

# Comet/PAW Pipeline

John Klimek

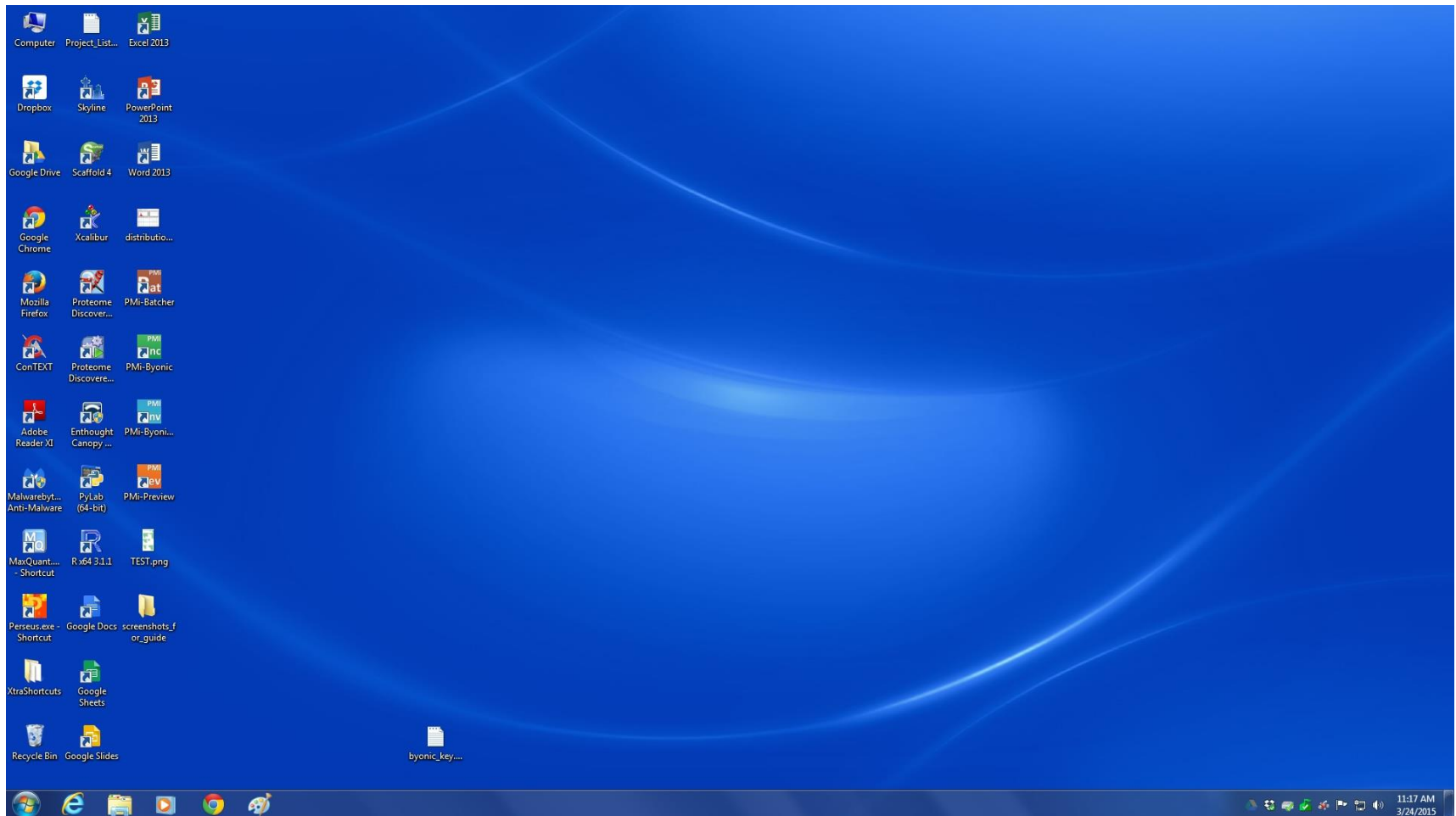
OHSU Proteomics Shared Resource

Mar 2015

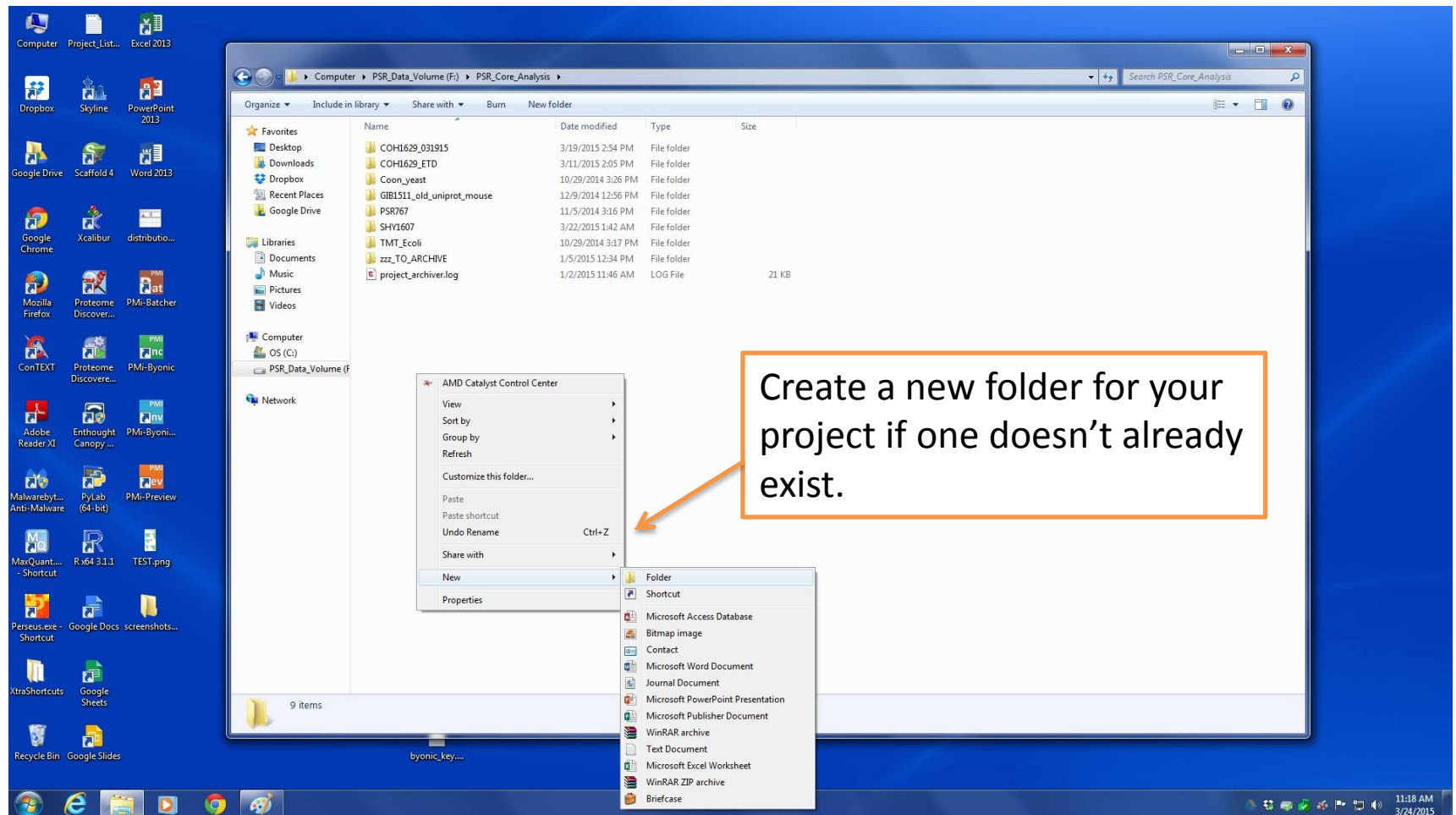
# Some notes before you start

- This outline is for analyzing data on the PSR computers. For other computers, you will need to install Python, Comet, and other software to use the workflow. You can contact PSR for a complete list of requirements.
- Three computers in room MRB521 are set up to use this pipeline. They are: MRBA608, MRBA610, & MRBA611.
- A whiteboard on the wall is set up to communicate computer usage. Please try to choose a computer that is less utilized, and note your project on the whiteboard if you plan to step away while your data is processing.
- Several of the programs will show a Windows dialog box upon starting them that will ask if it is okay to allow the program to run. You can press **ok**.
- When closing out a python program running a GUI, remember to close the GUI window first. If you do not, you may see warning when closing other windows.

The first step is to get your RAW data from the mass spectrometer onto a flash drive or other device and transfer those files to the computer you'll be doing the analysis on.

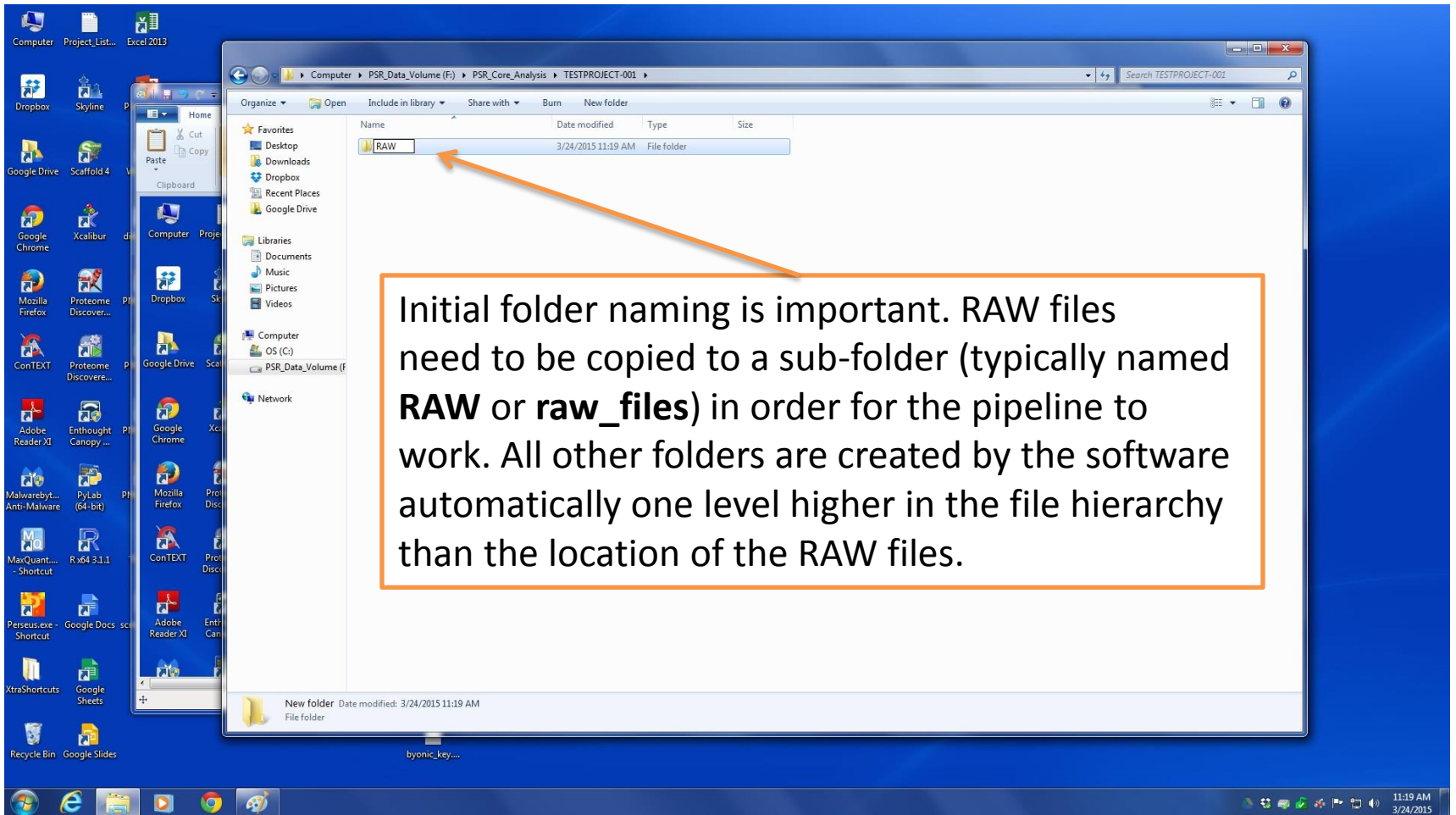


Each computer has a system hard drive (C), and a data analysis drive (F). You'll want to put the data on the **F:** drive. Data will need to be given a PSR project number and placed in the **F:\PSR\_Core\_Analysis** folder in order to be archived.

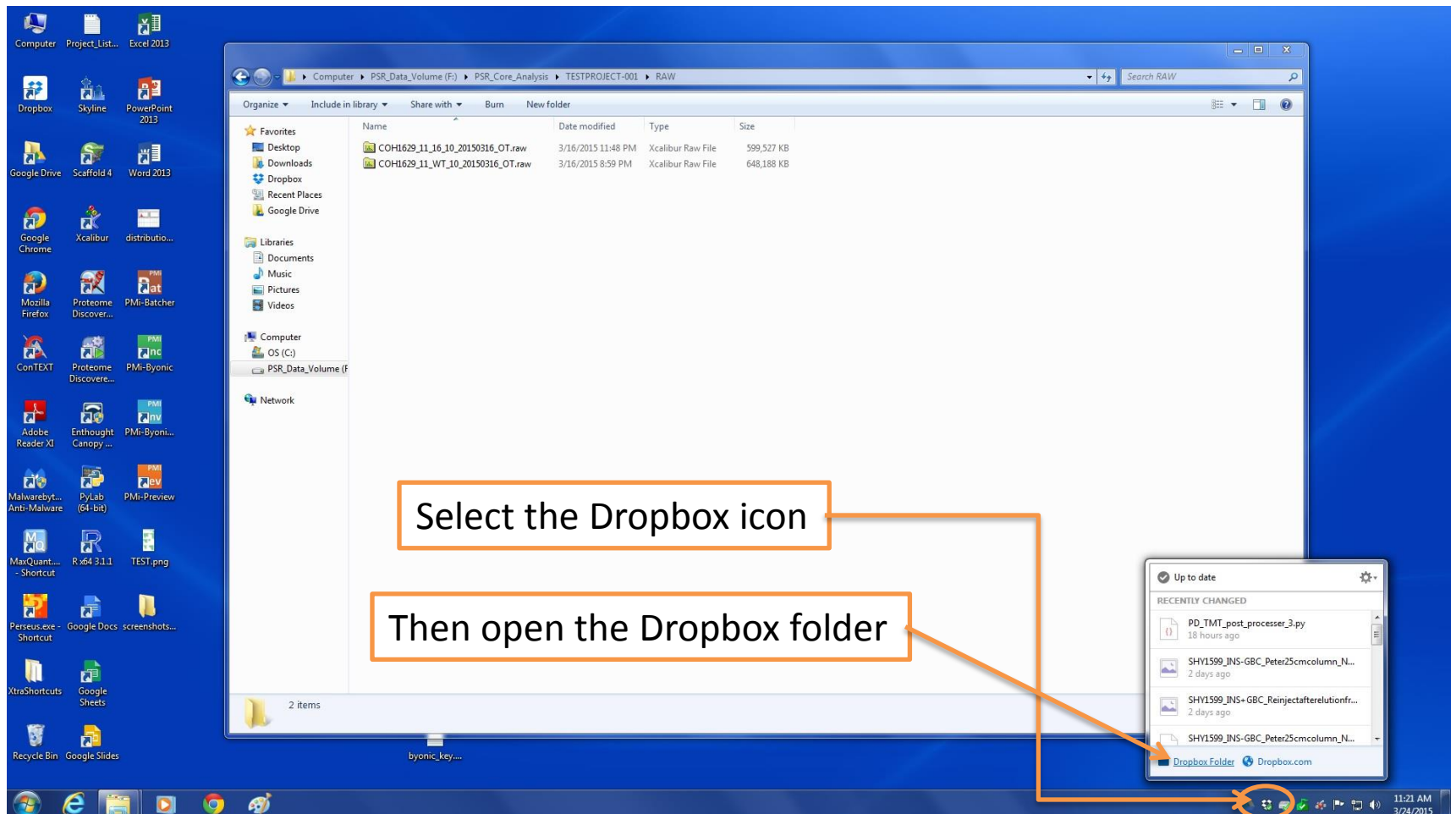




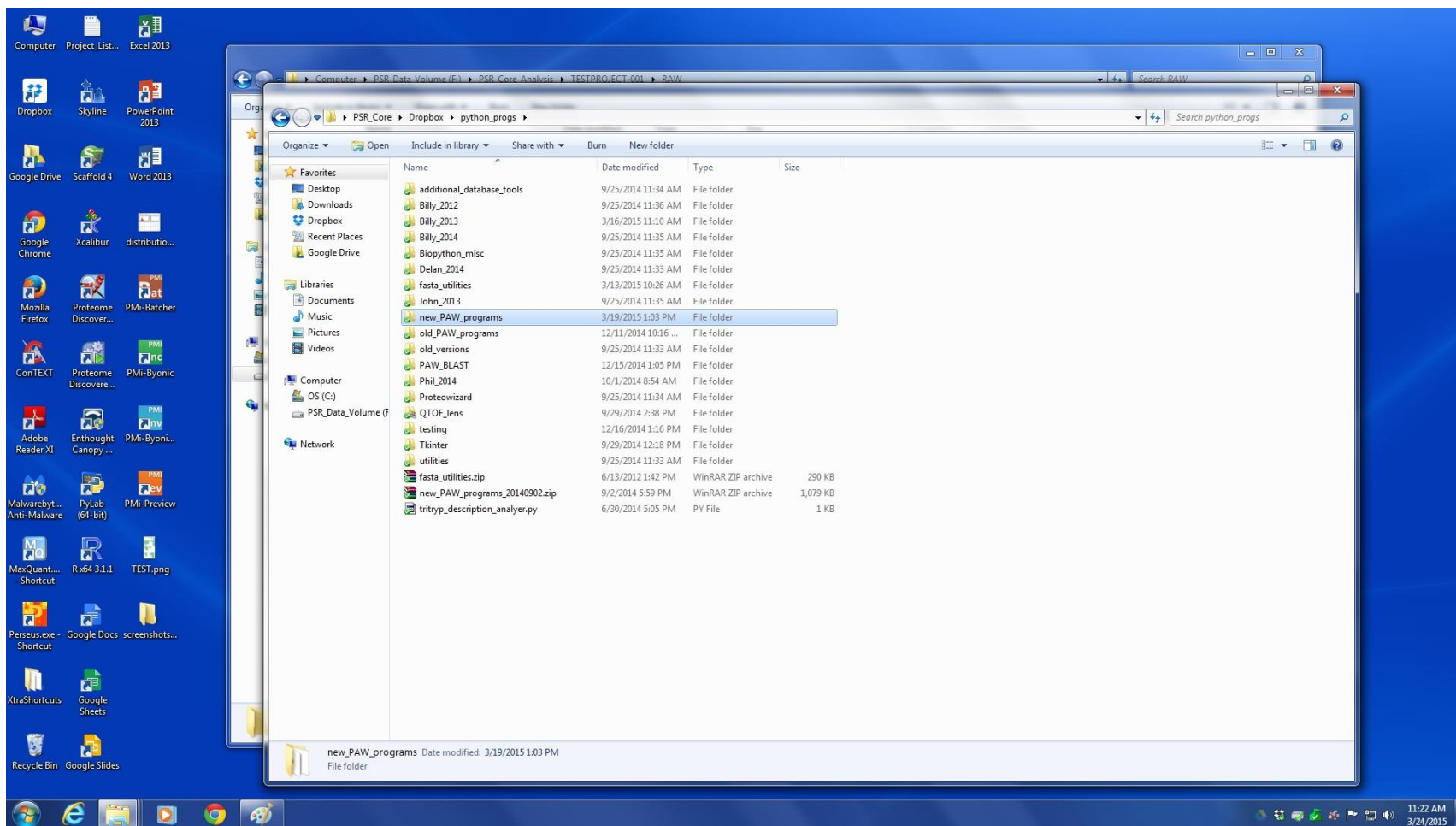
The folder name for a project should match the project ID in either ilab or Project Bubble. In either case, the code will start with the first 3 letters of the PI's last name. Project Bubble projects follow this with a 4-digit number (i.e. **DAV1620**), ilab projects will have a dash then the project number (i.e. **DAV-27**).



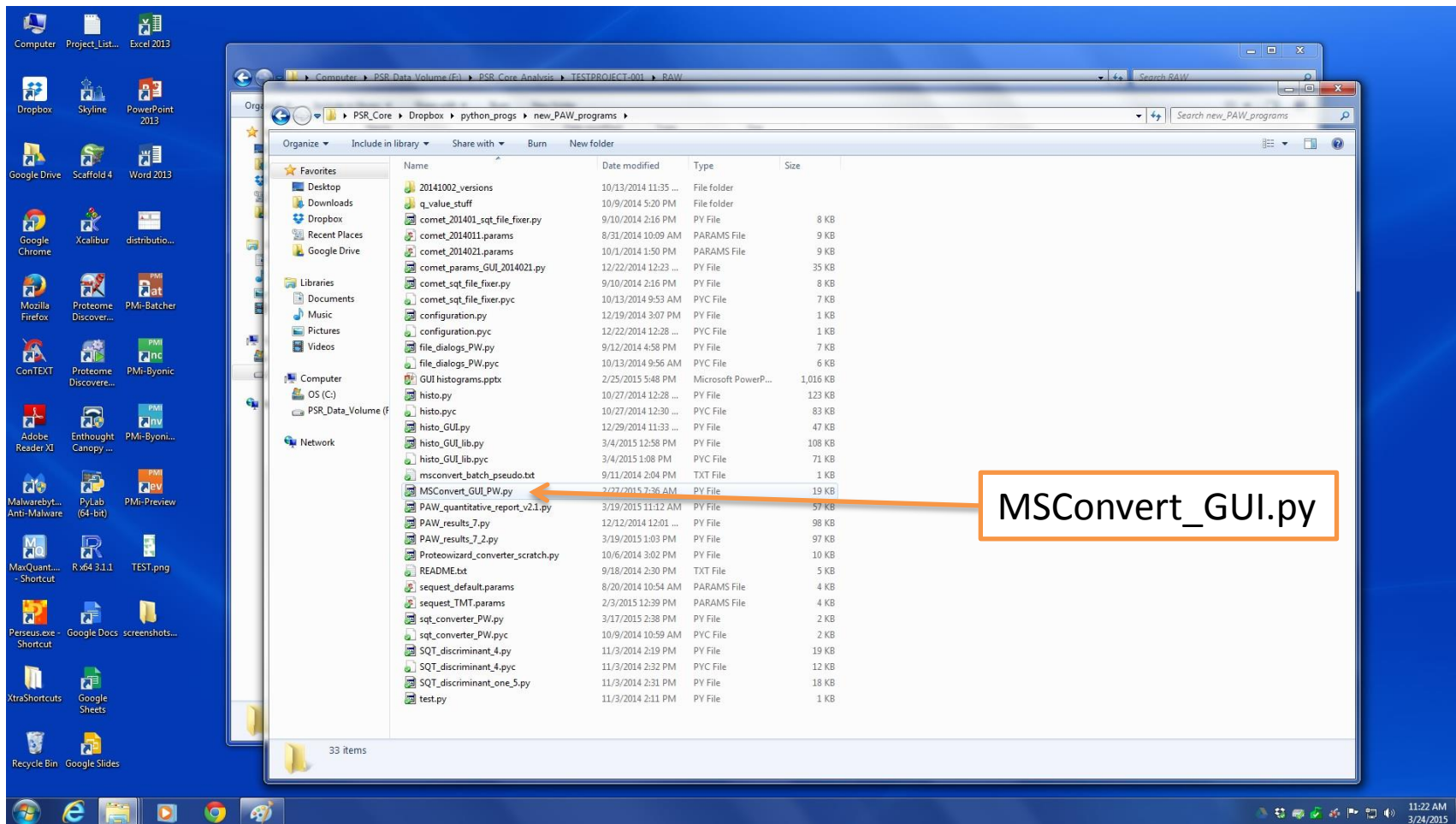
Once the RAW files are copied into the correct directory, then it's time to start analyzing the data. PAW Programs can be found in the **Dropbox** folder indicated below.



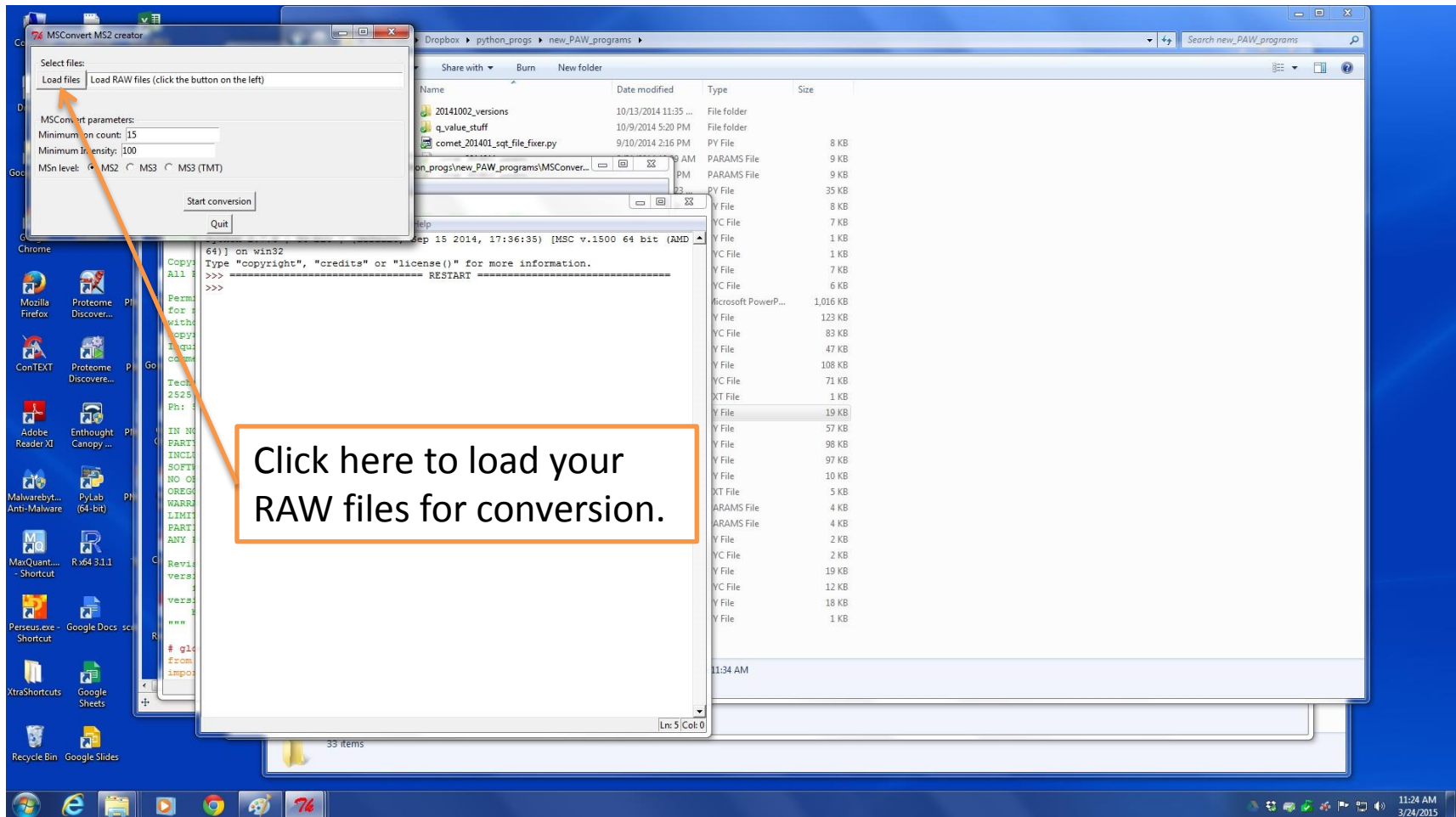
Once you open the **Dropbox** folder, open the **python\_progs** folder, and then the **new\_PAW\_programs** folder as shown below.



The first program to run is **MSConvert\_GUI.py**. This program will create ms2 files to be used by COMET, and place them in a **ms2\_files** folder. Double-click on the program to start it.

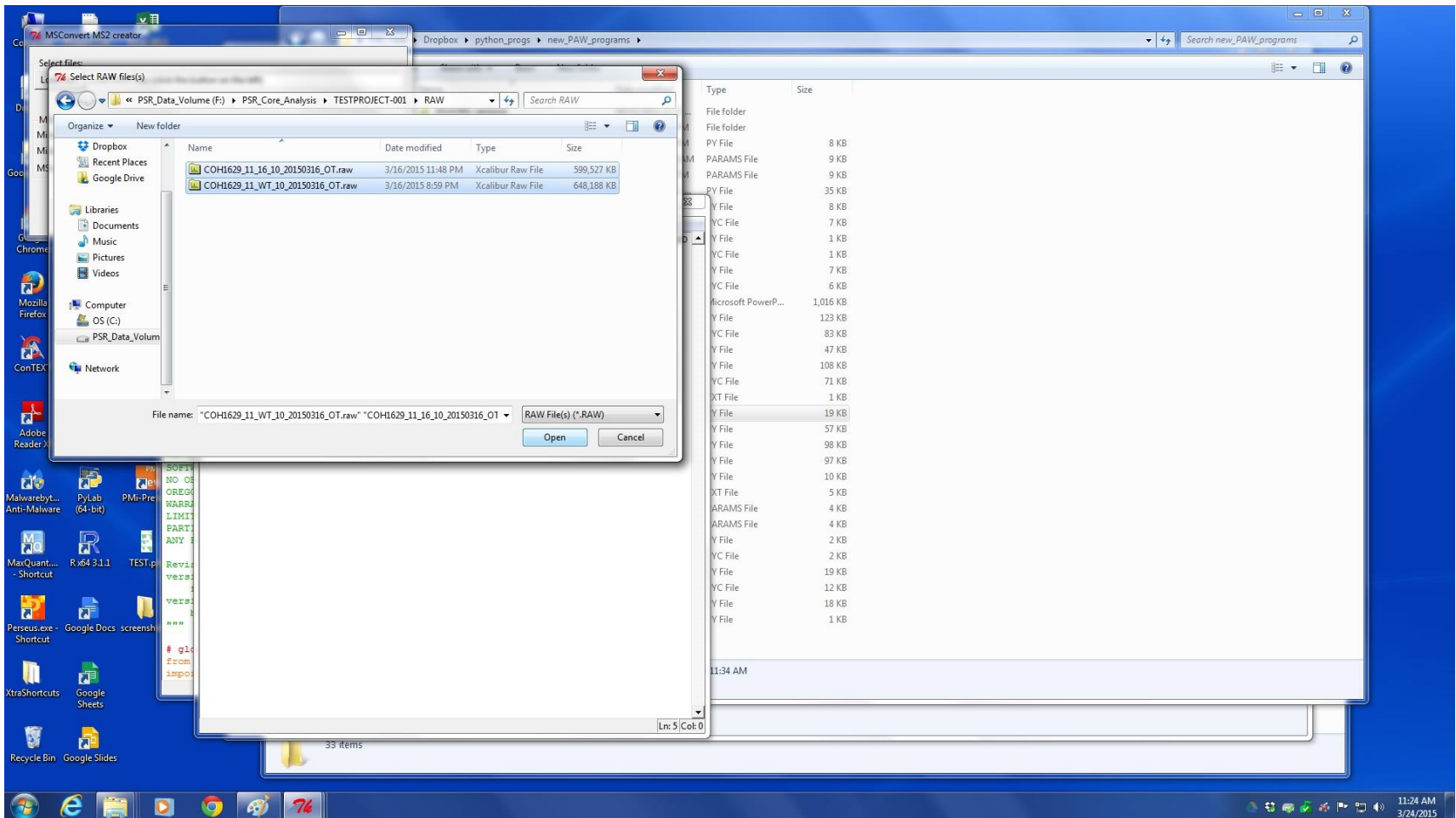


Once the program opens press **F5** to run (you shouldn't need to edit the source code). The GUI will pop open at this point. The MSConvert parameters should be OK in most cases, so you'll only be using the **Load files** and the **Start conversion** buttons.

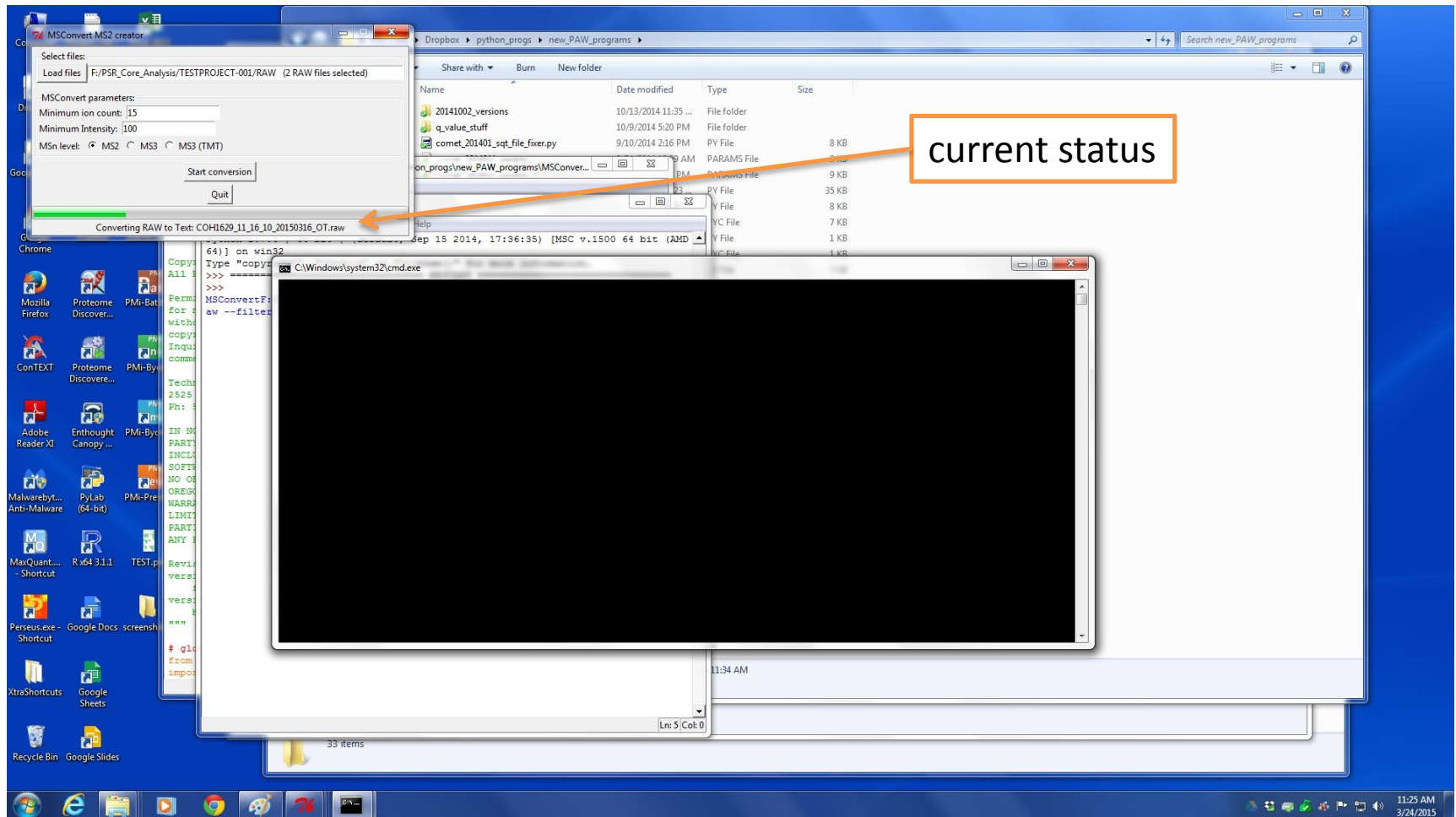




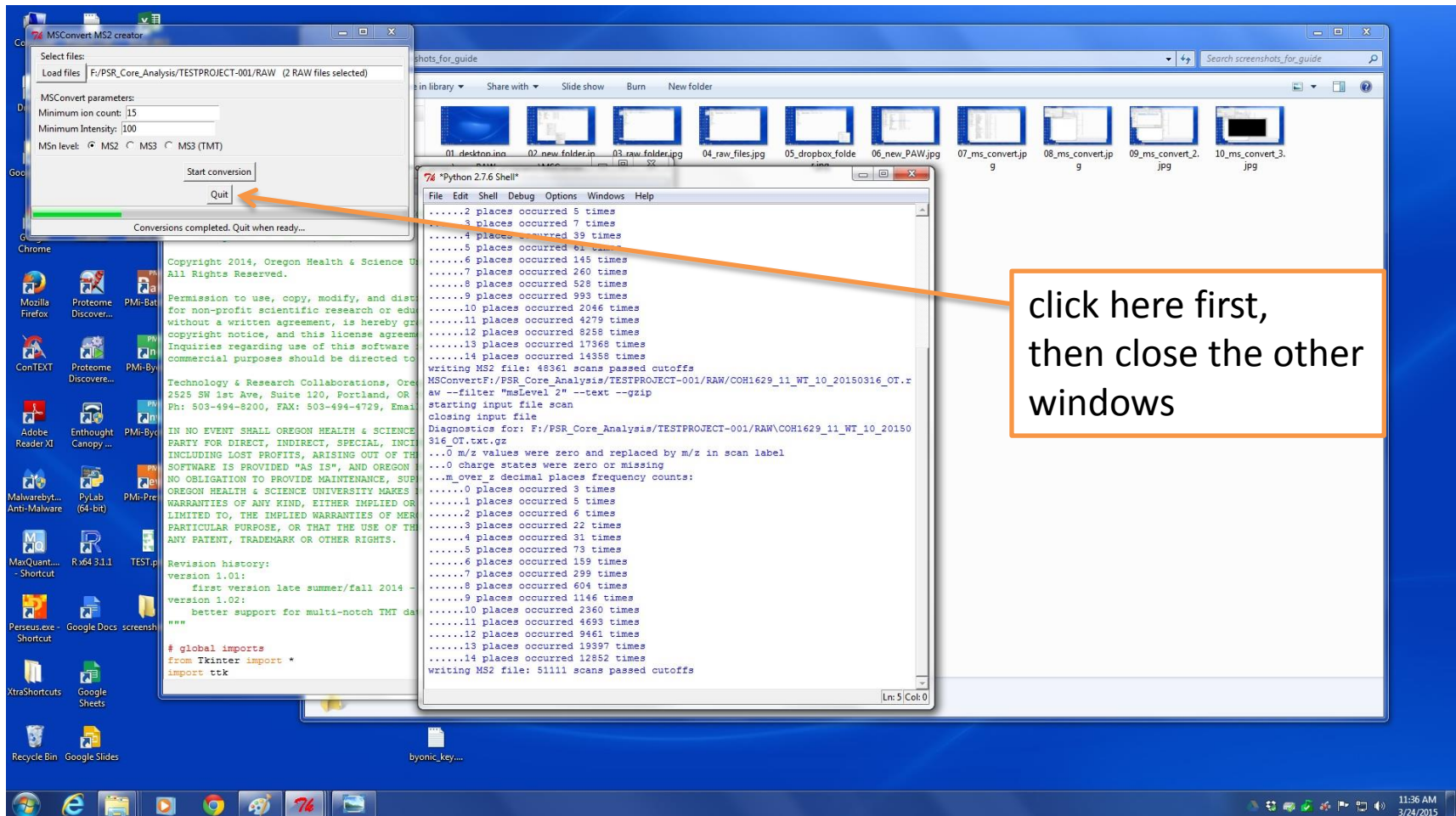
When the dialog window opens, navigate to your directory and select all the RAW files you wish to analyze. Then press **Open** to get back to the main GUI, and then press **Start Conversion** to process the files.



Once the conversion is started, a command window will pop up and the progress bar in the GUI will appear. The status is also listed at the bottom of the GUI window. The conversion can take several minutes for each RAW file, so you may want to step away for a few minutes while the data processes.

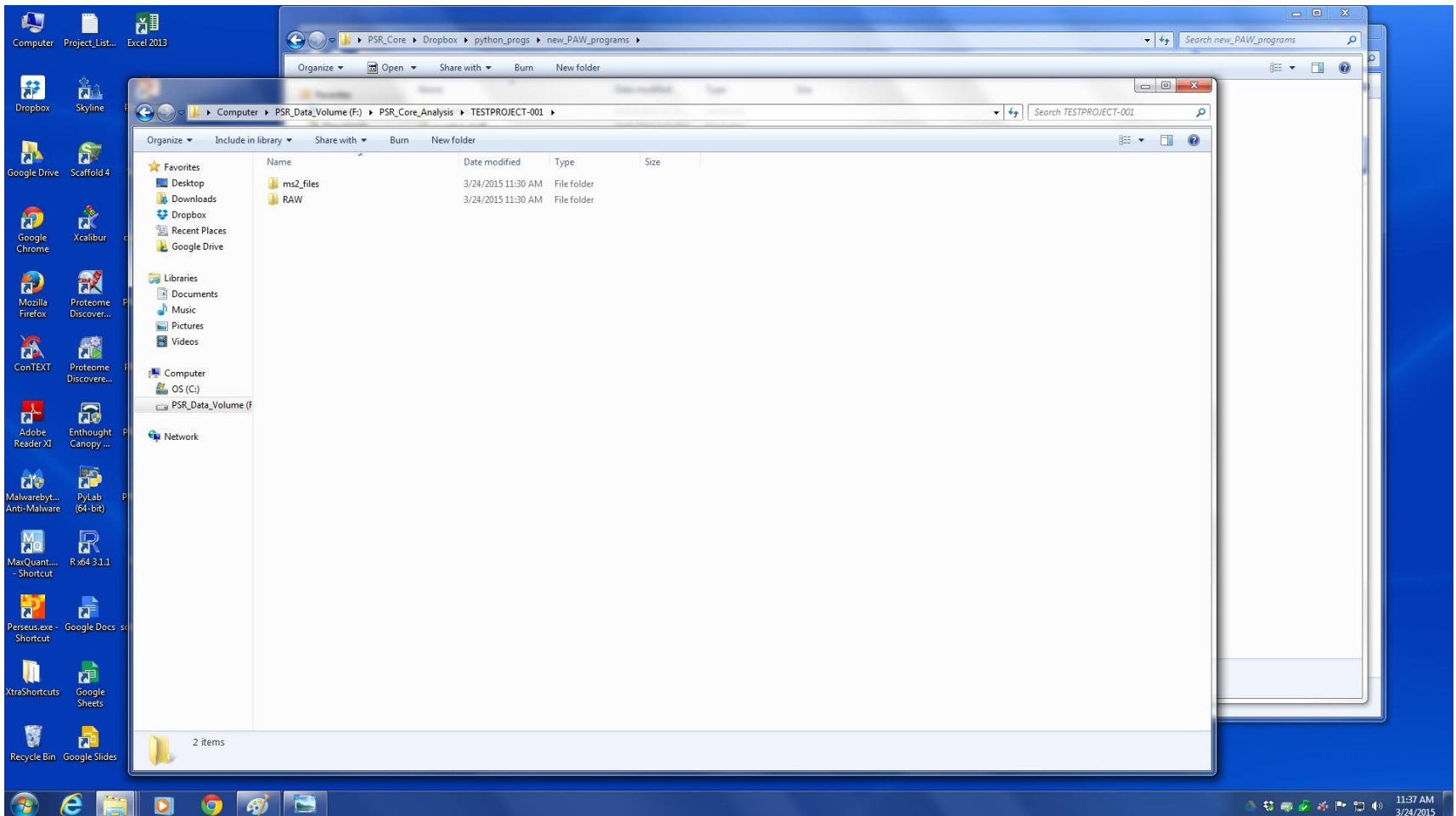


Once the conversion is complete, the windows will appear similar to below with some information about the conversion displayed, and the status bar in the GUI will read **Conversions completed. Quit when ready...** At this point, you can click the **Quit** button then close the other python windows.

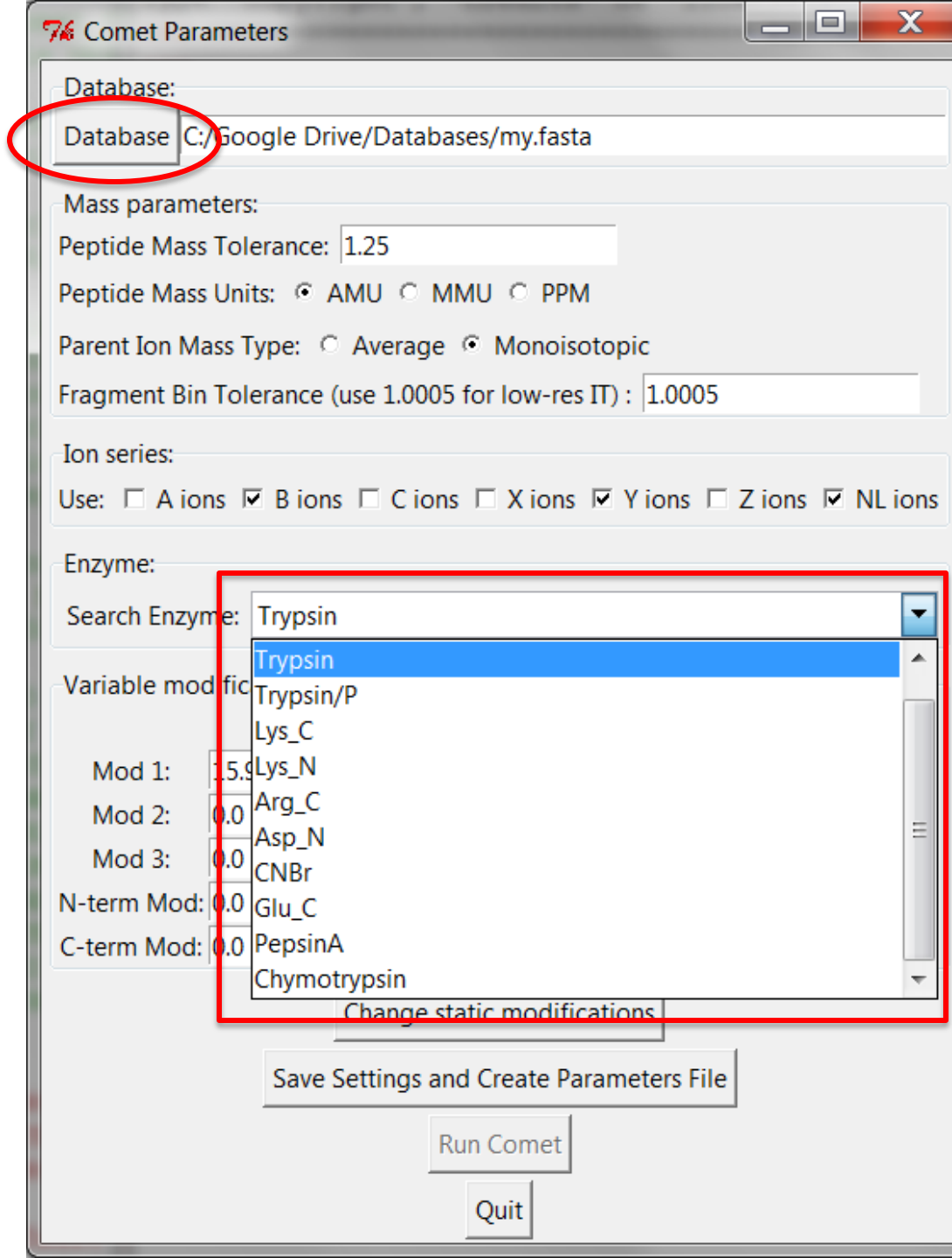




At this point, the project folder should look similar to below with two sub-folders: **ms2\_files** and **RAW**.



After ms2 file creation, Comet database searches can be run. Open the **comet\_GUI.py** program. It starts the same way as the others, double-click to open the program, and then press **F5** to run. A GUI window (see right) will open. Comet has many options. The GUI presents the most commonly used ones. Click the **Database** button and browse to the FASTA protein database location (usually on Google Drive). The Mass parameters pane is set for Orbitraps. Use **2.50**, **AMU**, and **Average** for low-resolution traps. Ion series is set for CID or HCD. Use **C** and **Z** ions instead of B and Y ions for ETD (neutral loss ions are OK). Select the enzyme used to digest your sample (usually **Trypsin**).



Variable modifications are specified next. The fewer mods the better. Oxidized Met is a common mod that happens during sample processing and is set by default. Accurate modification mass is important for Orbitraps. Consult with PSR staff for help. Static modifications are done through a separate pop up window (**Change static modifications** button). The default is alkylation of Cys (C+57). Once parameters are set, press the **Save Settings and Create Parameters File** button. A dialog box will pop up and ask you to locate the folder with the ms2 files created above. A **comet.params** file will be written in the folder with your ms2 files.

The screenshot shows the 'Comet Parameters' window. A red rectangle highlights the 'Variable modifications' section, which contains a table of modifications. A red oval highlights the 'Save Settings and Create Parameters File' button at the bottom. Other buttons visible include 'Change static modifications', 'Run Comet', and 'Quit'.

Database: C:/Google Drive/Databases/my.fasta

Mass parameters:

Peptide Mass Tolerance: 1.25

Peptide Mass Units: ☒ AMU ☐ MMU ☐ PPM

Parent Ion Mass Type: ☐ Average ☒ Monoisotopic

Fragment Bin Tolerance (use 1.0005 for low-res IT): 1.0005

Ion series:

Use: ☐ A ions ☒ B ions ☐ C ions ☐ X ions ☒ Y ions ☐ Z ions ☒ NL ions

Enzyme:

Search Enzyme: Trypsin

|             | Delta Mass | Residues |
|-------------|------------|----------|
| Mod 1:      | 15.9949    | M        |
| Mod 2:      | 0.0        | X        |
| Mod 3:      | 0.0        | X        |
| N-term Mod: | 0.0        | N-term   |
| C-term Mod: | 0.0        | C-term   |

Change static modifications

Save Settings and Create Parameters File

Run Comet

Quit

After the parameters file has been written, Comet searches can be launched by clicking the **Run Comet** button. The GUI window will lengthen to show a progress bar and status message. The console window will also display information during search progress. Black command windows will also come and go as the searches run (similar to the ms2 creation). After Comet searches are done, some post-processing of results will take place. The status message will let you know when everything is finished. Be patient, Comet searches and the post-processing can take several minutes to hours to complete depending on search parameters and database size/complexity.

Database:

Database //psf/Google Drive/Databases/special\_sequences/10kDa\_culture\_fit

Mass parameters:

Peptide Mass Tolerance: 1.25

Peptide Mass Units: ☒ AMU ☐ MMU ☐ PPM

Parent Ion Mass Type: ☐ Average ☒ Monoisotopic

Fragment Bin Tolerance (use 1.0005 for low-res IT) : 1.0005

Ion series:

Use: ☐ A ions ☒ B ions ☐ C ions ☐ X ions ☒ Y ions ☐ Z ions ☒ NL ions

Enzyme:

Search Enzyme: Trypsin

Variable modifications:

|             | Delta Mass | Residues |
|-------------|------------|----------|
| Mod 1:      | 15.9949    | M        |
| Mod 2:      | 0.0        | X        |
| Mod 3:      | 0.0        | X        |
| N-term Mod: | 0.0        | N-term   |
| C-term Mod: | 0.0        | C-term   |

Change static modifications

Save Settings and Create Parameters File

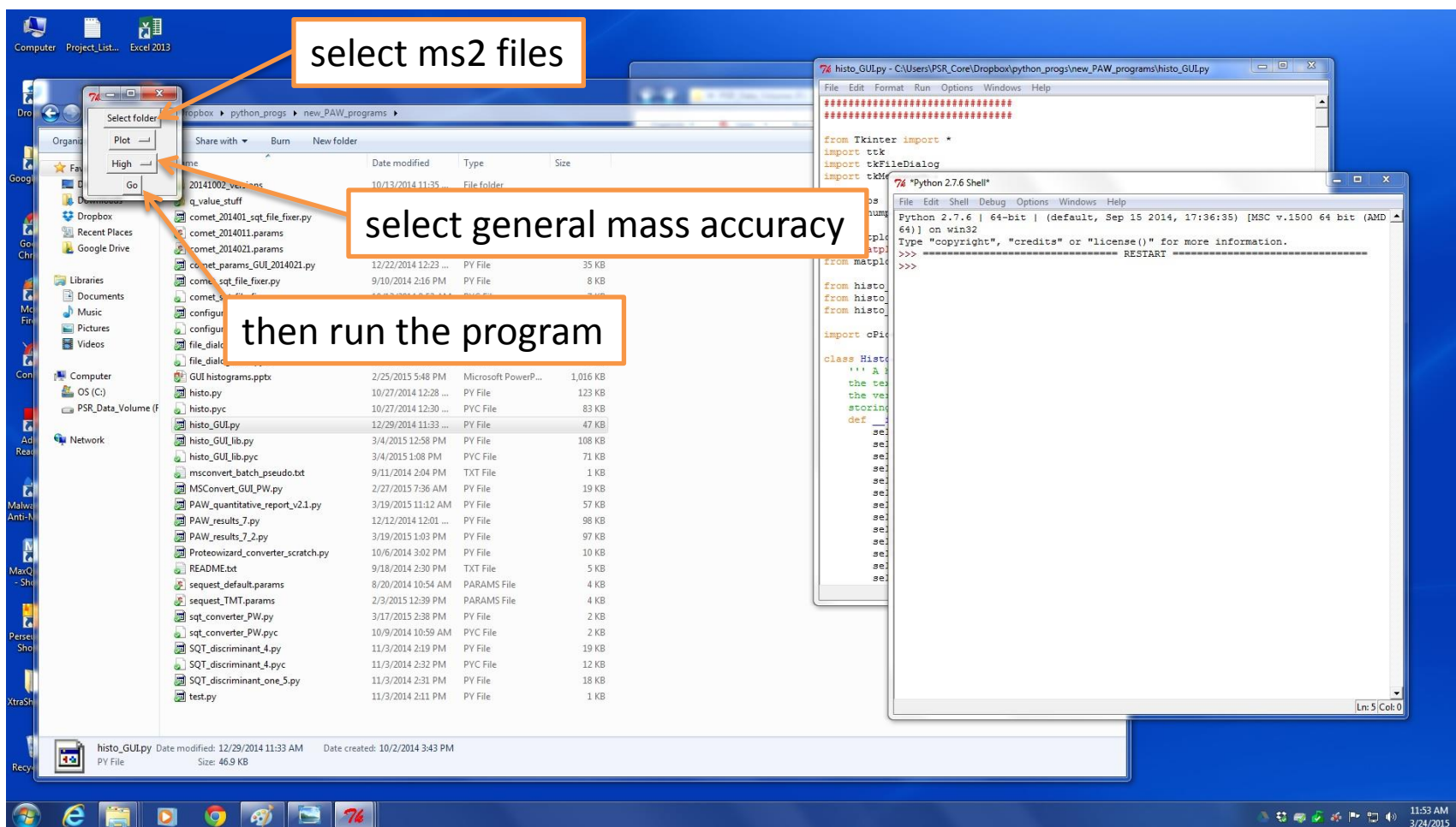
Run Comet

Quit

**NOTE:** The histogram viewing GUI program described next was written for SEQUEST searches (which are very similar to Comet searches) and requires a **sequest.params** file. A mock **sequest.params** file must be created at this point in time. Parsing of **comet.params** files is planned for the near future. Consult PSR staff for help with this step.

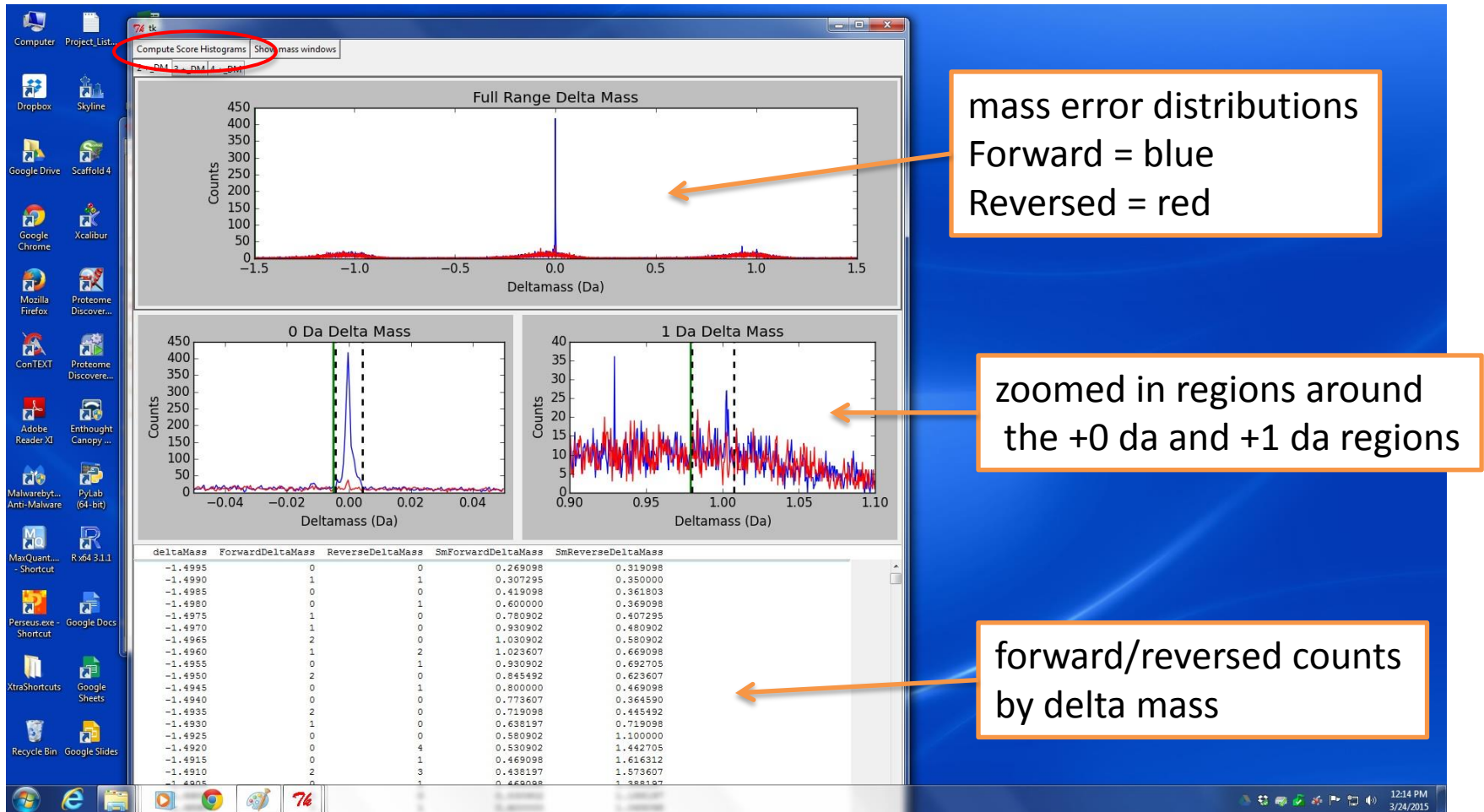
The **comet\_GUI.py** program makes configuring and running basic Comet searches pretty easy. It just executes some Windows command line calls and runs a couple of other Python programs. These steps can also be done manually and are described in slides at the end of the main tutorial. This can be handy when repeating a search with different parameters, or if more advanced Comet parameters are needed.

The next pipeline step is **histo\_GUI.py**. It starts the same way as the others, double-click to open the program, and then press **F5** to run. A small GUI window will open. You can use the top button to select the **ms2\_files** folder you wish to analyze, and the bottom button is for toggling between high mass accuracy (Orbitrap) data, and low mass accuracy data (everything else).





The GUI will open after a short loading time, and look something like below. The first step is to set the mass windows. If you are using low mass accuracy data you can skip this step and simply press the **Compute Score Histograms** button to jump ahead to the next window.



The first step in selecting the mass windows is to click on the tab for the charge state you wish to set (thresholds should be checked/adjusted for all charge states).

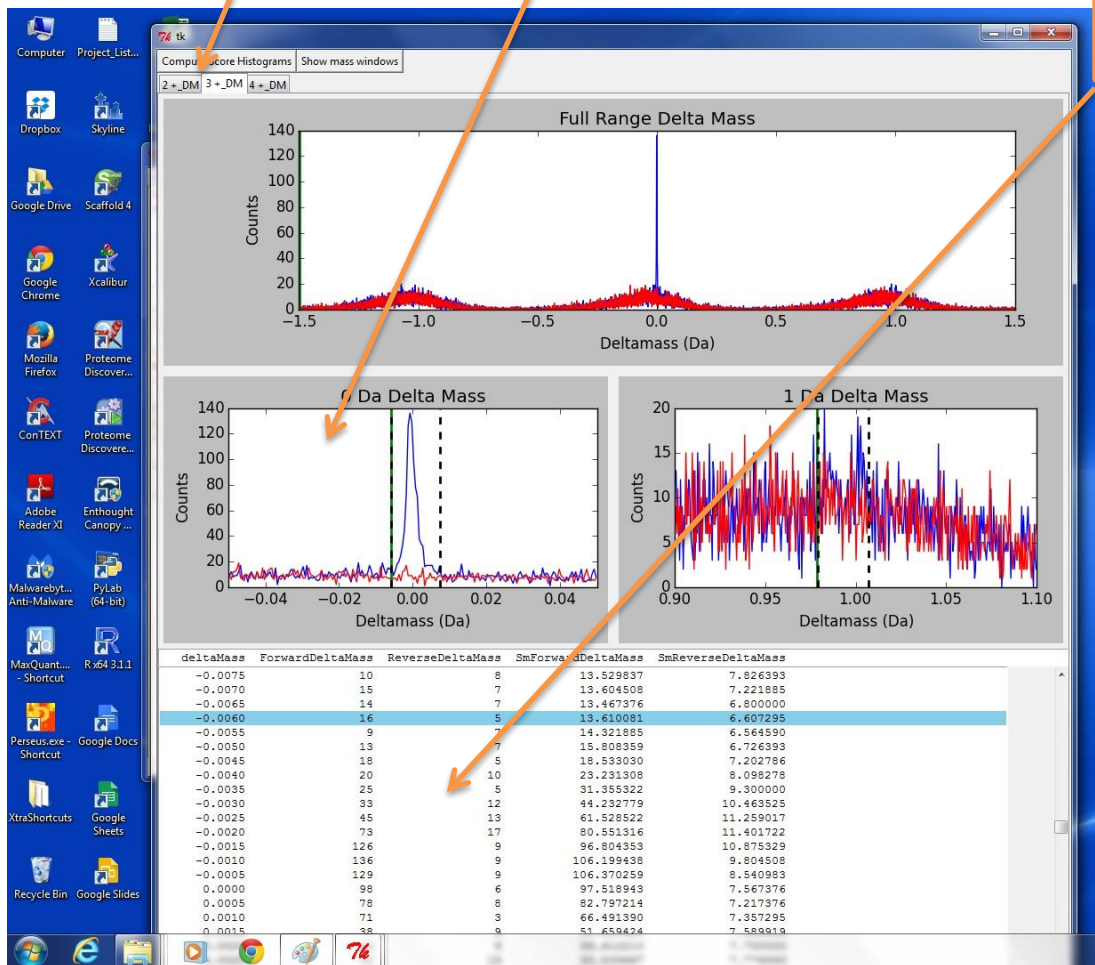
Next click on the window you want to edit (0 Da or 1 Da).

Then click on the table at the bottom.

Once the table is selected, the left and right arrow keys toggle between the upper and lower thresholds; the up and down arrow keys adjust the threshold higher or lower.

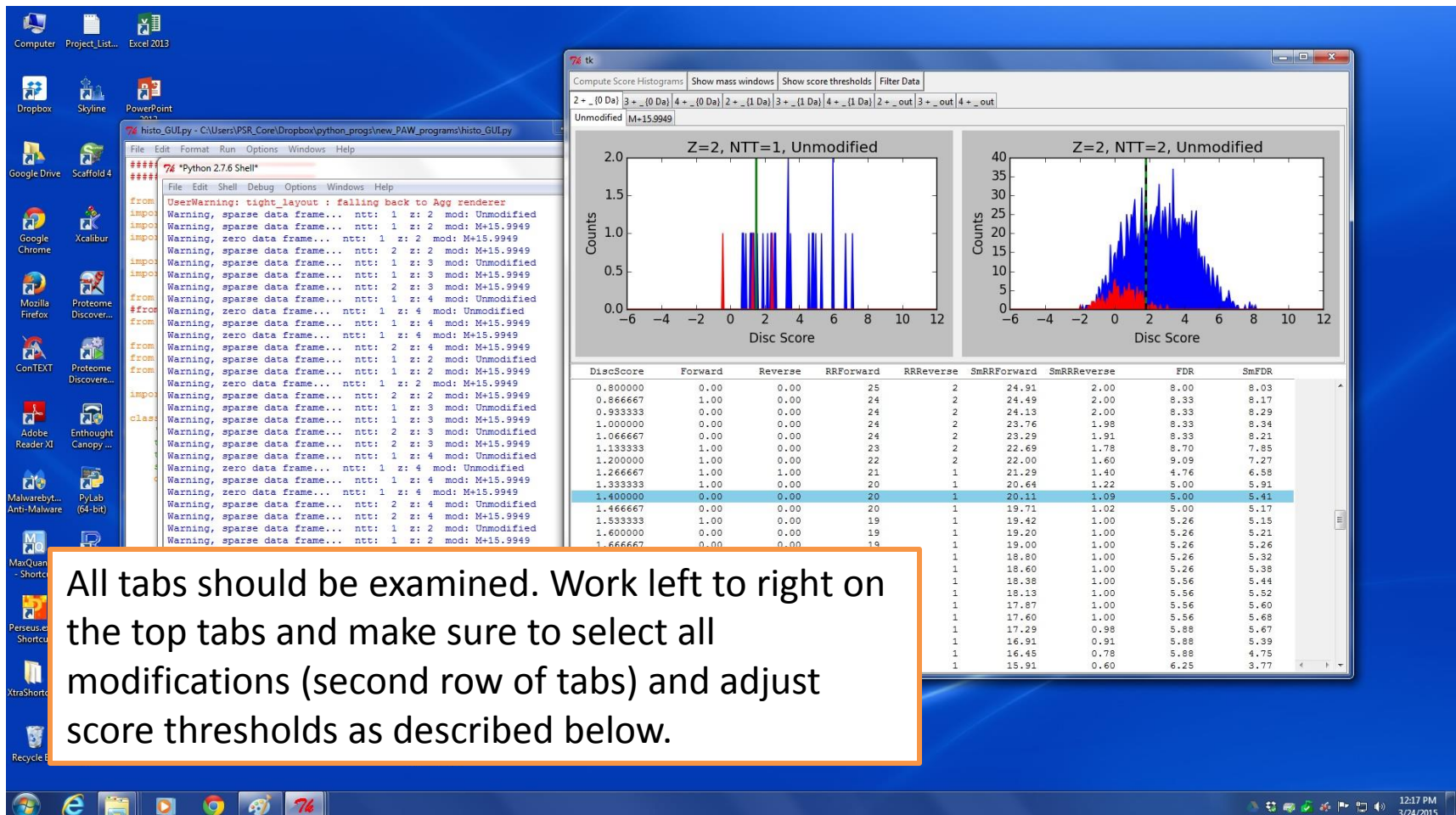
The **enter** key updates the threshold to the current dotted line location. You want to capture just the blue peaks associated with the forward distributions between the thresholds. The default windows may be OK.

Once all the thresholds are set, click **The Compute Scores Histograms** button to create the search score histograms.





After the mass accuracy filtering, next is filtering by discriminate scores. The tabs and windows work in similar ways as the mass accuracy window, with the exception of there only being a single minimum threshold for each plot. There are two rows of tabs above the score histograms. The top are different charge states and delta mass windows. The bottom tabs are for different modification states. There will be multiple score histograms (plots) for different enzymatic cleavage states.

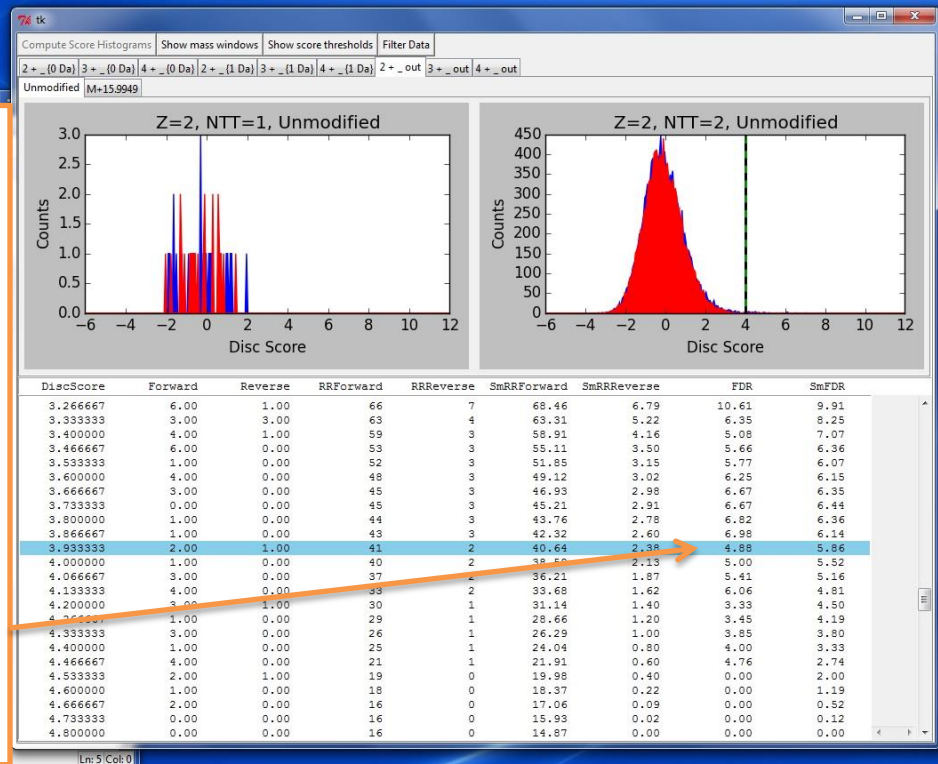


All tabs should be examined. Work left to right on the top tabs and make sure to select all modifications (second row of tabs) and adjust score thresholds as described below.

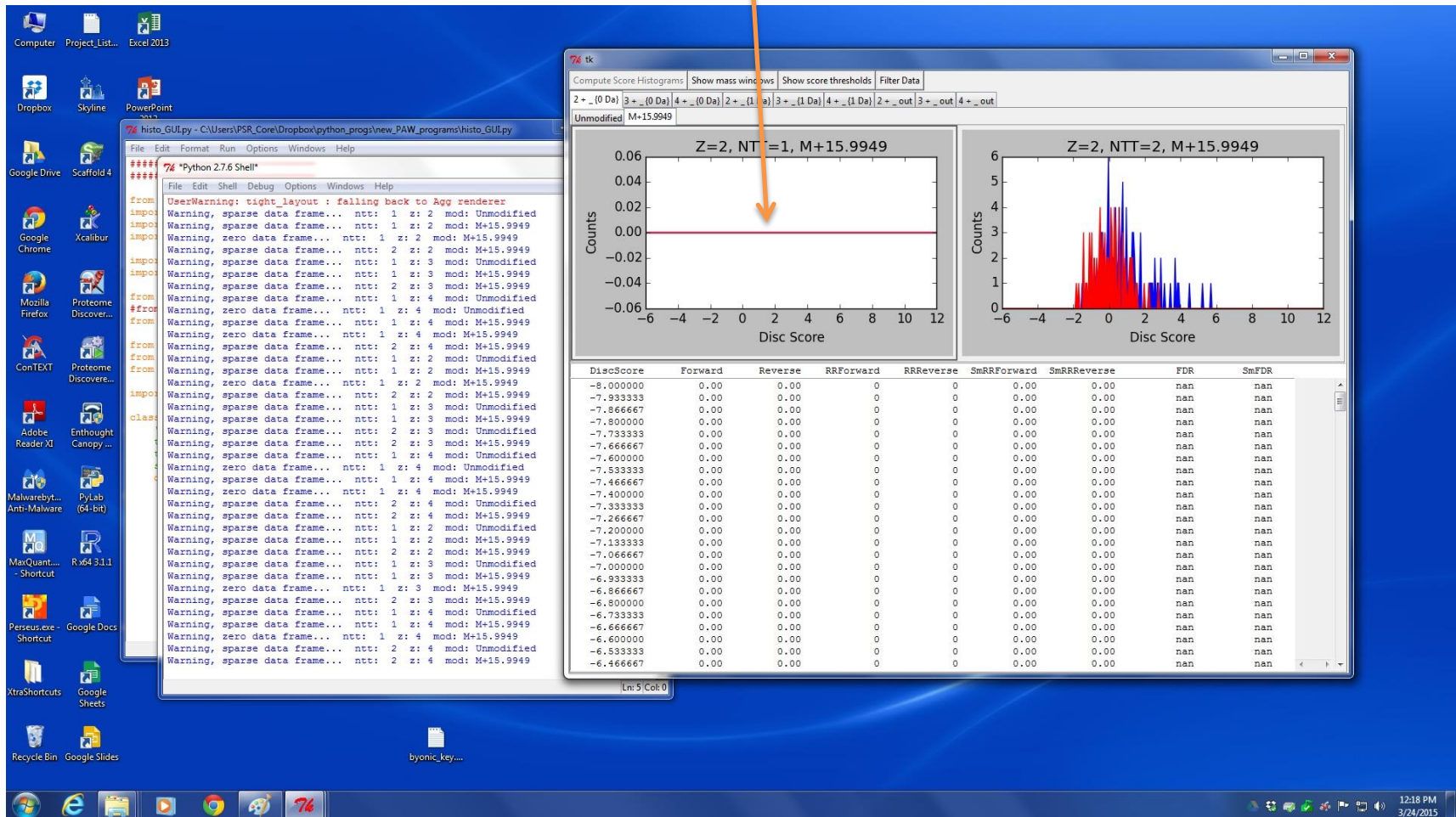
The goal when setting peptide thresholds is to end up with a reasonable final protein false discovery rate (FDR). How a peptide FDR translates to a protein-level FDR will vary depending on the size of the database and the complexity of your samples. What follows are some basic guidelines.

The default threshold locations are at 1% peptide FDR, which works well for larger datasets (MudPITs). For samples with single LC runs per sample, a lower peptide FDR like 5% may be better.

On the right is an example of a filter set at just under 5% (4.88%).



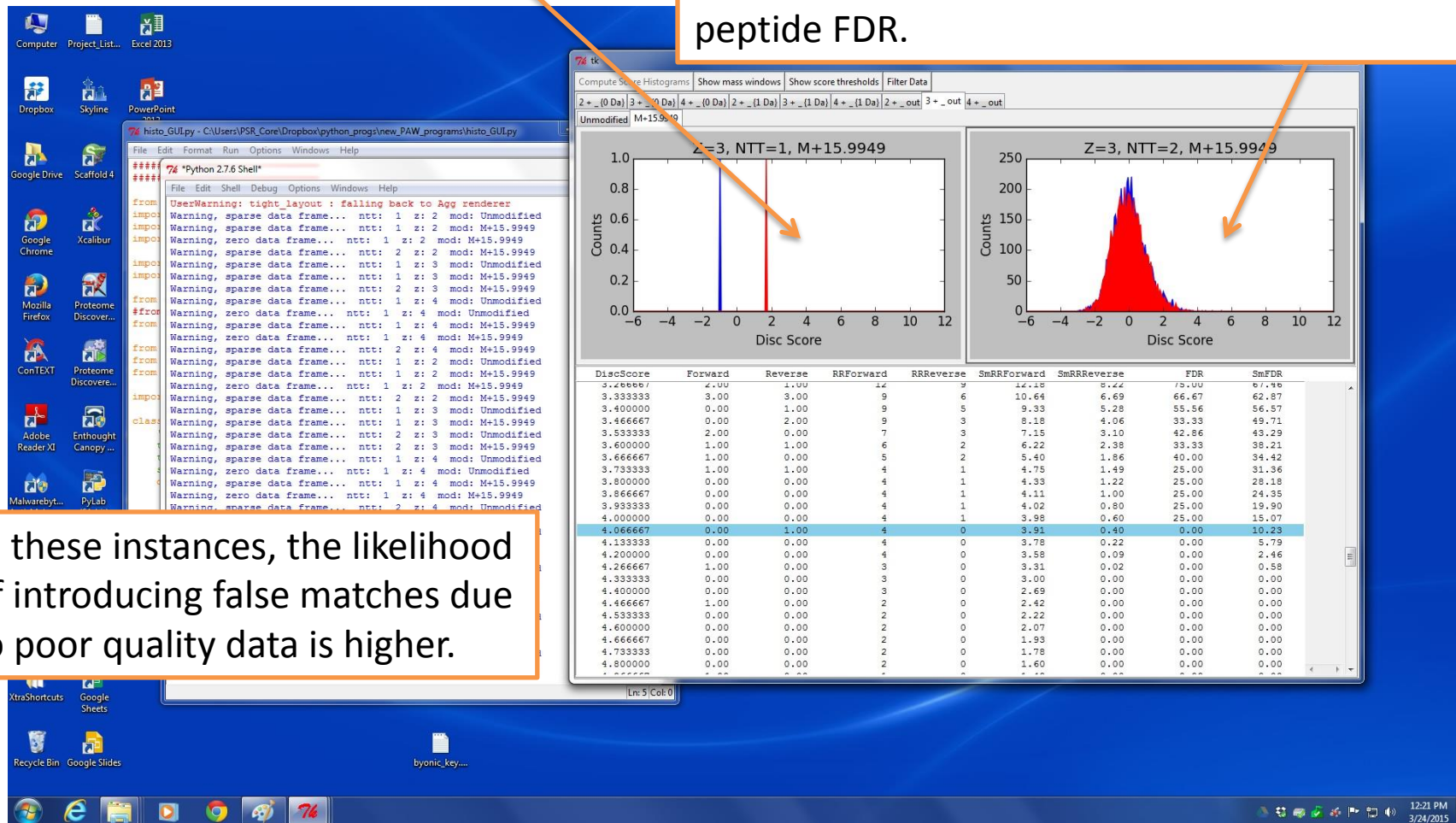
To accurately control peptide sequence errors, peptides are divided into many classes depending on accurate mass, search parameters, and charge states. In many datasets, several of these classes may be empty or have very few estimated correct identifications. There will be no dotted threshold lines for these classes.





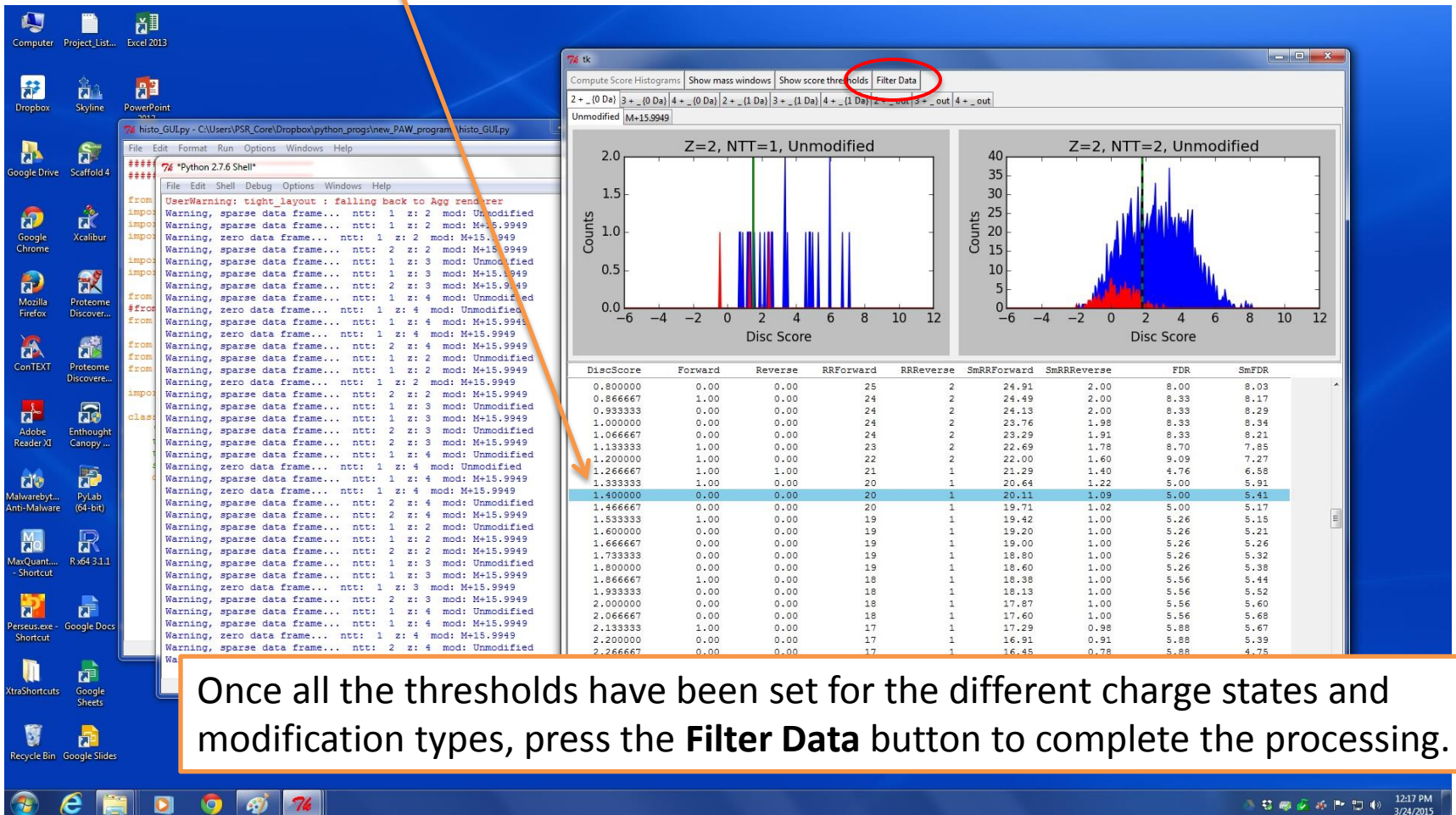
In addition to empty peptide classes, you can ignore sparse (nearly empty) classes such as the one on the left.

It is also a good idea to exclude peptide classes with few forward matches above the desired peptide FDR.



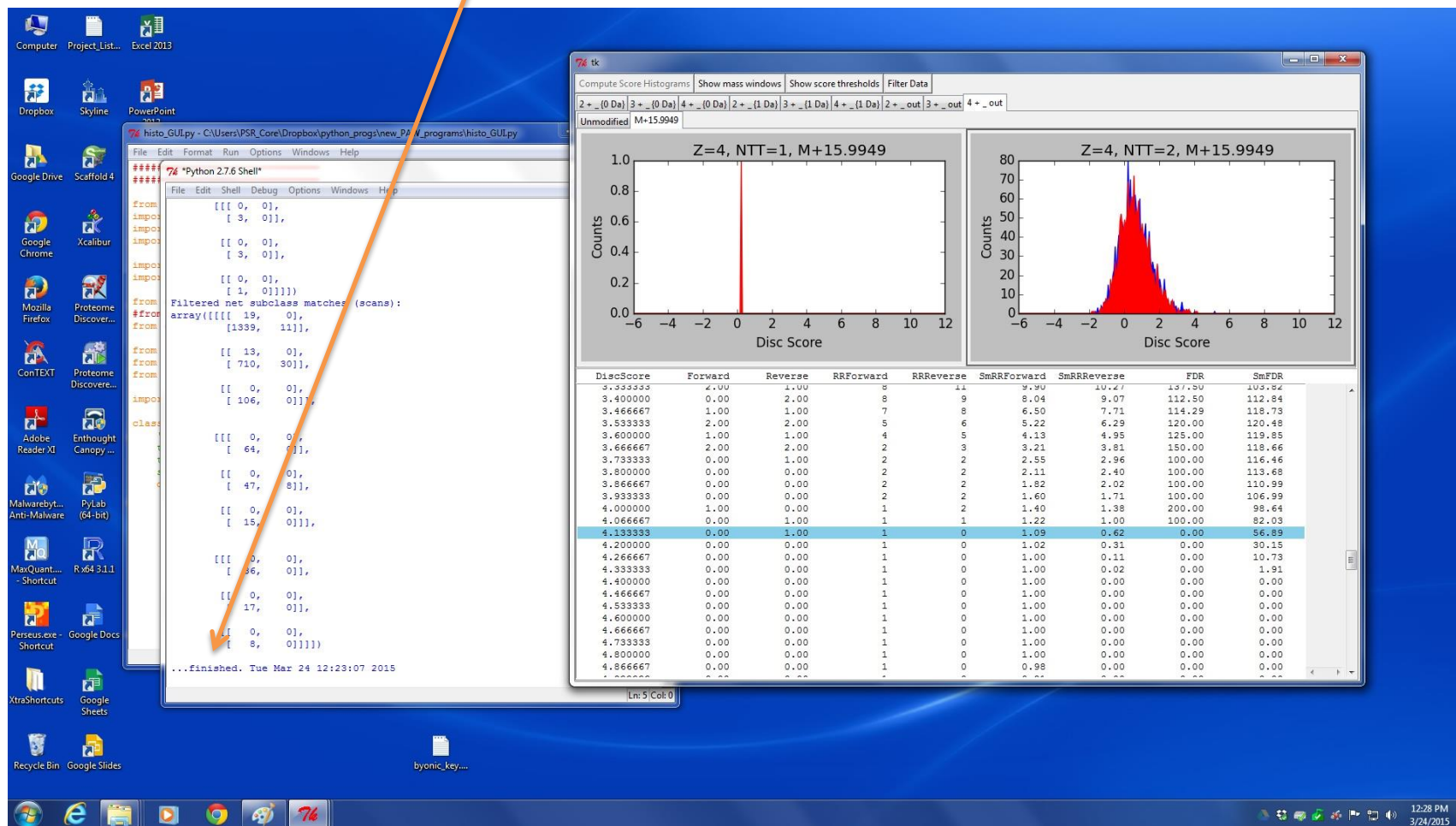
In these instances, the likelihood of introducing false matches due to poor quality data is higher.

Finally, setting a minimum threshold for the discriminate score (like the 1.4 that is used below) is also a good idea. These very low scores would likely fail manual inspection, and wouldn't meet accepted proteomics publication standards.



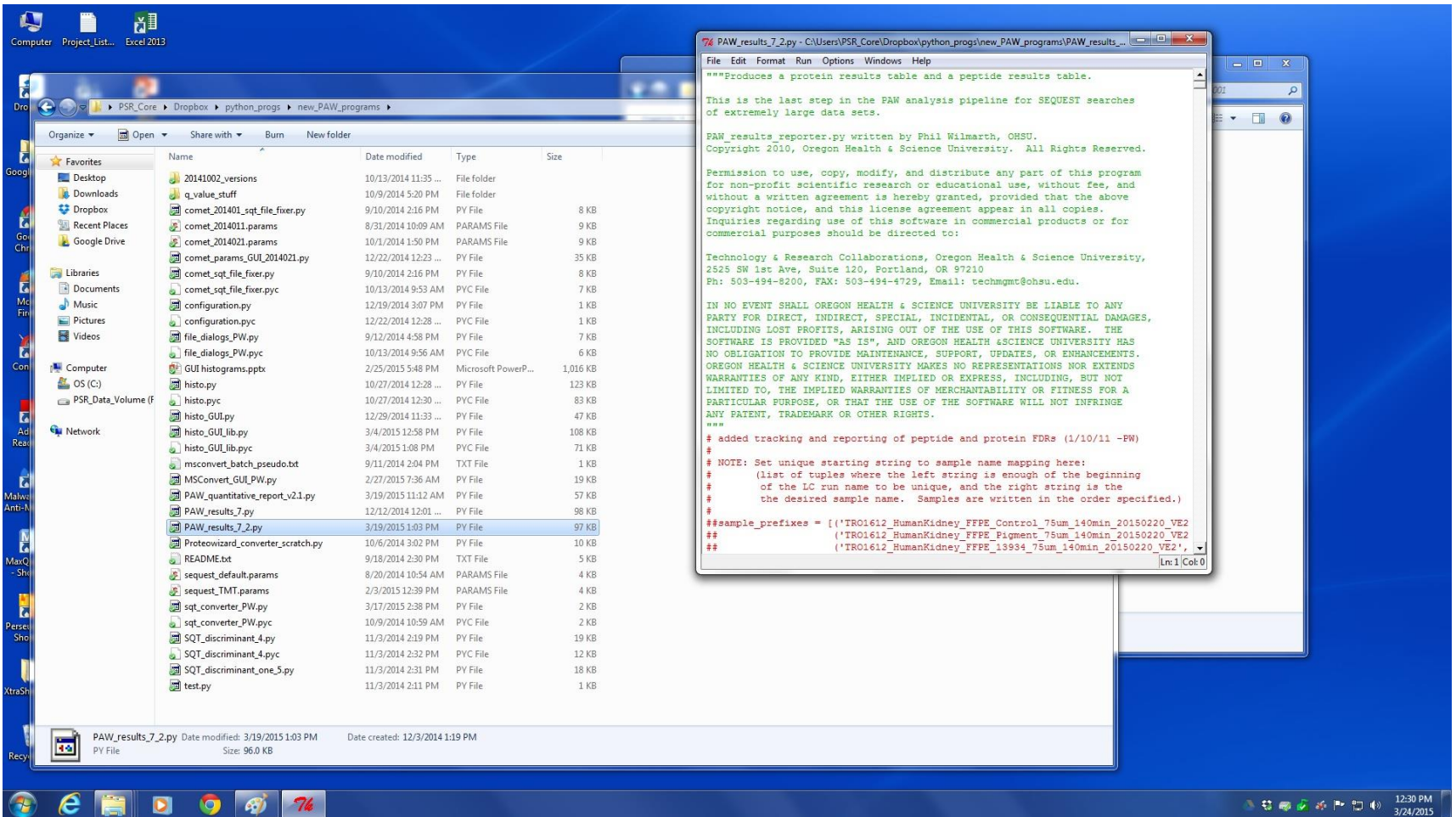
Once all the thresholds have been set for the different charge states and modification types, press the **Filter Data** button to complete the processing.

Filtering will usually take about 20-30 seconds per file (or longer) to process and write out the filtered files. When it is finished, the python shell will display **...finished.** You can close the GUI and python windows to exit.





The final step in the analysis pipeline in the **PAW\_results\_7.py** program (note: the number at the end may change with new versions). Unlike with previous programs you'll need to edit the source code after opening the program file. If you are unfamiliar with Python coding, consult with the PSR staff for assistance.

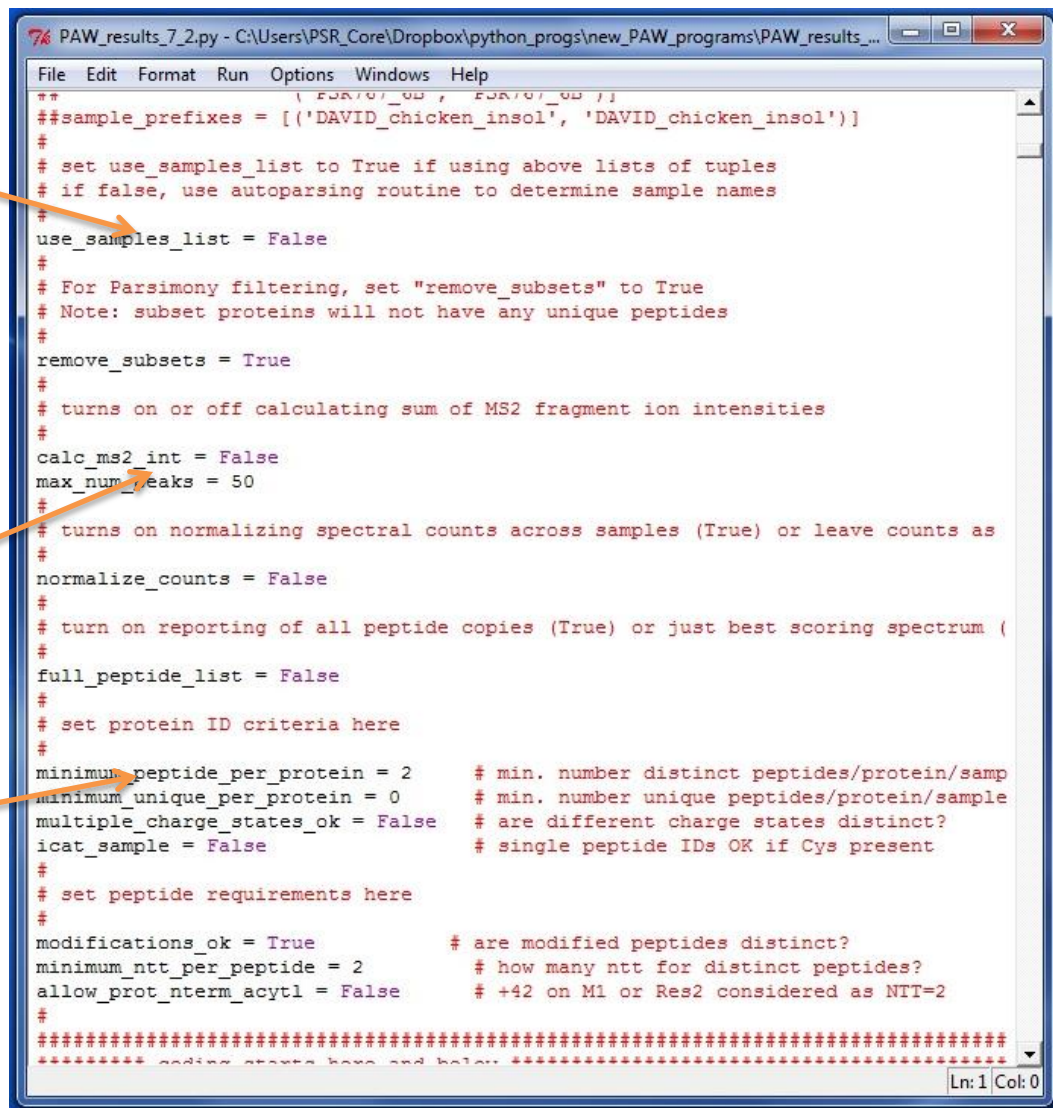


Below the initial list of sample names (which will be covered momentarily), there's a couple of potential flags that are commonly changed and should be checked and set properly before running the program.

The program will automatically try and determine how to combine the different files into sample sets. If you need to bypass that logic and define your sample names manually, set this to **True**.

This should be set to **False** unless you're specifically doing an ms2 intensity quantitative project.

This should be set to **2** (peptides per protein) for most analyses.



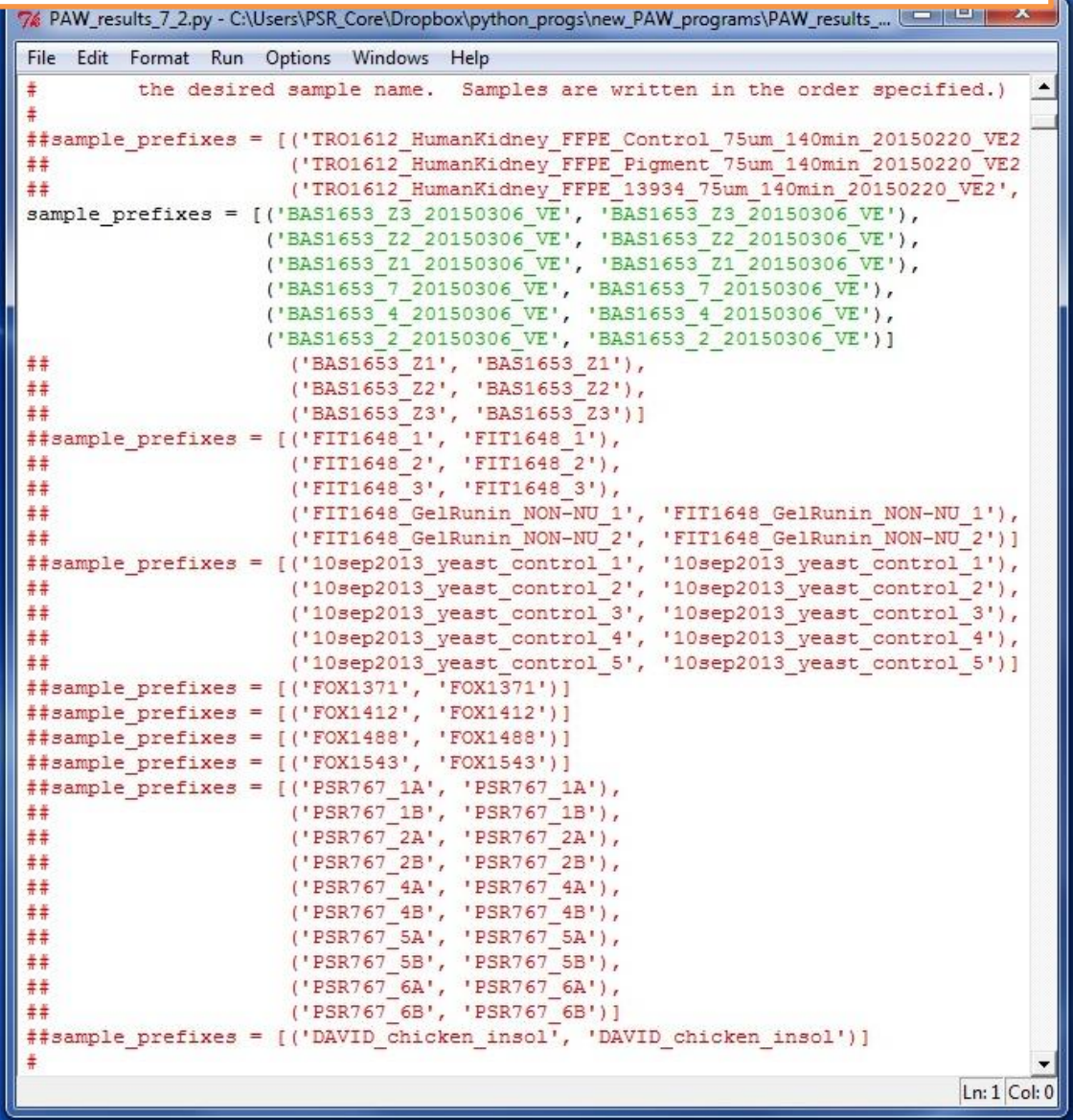
```
PAW_results_7_2.py - C:\Users\PSR_Core\Dropbox\python_progs\new_PAW_programs\PAW_results_...
File Edit Format Run Options Windows Help
** (FOR /G /QD, FOR /G /QD)
##sample_prefixes = [('DAVID_chicken_insol', 'DAVID_chicken_insol')]
#
# set use_samples_list to True if using above lists of tuples
# if false, use autoparsing routine to determine sample names
#
use_samples_list = False
#
# For Parsimony filtering, set "remove_subsets" to True
# Note: subset proteins will not have any unique peptides
#
remove_subsets = True
#
# turns on or off calculating sum of MS2 fragment ion intensities
#
calc_ms2_int = False
max_num_peaks = 50
#
# turns on normalizing spectral counts across samples (True) or leave counts as
#
normalize_counts = False
#
# turn on reporting of all peptide copies (True) or just best scoring spectrum (
#
full_peptide_list = False
#
# set protein ID criteria here
#
minimum_peptide_per_protein = 2 # min. number distinct peptides/protein/samp
minimum_unique_per_protein = 0 # min. number unique peptides/protein/sample
multiple_charge_states_ok = False # are different charge states distinct?
icat_sample = False # single peptide IDs OK if Cys present
#
# set peptide requirements here
#
modifications_ok = True # are modified peptides distinct?
minimum_ntt_per_peptide = 2 # how many ntt for distinct peptides?
allow_prot_nterm_acetyl = False # +42 on M1 or Res2 considered as NTT=2
#
##### coding starts here and below #####
Ln: 1 Col: 0
```



The program's logic for combining different instrument runs can be overridden by setting **use\_samples\_list** to **True**. In this case, you'll have to define how to combine the runs into samples.

The **sample\_prefixes** section of the code does this. Each line is a pairing. The first item is the beginning string the program will look for. Every file name that starts with what is listed in the first set of quotes (only enough to be unique is required) will be combined into a single column in the results spreadsheet with the column name given by the right item string.

Once everything is ready to go press **F5** to run the program as usual, and press **ok** if the program asks you if you'd like to save any changes.



```
PAW_results_7_2.py - C:\Users\PSR_Core\Dropbox\python_progs\new_PAW_programs\PAW_results_...
File Edit Format Run Options Windows Help
# the desired sample name. Samples are written in the order specified.)
#
##sample_prefixes = [('TRO1612_HumanKidney_FFPE_Control_75um_140min_20150220_VE2',
##                    ('TRO1612_HumanKidney_FFPE_Pigment_75um_140min_20150220_VE2',
##                    ('TRO1612_HumanKidney_FFPE_13934_75um_140min_20150220_VE2',
sample_prefixes = [('BAS1653_Z3_20150306_VE', 'BAS1653_Z3_20150306_VE'),
                    ('BAS1653_Z2_20150306_VE', 'BAS1653_Z2_20150306_VE'),
                    ('BAS1653_Z1_20150306_VE', 'BAS1653_Z1_20150306_VE'),
                    ('BAS1653_7_20150306_VE', 'BAS1653_7_20150306_VE'),
                    ('BAS1653_4_20150306_VE', 'BAS1653_4_20150306_VE'),
                    ('BAS1653_2_20150306_VE', 'BAS1653_2_20150306_VE')]
##                    ('BAS1653_Z1', 'BAS1653_Z1'),
##                    ('BAS1653_Z2', 'BAS1653_Z2'),
##                    ('BAS1653_Z3', 'BAS1653_Z3')]
##sample_prefixes = [('FIT1648_1', 'FIT1648_1'),
##                    ('FIT1648_2', 'FIT1648_2'),
##                    ('FIT1648_3', 'FIT1648_3'),
##                    ('FIT1648_GelRunin_NON-NU_1', 'FIT1648_GelRunin_NON-NU_1'),
##                    ('FIT1648_GelRunin_NON-NU_2', 'FIT1648_GelRunin_NON-NU_2')]
##sample_prefixes = [('10sep2013_yeast_control_1', '10sep2013_yeast_control_1'),
##                    ('10sep2013_yeast_control_2', '10sep2013_yeast_control_2'),
##                    ('10sep2013_yeast_control_3', '10sep2013_yeast_control_3'),
##                    ('10sep2013_yeast_control_4', '10sep2013_yeast_control_4'),
##                    ('10sep2013_yeast_control_5', '10sep2013_yeast_control_5')]
##sample_prefixes = [('FOX1371', 'FOX1371')]
##sample_prefixes = [('FOX1412', 'FOX1412')]
##sample_prefixes = [('FOX1488', 'FOX1488')]
##sample_prefixes = [('FOX1543', 'FOX1543')]
##sample_prefixes = [('PSR767_1A', 'PSR767_1A'),
##                    ('PSR767_1B', 'PSR767_1B'),
##                    ('PSR767_2A', 'PSR767_2A'),
##                    ('PSR767_2B', 'PSR767_2B'),
##                    ('PSR767_4A', 'PSR767_4A'),
##                    ('PSR767_4B', 'PSR767_4B'),
##                    ('PSR767_5A', 'PSR767_5A'),
##                    ('PSR767_5B', 'PSR767_5B'),
##                    ('PSR767_6A', 'PSR767_6A'),
##                    ('PSR767_6B', 'PSR767_6B')]
##sample_prefixes = [('DAVID_chicken_insol', 'DAVID_chicken_insol')]
#
```

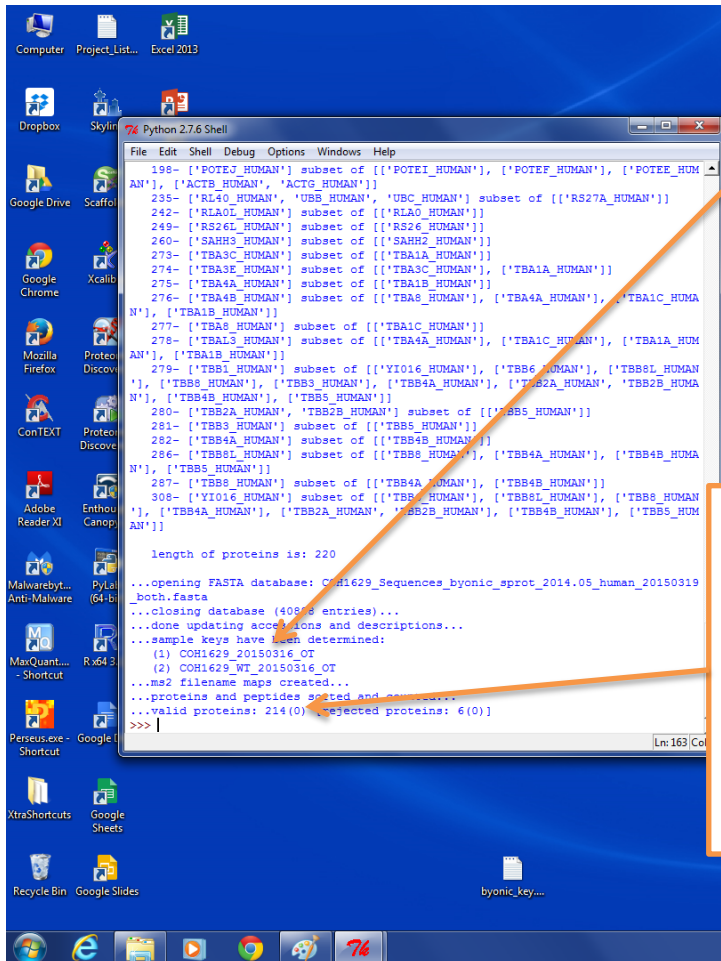
Ln: 1 Col: 0

After navigating to the **filtered\_ms2\_files** folder you wish to analyze, the program will churn through the list of potential proteins and create the final results spreadsheets, as well as providing you with two important pieces of information.

The first thing to check upon completion of the program is the **sample keys** that were determined. You want to make sure that the RAW files were combined into the sample groups you expected to see.

Commercial purposes should be directed to:  
Technology & Research Collaborations, Oregon Health & Science University,  
2525 SW 1st Ave, Suite 120, Portland, OR 97210  
Ph: 503-494-8200, FAX: 503-494-4729, Email: techmgmt@ohsu.edu.  
  
IN NO EVENT SHALL OREGON HEALTH & SCIENCE UNIVERSITY BE LIABLE TO ANY  
PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES,  
INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE. THE  
SOFTWARE IS PROVIDED "AS IS", AND OREGON HEALTH & SCIENCE UNIVERSITY HAS  
NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, OR ENHANCEMENTS.  
OREGON HEALTH & SCIENCE UNIVERSITY MAKES NO REPRESENTATIONS NOR EXTENDS

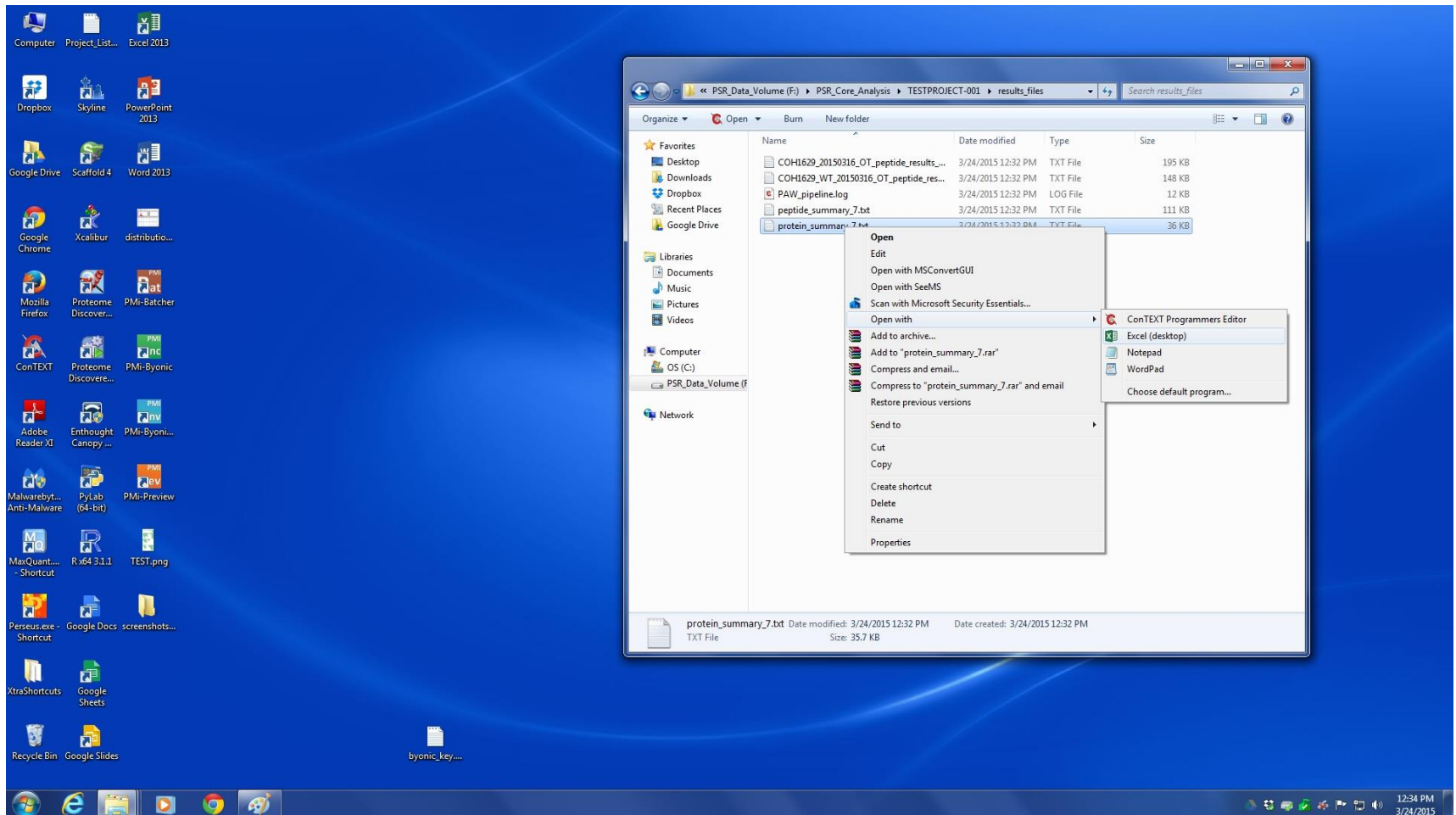
The second thing to check is the **valid proteins**. This is displayed in a forward (reversed) format. In most Experiments, you'll want a FDR in the range of 1 to 5% or so. If the values you get here fall outside that range, you'll need to return to the **histo\_GUI.py** program and adjust the filtering thresholds.



```
File Edit Shell Debug Options Windows Help
198- ['POTEF_HUMAN'] subset of [['POTEF_HUMAN'], ['POTEF_HUMAN'], ['POTEF_HUMAN']]
235- ['RLA40_HUMAN'] subset of [['RLA40_HUMAN'], ['RLA40_HUMAN']]
242- ['RLA40_HUMAN'] subset of [['RLA40_HUMAN']]
249- ['RS26L_HUMAN'] subset of [['RS26L_HUMAN']]
260- ['SAHH3_HUMAN'] subset of [['SAHH3_HUMAN']]
273- ['TBA3C_HUMAN'] subset of [['TBA1A_HUMAN']]
274- ['TBA3E_HUMAN'] subset of [['TBA3C_HUMAN'], ['TBA1A_HUMAN']]
275- ['TBA4A_HUMAN'] subset of [['TBA1B_HUMAN']]
276- ['TBA4B_HUMAN'] subset of [['TBA8_HUMAN'], ['TBA4A_HUMAN'], ['TBA1C_HUMAN'], ['TBA1B_HUMAN']]
277- ['TBA8_HUMAN'] subset of [['TBA1C_HUMAN']]
278- ['TBA15_HUMAN'] subset of [['TBA4A_HUMAN'], ['TBA1C_HUMAN'], ['TBA1A_HUMAN'], ['TBA1B_HUMAN']]
279- ['TBB1_HUMAN'] subset of [['YI016_HUMAN'], ['TBB6_HUMAN'], ['TBB8L_HUMAN'], ['TBB8_HUMAN'], ['TBB3_HUMAN'], ['TBB4A_HUMAN'], ['TBB2A_HUMAN'], ['TBB2B_HUMAN'], ['TBB4B_HUMAN'], ['TBB5_HUMAN']]
280- ['TBB2A_HUMAN', 'TBB2B_HUMAN'] subset of [['TBB5_HUMAN']]
281- ['TBB3_HUMAN'] subset of [['TBB5_HUMAN']]
282- ['TBB4A_HUMAN'] subset of [['TBB4B_HUMAN']]
286- ['TBB6L_HUMAN'] subset of [['TBB6_HUMAN'], ['TBB4A_HUMAN'], ['TBB4B_HUMAN'], ['TBB5_HUMAN']]
287- ['TBB8_HUMAN'] subset of [['TBB4A_HUMAN'], ['TBB4B_HUMAN']]
308- ['YI016_HUMAN'] subset of [['TBB6_HUMAN'], ['TBB8L_HUMAN'], ['TBB8_HUMAN'], ['TBB4A_HUMAN'], ['TBB2A_HUMAN', 'TBB2B_HUMAN'], ['TBB4B_HUMAN'], ['TBB5_HUMAN']]

length of proteins is: 220
...opening FASTA database: COH1629_Sequences_byonic_sprot_2014.05_human_20150319_both.fasta
...closing database (4083 entries)...
...done updating accession and descriptions...
...sample keys have been determined:
(1) COH1629_20150316_OT
(2) COH1629_WT_20150316_OT
...ms2 filename maps created...
...proteins and peptides sorted and counted...
...valid proteins: 214(0) rejected proteins: 6(0)
>>>
```

The final output from the pipeline will be a **protein\_summary\_7.txt** file and a **peptide\_summary\_7.txt** file. They can be found in the **results\_files** folder that's created by the **PAW\_Results\_7.py** program. To view these files, right click on them and choose to open them in Microsoft Excel (or OpenOffice).





The **protein\_summary** file will contain the proteins identified, and the **peptide\_summary** file will list the identified amino acid sequences. Information regarding the formatting and layout of the spreadsheet can be found in the “PAW Results Guide” in the Educational Links section of the PSR Website.

| ProtGroup | Counter | Accession Link | Filter    | Coverage | SeqLength | MW     | Descriptic  | CountsTot | UniqueTo | UniqFrac | Total | Total | Unique | Unique | Corrected | Corrected | OtherLoci  |
|-----------|---------|----------------|-----------|----------|-----------|--------|-------------|-----------|----------|----------|-------|-------|--------|--------|-----------|-----------|--|
| 1         | 1       | 1433T_HU       |           | 14.7     | 245       | 27765  | 14-3-3 pro  | 3         | 2        | 0.667    | 2     | 1     | 2      | 0      | 2         | 0.333     | 1433E_HUMAN, 1433T_HUMAN, 1433Z_HUMAN                                      |
| 2         | 2       | 1ABCE1_HU      |           | 5.8      | 599       | 67315  | ATP-bindi   | 2         | 2        | 1        | 2     | 0     | 2      | 0      | 2         | 0         |  |
| 3         | 3       | 1ABCF1_HU      |           | 2.7      | 845       | 95927  | ATP-bindi   | 2         | 2        | 1        | 2     | 0     | 2      | 0      | 2         | 0         |  |
| 4         | 4       | 1ACACA_HI      |           | 34.9     | 2346      | 265555 | Acetyl-Co   | 105       | 105      | 1        | 32    | 73    | 32     | 73     | 32        | 73        |  |
| 5         | 5       | 1ACADM_H       |           | 6.7      | 421       | 46589  | Medium-c    | 2         | 2        | 1        | 2     | 0     | 2      | 0      | 2         | 0         |  |
| 6         | 6       | 1ACLY_HUN      |           | 5.5      | 1101      | 120840 | ATP-citrat  | 5         | 5        | 1        | 4     | 1     | 4      | 1      | 4         | 1         |  |
| 7         | 7       | 1ACTB_HUM      |           | 49.3     | 375       | 41738  | Actin, cyto | 21        | 21       | 1        | 11    | 10    | 11     | 10     | 11        | 10        |  |
| 8         | 7.01    | 1ACTG_HUI      | redundant | 49.3     | 375       | 41794  | Actin, cyto | 21        | 21       | 1        | 11    | 10    | 11     | 10     | 11        | 10        |  |
| 9         | 8       | 1ADT2_HUM      |           | 29.9     | 298       | 32853  | ADP/ATP     | 15        | 6        | 0.4      | 11    | 4     | 4      | 2      | 8.333     | 4         | ADT2_HUMAN, ADT3_HUMAN   |
| 10        | 9       | 1ADT3_HUM      |           | 26.5     | 298       | 32867  | ADP/ATP     | 11        | 2        | 0.182    | 9     | 2     | 2      | 0      | 4.667     | 0         | ADT2_HUMAN, ADT3_HUMAN   |
| 11        | 10      | 1AGK_HUM       |           | 8.8      | 422       | 47138  | Acylglycer  | 2         | 2        | 1        | 2     | 0     | 2      | 0      | 2         | 0         |  |
| 12        | 11      | 1ALDOA_HI      |           | 10.2     | 364       | 39421  | Fructose-1  | 4         | 4        | 1        | 3     | 1     | 3      | 1      | 3         | 1         |  |
| 13        | 12      | 1AN32A_HI      |           | 9.6      | 249       | 28586  | Acidic leu  | 2         | 1        | 0.5      | 2     | 0     | 1      | 0      | 1.5       | 0         | AN32A_HUMAN, AN32B_HUMAN   |
| 14        | 13      | 1AN32B_HI      |           | 11.6     | 251       | 28789  | Acidic leu  | 2         | 1        | 0.5      | 2     | 0     | 1      | 0      | 1.5       | 0         | AN32A_HUMAN, AN32B_HUMAN   |
| 15        | 14      | 1AN32E_HI      |           | 18.3     | 268       | 30693  | Acidic leu  | 4         | 4        | 1        | 4     | 0     | 4      | 0      | 4         | 0         |  |
| 16        | 15      | 1ASN5_HUI      |           | 5.2      | 561       | 64371  | Asparagin   | 2         | 2        | 1        | 2     | 0     | 2      | 0      | 2         | 0         |  |
| 17        | 16      | 1AT1A1_HI      |           | 14.1     | 1023      | 112897 | Sodium/p    | 12        | 12       | 1        | 11    | 1     | 11     | 1      | 11        | 1         |  |
| 18        | 17      | 1AT2A2_HI      |           | 6.3      | 1042      | 114758 | Sarcoplas   | 4         | 4        | 1        | 4     | 0     | 4      | 0      | 4         | 0         |  |
| 19        | 18      | 1ATPA_HUI      |           | 13.4     | 553       | 59752  | ATP synth   | 9         | 9        | 1        | 4     | 5     | 4      | 5      | 4         | 5         |  |
| 20        | 19      | 1C1QBP_HI      |           | 33.7     | 282       | 31363  | Complem     | 9         | 9        | 1        | 9     | 0     | 9      | 0      | 9         | 0         |  |
| 21        | 20      | 1C1TC_HUN      |           | 4.1      | 935       | 101560 | C-1-tetra   | 3         | 3        | 1        | 2     | 1     | 2      | 1      | 2         | 1         |  |
| 22        | 21      | 1CH60_HUN      |           | 12.9     | 573       | 61056  | 60 kDa he   | 7         | 7        | 1        | 4     | 3     | 4      | 3      | 4         | 3         |  |
| 23        | 22      | 1CLH1_HUN      |           | 3.3      | 1675      | 191616 | Clathrin h  | 4         | 4        | 1        | 4     | 0     | 4      | 0      | 4         | 0         |  |
| 24        | 23      | 1CMC2_HU       |           | 8.7      | 675       | 74177  | Calcium-b   | 5         | 5        | 1        | 5     | 0     | 5      | 0      | 5         | 0         |  |
| 25        | 24      | 1COF1_HUM      |           | 25.3     | 166       | 18503  | Cofilin-1   | 3         | 3        | 1        | 2     | 1     | 2      | 1      | 2         | 1         |  |
| 26        | 25      | 1CONT_005      | contamin  | 100      | 50        | 5561   | Promega f   | 15        | 11       | 0.733    | 5     | 10    | 3      | 8      | 3.316     | 8.471     | CONT_005, CONT_010   |
| 27        | 26      | 1CONT_010      | contamin  | 25.1     | 231       | 24410  | TRYPSIN P   | 46        | 42       | 0.913    | 18    | 28    | 16     | 26     | 17.684    | 27.529    | CONT_005, CONT_010   |
| 28        | 27      | 1CONT_015      | contamin  | 30.8     | 581       | 65798  | albumin (l  | 23        | 2        | 0.087    | 8     | 15    | 0      | 2      | 0         | 5.714     | CONT_015, CONT_016   |
| 29        | 28      | 1CONT_016      | contamin  | 34.6     | 607       | 69271  | SERUM AL    | 28        | 7        | 0.25     | 10    | 18    | 2      | 5      | 10        | 14.286    | CONT_015, CONT_016   |
| 30        | 29      | 1CONT_015      | contamin  | 26.2     | 214       | 24530  | ALPHA-S1    | 6         | 6        | 1        | 3     | 3     | 3      | 3      | 3         | 3         |  |
| 31        | 30      | 1CONT_021      | contamin  | 8.5      | 224       | 25108  | BETA CASI   | 3         | 3        | 1        | 1     | 2     | 1      | 2      | 1         | 2         |  |
| 32        | 31      | 1CONT_050      | contamin  | 12.1     | 420       | 45866  | keratin 13  | 15        | 1        | 0.067    | 7     | 8     | 1      | 0      | 3.115     | 0         | CONT_050, CONT_082, CONT_094, CONT_101, CONT_156, K1C10_HUMAN, K1C13_HUMAN |
| 33        | 31.01   | 1CONT_098      | redundant | 11.1     | 458       | 49645  | KERATIN,    | 15        | 1        | 0.067    | 7     | 8     | 1      | 0      | 3.115     | 0         | CONT_050, CONT_082, CONT_094, CONT_101, CONT_156, K1C10_HUMAN, K1C13_HUMAN |
| 34        | 32      | 1CONT_072      | contamin  | 62.2     | 645       | 65866  | KERATIN,    | 63        | 2        | 0.032    | 17    | 46    | 0      | 2      | 0         | 13.882    | CONT_072, CONT_073, CONT_129, CONT_133, K22E_HUMAN, K2C1_HUMAN, K2C5_HUMAN |
| 35        | 33      | 1CONT_073      | contamin  | 5.8      | 482       | 53563  | cytokera    | 9         | 0        | 0        | 1     | 8     | 0      | 0      | 0         | 0         | CONT_072, CONT_073, CONT_129, CONT_133, K22E_HUMAN, K2C1_HUMAN, K2C5_HUMAN |
| 36        | 33.01   | 1CONT_103      | redundant | 5.8      | 483       | 53705  | Keratin 8-  | 9         | 0        | 0        | 1     | 8     | 0      | 0      | 0         | 0         | CONT_072, CONT_073, CONT_129, CONT_133, K22E_HUMAN, K2C1_HUMAN, K2C5_HUMAN |
| 37        | 33.02   | 1CONT_124      | redundant | 5.8      | 483       | 53749  | cytokera    | 9         | 0        | 0        | 1     | 8     | 0      | 0      | 0         | 0         | CONT_072, CONT_073, CONT_129, CONT_133, K22E_HUMAN, K2C1_HUMAN, K2C5_HUMAN |

The End



The following slides describe running Comet searches via command line and performing the comet results post-processing steps manually.

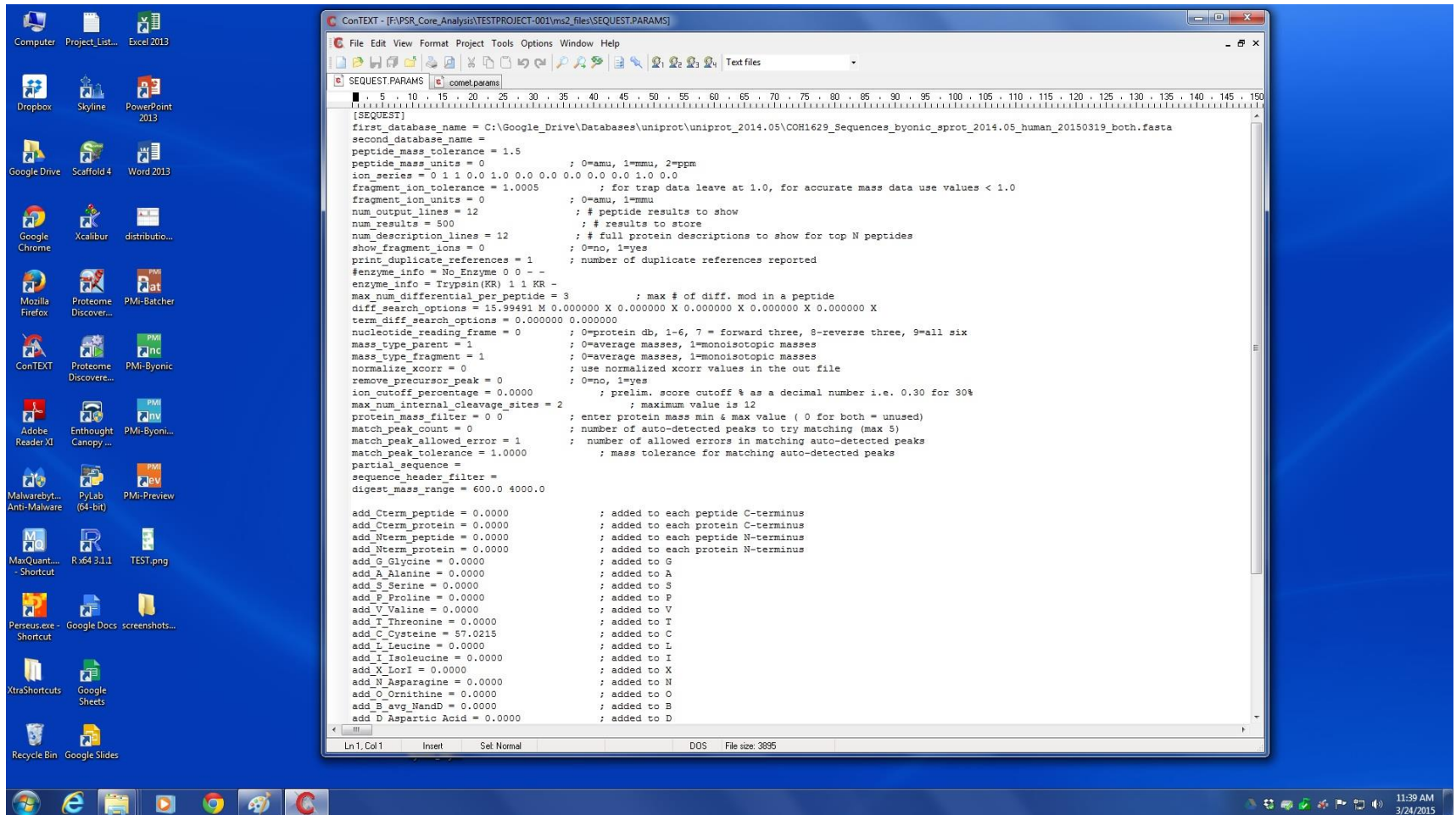
Situations where this might be useful are:

Doing a search using the same parameters as in a previous analysis where the **comet.params** file can be copied to the new **ms2\_files** location.

Making changes to the **comet.params** file that are beyond the basic parameters available in the GUI program.

**NOTES:** **sequest.params** files are discussed. The need for these files may change in the near future. Also, **comet.params** files can change format with different versions of Comet and contain a version key. If the version key does not match the Comet version, Comet will issue an error message and terminate.

Once the parameter files have been copied over, they can be opened in any text editor. There are a few different places you'll likely need to make changes: database, fragment and parent ion mass tolerances, parent ion type, differential and static modifications, enzyme info, and ion series.





# SEQUEST Parameters

```
[SEQUEST]
first_database_name = C:\Google_Drive\Databases\uniprot\uniprot_2014.05\COH1629_Sequences_byonic_sprot_2014.05_human_20150319_both.fasta
second_database_name =
peptide_mass_tolerance = 1.5 ; 0=amu, 1=mmu, 2=ppm
peptide_mass_units = 0
ion_series = 0 1 1 0.0 1.0 0.0 0.0 0.0 0.0 0.0 1.0 0.0
fragment_ion_tolerance = 1.0005 ; for trap data leave at 1.0, for accurate mass data use values < 1.0
fragment_ion_units = 0 ; 0=amu, 1=mmu
num_output_lines = 12 ; # peptide results to show
num_results = 500 ; # results to store
num_description_lines = 12 ; # full protein descriptions to show for top N peptides
show_fragment_ions = 0 ; 0=no, 1=yes
print_duplicate_references = 1 ; number of duplicate references reported
#enzyme_info = No Enzyme 0 0 - -
enzyme_info = Trypsin(KR) 1 1 KR -
max_num_differential_per_peptide = 3 ; max # of diff. mod in a peptide
diff_search_options = 15.99491 M 0.000000 X 0.000000 X 0.000000 X 0.000000 X 0.000000 X
term_diff_search_options = 0.000000 0.000000
nucleotide_reading_frame = 0 ; 0=protein db, 1-6, 7 = forward three, 8=reverse three, 9=all six
mass_type_parent = 1 ; 0=average masses, 1=monoisotopic masses
mass_type_fragment = 1 ; 0=average masses, 1=monoisotopic masses
normalize_xcorr = 0 ; use normalized xcorr values in the out file
remove_precursor_peak = 0 ; 0=no, 1=yes
ion_cutoff_percentage = 0.0000 ; prelim. score cutoff % as a decimal number i.e. 0.30 for 30%
max_num_internal_cleavage_sites = 2 ; maximum value is 12
protein_mass_filter = 0 0 ; enter protein mass min & max value ( 0 for both = unused)
match_peak_count = 0 ; number of auto-detected peaks to try matching (max 5)
match_peak_allowed_error = 1 ; number of allowed errors in matching auto-detected peaks
match_peak_tolerance = 1.0000 ; mass tolerance for matching auto-detected peaks
partial_sequence =
sequence_header_filter =
digest_mass_range = 600.0 4000.0

add_Cterm_peptide = 0.0000 ; added to each peptide C-terminus
add_Cterm_protein = 0.0000 ; added to each protein C-terminus
add_Nterm_peptide = 0.0000 ; added to each peptide N-terminus
add_Nterm_protein = 0.0000 ; added to each protein N-terminus
add_G_Glycine = 0.0000 ; added to G
add_A_Alanine = 0.0000 ; added to A
add_S_Serine = 0.0000 ; added to S
add_P_Proline = 0.0000 ; added to P
add_V_Valine = 0.0000 ; added to V
add_T_Threonine = 0.0000 ; added to T
add_C_Cysteine = 57.0215 ; added to C
add_L_Leucine = 0.0000 ; added to L
add_I_Isoleucine = 0.0000 ; added to I
add_X_LorI = 0.0000 ; added to X
add_N_Asparagine = 0.0000 ; added to N
add_O_Ornithine = 0.0000 ; added to O
add_B_avg_NandD = 0.0000 ; added to B
add_D_Aspartic_Acid = 0.0000 ; added to D
```

database location

1.25 for the Orbitrap  
2.5 for everything else

The **ion\_series** will need to be changed for ETD data. The two 1.0 flags will need to be moved to the right one spot in the series. This will toggle c/z ions to be searched for instead of b/y ions.

# SEQUEST Parameters

```
ConTEXT - [F:\PSR_Core_Analysis\TESTPROJECT-001\ms2_files\SEQUEST.PARAMS]
File Edit View Format Project Tools Options Window Help
SEQUEST.PARAMS comet.params
5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110 115 120 125 130 135 140 145 150
[SEQUEST]
first_database_name = C:\Google_Drive\Databases\uniprot\uniprot_2014.05\COH1629_Sequences_byonic_sprot_2014.05_human_20150319_both.fasta
second_database_name =
peptide_mass_tolerance = 1.5
peptide_mass_units = 0 ; 0=amu, 1=mmu, 2=ppm
ion_series = 0 1 1 0.0 1.0 0.0 0.0 0.0 0.0 1.0 0.0
fragment_ion_tolerance = 1.0005 ; for trap data leave at 1.0, for accurate mass data use values < 1.0
fragment_ion_units = 0 ; 0=amu, 1=mmu
num_output_lines = 12 ; # peptide results to show
num_results = 500 ; # results to store
num_description_lines = 12 ; # full protein descriptions to show for top N peptides
show_fragment_ions = 0 ; 0=no, 1=yes
print_duplicate_references = 1 ; number of duplicate references reported
#enzyme_info = No Enzyme 0 0 - -
enzyme_info = Trypsin(KR) 1 1 KR -
max_num_differential_per_peptide = 3 ; max # of diff. mod in a peptide
diff_search_options = 15.99491 M 0.000000 X 0.000000 X 0.000000 X 0.000000 X 0.000000 X
term_diff_search_options = 0.000000 0.000000
nucleotide_reading_frame = 0 ; 0=protein db, 1-6, 7 = forward three, 8-reverse three, 9=all six
mass_type_parent = 1 ; 0=average masses, 1=monoisotopic masses
mass_type_fragment = 0 ; 0=average masses, 1=monoisotopic masses
normalize_xcorr = 0 ; use normalized xcorr values in the out file
remove_precursor_peak = 0 ; 0=no, 1=yes
ion_cutoff_percentage = 0.0000 ; prelim. score cutoff % as a decimal number (i.e. 0.30 for 30%)
max_num_internal_cleavage_sites = 1 ; maximum value is 12
protein_mass_filter = 0 0 ; enter protein mass min & max value ( 0 for both = unused)
match_peak_count = 0 ; number of auto-detected peaks to try matching (max 5)
match_peak_allowed_error = 1 ; number of allowed errors in matching auto-detected peaks
match_peak_tolerance = 1.0000 ; mass tolerance for matching auto-detected peaks
partial_sequence =
sequence_header_filter =
digest_mass_range = 600.0 4000.0

add_Cterm_peptide = 0.0000 ; added to each peptide C-terminus
add_Cterm_protein = 0.0000 ; added to each protein C-terminus
add_Nterm_peptide = 0.0000 ; added to each peptide N-terminus
add_Nterm_protein = 0.0000 ; added to each protein N-terminus
add_G_Glycine = 0.0000 ; added to G
add_A_Alanine = 0.0000 ; added to A
add_S_Serine = 0.0000 ; added to S
add_P_Proline = 0.0000 ; added to P
add_V_Valine = 0.0000 ; added to V
add_T_Threonine = 0.0000 ; added to T
add_C_Cysteine = 57.0215 ; added to C
add_L_Leucine = 0.0000 ; added to L
add_I_Isoleucine = 0.0000 ; added to I
add_X_LorI = 0.0000 ; added to X
add_N_Asparagine = 0.0000 ; added to N
add_O_Ornithine = 0.0000 ; added to O
add_B_avg_NandD = 0.0000 ; added to B
add_D_Aspartic_Acid = 0.0000 ; added to D
```

Enzyme information: the '#' comments out unused lines, and other enzymes can be entered here.

Differential modifications: mass followed by amino acid location(s). Please use at least 4 digits after the decimal point for Orbitrap data.

mass\_type\_parent should be '1' for Orbitrap data, '0' for everything else.

static modifications by amino acid: please use at least 4 digits after the decimal for Orbitrap data.

The **comet.params** file is very similar to the **sequest.params** file; however, it is longer and better annotated. All of the same options are there just in a different format. The two files should be checked to make sure the parameters are identical prior to analyzing the data. The various considerations for the parameters are the same as with the **sequest.params** file.

The screenshot displays the ConTEXT software interface with the **comet.params** file open. The file is a text-based configuration file for mass spectrometry data analysis. Several key parameters are highlighted with orange callouts:

- comet\_version**: 2014.02 rev. 2 (circled in red)
- database\_name**: C:\Google\_Drive\Databases\uniprot\uniprot\_2014.05\COH1629\_Sequences\_byonic\_sprot\_2014.05\_human\_20150319\_both.fasta (labeled "database")
- peptide\_mass\_tolerance**: 1.5 (labeled "2.5 or 1.5")
- search\_enzyme\_number**: 1 (labeled "'1' for Orbitrap data only")
- variable\_mods**: A table of variable modifications (labeled "variable mods table")
- search\_enzyme**: 1 (labeled "set enzyme number here")

The **variable\_mods** table is as follows:

| variable_mod   | mass    | residues | variable | binary | max_mods_per_peptide |   |
|----------------|---------|----------|----------|--------|----------------------|---|
| variable_mod01 | 15.9949 | M        | 0        | 3      | -1                   | 0 |
| variable_mod02 | 0.0000  | X        | 0        | 3      | -1                   | 0 |
| variable_mod03 | 0.0000  | X        | 0        | 3      | -1                   | 0 |
| variable_mod04 | 0.0000  | X        | 0        | 3      | -1                   | 0 |
| variable_mod05 | 0.0000  | X        | 0        | 3      | -1                   | 0 |
| variable_mod06 | 0.0000  | X        | 0        | 3      | -1                   | 0 |
| variable_mod07 | 0.0000  | X        | 0        | 3      | -1                   | 0 |
| variable_mod08 | 0.0000  | X        | 0        | 3      | -1                   | 0 |
| variable_mod09 | 0.0000  | X        | 0        | 3      | -1                   | 0 |



Computer Project\_List... Excel 2013

Dropbox Skyline PowerPoint 2013

Google Drive Scaffold 4 Word 2013

Google Chrome Xcalibur distributio...

Mozilla Firefox Proteome Discover... PMi-Batcher

ConTEXT Proteome Discover... PMi-Byonic

Adobe Reader XI Enthoutght Canopy... PMi-Byoni...

Malwarebyt... PyLab (64-bit) PMi-Preview

MaxQuant... R64 3.1.1 TEST.png

Perseus.exe - Shortcut Google Docs screenshots...

XtraShortcuts Google Sheets

Recycle Bin Google Slides

ConTEXT - [F:\PSR\_Core\_Analysis\TESTPROJECT-001\ms2\_files\comet.params]

File Edit View Format Project Tools Options Window Help

SEQUEST.PARAMS comet.params

max\_variable\_mods\_in\_peptide = 5

# fragment ions

# ion trap ms/ms: 1.0005 tolerance, 0.4 offset (mono masses), theoretical\_fragment\_ions = 1

# high res ms/ms: 0.02 tolerance, 0.0 offset (mono masses), theoretical\_fragment\_ions = 0

fragment\_bin\_tol = 1.0005 # binning to use on fragment ions

fragment\_bin\_offset = 0.4 # offset position to start the binning (0.0 to 1.0)

theoretical\_fragment\_ions = 1 # 0=use flanking peaks, 1=M peak only

use\_A\_ions = 0

use\_B\_ions = 1

use\_C\_ions = 0

use\_X\_ions = 0

use\_Y\_ions = 1

use\_Z\_ions = 0

use\_NL\_ions = 1 # 0=no, 1=yes to consider NHS/H2O neutral loss peaks

use\_sparse\_matrix = 0

# output

output\_sqtstream = 0 # 0=no, 1=yes write sqt to standard output

output\_sqtfile = 1 # 0=no, 1=yes write sqt file

output\_txtfile = 0 # 0=no, 1=yes write tab-delimited txt file

output\_pepxmlfile = 0 # 0=no, 1=yes write pep.xml file

output\_percolatorfile = 0 # 0=no, 1=yes write Percolator tab-delimited input file

output\_outfiles = 0 # 0=no, 1=yes write .out files

print\_expect\_score = 1 # 0=no, 1=yes to replace Sp with expect in out & sqt

num\_output\_lines = 5 # num peptide results to show

show\_fragment\_ions = 0 # 0=no, 1=yes for out files only

sample\_enzyme\_number = 1 # Sample enzyme which is possibly different than the one applied to the search.

# Used to calculate NTT & NMC in pepXML output (default=1 for trypsin).

# mzXML parameters

scan\_range = 0 0 # start and scan scan range to search: 0 as 1st entry ignores parameter

precursor\_charge = 0 0 # precursor charge range to analyze; does not override any existing charge: 0 as 1st entry ignores parameter

override\_charge = 0 # 0=no, 1=yes to override existing precursor charge states with precursor\_charge parameter

ms\_level = 2 # MS level to analyze, valid are levels 2 (default) or 3

activation\_method = ALL # activation method; used if activation method set; allowed ALL, CID, ECD, ETD, PQD, HCD, IRMPD

# misc parameters

digest\_mass\_range = 600.0 5000.0 # MH+ peptide mass range to analyze

num\_results = 50 # number of search hits to store internally

skip\_researching = 1 # for '.out' file output only, 0=search everything again (default), 1=don't search if .out exists

Ln1, Col1 Insert Set Normal DOS File size: 8954

ion series, set c/z = 1 and b/y = 0, for ETD data

11:40 AM 3/24/2015

Computer Project\_List... Excel 2013

Dropbox Skyline PowerPoint 2013

Google Drive Scaffold 4 Word 2013

Google Chrome Xcalibur distributio...

Mozilla Firefox Proteome Discover... PMi-Batcher

ConTEXT Proteome Discover... PMi-Byonic

Adobe Reader XI Enthought Canopy... PMi-Byoni...

Malwarebyt... PyLab (64-bit) PMi-Preview

MaxQuant... R64 3.1.1 TEST.png

Perseus.exe - Shortcut Google Docs screenshots...

XtraShortcuts Google Sheets

Recycle Bin Google Slides

ConTEXT - [F:\PSR\_Core\_Analysis\TESTPROJECT-001\ms2\_files\comet.params]

File Edit View Format Project Tools Options Window Help

SEQUEST PARAMS comet.params

add\_Cterm\_peptide = 0.0000  
add\_Nterm\_peptide = 0.0000  
add\_Cterm\_protein = 0.0000  
add\_Nterm\_protein = 0.0000

add\_G\_glycine = 0.0000 # added to G - avg. 57.0513, mono. 57.02146  
add\_A\_alanine = 0.0000 # added to A - avg. 71.0779, mono. 71.03711  
add\_S\_serine = 0.0000 # added to S - avg. 87.0773, mono. 87.03203  
add\_P\_proline = 0.0000 # added to P - avg. 97.1152, mono. 97.05276  
add\_V\_valine = 0.0000 # added to V - avg. 99.1311, mono. 99.06841  
add\_T\_threonine = 0.0000 # added to T - avg. 101.1038, mono. 101.04768  
add\_C\_cysteine = 57.0215 # added to C - avg. 103.1429, mono. 103.00918  
add\_L\_leucine = 0.0000 # added to L - avg. 113.1576, mono. 113.08406  
add\_I\_isoleucine = 0.0000 # added to I - avg. 113.1576, mono. 113.08406  
add\_N\_asparagine = 0.0000 # added to N - avg. 114.1026, mono. 114.04293  
add\_D\_aspartic\_acid = 0.0000 # added to D - avg. 115.0874, mono. 115.02694  
add\_Q\_glutamine = 0.0000 # added to Q - avg. 128.1292, mono. 128.05858  
add\_K\_lysine = 0.0000 # added to K - avg. 128.1723, mono. 128.09496  
add\_E\_glutamic\_acid = 0.0000 # added to E - avg. 129.1140, mono. 129.04259  
add\_M\_methionine = 0.0000 # added to M - avg. 131.1961, mono. 131.04048  
add\_O\_ornithine = 0.0000 # added to O - avg. 132.1610, mono. 132.08988  
add\_H\_histidine = 0.0000 # added to H - avg. 137.1393, mono. 137.05891  
add\_F\_phenylalanine = 0.0000 # added to F - avg. 147.1739, mono. 147.06841  
add\_R\_arginine = 0.0000 # added to R - avg. 156.1857, mono. 156.10111  
add\_Y\_tyrosine = 0.0000 # added to Y - avg. 163.0633, mono. 163.06333  
add\_W\_tryptophan = 0.0000 # added to W - avg. 186.0793, mono. 186.07931  
add\_B\_user\_amino\_acid = 0.0000 # added to B - avg. 0.0000, mono. 0.00000  
add\_U\_user\_amino\_acid = 0.0000 # added to U - avg. 0.0000, mono. 0.00000  
add\_X\_user\_amino\_acid = 0.0000 # added to X - avg. 0.0000, mono. 0.00000  
add\_Z\_user\_amino\_acid = 0.0000 # added to Z - avg. 0.0000, mono. 0.00000

# COMET\_ENZYME\_INFO\_must\_be at the end of this parameters file

[COMET\_ENZYME\_INFO]

|                  |   |      |   |
|------------------|---|------|---|
| 0. No_enzyme     | 0 | -    | - |
| 1. Trypsin       | 1 | KR   | P |
| 2. Trypsin/P     | 1 | KR   | - |
| 3. Lys_C         | 1 | K    | P |
| 4. Lys_N         | 0 | K    | - |
| 5. Arg_C         | 1 | R    | P |
| 6. Asp_N         | 0 | D    | - |
| 7. CNBr          | 1 | M    | - |
| 8. Glu_C         | 1 | DE   | P |
| 9. PepsinA       | 1 | FL   | P |
| 10. Chymotrypsin | 1 | FWYL | P |

static modifications table

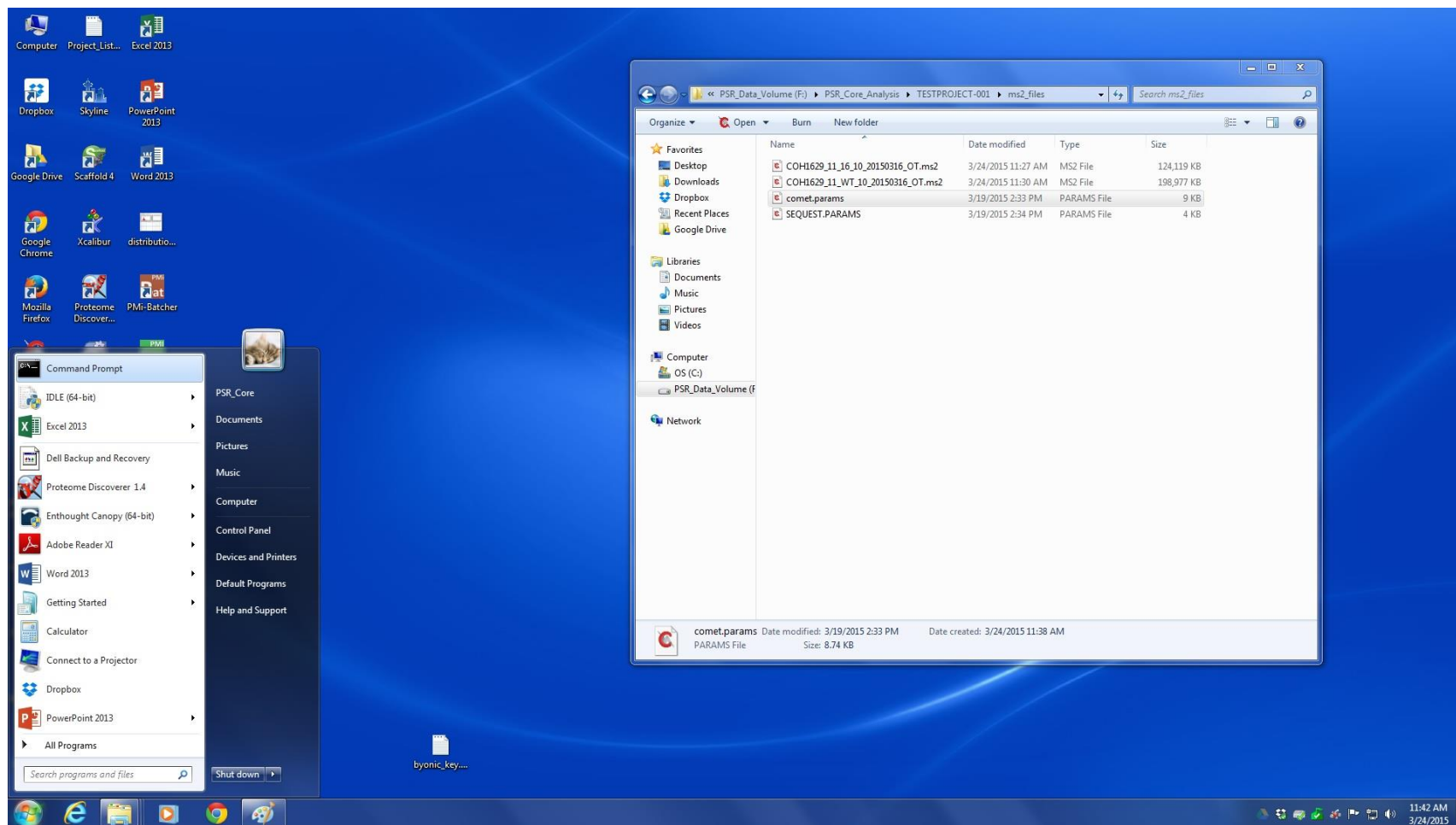
Enzyme information, use the reference number here in the location marked on a previous slide.

Ln 1, Col 1 Insert Set Normal DOS File size: 8954

11:41 AM 3/24/2015

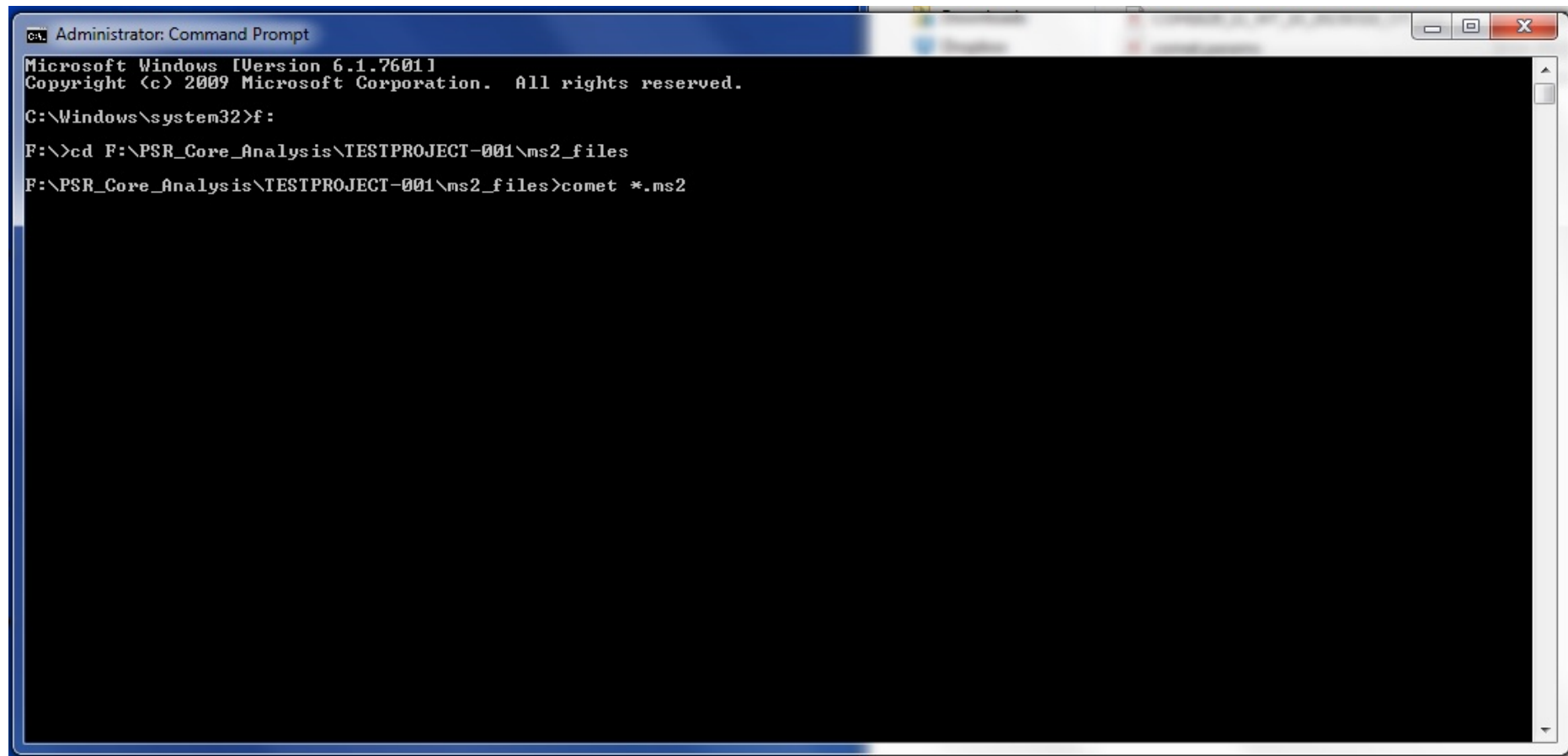


After the parameters files are in place, the next step is to open up the command window and launch the Comet searches. The command prompt shortcut can be found on the PSR computers as indicated in the picture below. A **<shift> right-mouse-click** on a folder will also open a command window at that location.



The three commands you'll need to use to run your search are:

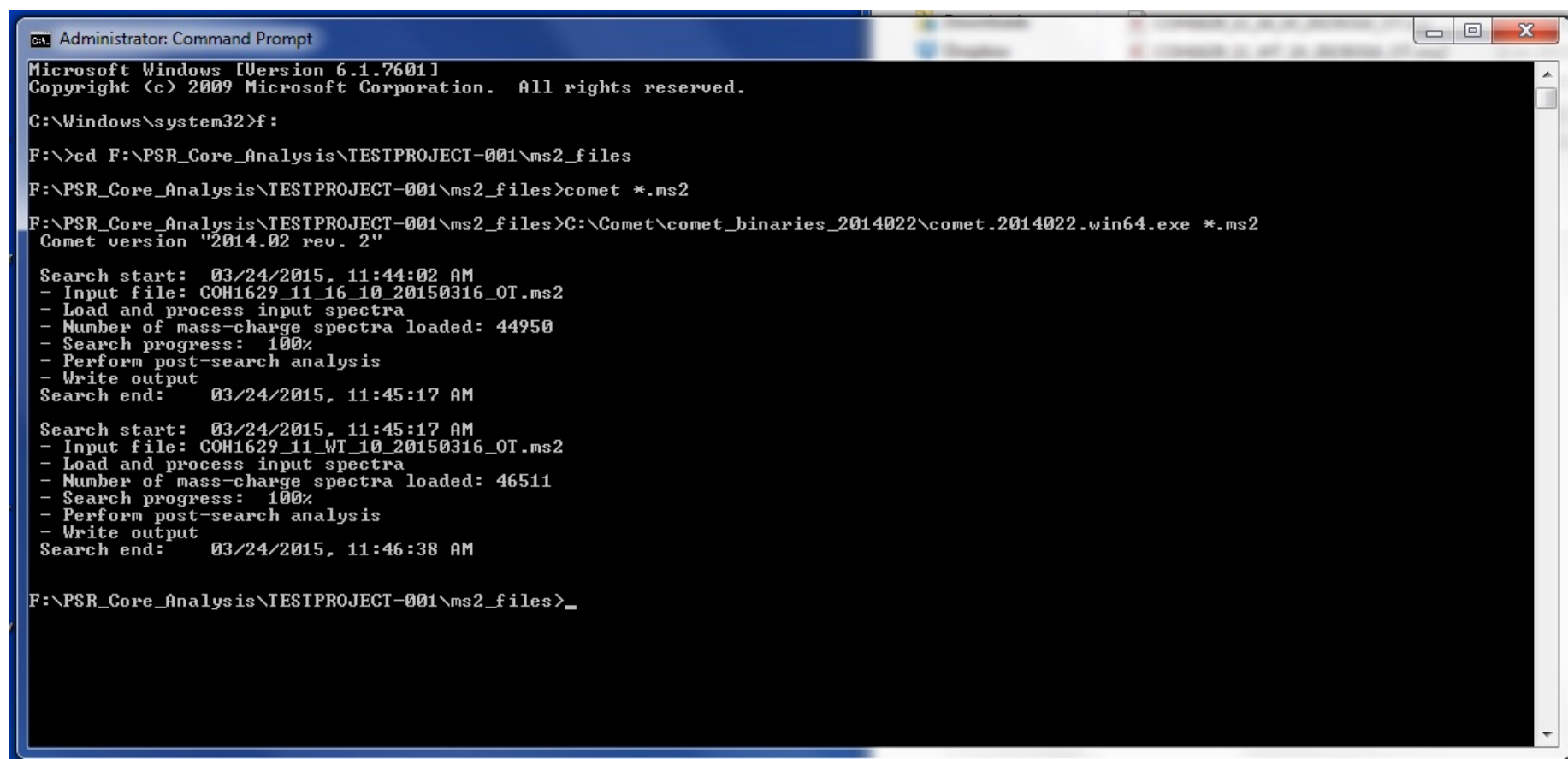
- 1) switch to the **F:** drive
- 2) **cd** (change directory) to your file location. This path can be pasted in with a right click menu. The tab key can also be used to auto-complete the path name.
- 3) **comet \*.ms2** searches all the ms2 files in the folder; files can also be searched individually.



```
Administrator: Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>f:
F:\>cd F:\PSR_Core_Analysis\TESTPROJECT-001\ms2_files
F:\PSR_Core_Analysis\TESTPROJECT-001\ms2_files>comet *.ms2
```

A completed Comet search will display as seen below. Comet often runs much faster than SEQUEST searches used to. A search with a single differential modification against a typical database should only take about one minute or so. Once the searches have completed you can close the command prompt window.



```
Administrator: Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

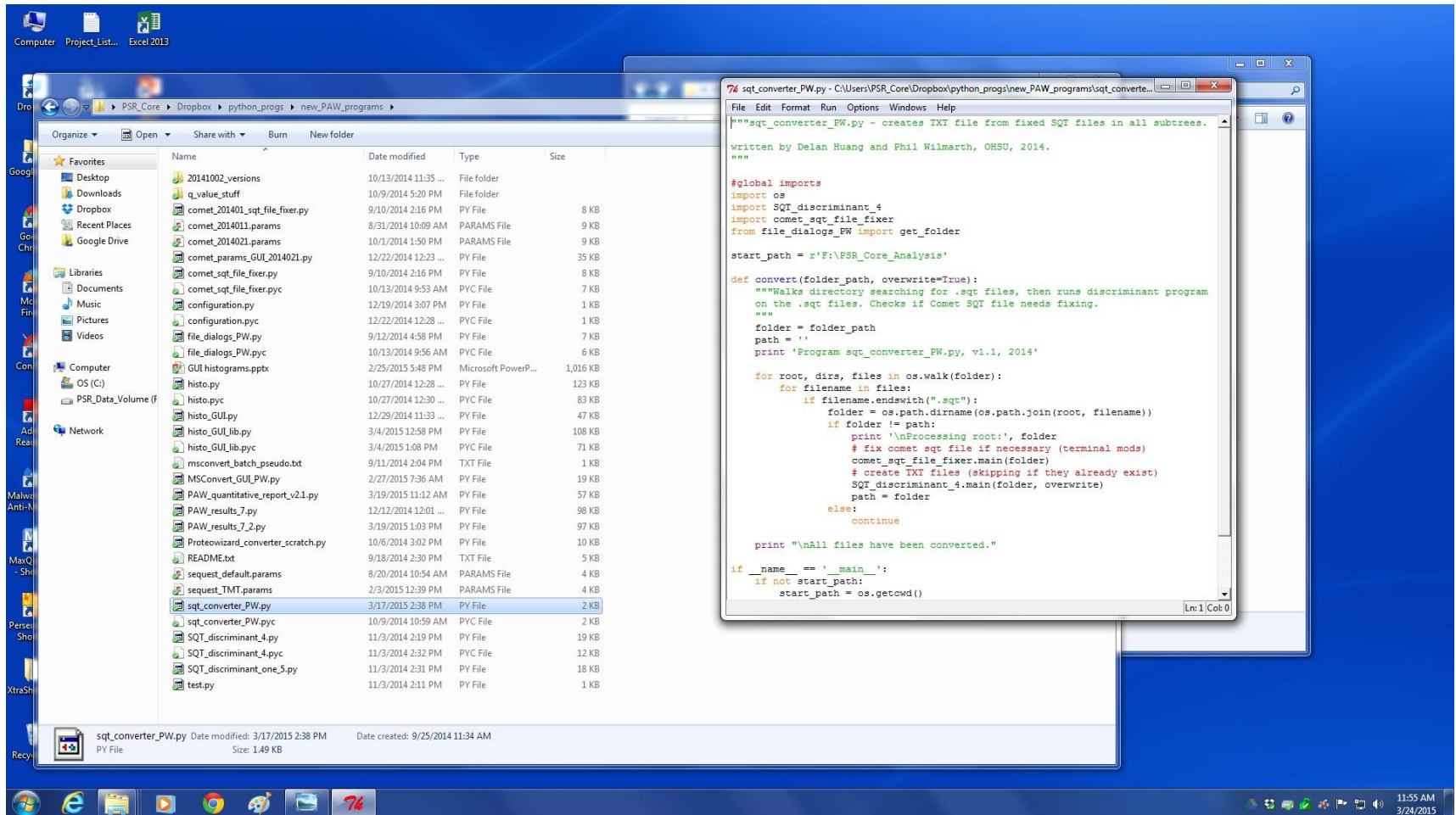
C:\Windows\system32>f:
F:\>cd F:\PSR_Core_Analysis\TESTPROJECT-001\ms2_files
F:\PSR_Core_Analysis\TESTPROJECT-001\ms2_files>comet *.ms2
F:\PSR_Core_Analysis\TESTPROJECT-001\ms2_files>C:\Comet\comet_binaries_2014022\comet.2014022.win64.exe *.ms2
Comet version "2014.02 rev. 2"

Search start: 03/24/2015, 11:44:02 AM
- Input file: COH1629_11_16_10_20150316_OT.ms2
- Load and process input spectra
- Number of mass-charge spectra loaded: 44950
- Search progress: 100%
- Perform post-search analysis
- Write output
Search end: 03/24/2015, 11:45:17 AM

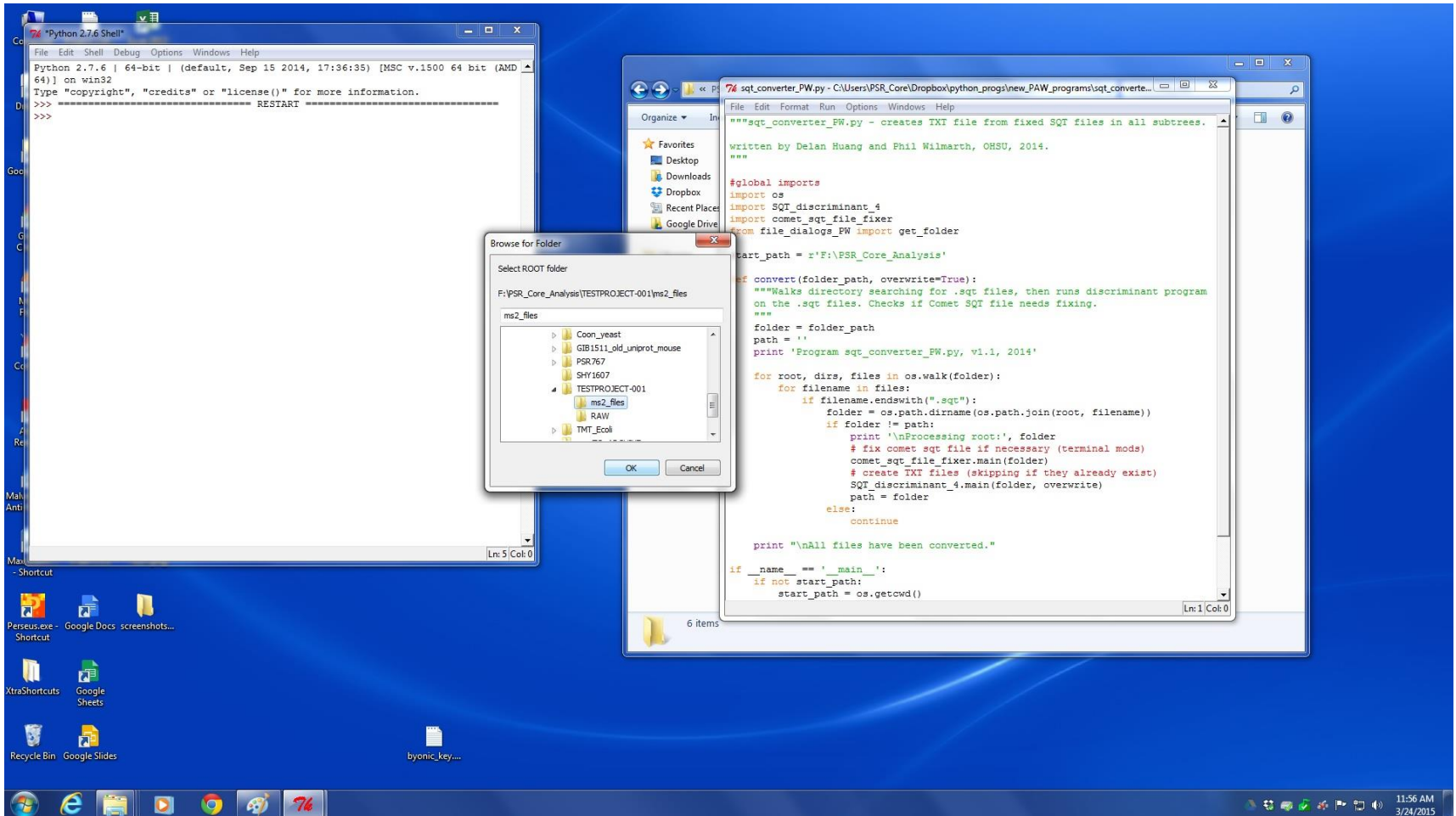
Search start: 03/24/2015, 11:45:17 AM
- Input file: COH1629_11_WT_10_20150316_OT.ms2
- Load and process input spectra
- Number of mass-charge spectra loaded: 46511
- Search progress: 100%
- Perform post-search analysis
- Write output
Search end: 03/24/2015, 11:46:38 AM

F:\PSR_Core_Analysis\TESTPROJECT-001\ms2_files>_
```

The next step is to run the **sqt\_converter.py** program. The program takes the sqt files Comet creates and, after some processing of the data, it creates txt files that the rest of the PAW Pipeline will process. The program can be launched with a double-click and run by pressing **F5** once the python window opens.



After launch, an open folder dialog box will ask you to browse to the **ms2\_files** folder containing the files you wish to analyze. After selecting the proper folder and pressing okay, the program will run. This is probably the most time-intensive part of the pipeline at the moment, and can easily take 15-20 minutes per file to do the conversion.





When the processing is complete, the output window will display **All files have been converted**. At this point, it's safe to close both windows and move on to the next step. The output will also display scans with missing data. There are commonly a couple of these scans in each analysis; however, lots of missing data could indicate a problem.

