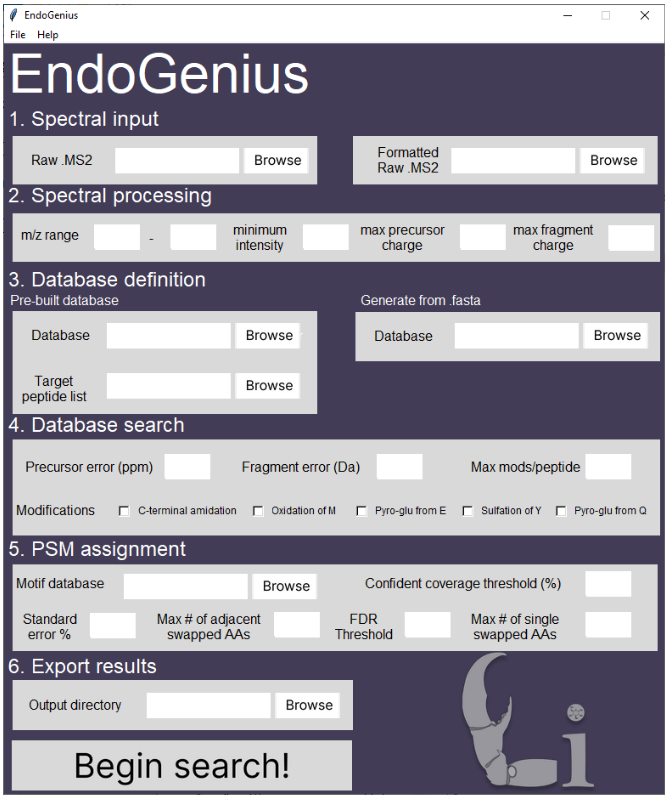
Output
Destination Directory: C:\Users\lawashburn\Documents\DBpep_v2\validation\Nhu_RAW_files
Output Formats:
□ MS1 □ MS2 □ MS3
□ MGF □ mzXML □ mzML
□ Log

File 0 / 0

Home Screen functionalities:

Phase I: Spectral input

- Spectra can be input in one of two formats, either as a .MS2 file, or as a formatted .MS2 file (.txt format):
    - **Raw .MS2 (recommended):** Upon initial run, input a .MS2 file into the “raw .MS2” input field. The input file will be the .MS2 generated from RawConverter.
    - **Formatted Raw .MS2:** If rerunning a previously analyzed file, a formatted .MS2 file can be selected to reduce analysis time. This is located in the output folder of the previous run.



Phase II: Spectral processing

- **m/z range:** minimum and maximum m/z to analyze
- **Minimum intensity:** minimum intensity of precursor peaks for consideration
- **Max precursor charge**: maximum precursor charge for consideration
- **Max fragment charge:** maximum fragment charge for consideration

Phase III: Database definition

- A database can be input in one of two formats, either as a .fasta file, or as a .csv with a target peptide list:
    - **Option 1:** for users who would like to generate a database independently, for instances in which a specific target-decoy approach is desired, users can include a .csv file with a list of peptides (target & decoy), as well as a target list. *File formats can be located on the GitHub page. Note: if this database includes PTMs, any PTM selections elsewhere will not be applied.*

    - **Option 2 (recommended):** input a .fasta file with all peptide sequences, from which a target-decoy database will be generated using a decoy-shuffle approach.

Phase IV: Database search

- **Precursor error (ppm):** maximum precursor error considered for match
- **Fragment error (ppm):** maximum fragment error considered for match
- **Maximum mods/peptide:** maximum number of post-translational modifications (PTMs) to be considered per peptide
- **PTMs:**

Modification	Amino Acid	Monoisotopic Mass	Composition
C-terminal amidation	-	-0.984016	H(1)N(1)O(-1)
Oxidation	M	15.994915	O(1)
Pyro-glu (Glu→pyro-Glu)	E	-18.010565	H(-2)O(-1)
Pyro-glu (Gln→pyro-Glu)	Q	-17.026549	H(-3)N(-1)
Sulfation	Y	79-956815	O(3)S(1)

Phase V: PSM assignment

- **Motif database:** List of desired motifs consistent with peptide-type of interest. *Example files can be found on the GitHub page.*
- **Confident coverage threshold %:** percent sequence coverage above which is considered a probable match. *Recommended value is 70%.*
- **Standard error %:** Used to differentiate two potential PSM assignments. Within this threshold, scores are considered indifferentiable. *Recommended value is 0.10.*
- **Maximum # of adjacent swapped AAs:** the number of instances in which two amino acids can be swapped and considered chimeric. *Recommended value is 2.*
- **FDR threshold:** false-discovery rate threshold. *Recommended value is 0.01 (1%).*
- **Maximum # of single swapped AAs:** number of AAs by which two peptides can differ and be considered chimeric. *Recommended value is 1.*

<u>Output files:</u>

For a standard run, exports will include:

- A formatted spectral file (if input was an unformatted .MS2 file, see above)
- Precursor AMM results: preliminary matches at the precursor level only
- A folder called “fragment_matches”, which will contain the fragments identified for each putative peptide-spectrum match (PSM)
- A formatted database and target peptide file (if input was an unformatted .FASTA file, see above)

- “Final_psm_report_out”: a file with all target and decoy peptides identified at the specified FDR

Troubleshooting:

The program displays “not responding” when I click a command:

This occurs when the program is conducting a time-consuming task, the program will begin to respond again when the task is complete.

Other issues:

This document will be routinely updated as user issues arise. If your issue is not displayed in this document, feel free to either contact the current maintainer (lawashburn@wisc.edu), or open an issue on the GitHub page for this program (https://github.com/lingjunli-research/EndoGenius/issues).