**Required Packages: biopython numpy pandas openpyxl time pickle itertools collections**

**1) Parse an annotation file into proteome and gene files.**

Annotation files here are genbank formatted, (.gb, .gbk, .gbff)

python parse\_genbank.py --gb wt\_ecoli.gbff --tt 11 –parse lt

--gb recognizes the file

--tt indicates the NCBI translation table, for bacteria, this is 11.

--parse indicates the IDs that should be used for parsing the annotation file into pairs of gene and protein sequences. This ID must be unique. Options include lt: locus\_tag, g: gene names, p: protein IDs.

*Using Proteome Discoverer perform a search with the produced proteome file. Export proteins from Proteome Discoverer to Excel. (Skip for tutorial).*

**2) Generate Custom Database**

python generate\_custom\_database.py --gen\_xlsx wt\_ecoli\_generic.xlsx --proteome\_file wt\_ecoli\_proteome.fasta --gene\_file wt\_ecoli\_genes.fasta

This will perform the internal sequence homology check and produce the mutant database.

--gen\_xlsx is the generic search .xlsx file exported from the previous step.

--proteome\_file is the generated proteome FASTA from parse\_genbank

--gene\_file is the generated FASTA file of the gene sequences from parse\_genbank

*Using Proteome Discoverer perform a search with the mutant database. Export peptide groups to an Excel document. (Skip for tutorial).*

**3) Perform Error Analysis**

This will perform filtering, pre-process peptides for chemical artifacts, correct for Ile/Leu, do the analysis, and generate both amino acid and codon substitution spectra. It will also determine the likely position of codon mistranslation and generate a stats file containing data for individual codons (the stats file is only generated automatically when processing a list of files). The resultant files with “processed” in the file name are the data which have been filtered to exclude peptides that contained a chemical artifact.

For processing of multiple files via a list:

python run\_error\_analysis.py --mut\_fasta ih\_mut\_custom\_wt\_ecoli\_proteome.fasta --file\_list wt\_ecoli\_file\_list.txt --gene\_file wt\_ecoli\_genes.fasta --protein\_file wt\_ecoli\_proteome.fasta --tt 11 --usage abs

For processing of a single file:

python run\_error\_analysis.py --mut\_fasta ih\_mut\_custom\_wt\_ecoli\_proteome.fasta --workbook\_input wt\_ecoli-1.xlsx --gene\_file wt\_ecoli\_genes.fasta --protein\_file wt\_ecoli\_proteome.fasta --tt 11 --usage abs

--mut\_fasta is the generated mutant fasta file

--workbook\_input is the exported peptide groups .xlsx file from the mutant database search

--file\_list is a .txt file of multiple exported peptide groups .xlsx files, with each file on a new line

--gene\_file is the generated FASTA file of the gene sequences from parse\_genbank

--proteome\_file is the generated proteome FASTA from parse\_genbank

--tt indicates the NCBI translation table, for bacteria, this is 11.

--usage is a string argument of either “abs” or “rel” indicating whether to use absolute codon usage values or relative values. Absolute codon usage is the fraction of usage of a specific codon over the total number of codons in CDSs. Relative usage is the fraction of usage of a codon within an amino acid family.

Note, non-standard trypsin products (NSPs) searching is somewhat time-consuming. NSPs that are found are saved to the custom mutant FASTA file and will improve search speeds during re-analysis of the data, or when there is an instance of an NSP appearing in multiple samples. So, try to avoid deleting the custom mutant file if you want to save time and not perform the NSP search again.