

# XTOLM use case - compute brain volumes from an MRI experiment and export data to spreadsheet

In this example, we will use [XTOLM](#) use case - compute brain volumes from an MRI experiment and export data to spreadsheet to compute whole brain volume using the [Brain Extraction Tool](#) for all MPRAGE scans in an MRI project in XNAT, and save the resulting statistics in a spreadsheet.

## Prerequisites

- [FSL](#) and [dcm2nii](#) on your path
- We'll use data from surfmask\_smpl project on XNAT central so you'll need an [XNAT Central](#) account.

## Step by Step Instructions

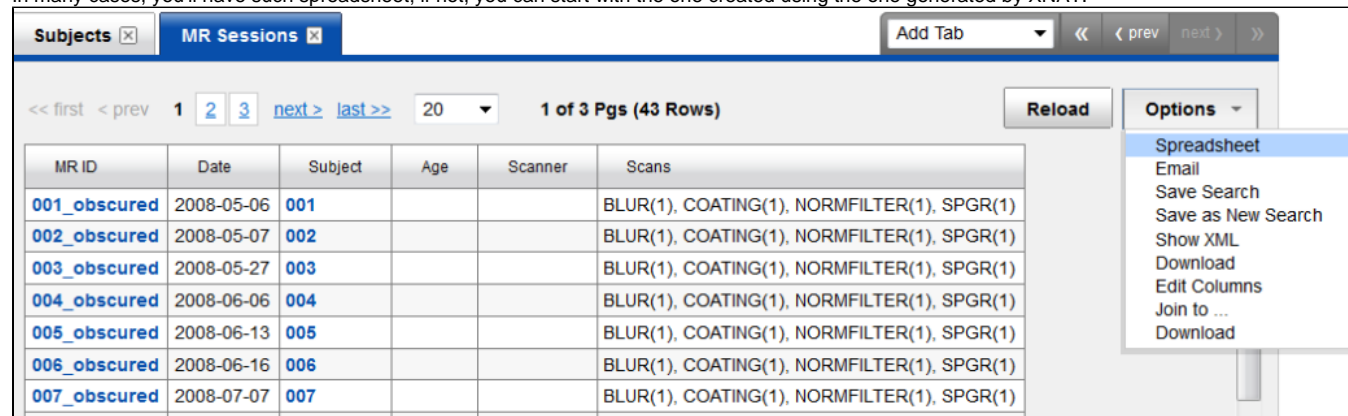
1. Write a bash script that computes human brain volume from a T1-weighted MR image in DICOM format (assuming FSL v4+ is on the path).

### brain\_volume.sh

```
#!/bin/bash
in=$1 #input DICOM directory
dcm2nii $in
cp $in/[0-9]*.nii head.nii
if [ ! -f "head.nii" ]; then exit -1; fi
bet head brain -m
vol=$(fslstats brain_mask -V); vol=${vol[1]}
#report the volume in mm3 to console.
echo $vol
```

2. Create a CSV or XLSX spreadsheet listing all scans that you want to process.

In many cases, you'll have such spreadsheet; if not, you can start with the one created using the one generated by XNAT:



MR ID	Date	Subject	Age	Scanner	Scans
<a href="#">001_obscured</a>	2008-05-06	001			BLUR(1), COATING(1), NORMFILTER(1), SPGR(1)
<a href="#">002_obscured</a>	2008-05-07	002			BLUR(1), COATING(1), NORMFILTER(1), SPGR(1)
<a href="#">003_obscured</a>	2008-05-27	003			BLUR(1), COATING(1), NORMFILTER(1), SPGR(1)
<a href="#">004_obscured</a>	2008-06-06	004			BLUR(1), COATING(1), NORMFILTER(1), SPGR(1)
<a href="#">005_obscured</a>	2008-06-13	005			BLUR(1), COATING(1), NORMFILTER(1), SPGR(1)
<a href="#">006_obscured</a>	2008-06-16	006			BLUR(1), COATING(1), NORMFILTER(1), SPGR(1)
<a href="#">007_obscured</a>	2008-07-07	007			BLUR(1), COATING(1), NORMFILTER(1), SPGR(1)

Now, modify the spreadsheet so that column names don't contain spaces, empty and irrelevant cells are removed, and set "Scans" to "SPGR", the scan type that we want to process for each session. If any fields have trailing zeroes, you'll need to modify the csv outside of Excel to add them.

```
MR_ID,Subject,Scans,
001_obscured,001,SPGR,
002_obscured,002,SPGR,
003_obscured,003,SPGR,
004_obscured,004,SPGR,
005_obscured,005,SPGR,
006_obscured,006,SPGR,
```

3. Adapt the brain\_volume.sh to process all sessions from your CSV file.

### brain\_volume.xt

```
#!/bin/bash
#this line will convert test.csv to a bash source test.params that assigns corresponding arrays.
load_data test.csv
#as a result, the following arrays are assigned:
#MR_ID=("001_obscured" "002_obscured" ...)
#Subject=("001" "002" "003" ...)
#Scans=("SPGR" "SPGR" "SPGR" ...)
#now, iterate over all MR sessions.
xt_for Subject MR_ID $1
    #load specified scan from XNAT.
    load_type $Scans
    #find the scan that was created by load_type.
    in=`ls -d study*`
    #compute brain volume, similarly to brain_volume.sh
    dcm2nii $in
    #select the unmodified NIFTI volume to process.
    cp $in/[0-9]*[0-9].nii head.nii
    if [ ! -f "head.nii" ]; then exit -1; fi
    bet head brain -m
    vol=(`fslstats brain_mask -V`); vol=${vol[1]}
    #now we use special command to save a variable with this offline session.
    save_vars vol
xt_done
#this will go through all sessions and generate a spreadsheet summarizing variables saved with "save_vars"
command.
summary mpr_vol MR_ID
```

#### 4. Test the batch processing.

We will run the developed script on a single session to see if it works. For that, uncomment break statement on line 26. Now we are ready to run the script:

### running brain\_volume.xt

```
xtolm -sr https://central.xnat.org -u your_xnat_central_user -pr surfmask_smpl -o brain_volume.xt
cat mpr_vol.csv
```

Now we'll have the BET calculations saved in mpr\_vol.csv file ready for analysis (note that Excel removed trailing zeroes from the subject again).

	A	B	C	
1	session	subject	vol	
2	001_obscured	1	1376148.75	
3	002_obscured	2	1393418	
4	003_obscured	3	1470579.75	
5	004_obscured	4	1452396.75	
6	005_obscured	5	1410483.875	
7	006_obscured	6	1709883.75	
8	007_obscured	7	1442823.25	
9	008_obscured	8	1616453.5	