3T Magnetic Resonance Spectroscopy Manual - Bartha Lab

By Scott Kevin Wilson

Introduction

The following will be an overview of how to complete magnetic resonance spectroscopy analysis on the 3T scanners at Robarts Research Institute. This manual was created in 2025, so please keep in mind that some changes may occur. This is not an explanation of the physics or background in spectroscopy, but merely a rundown of how to complete the analysis.

Disclaimer: I did not create any code for MAGIQ from scratch, nor do I intend to take credit for the amazing work done by Dickson Wong.

To complete the analysis for MRS you will need:

1) MAGIQ installed or accessible (currently available on Bartha Lab computer). Find the installation instructions located here under github.com/swils256/MAGIQ. See Dickson Wong's Github for more info on MAGIQ.
2) T1 Weighted Anatomicals
3) Spectroscopy data
   a) The spec data is currently saved to the DICOM server
4) A functioning .ges and .cst file for your scanner and echo time (created through PINTS)

Overview of Analysis:

Spectroscopy:

Water suppressed: Download from DICOM server -> Convert to .dat files -> Run water removal(s) -> Align .ges file to .dat file -> Run fits -> Check all fits

Water Unsuppressed: Download from DICOM server -> Convert to .dat files -> ensure water .ges and .cst align at 0 -> Run water fits

T1 Anatomicals:

Download from DICOM server -> Convert to NIFTI File (.nii.gz) -> Reorient (fslreorient2std) -> Brain Extraction (ROBEX) -> Brain Segmentation (fslfast)

To Finish:

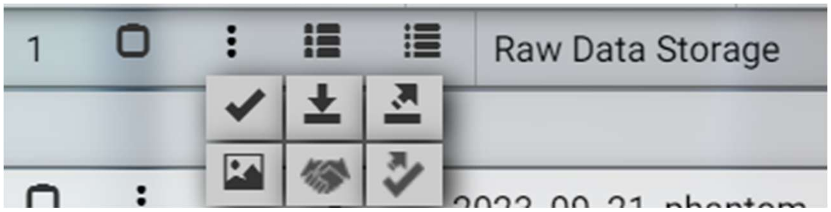
Run Quantification in Barstool -> Look over all images created to ensure brain extraction includes the entire voxel (adjust mask as necessary; fslmaths) -> Look over metabolite levels with Dr. Bartha to detect any issues

In Depth Analysis

Spectroscopy:

Acquiring data:

The current method of saving spectroscopy data is through the DICOM server. Spectroscopy data is listed under the modality **OT**, meant for 'other'. The name often includes some form of 'sup'. To download the datasets, you must individually follow the arrows to the spectroscopy data, select the three dots, and click "View DICOM Object". The data will then save, embedded into two ZIP files. After extraction, you should have two .rda files, one labelled as 'unsup', corresponding to the unsuppressed dataset.



(bottom left in this image)

If the data is not saved on the DICOM server, you must find a way to get it into the correct .rda format that fitMAN can recognize.

Conversion to .dat file:

Now that you have two .rda files, you must convert them to .dat files using fitMAN. .dat files are those that can be read into fitMAN and displayed. To do this individually for a dataset, within fitMAN, you may navigate to File -> Convert Raw Data, and a new popup should appear. Input your parameters, these are the parameters that you must change:

1) Input File Type: Select your scanner. As of 2024 the 3T scanner at Robarts is a SIEMENS
2) Input Filename: Water suppressed data
3) Reference Filename: What you want the .dat files to be named. Often the same file path as the suppressed .rda file, but with .rda switched to .dat
4) Baseline correction: Turn on
5) Select between QUECC (quality and eddy current correction) or ECC (eddy current correction). QUECC requires you to specify quality points to use - often set at 100, 200, 300, or 400.
   a) You should use more quality points if possible (start with 400). Look at the spectrum after the conversion to see if there is a quality correction error (ask Rob what this may look like – It is beneficial to run the conversion at multiple values, such as 100, 200, 300, etc. to the show Rob Bartha to decide what to use for your data). In this situation, you

should try to use less quality points, and if there are still issues at 100, use only ECC correction.

6) Reference File: Water unsuppressed .rda file path
7) Time Delay: Set to zero

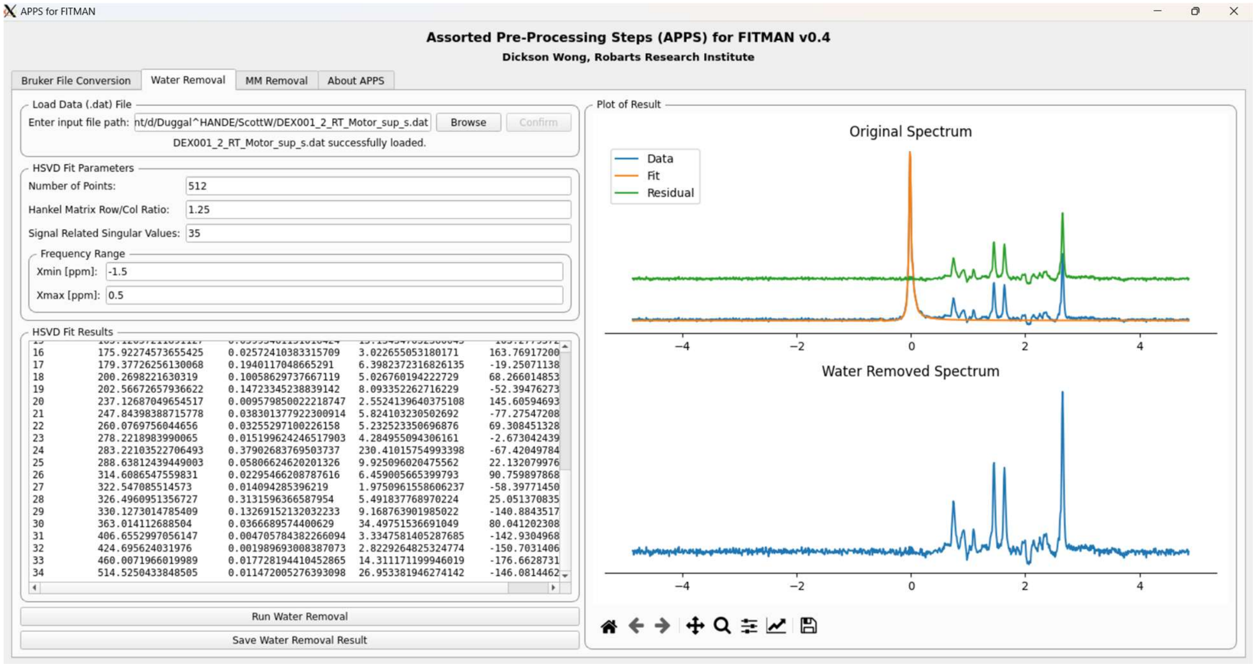
After running the conversion, you must go to the background terminal to specify the GAIN parameter for the suppressed and unsuppressed files. Set these to 0. Once run, the output .dat files will end in _s.dat or _uns.dat, for the suppressed and unsuppressed .dat files, respectively.

This can also be done (in bulk using scripts, or individually) through the wsl command line using fitMAN's **sim2fitman** function. Here are examples for QUECC 400 and just ECC, with the same parameters as labelled above:

1) QUECC 400
   a)
   ```
   /path/to/sim2fitman -ow "/path/to/suppressed.rda" -scale -bc -if -quecc 400 0.00000 "/path/to/unsuppressed.rda" -norm -rscale -rbc -rif "/path/to/suppressed.dat”
   ```
   b)
   ```
   /mnt/c/users/scott/MAGIQ/bin/fitman/Linux/sim2fitman -ow "/mnt/d/Duggal^Hande/ScottW/DEX001_2_RT_Motor_sup.rda" -scale -bc -if -quecc  400  0.00000 "/mnt/d/Duggal^Hande/ScottW/DEX001_2_RT_Motor_unsup.rda"  -norm -rscale -rbc -rif "/mnt/d/Duggal^Hande/ScottW/DEX001_2_RT_Motor_sup.dat"
   ```
2) ECC
   a)
   ```
   /path/to/sim2fitman -ow “/path/to/suppressed.rda” -scale -bc -if -ecc “/path/to/unsuppressed.rda” -norm -rscale -rbc -rif “/path/to/suppressed.dat”
   ```
   b)
   ```
   /mnt/c/users/scott/MAGIQ/fitman/bin/Linux/sim2fitman -ow "/mnt/d/Duggal^Hande/ScottW/DEX001_2_RT_Motor_sup.rda" -scale -bc -if -ecc "/mnt/d/Duggal^Hande/ScottW/DEX001_2_RT_Motor_unsup.rda"  -norm -rscale -rbc -rif "/mnt/d/Duggal^Hande/ScottW/DEX001_2_RT_Motor_sup.dat"
   ```

Running Water Removal – Water Suppressed Only:

Despite the water being suppressed in the water suppressed file, there is still some residual water that must be removed. This is done through the program APPS, which is built into MAGIQ. Navigate to the ‘Water Removal” tab within APPS, and input your water suppressed .dat file. The parameters are all preset to do a basic water removal, so click the water removal button and save the new .dat file. The new file name will end in _wr.dat. An example image is shown below:



Assorted Pre-Processing Steps (APPS) for FITMAN v0.4

Dickson Wong, Robarts Research Institute

Load Data (.dat) File

Enter input file path: nt/d/Duggal^HANDE/ScottW/DEX001_2_RT_Motor_sup_s.dat

DEX001_2_RT_Motor_sup_s.dat successfully loaded.

HSVD Fit Parameters

Number of Points: 512

Hankel Matrix Row/Col Ratio: 1.25

Signal Related Singular Values: 35

Frequency Range

Xmin [ppm]: -1.5

Xmax [ppm]: 0.5

HSVD Fit Results

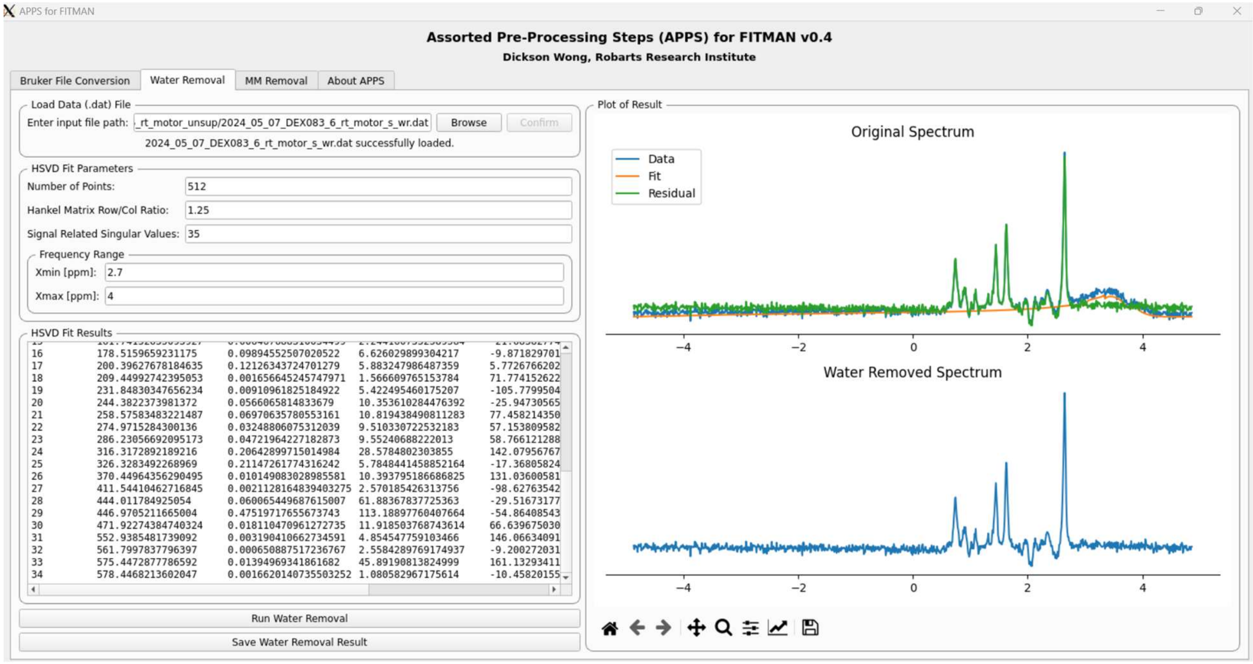
```
16          175.92274573655425    0.02572410383315709   3.022655053180171     163.76917200
17          179.37726256130068    0.1940117048665291    6.3982372316826135    -19.25071138
18          200.2698221630319     0.10058629737667119   5.026760194222729     68.266014853
19          202.56672657936622    0.14723345238839142   8.093352262716229     -52.39476273
20          237.12687049654517    0.009579850022218747  2.5524139640375108    145.60594693
21          247.84398388715778    0.038301377922300914  5.824103230502692     -77.27547208
22          260.0769756044656     0.03255297100226158   5.232523350696876     69.308451328
23          278.2218983990065     0.015199624246517903  4.284955094306161     -2.673042439
24          283.22103522706493    0.37902683769503737   230.41015754993398    -67.42049784
25          288.63812439449003    0.05806624620201326   9.925096020475562     22.132079976
26          314.6086547559831     0.02295466208787616   6.459005665399793     90.759897868
27          322.547085514573      0.014094285396219     1.9750961558606237    -58.39771450
28          326.4960951356727     0.3131596366587954    5.491837768970224     25.051370835
29          330.1273014785409     0.13269152132032233   9.168763901985022     -140.8843517
30          363.014112688504      0.0366689574400629    34.49751536691049     80.041202308
31          406.6552997056147     0.004705784382266094  3.3347581405287685    -142.9304968
32          424.695624031976      0.001989693008387073  2.8229264825324774    -150.7031406
33          460.0071966019989     0.017728194410452865  14.311171199946019    -176.6628731
34          514.5250433848505     0.011472005276393098  26.953381946274142    -146.0814462
```

Run Water Removal

Save Water Removal Result

You will notice that the water is always the giant peak situated at 0 ppm in the _s.dat files.

On some occasions, you may see that there is some residual lipid signal that is showing after the NAA peak (the largest, and last peak in 3T). If this is the case, you must change the Xmin and Xmax values to remove this lipid signal, after the regular water removal. Be sure to get as close as possible to the NAA peak but DO NOT accidentally take away any NAA signal. An example of this case is shown below:



Assorted Pre-Processing Steps (APPS) for FITMAN v0.4

Dickson Wong, Robarts Research Institute

Load Data (.dat) File

Enter input file path: _rt_motor_unsup/2024_05_07_DEX083_6_rt_motor_s_wr.dat

2024_05_07_DEX083_6_rt_motor_s_wr.dat successfully loaded.

HSVD Fit Parameters

Number of Points: 512

Hankel Matrix Row/Col Ratio: 1.25

Signal Related Singular Values: 35

Frequency Range

Xmin [ppm]: 2.7

Xmax [ppm]: 4

HSVD Fit Results

```
16          178.5159659231175     0.09894552507020522   6.626029899304217     -9.871829701
17          200.39627678184635    0.12126343724701279   5.883247986487359     5.7726766202
18          209.44992742395053    0.001656645245747971  1.566609765153784     71.774152622
19          231.84830347656234    0.00910961825184922   5.422495460175207     -105.7799504
20          244.3822373981372     0.0566065814833679    10.353610284476392    -25.94730565
21          258.57583483221487    0.06970635780553161   10.819438490811283    77.458214350
22          274.9715284300136     0.03248806075312039   9.510330722532183     57.153809582
23          286.23056692095173    0.04721964227182873   9.55240688222013      58.766121288
24          316.3172892189216     0.20642899715014984   28.5784802303855      142.07956767
25          326.3283492268969     0.21147261774316242   5.7848441458852164    -17.36805824
26          370.44964356290495    0.010149083028985581  10.393795186686825    131.03600581
27          411.54410462716845    0.0021128164839403275 2.570185426313756     -98.62763542
28          444.011784925054      0.060065449687615007  61.88367837725363     -29.51673177
29          446.9705211665004     0.47519717655673743   113.18897760407664    -54.86408543
30          471.92274384740324    0.018110470961272735  11.918503768743614    66.639675030
31          552.9385481739092     0.003190410662734591  4.854547759103466     146.06634091
32          561.7997837796397     0.000650887517236767  2.5584289769174937    -9.200272031
33          575.4472877786592     0.01394969341861682   45.89190813824999     161.13293411
34          578.4468213602047     0.0016620140735503252 1.080582967175614     -10.45820155
```

Run Water Removal

Save Water Removal Result

These files will be saved as _wr_wr.dat, since the ‘water’ removal was done twice, once for the water, and once for the lipid.

Align .ges and .cst Files:

Water suppressed data

The goal here is to create a series of peaks using prior knowledge templates so that fitMAN can run its fitting program to match the manually generated peaks to that of the data, therefore labelling each peak with a specific metabolite for quantification. The .cst file is used for all datasets in a project and it is used as an overarching background information provider, while the .ges file is different for each dataset, since it allows for small changes to better match the created peaks to the actual data. Upload the water removed .dat file, .ges, and .cst files into fitMAN. Assuming you have proper .ges and .cst files (ask Rob Bartha if unsure), you should change the parameters at the top of the .ges file:

• afactor 0.0000026	Changes amplitudes of peaks
• pfactor 1.7	Changes phase of peaks
• wfactor 3.2	Changes width of peaks
• dfactor 0.00085	Changes the delay time of peaks
• sfactor 0.02	Changes the shift of peaks

You are changing these values to try and get the peaks created by the .ges and .cst files as close to that of the acquired data. Focus on matching the NAA peak (the largest and last peak), with the shift being a super important parameter. Amplitudes are less important since they are adjusted quite well when fitting (but still good to change so you can look at the shift more easily). Once you are content that the created peaks look like that of the acquired data, you are ready for the next step.

Occasionally, you may need to change the .cst file to adjust for a couple of weird datasets. If this is the case, ask Dr. Bartha for guidance, but make sure to make a copy of the original .cst file to change and label is according to the dataset it is being changed for.

There is no script for this step, and it must be completed manually.

Water unsuppressed data

The water unsuppressed files only have one very large peak at 0 ppm: water. Thus, there is no need to change any parameters in a water .ges or .cst file since there is only one peak at 0 ppm, and fitMAN will adjust the amplitudes as required. So, just ensure that you have a proper water .ges and .cst files.

Running Fits

Water suppressed data

Generating a fit for an individual dataset can be completed in fitMAN using through File -> Generate Fit. This does not actually work on my computer for some reason, so I cannot go into the specific inputs for this built in generator. However, there is a command line function that can be ran to generate the fit in the command prompt, which uses the **ultra_fitman** function. The format for this is as follows:

```
/path/to/ultrafitman /path/to/.dat /path/to/.ges /path/to/.cst /path/to/.out
```

This format calls on the ultra_fitman function and provides it with the three necessary files needed for the fitting, as well as what you would like the output file to be named and where it should be saved. For simplicity's sake, keep the output file name the same as the .dat file, and save it in the exact same place. Separate each file path by only one space each. Here is one example of this format:

```
/mnt/c/users/scott/magiq/fitman/bin/linux/ultra_fitman
/mnt/d/Duggal^HANDE/Scott/20190326_2019_03_26_DEX001_2/DEX001_2_RT_Motor_sup/D
EX001_2_RT_Motor_sup_s_wr.dat
/mnt/d/Duggal^HANDE/Scott/20190326_2019_03_26_DEX001_2/DEX001_2_RT_Motor_sup/te
st.ges /mnt/d/Duggal^HANDE/Scott/gesandcst/FINALscylloed.cst
/mnt/d/Duggal^HANDE/Scott/20190326_2019_03_26_DEX001_2/DEX001_2_RT_Motor_sup/D
EX001_2_RT_Motor_sup_s_wr.out
```

Recall that the .ges file will be different for each dataset, while the .cst file will be the same for all data in one study (assuming TE/TR and acquisition sequence is the same).

Water unsuppressed data

You will also need to generate fits for all of the water unsuppressed data. This is much easier since you should really only have one .ges and .cst file being used, and no adjustments are needed as long as the peak created by these two files lies at 0 ppm (the fitting function will adjust the amplitudes as needed).

The same command line format will be used for individual dataset fitting:

```
/path/to/ultrafitman /path/to/.dat /path/to/.ges /path/to/.cst /path/to/.out
```

Example:

```
/mnt/c/users/scott/magiq/fitman/bin/linux/ultra_fitman
/mnt/d/Duggal^HANDE/Scott/20190326_2019_03_26_DEX001_2/DEX001_2_RT_Motor_sup/D
EX001_2_RT_Motor_sup_uns.dat /mnt/d/Duggal^HANDE/Scott/gesandcst/water.ges
```

/mnt/d/Duggal^HANDE/Scott/gesandcst/water.cst
/mnt/d/Duggal^HANDE/Scott/20190326_2019_03_26_DEX001_2/DEX001_2_RT_Motor_sup/DEX001_2_RT_Motor_sup_uns.out

At this point, you should have two .out files, one for each of the suppressed and unsuppressed .dat files.

Check all fits:

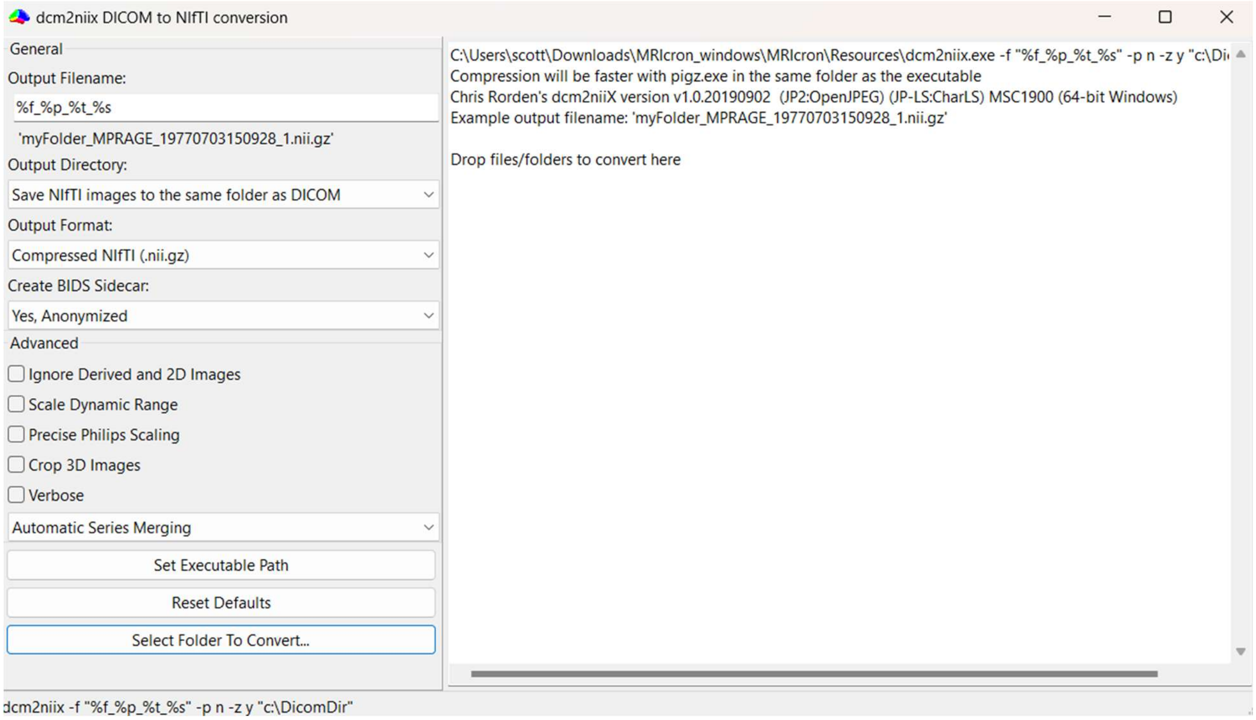
You should now set up a meeting with Dr. Bartha to sit down and go through all of the water suppressed fits that you have created. He will let you know if there are any issues and how to fix them if there are. In the meantime, you can move onto the Anatomicals section.

I have a script that goes through all of the fits that you have generated and takes a screenshot of each, so that you can just go through the .png's instead of pulling up every .dat and .out file in fitMAN to show Rob Bartha. This script is mentioned at the end of this manual and is very helpful.

Anatomicals:

Acquiring Data and Conversion to .nii.gz:

The data acquisition for the T1 anatomicals is similar to that of the spec data. However, if the data is on the DICOM server, you can now use the regular download button, and do not have to navigate through to the "View DICOM object". The T1s can sometimes not be labled on the DICOM server as such. I have seen the names start with 'mprrage' sometimes, ask Dr. Bartha if you are unsure of the names of the T1's. When downloading the data from the DICOM server manually (without the script, see Spectroscopy->Acquiring Data, and below), you will download multiple .dcm (DICOM) files in a folder. These must be converted to nifty (.nii.gz) files, which can be done using the app mricron.exe. In mricron, go to Import-> Convert DICOM to Nifti, change the "Output File" to be .nii.gz, and drag the entire folder containing all of the .dcm files into the conversion window.


dcm2niix DICOM to NIfTI conversion
General
Advanced
Set Executable Path
Reset Defaults
Select Folder To Convert...

Output Filename: %f_%p_%t_%s

'myFolder_MPRAGE_19770703150928_1.nii.gz'

Output Directory: Save NIfTI images to the same folder as DICOM

Output Format: Compressed NIfTI (.nii.gz)

Create BIDS Sidecar: Yes, Anonymized

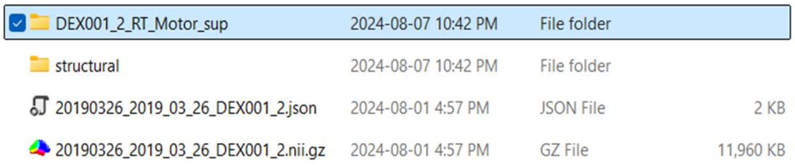
☐ Ignore Derived and 2D Images
☐ Scale Dynamic Range
☐ Precise Philips Scaling
☐ Crop 3D Images
☐ Verbose

Automatic Series Merging

```
C:\Users\scott\Downloads\MRIcron_windows\MRIcron\Resources\dcm2niix.exe -f "%f_%p_%t_%s" -p n -z y "c:\Di
Compression will be faster with pigz.exe in the same folder as the executable
Chris Rorden's dcm2niiX version v1.0.20190902  (JP2:OpenJPEG) (JP-LS:CharLS) MSC1900 (64-bit Windows)
Example output filename: 'myFolder_MPRAGE_19770703150928_1.nii.gz'

Drop files/folders to convert here
```

```
dcm2niix -f "%f_%p_%t_%s" -p n -z y "c:\DicomDir"
```

The new files will be created in the same directory the .dcm containing folder is located. If you use the download_datasets_3T.py script, the correct prefix for all T1's should be labeled in the config.yml file (hopefully they all start the same way). The .dcm files will be saved to the folder labeled 'structural', and the conversion to .nii.gz files will be done automatically, and saved alongside the 'structural' folder and the spectroscopy data folder, as seen below:



☑ DEX001_2_RT_Motor_sup	2024-08-07 10:42 PM	File folder	
structural	2024-08-07 10:42 PM	File folder	
20190326_2019_03_26_DEX001_2.json	2024-08-01 4:57 PM	JSON File	2 KB
20190326_2019_03_26_DEX001_2.nii.gz	2024-08-01 4:57 PM	GZ File	11,960 KB

If you get the data from outside of the DICOM server, you will need to find a way to get to a .nii.gz file for the T1 weighted image. There are likely to be file convertors online, so check there.

Anatomical Reorientation:

The next step is to reorient each anatomical images. This is accomplished using an FSL function. The function used is *fslreorient2std*. The function must be used in the environment that you have FSL downloaded in (whether it be terminal, wsl, etc.). It should also be noted that I have had some trouble in the past with using this function for files on an external hard drive, so you may need to transfer any files to your computers local drive. The correct syntax for this function is

```
Fslreorient2std input_filename output_filename
```

The output filename should have the same name as the input filename, with _std at the end. An example of what I input into wsl for the function is shown below:

```
fslreorient2std "C:\Users\scott\Documents\FSLSTD\DEX059_6.nii.gz"
"C:\Users\scott\Documents\FSLSTD\DEX059_6_std.nii.gz"
```

Brain Extraction Using ROBEX:

The brain extraction step can be done using FSL, and this tool is built into MAGIQ (see BARSTOOL -> Brain Extraction). However, ROBEX offers an easier and quicker way to extract the brain from anatomical images. ROBEX is a tool that can be downloaded from the internet, and is run using the command line. See here for the download. ROBEX can be run in the command line, after setting the directory to that of the location of ROBEX. The syntax that will be used for ROBEX is:

```
ROBEX.exe original_filename brain_extracted_filename mask_filename
```

The brain extracted filename should be that of the original filename, with _brain, and the mask filename should be that of the original filename, with _mask at the end. An example of what this looks like is shown below:

```
cd C:\Users\scott\Documents\ROBEX
```

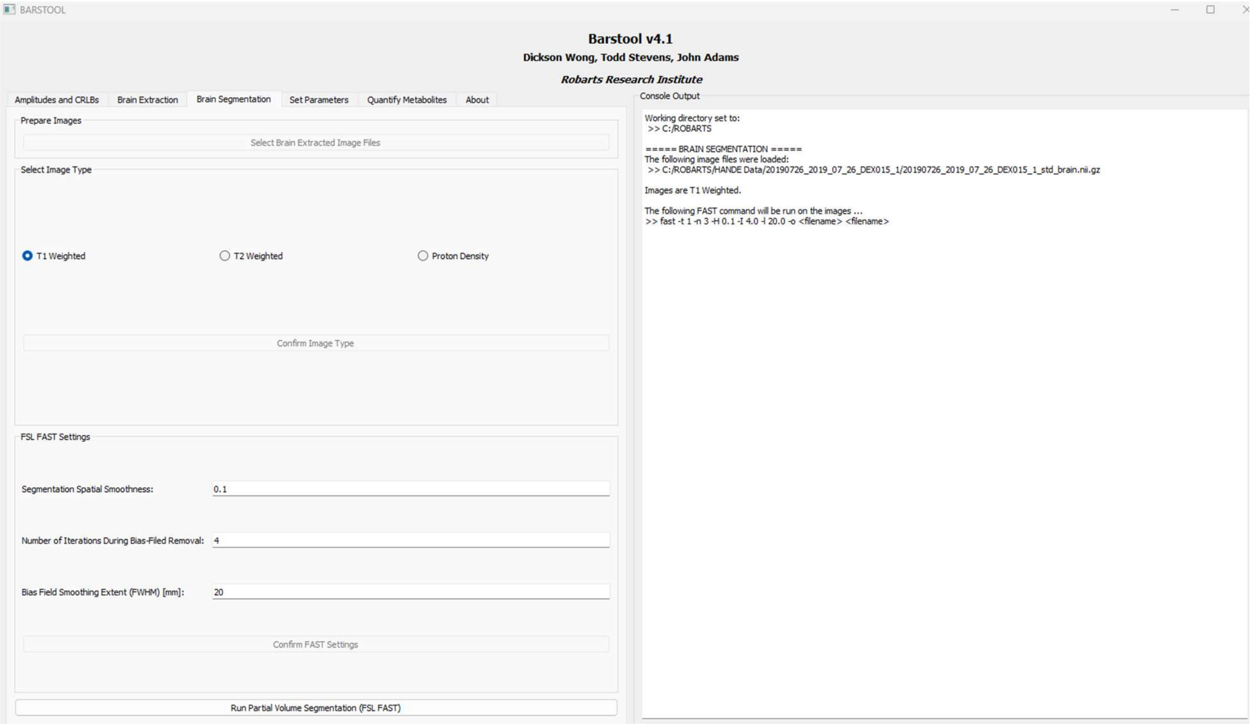
```
ROBEX.exe "C:\Users\scott\Documents\Data\DEX001_20181126_std.nii.gz"
"C:\Users\scott\Documents\Data\DEX001_20181126_std_brain.nii.gz"
"C:\Users\scott\Documents\Data\DEX001_20181126_std_brain_mask.nii.gz"
```

The _mask file will also be created using this. It is not critical at this time but may be needed later if there is part of the voxel being removed during extraction. This will be determined following final quantification.

Brain Segmentation

The final step for the anatomicals is to segment them into grey matter, white matter, and CSF. This can be done using MAGIQ, as well as in the command line. To perform this in MAGIQ, navigate to BARSTOOL, and to the tab labelled 'Brain Segmentation'. This tool uses an FSL function *fslFAST* to perform the segmentation.

To have permission to select files in this tab (and all tabs in BARSTOOL), you must first set the working directory under the 'Amplitudes and CRLBs' tab. Once the directory is set, use the 'Brain Segmentation', select the image(s) that were created during the brain extraction tool, with the suffix _brain.nii.gz. Once this is done, confirm the image type (should be a T1 weighted image), confirm the FAST settings (I do not know what these do, the default settings have worked with no issues for me), and then click the button "Run Partial Volume Segmentation (FSL FAST)". An image of what this should look like is shown below:



● T1 Weighted ○ T2 Weighted ○ Proton Density

Segmentation Spatial Smoothness: 0.1

Number of Iterations During Bias-Filed Removal: 4

Bias Field Smoothing Extent (FWHM) [mm]: 20

```
Working directory set to:
 >> C:/ROBARTS

===== BRAIN SEGMENTATION =====
The following image files were loaded:
 >> C:/ROBARTS/HANDE Data/20190726_2019_07_26_DEX015_1/20190726_2019_07_26_DEX015_1_std_brain.nii.gz

Images are T1 Weighted.

The following FAST command will be run on the images ...
>> fast -t 1 -n 3 -H 0.1 -I 4.0 -l 20.0 -o <filename> <filename>
```

This can also be done in bulk using the environment where FSL is installed. To do this, you must use the command:

Path/to/fsl/bin/fast -t 1 -n 3 -H 0.1 -I 4 -l 20.0 -v -o input_filename output_filename

It is important that the first part is the path in your computers files to locate the bin/fast portion of FSL. The middle portion is the parameter settings that are preset in BARSTOOL. The input filename should be your _brain.nii.gz file, while the output should have the same name as the input, with _seg following _brain (_brain_seg.nii.gz). An example of me using this function is seen below (in wsl):

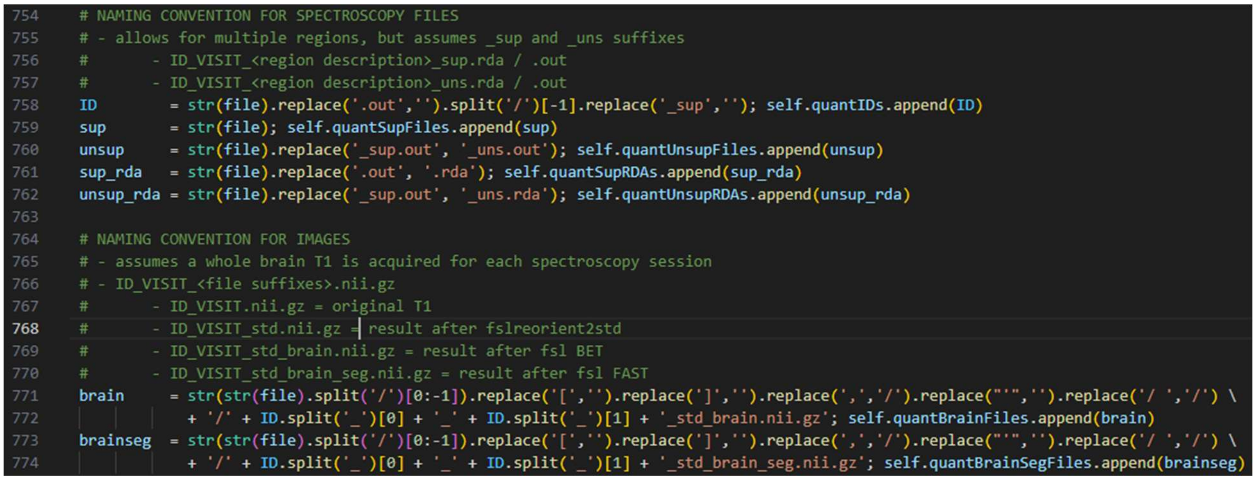
/home/swils256/fsl/bin/fast -t 1 -n 3 -H 0.1 -I 4 -l 20.0 -v -o
"D:\HANDE\Scott\20220201\DEX059_6_RT\DEX059_6_std_brain.nii.gz"
"D:\HANDE\Scott\20220201\DEX059_6_RT\DEX059_6_std_brain_seg.nii.gz"

You should then have a new _brain_seg.nii.gz file alongside your _brain.nii.gz file.

Final Steps:

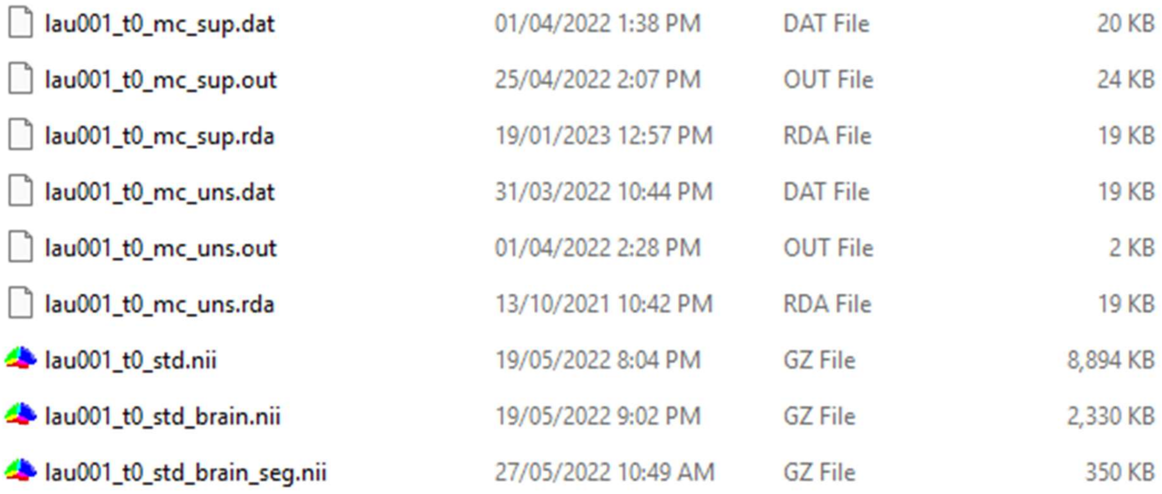
Filename Setup:

You have now reached a point where you should have a reoriented, brain extracted, and segmented anatomical. You should also have .out, .rda, and .dat files for the suppressed and unsuppressed data. At this point, you will need the *.rda*, *.out*, *_std_brain.nii.gz*, and *_std_brain_seg.nii.gz* files. You will need to rename each of these files according to a strict naming convention. This convention is outlined below using the code from barstool.py.



```
# NAMING CONVENTION FOR SPECTROSCOPY FILES
# - allows for multiple regions, but assumes _sup and _uns suffixes
#       - ID_VISIT_<region description>_sup.rda / .out
#       - ID_VISIT_<region description>_uns.rda / .out
ID        = str(file).replace('.out','').split('/')[-1].replace('_sup',''); self.quantIDs.append(ID)
sup       = str(file); self.quantSupFiles.append(sup)
unsup     = str(file).replace('_sup.out', '_uns.out'); self.quantUnsupFiles.append(unsup)
sup_rda   = str(file).replace('.out', '.rda'); self.quantSupRDAs.append(sup_rda)
unsup_rda = str(file).replace('_sup.out', '_uns.rda'); self.quantUnsupRDAs.append(unsup_rda)

# NAMING CONVENTION FOR IMAGES
# - assumes a whole brain T1 is acquired for each spectroscopy session
# - ID_VISIT_<file suffixes>.nii.gz
#       - ID_VISIT.nii.gz = original T1
#       - ID_VISIT_std.nii.gz = result after fslreorient2std
#       - ID_VISIT_std_brain.nii.gz = result after fsl BET
#       - ID_VISIT_std_brain_seg.nii.gz = result after fsl FAST
brain     = str(str(file).split('/')[0:-1]).replace('[','').replace(']','').replace(',','/').replace("'",'').replace('/ ','/') \
            + '/' + ID.split('_')[0] + '_' + ID.split('_')[1] + '_std_brain.nii.gz'; self.quantBrainFiles.append(brain)
brainseg  = str(str(file).split('/')[0:-1]).replace('[','').replace(']','').replace(',','/').replace("'",'').replace('/ ','/') \
            + '/' + ID.split('_')[0] + '_' + ID.split('_')[1] + '_std_brain_seg.nii.gz'; self.quantBrainSegFiles.append(brainseg)
```

The comments in the code accurately describe how the files should be named. That is, they should be named according their ID number (i.e. DEX059), then their visit (i.e. 1, 2, or t0, t1, etc.), and finally the respective suffixes for each filename. The spectroscopy files should also have an extra area that describes the brain region (i.e. mc for motor cortex). It is important that you name all flies in this manner. A screenshot of what the files should look like is shown below.



lau001_t0_mc_sup.dat	01/04/2022 1:38 PM	DAT File	20 KB
lau001_t0_mc_sup.out	25/04/2022 2:07 PM	OUT File	24 KB
lau001_t0_mc_sup.rda	19/01/2023 12:57 PM	RDA File	19 KB
lau001_t0_mc_uns.dat	31/03/2022 10:44 PM	DAT File	19 KB
lau001_t0_mc_uns.out	01/04/2022 2:28 PM	OUT File	2 KB
lau001_t0_mc_uns.rda	13/10/2021 10:42 PM	RDA File	19 KB
lau001_t0_std.nii	19/05/2022 8:04 PM	GZ File	8,894 KB
lau001_t0_std_brain.nii	19/05/2022 9:02 PM	GZ File	2,330 KB
lau001_t0_std_brain_seg.nii	27/05/2022 10:49 AM	GZ File	350 KB

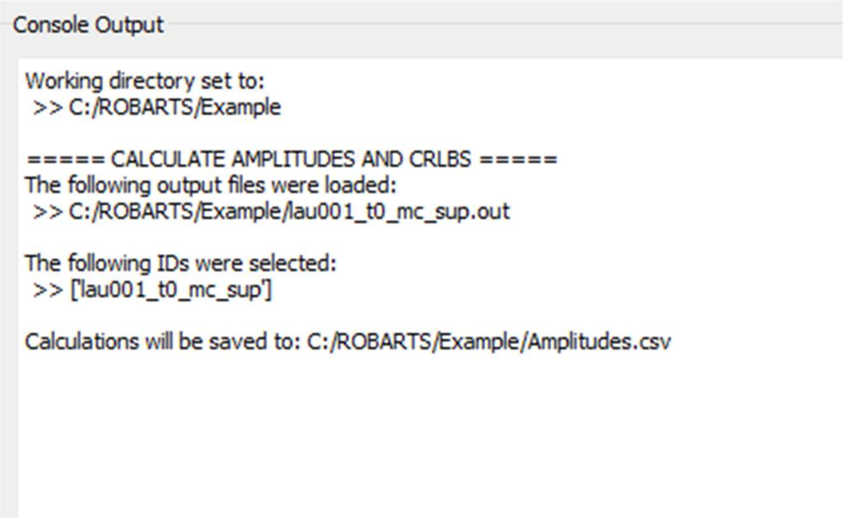
I should note that this code can be changed to alleviate some of the restrictions in naming. I actually made changes myself, but lost that version of MAGIQ in a computer crash. In my version, I had the program use every _sup.out file as a basename, and found each suffix that contained the same prefix (i.e. whatever was found before the _sup.out file). You may benefit from using AI to create a script to rename the files for you.

If you have a large number of spectra that need quantification, you should add all of the renamed files for each dataset into a new folder, so that they can all be selected at once in the following steps.

I know this is a very tedious process, sorry.

Barstool Quantification:

You are now at the final stretch of the analysis. To complete this, you will need to use BARSTOOL. In BARSTOOL, you must first set the working directory in the 'Amplitudes and CRLBs' tab. Here, you will also want to find the amplitudes and CRLBs. To do this, select the _sup.out file you would like to use. If you have all of the files for your data in one folder, you can select multiple _sup.out files. After this, select "Confirm Study IDs", choose a convenient filename, confirm it, and click the final button at the bottom. The console output side of barstool (right side) should look something like this:

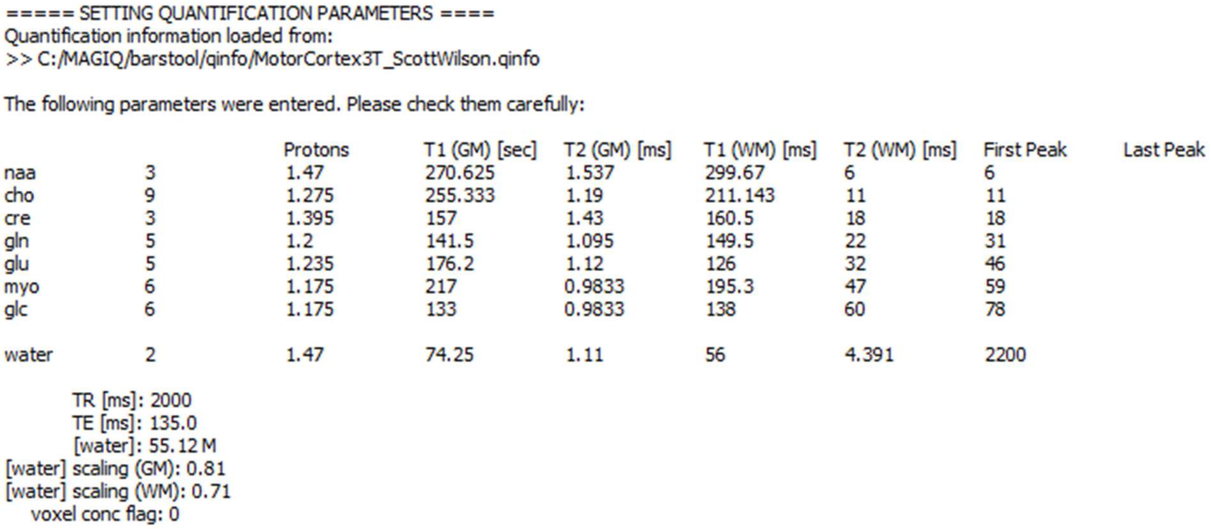


```
Working directory set to:
 >> C:/ROBARTS/Example

===== CALCULATE AMPLITUDES AND CRLBS =====
The following output files were loaded:
 >> C:/ROBARTS/Example/lau001_t0_mc_sup.out

The following IDs were selected:
 >> ['lau001_t0_mc_sup']

Calculations will be saved to: C:/ROBARTS/Example/Amplitudes.csv
```

Now, you will have these values located in the filename and path that you chose.

Set Parameters

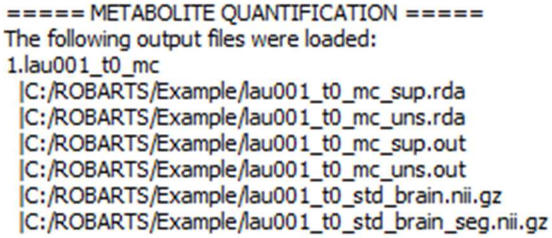
This is a very simple, but very important step. Here, under the 'Set Parameters' tab, you will need to select a filename with predetermined values using the 'Load' button at the bottom right.. In the version of MAGIQ attached, there are a few files. If you are working with the motor cortex, you are in luck – select the file named *MotorCortex3T_ScottWilson.qinfo*. Many people have incorrectly selected other files that were not for the motor cortex, and I had to find the correct values and rerun their analyses. If you are not working with the motor cortex, stop here. Speak with Rob Bartha, and discuss doing a literature search to compile some values for your region of interest. Once you have selected the correct file, determine if you are doing a tissue concentration calculation or voxel concentration calculation (ask Rob Bartha if unsure). I suspect you will be doing the former. Finally, click ‘Confirm Quantification Parameters and ensure something like the following appears in the Console Output:



```
===== SETTING QUANTIFICATION PARAMETERS ====
Quantification information loaded from:
>> C:/MAGIQ/barstool/qinfo/MotorCortex3T_ScottWilson.qinfo

The following parameters were entered. Please check them carefully:

                    Protons   T1 (GM) [sec]  T2 (GM) [ms]  T1 (WM) [ms]  T2 (WM) [ms]  First Peak   Last Peak
naa        3        1.47      270.625        1.537         299.67        6             6
cho        9        1.275     255.333        1.19          211.143       11            11
cre        3        1.395     157            1.43          160.5         18            18
gln        5        1.2       141.5          1.095         149.5         22            31
glu        5        1.235     176.2          1.12          126           32            46
myo        6        1.175     217            0.9833        195.3         47            59
glc        6        1.175     133            0.9833        138           60            78

water      2        1.47      74.25          1.11          56            4.391         2200

          TR [ms]: 2000
          TE [ms]: 135.0
          [water]: 55.12 M
[water] scaling (GM): 0.81
[water] scaling (WM): 0.71
    voxel conc flag: 0
```

Quantifiy Metabolites

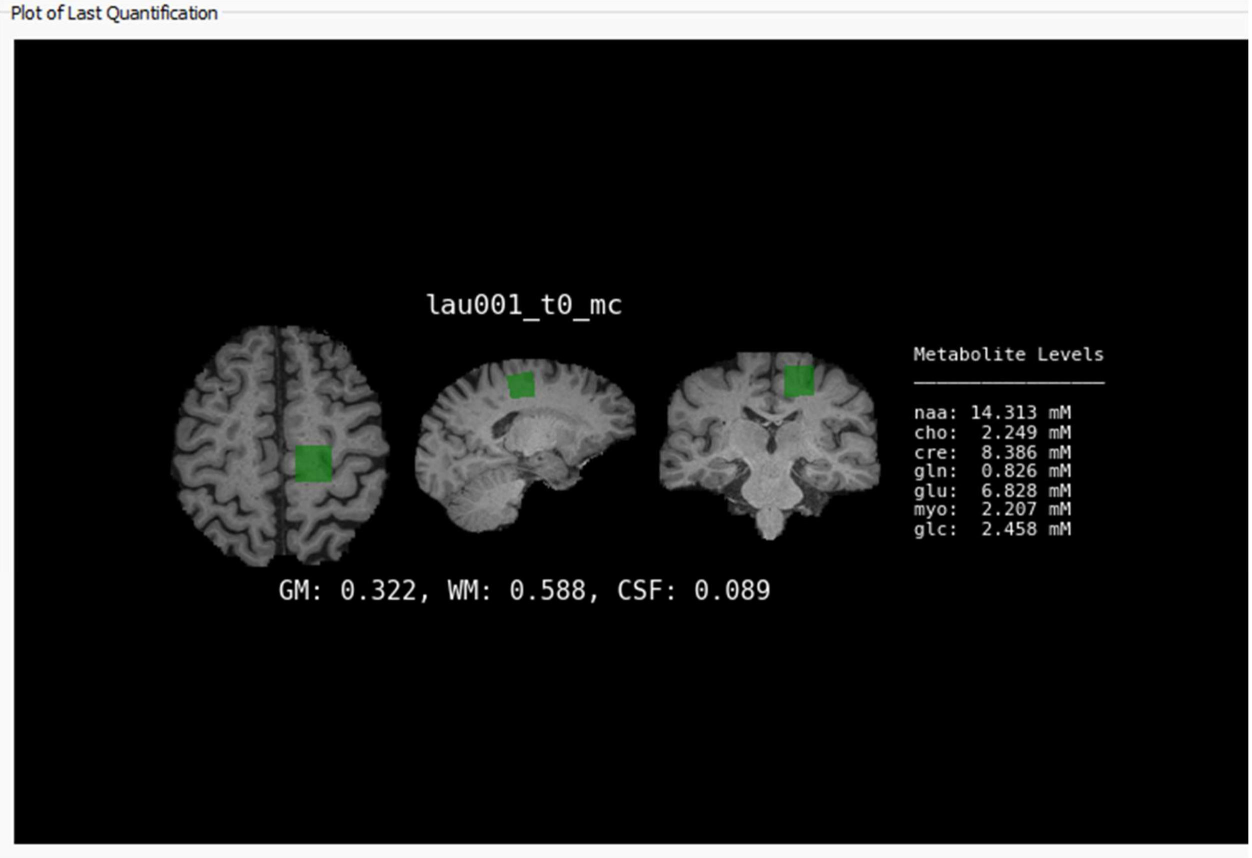
Navigate to the ‘Quantify Metabolites’ tab. Here, you will select ‘Load Suppressed Output Files’, and select the _sup.out file(s), as you may be doing many datasets in bulk. Here it is crucial that you have the correct naming convention described above. Once you have selected the files, ensure that 6 files were displayed in the console for each dataset you are doing, as seen below:



```
===== METABOLITE QUANTIFICATION =====
The following output files were loaded:
1.lau001_t0_mc
  |C:/ROBARTS/Example/lau001_t0_mc_sup.rda
  |C:/ROBARTS/Example/lau001_t0_mc_uns.rda
  |C:/ROBARTS/Example/lau001_t0_mc_sup.out
  |C:/ROBARTS/Example/lau001_t0_mc_uns.out
  |C:/ROBARTS/Example/lau001_t0_std_brain.nii.gz
  |C:/ROBARTS/Example/lau001_t0_std_brain_seg.nii.gz
```

Now, select a path and name for the .csv file that will be created with all the final metabolite levels. Click ‘Confirm Save File Name’, select the scanner type (I believe that only the Siemens version works), and click ‘Run Quantification’.

The quantification will run, and may take a few minutes if you are doing multiple datasets. You will know it has complted when an image of the brain with the voxel overlay, and some metabolites values is shown, like so:



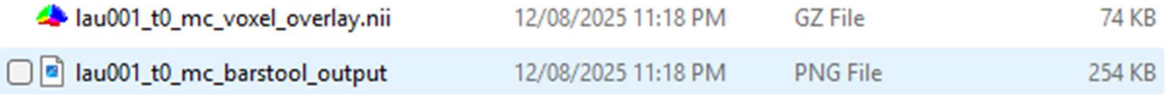
Plot of Last Quantification

lau001_t0_mc

```
Metabolite Levels
────────────────
naa: 14.313 mM
cho:  2.249 mM
cre:  8.386 mM
gln:  0.826 mM
glu:  6.828 mM
myo:  2.207 mM
glc:  2.458 mM
```

GM: 0.322, WM: 0.588, CSF: 0.089

This image, alongside a voxel overlay .nii.gz file, will appear with the same file naming convention in your folder:



lau001_t0_mc_voxel_overlay.nii	12/08/2025 11:18 PM	GZ File	74 KB
□ lau001_t0_mc_barstool_output	12/08/2025 11:18 PM	PNG File	254 KB

Ensure the Extracted Brain Includes the Entire Voxel

The final step is to ensure that none of the voxel was cut out during the brain extraction process. To do this, carefully inspect each voxel overlay/extracted brain in FSLeyes. Have the extracted brain be the top layer, with the voxel overlay in the back with a visible color. The idea is to ensure that the entirety of the voxel is contained in the extracted brain (in this case you want the voxel to be covered by the extracted brain). Sometimes, too much of the brain will be trimmed off in the extraction, leaving the voxel behind with no brain.

If too much of the brain was trimmed, indicated by the voxel hanging partially off the brain, you must edit the mask that was created earlier to include the entire voxel. To do this, you should pull up the mask in FSL, along side the voxel in the back. Use the 'Edit' tool, alonside the pencil to extend the mask to cover any of the voxel that was not previously covered. A detailed explanation of each edit tool in FSL can be found here. Note that you want the entire mask to have a fill value of 1.

Once you have a new mask, you will need to use fslmaths to multiply the mask onto the _std.nii.gz file to create a new _std_brain.nii.gz file. To do this, you should use the environment where fsl is found, and use:

```
Fslmaths mask_file -mul std_file brain_file
```

Example seen below:

```
fslmaths /mnt/d/HANDE/Scott/20221130/DEX064_20221130_std_brain_mask.nii.gz -mul
/mnt/d/HANDE/Scott/20221130/DEX064_20221130_std.nii.gz
/mnt/d/HANDE/Scott/20221130/DEX064_20221130_std_brain.nii.gz
```

This will create a new _brain.nii.gz file that contains the entire voxel. From here, you must redo the segmentation with the new brain extracted image. Then, ensuring the proper file naming convention, rerun the quantification.

Inspect the Final Values With Rob Bartha

You are now at the end! You should meet with Rob Bartha to look over the final values you have created. He will be able to find any inconstencies or things that do not look right. Congratulations!

Thank you for reading through and following the instruction portion of this manual, I hope it was of assistance!

Script Information

The following is a breif description of some scripts that I have created along the way. These scripts sped up my process exponentially as lots of tedioius tasks could be done super fast, and all command line tasks could be automated. I recommend you look through some of them, and use AI to tweak them to your needs, or even create your own. Note that not all scripts that are attached to this are explained.

Acquiring Spectroscopy Data

I have a script that can download the datasets in bulk, so that you do not have to do all this clicking through. It is called **download_datasets_3T.py**. This script is an adapted version that was created by Igor Solovey for the 9.4T data. This program also downloads the anatomicals and converts them to .nii.gz files (more info in anatomicals. For this script to work, the proper parameters should be specified in the required config.yml file. In this file, you must specify your username to access the DICOM server, the study description, the dates that you want scans downloaded from, and the names of patients (I put 20* since all patients names start with the date). Also specify the names spectroscopy and T1 files. From here, run the code and it will download the spectroscopy data and anatomicals. Note that this script runs using a keycloak connection to the DICOM server, and thus loses that connection over time due to suspected inactivity, so you will have to change the study dates to reflect the remaining undownloaded files and rerun the script until it breaks again.

Conversion to .dat File

I have created a script that can run the conversion automatically, as well as inputting the 0's in the terminal for the GAIN parameters. This script is called 'batchRDAtoDAT.py'. This script looks through folders from a root directory and uses folders that contain two .rda files, with the one that contains 'uns' in it being the unsuppressed file, and the other as suppressed. You may need to change the file detection process if your file names are wonky (for me, all the unsuppressed files had uns in the filename).

Water Removal

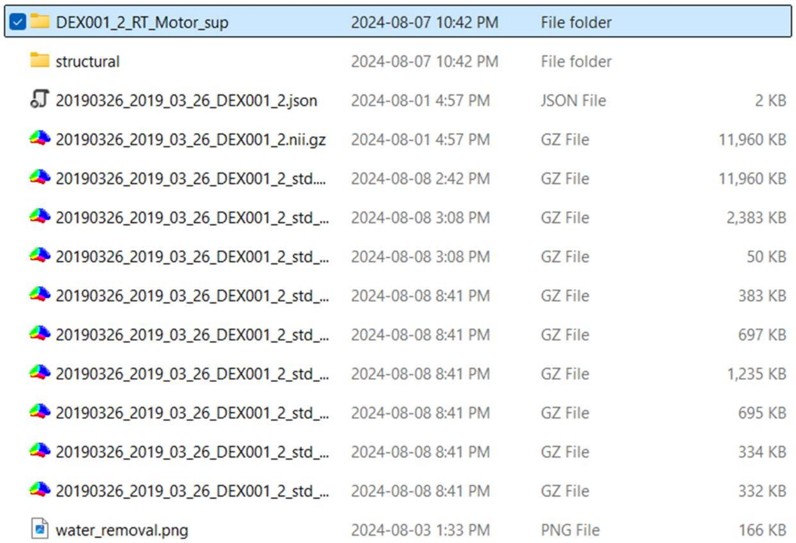
I once again have a script (adapted from Igor's for 9.4T) that can run this in batch, called 'waterremoval.py'. This script requires one additional file named 'config.yml', which is where the root directory is specified. After listing the directory in the 'output_directory' parameter (at the very top), run the program, which will then go through all the files in the listed directory and run the basic water conversion on those that end in _s.dat. This file ending is what the RDA to

DAT file conversion creates by default, so if the file has been converted using fitMAN, then it should be named correctly for the conversion.

Water Suppressed Fitting

Once again, I have created a batch script called **Scott_batch_fits.py** that can run all the water suppressed fits at once. It relies on a very specific format but can be adapted relatively easily to suit your needs. Here is the format that it uses as well as what you may need to change:

- State the base directory that all of your data is found, as the parameter base_dir found at the start of the script
- State your path to ultra_fitman
- State your path to the .cst file that is being used
- Ideally, your data should be setup so that there is one main folder for each dataset, which are found within the base directory. Each main folder should then have one more subfolder with all of the spec data found within. See image as an example:

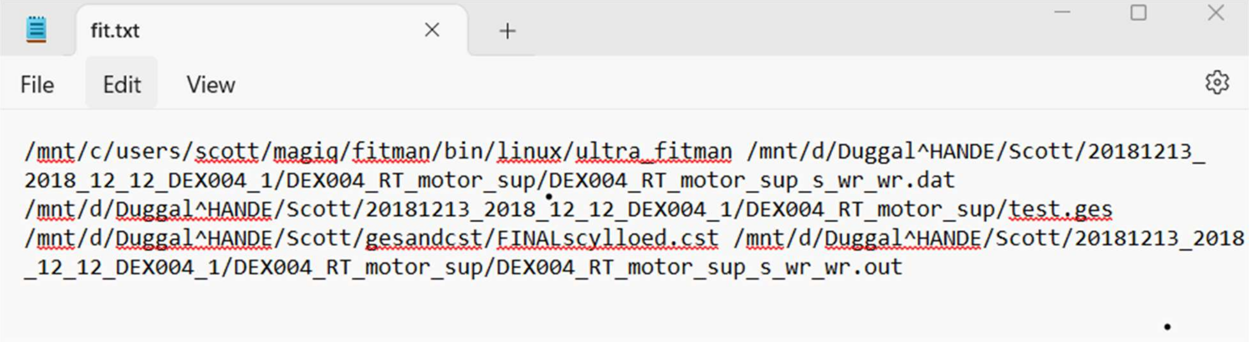


☑ DEX001_2_RT_Motor_sup	2024-08-07 10:42 PM	File folder	
structural	2024-08-07 10:42 PM	File folder	
20190326_2019_03_26_DEX001_2.json	2024-08-01 4:57 PM	JSON File	2 KB
20190326_2019_03_26_DEX001_2.nii.gz	2024-08-01 4:57 PM	GZ File	11,960 KB
20190326_2019_03_26_DEX001_2_std....	2024-08-08 2:42 PM	GZ File	11,960 KB
20190326_2019_03_26_DEX001_2_std_...	2024-08-08 3:08 PM	GZ File	2,383 KB
20190326_2019_03_26_DEX001_2_std_...	2024-08-08 3:08 PM	GZ File	50 KB
20190326_2019_03_26_DEX001_2_std_...	2024-08-08 8:41 PM	GZ File	383 KB
20190326_2019_03_26_DEX001_2_std_...	2024-08-08 8:41 PM	GZ File	697 KB
20190326_2019_03_26_DEX001_2_std_...	2024-08-08 8:41 PM	GZ File	1,235 KB
20190326_2019_03_26_DEX001_2_std_...	2024-08-08 8:41 PM	GZ File	695 KB
20190326_2019_03_26_DEX001_2_std_...	2024-08-08 8:41 PM	GZ File	334 KB
20190326_2019_03_26_DEX001_2_std_...	2024-08-08 8:41 PM	GZ File	332 KB
water_removal.png	2024-08-03 1:33 PM	PNG File	166 KB

  -
  - This folder is found within D:\Duggal^HANDE\Scott\20190326_2019_03_26_DEX001_2, where D:\Duggal^HANDE\Scott\ is the base_dir, and 20190326_2019_03_26_DEX001_2 is one dataset found inside
  - Spec data is found within DEX001_2_RT_Motor_sup
  - If you have used the download script that I mention above, the data SHOULD already be in this format, assuming you added the .ges files alongside the .dat files
  - Additional info: the structural folder is where the raw anatomicals are downloaded to, while the .nii.gz files are the anatomicals we work with (see anatomicals section).

- The .ges files are assumed to be called “test.ges” and there should be only one found within each folder that contains a .dat file to be used. If you change the name of your .ges files, you can tweak the code to allow for any .ges file names, as long as there is only one per corresponding .dat file.
- As you may recall, you will have more that one .dat file in a spec data folder due to having the original suppressed and unsuppressed .dat file, the _wr.dat file, and the _wr_wr.dat file if lipid removal was needed. You may also need to do additional modification to the .dat file such as phasing (speak with Rob if you suspect any issues like this). These additional modifications should end in _test.dat.
    - The code now goes through each spec subfolder and pulls the .dat file in this hierarchal order _test.dat>_wr_wr.dat>_wr.dat, to ensure that the most modified .dat file is always used.
        - Simply, if a _test.dat file exist, it uses that over all others. If there is no _test.dat, it looks for _wr_wr.dat, and if that does not exist, it just uses a simple _wr.dat file

The script also creates a file called fit.txt file in each spec subfolder that was fit, where it puts the command that was used to generate the fit. This way, if you need to adjust the fit just for one dataset, you can go through and copy/paste the command from this file instead of typing it all out. Example:



```
/mnt/c/users/scott/magiq/fitman/bin/linux/ultra_fitman /mnt/d/Duggal^HANDE/Scott/20181213_
2018_12_12_DEX004_1/DEX004_RT_motor_sup/DEX004_RT_motor_sup_s_wr_wr.dat
/mnt/d/Duggal^HANDE/Scott/20181213_2018_12_12_DEX004_1/DEX004_RT_motor_sup/test.ges
/mnt/d/Duggal^HANDE/Scott/gesandcst/FINALscylloed.cst /mnt/d/Duggal^HANDE/Scott/20181213_2018
_12_12_DEX004_1/DEX004_RT_motor_sup/DEX004_RT_motor_sup_s_wr_wr.out
```

Without this batch script, you will have to manually input a command for each dataset and allow it to run. Feasible, but takes a while.

Water Unsuppressed Fitting

The script to be used to batch fit the unsuppressed data is called **Scott_batch_unsfits.py.** This script relies on the same file format and similar required parameters as the suppressed data fitting script:

- At the top of the script state your: base_dir, ultra_fitman path, .cst file path, and .ges file path

- The .ges file path is stated at the top now since there is only one .ges file for all unsuppressed datasets
- In the base directory, there should be a folder for each dataset, with a spectroscopy subfolder found within each of these, where the unsuppressed .dat files can be found.
- The script uses the .dat file that ends in _uns.dat, which is the file ending given to the unsuppressed .dat files following RDA to DAT conversion.

Fit Screenshots

fitMAN_Screeshot_development_pastedatasets.py

What I believe to be the most valuable script is that which opens every .dat and .out combination, screenshots them, and stores them as a .png that you can easily show Rob. This way, you save lots of time having to tediously pull up everything in fitMAN to show him. The script navigates through fitMAN to input the phase and filter values as well. To have it work for you, input the .dat and .out files in the same order. Then, do a few test runs, editing the coordinates for the screenshot each time. Many other have also manually screenshotted each fit as well, so if this does not work, that is an option.

There are additional scripts for 9.4T data found under Igor Solovey's GitHub repository found here. Some of these scripts are adaptations of those found there.