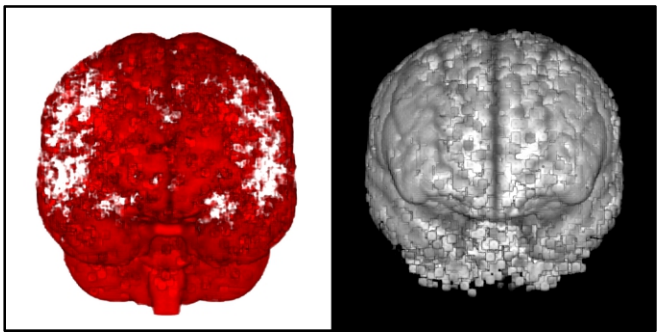
Group Level Imputation of Statistic Maps, Version 1.0



User Manual by Kenny Vaden, Ph.D.
(vaden @ musc . edu)

Notice for use in Academic Work: If our toolkit is used in an academic work (e.g., journal article or conference proceedings), we would like to be explicitly cited. Please cite both of these papers:

1. Vaden, K.I., Gebregziabher, M., Kuchinsky, S.E., Eckert, M.A. (2012). Multiple imputation of missing fMRI data in whole brain analysis. *NeuroImage, 60(3),* 1843-1855. (toolkit to implement multiple imputation for fMRI statistics)
2. van Buuren, S., Groothis-Oudshoorn, C.G.M. (2011): mice: Multivariate Imputation by Chained Equations in R. Journal of Statistical Software, 45(3), ISSN 1548-7660. (MICE - multiple imputation package for R).

Disclaimer: This toolkit is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

Features: Three command line scripts used to measure missingness, evaluate predictors, and perform single sample t-tests on group level fMRI datasets.

- Identifies voxels with missing data.
- Performs multiple imputation in voxels with missing data.
- Creates statistic maps with and without multiple imputation.
- Uses the multiple imputation method from Vaden et al. (2012).

Software requirements: This toolkit is freely available from: http://www.nitrc.org/projects/multimpute. Before using our toolkit, you will need to install Matlab (commercial software) and R statistics (freeware) on your machine (both run on MAC/Windows/Linux operating systems). One free Matlab toolbox is required: SPM5 or SPM8 (www.fil.ion.ucl.ac.uk/spm), and one freely distributed package is also required for R: MICE (van Buuren & Groothis-Oudshoorn, 2011; http://www.multiple-imputation.com). Our scripts were written and tested on a Linux machine, but to our knowledge the code does not present operating system based conflicts (e.g., incorrect file path conventions could result in error messages).

Summary: Group Level Imputation of Statistic Maps (version 1.0) is a toolkit that performs multiple imputation for group level, single sample t-tests. Whole brain group level statistic maps from fMRI rarely cover the entire brain as a result of missing data. Missingness between subjects in fMRI datasets can result from susceptibility artifacts, bounding box (acquisition parameters), and small differences in post-normalized morphology. The toolkit consists of several interactive command line scripts that guide the user to map the spatial distribution of missing data across contrast images, calculate spatial neighborhood averages that help impute values, perform conventional and multiple imputed t-statistics, save the results to brain maps, and create result tables. The toolkit contains the manual (this file), two Matlab scripts and one R-Statistics script, which depend on functions defined in the popular SPM toolbox and functions defined in the MICE package for [R].

Missing Data in Group fMRI Statistics

Group level fMRI analyses rarely extend across the entire brain because of missing data that can result from subtle differences in brain size or shape, data acquisition limits, and susceptibility artifacts (Vaden et al., 2012). Normally, voxels that do not contain a value in every subject's contrast image (individual level statistic parametric maps) are omitted from the group level tests. Excluding those voxels that are of theoretical interest from group level analyses increases the potential for Type II error at cortical boundaries or Type I error when spatial thresholds are used to establish significance. The practice of omitting voxel datasets that contain missing data can result in a significant proportion of untested voxels and obscures significant activation at the boundaries.

Figure 1: Apparent Missingness at the Cortical Boundary

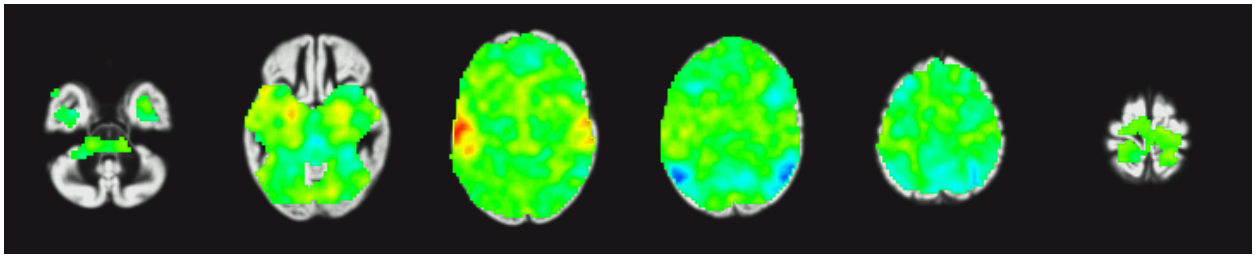


Fig 1. Missingness can be seen in spatial inconsistencies between the group-level statistic and group-defined anatomical image. When the statistic map is thresholded with p-value = 1, the overlay reveals many voxels along the cortical edges and susceptibility regions were probably omitted from group-level analysis. In Step 1, we map the proportion of missing data (one or more subjects) in voxels that were omitted from statistical tests (Fig 2).

Multiple Imputation

Multiple imputation is a principled "filling in" statistical technique for inferences based on data with missing values assumed to be at random (Rubin, 1987). In the first step, a regression model with predictors is fitted to the observed contrast values (e.g., age, gender, motion, neighbors: mean observed value within 18 mm of each voxel with missing data). The second step is to generate *m* datasets by replacing missing values according to the regression equation. A statistic (e.g., t-test) can then be computed for each version of the dataset with different replacement values, then results are pooled to approximate the statistic with no missing data. Multiple imputation reduces bias, Type I errors, and Type II errors in voxels with predictable missingness and fewer than 33% missing cases, compared to omitting voxels with missing data or alternative strategies that include available case or mean replacement analyses (Vaden et al., 2012).

Multiple imputation can result in expanded statistical map coverage, increase power, and enhance interpretation of fMRI results. Including missing data voxels in the group statistic maps instead of omitting them from analysis increases the spatial coverage of the map and can reveal clusters that extended into regions with some missing values. We note that the success of multiple imputed statistics depends on correct imputation model specification and data that contain predictable missingness (Missing at Random; MAR).

Multiple Imputation Can be Biased by Data that are Missing Not At Random

Correctly performing multiple imputation on fMRI data requires familiarity with the limitations of multiple imputation, especially the important distinction between data that are missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR).

A general rule for determining whether your design contains MAR or MNAR data is to consider all potential missingness mechanisms in the dataset. If there are factors that systematically predict missingness but were not measured or related to predictors that were included in the imputation model, then those factors are potential MNAR mechanisms. For example, in a group level analysis, males tend to have a larger intracranial volume than females on average and consequently fMRI image boundaries may reduce coverage of the brain, especially for men. If the imputation model did not include the gender of each subject, then values from female subjects could have a disproportionate influence on replacement values. Including the gender of each subject in the model allows that factor to influence the estimated replacement values, thereby avoiding a biased estimate. An alternative would be to include intracranial volumes as predictor variables, since that more directly relates to the missingness mechanism (i.e., larger brains extending beyond the bounding box edges, while smaller brains do not).

Multiple imputation produces unbiased statistic estimates for MAR data and can also be effective for MCAR data (Vaden et al., 2012), but may be strongly biased by MNAR data. Performing multiple imputation for group level statistics limits the risk of condition- or task-related MNAR, because those missing data are more likely to occur in the time series data and those affected voxels are omitted from subject-level contrast images for all conditions. Nevertheless, it is always important to assess your design and subjects to identify MAR factors that can predict missingness to reduce the risk of biased results. (For more details on the topic of missingness types, see Gebregziabher and DeSantis, 2010; Ibrahim and Molenberghs, 2009; Little and Rubin 2002; Schafer 1997; Vaden et al., 2012.)

Recommendations for Authors

We recommend that the following information should be reported to help reviewers and readers evaluate the validity of the multiple imputation method and assess its impact on your results:

1. Report the maximum proportion of missing data that was imputed (e.g., voxels missing up to 30% of missing subjects; specified in step 1).
2. Report the results of a sensitivity analysis performed on the missing data voxels; how many voxels that were imputed had a missingness indicator that was significantly predicted by variables in the imputation model? (step 2, evaluation of predictors)
3. Report the spatial extent of group results with and without multiple imputation (step 1 output).
4. Report results with and without multiple imputation, or indicate which results were significant only after using multiple imputation (steps 3 and 4).
5. Disclose potential mechanisms resulting in MNAR data (see description above).

Summary of the Current Method

Our toolkit uses functions from the SPM toolbox (Matlab) and MICE package (R) to perform multiple imputation of missing values for group-level whole brain analyses, using the same method as in Vaden et al. (2012). These scripts are streamlined versions of the code that we used to implement multiple imputation of single sample t-tests. They are not designed to impute statistics, compare missing data methods, or perform bootstrap simulations.

The toolkit consists of four scripts that each perform a stage of the multiple imputation process:

Step 1: Prepare contrast image data for multiple imputation in Matlab.

Step 2: Evaluate predictors and perform multiple imputation on contrast data in R.

Step 3: Export statistic results into brain map images (nifti format, *.nii) in Matlab.

Step 4: View results in SPM (Matlab).

Required Software and Packages

1. Matlab software: http://www.mathworks.com/products/matlab
2. SPM (freely distributed toolbox; v.5 or v.8 will work): http://www.fil.ion.ucl.ac.uk/spm
3. R statistics freeware: http://www.r-project.org
4. MICE (freely distributed toolbox for R): http://www.multiple-imputation.com. Note: the mice package can be downloaded and installed from CRAN within the R interface.

Required Input Data

1. Subject-Level Contrast Images
   - Same contrast files that are normally submitted to group level t-tests in SPM (these might also be generated in AFNI or FSL - not confirmed).
   - Contrast files (con*.img or con*.nii) are produced by SPM (Matlab toolbox). Each contrast is either a subject-level GLM estimated parameter β or within-subject comparison of β parameters (e.g., $\beta_1 - \beta_2$).
   - Note that paired samples t-tests can be performed using group level single sample t-tests on difference scores that are calculated by subtracting beta/contrast images within subjects. Difference maps may be submitted to single sample t-test to achieve that end.
   - Increased sample size (N) improves multiple imputation results (N > 15 recommended).

2. List of Contrast Image Files (see example below)
   - Tab-delimited textfile (no header row, no column names) listing subject ID numbers in the first column and filenames of each corresponding contrast image in the second column (including full directory paths to each contrast image file).
   - Contrast images in the list should only represent one condition or one comparison. Separate imputations can be performed to render t-statistics, using separate file lists.
   - **Example Contrast File List**
     Tab-delimited text file, no header, col 1= subject ID, col 2 = contrast file (entire filepath.)

```
1003   /images/vaden/wordlisten/individuals/1003/STATS_34/con_0001.img
1720   /images/vaden/wordlisten/individuals/1720/STATS_34/con_0001.img
...    ...
```

3. Predictor Variable List (see example, below)
   - Tab-delimited textfile with a header row (column names) that lists subject ID numbers in a column beneath “SID”. Note that the SID column is used to match predictors to fMRI data, so that column name is very important. Additional columns contain demographic and descriptive variables that may be predictive of missingness.
   - Predictor columns can include data from each subject, which could include intracranial volume, summary measures of motion, age, scanner operator, etc. (categorical variables should be coded numerically, e.g. 1=Kelly, 2=Noam, 3=Stef; our scripts prompt you to specify whether each variable is categorical or numeric).
   - The imputation model uses predictors to estimate missing value by initially performing a regression, so the number of predictors must be less than the degrees of freedom. Including more predictors than degrees of freedom values (DF) will result in computation errors, $DF = N_{observed} - 2$, given that $N_{observed} = N_{subjects} - N_{missing}$. Step 1 generates an average

neighborhood predictor, which is automatically included with the predictors that you specify - so choose your predictors wisely.

- A general rule is to include as many variables as possible that may predict missingness, separately or in combination. However, too many predictors can result in regression failure or result in over-specification errors where variance is underestimated. Too few predictors (or none) can result in MNAR-biased imputation by failing to use sufficient information to predict missing values. A useful step prior to multiple imputation is to assess which predictors are significantly correlated with the missingness indicator within voxels (option within Step 2) and remove predictors that do not contribute to the logistic regression.
- **Example Predictor File**

  Tab-delimited text file with a header row, col 1: Subject ID, col 2+: predictor variables. **Important:** subject ID column must have "SID" in the header row (no quotes). In the example analysis, the following predictors were included for each subject: Age, Gender, MotT (max-min translation-motion), MotR (max-min of rotation-motion).

SID	Age	Gender	MotT	MotR
1003	73	1	2.9	0.89
1720	28	2	1.0	0.54
…	…	…	…	…

After installing the required software and packages, and preparing the input files (contrast images, a list of the contrast image files, and a list of predictor variables), then follow the directions below to perform multiple imputation of a group level t-test.

Instructions – Step 1: Prepare contrast image data for multiple imputation in Matlab

Each of these steps gives an example of what to type at the command line in **courier** font:

1. Create a directory for the imputation scripts and uncompress the archive to save files there:
   > **mkdir /images/vaden/imputation/batchscripts/**
   > **mv multimpute*.zip /images/vaden/imputation/batchscripts/**
   > **cd /images/vaden/imputation/batchscripts/**
   > **unzip multimpute*.zip**
2. Create a working directory, move the contrast files list and predictor list there:
   > **mkdir /images/vaden/imputation/Analysis1/**
   > **mv Contrast_Filenames.txt /images/vaden/imputation/Analysis1/**
   > **mv Predictor_List.txt /images/vaden/imputation/Analysis1/**
3. Start a matlab session:
   > **matlab**
4. At the matlab command line, change the current directory to the location where you saved the Step1_MICE_inputs.m matlab script:
   > **cd /images/vaden/imputation/batchscripts/**
5. Execute Step1_MICE_inputs.m
   > **Step1_MICE_inputs**
6. Enter parameters and filenames as the script prompts you for them. The script will display explanations and examples for user input prompt. For example:

   USER-DEFINED OPTIONS: imputation parameters
   Specify max proportion of missing subjects for imputation: **0.3**
   Specify radius for mean neighborhood contrast value (mm): **18**

   USER-DEFINED OPTIONS: file names
   Working directory is where the input and output textfiles are located.
   Specify working directory: …
   > **/images/vaden/imputation/Analysis1**
   Specify input contrast list file: …
   > **Contrast_Filenames.txt**
   Specify input mask image file: …
   >
   (In this example, blank = no mask selected, but you could enter a filename from inside the working directory if you have a binary mask located there. A mask could be used to exclude voxels from potentially being multiple imputed, e.g. in eyeballs or other non-brain voxels.)

Output Files: Step 1

1. Step 1 creates an output textfile: **CON_imported.txt**, which contains contrast values for voxel-by-voxel multiple imputation (missing data voxels with proportion missing data less than the user-specified limit). The contents are organized into the following columns:
   - MIS: Voxel that contains missing data in some subjects (1) or not (0)
   - VID: Voxel ID Number
   - SID: Subject ID Number
   - CVal: Observed Contrast Value (NaN, if missing)
   - NMean: Mean contrast value within a voxel's neighborhood in that subject's map

Example CON_imported.txt:

MIS	VID	SID	CVal	NMean
1	55772	1030	-1.039987	0.000000
1	55773	1030	-0.528716	0.000000
1	55774	1030	NaN	0.000000
…	…	…	…	…

Note that "missing" voxels contain values for > 70% subjects in our example, which is why only one row in this selection contains an empty cell (NaN).

2. Step 1 also creates a statistic map (Fig 2), **`Proportion_missing_subjects.nii`**, which displays the proportion of subjects in each voxel that are missing data (contrast values). Multiple imputation will be performed where this proportion is less than the user specified cut off. A limit of 30% missingness was specified in the example above.

Figure 2: Proportion of Missing Subjects per Voxel

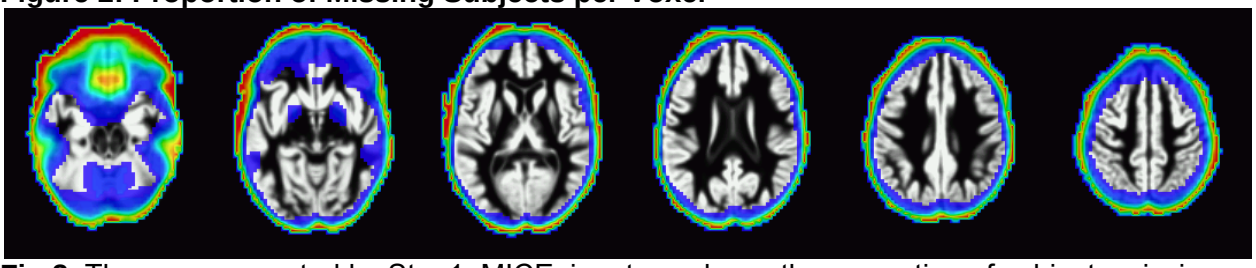


Fig 2. The map generated by Step1_MICE_inputs.m shows the proportion of subjects missing values in each voxel, superimposed on the anatomical image (color scale: blue = 1/36, yellow=18/36, red=35/36 missing). Typical analyses omit voxels that contain any nonzero number of missing subjects from the group-level statistic tests, so all of the colored voxels in this image would be excluded from typical statistical tests.

Instructions – Step 2: Evaluate predictors and perform multiple imputation in R

Step 2 separately performs two functions that are needed for the multiple imputation analysis. First, the script is used to evaluate predictor variables by performing a logistic regression on the missingness indicator within each voxel with missing data. Second, the script is used to perform multiple imputation and save each missing-replaced version of the voxel datasets. Splitting the job into two pieces seemed like a good idea to avoid long waiting times and overheating computers. More importantly, computers with multiple processors can run both options 1 and 2 at the same time, to complete the job more quickly. Importantly, both options should be executed for multiple imputed datasets.

Each of these steps gives an example of what I type at the command line in `courier` font:

1. Start an R session (Windows: click the R icon; Linux: type 'R' at the command line)
   > **R**

2. Within R, change directory to the directory where Step2_MICE_Rscript.R is located:
   > **setwd('/images/vaden/imputation/batchscripts');**

3. Execute Step2_MICE_Rscript.R with the following line:
   > **source('Step2_MICE_Rscript.R');**

4. Enter information about input files and data directory, when prompted. The program will look there for the output files that were created in Step 1.

   Enter working directory name:
   /images/vaden/imputation/Analysis1

5. **Both of these choices must be performed; start by selecting option # 1:**
   5A. When '1' is chosen at the next prompt, a logistic regression is performed to determine whether the missingness indicator was related to the predictors within the voxels that contain missing data. If there are too many known predictors for the imputation model (i.e. more variables than degrees of freedom) and some need to be removed, this could be used identify predictors that are unrelated to missingness. More importantly, if imputed voxels contain predictably missing values, then we can reject that missing data were unpredictable (MCAR). When every known significant/perfect predictor is included in the imputation model, then multiple imputation can be performed under the assumption of data that are predictably missing (MAR) and not MNAR.

   1) evaluate predictors (logistic regression on the missingness indicator)
   2) generate m multiple imputed datasets
   Enter 1 or 2 to indicate your choice: **1** ← *picking '1'*
   Enter predictors filename:
   Predictor_List.txt
   FOR EACH PREDICTOR VARIABLE, SELECT (1) NUMERIC or (2) FACTOR.
   predictor variable, SID: **2**
   predictor variable, Age: **1** ← 1 ("numeric" :: continuous number scale)
   predictor variable, Gender: **2** ← 2 ("factor" :: binary or multi-level categorical)
   predictor variable, MotT: **1**
   predictor variable, MotR: **1**

   5B. When you select '2', the script performs multiple imputations and saves the voxel datasets with replaced missing values in a tab-delimited text file.

   1) evaluate predictors (logistic regression on the missingness indicator)
   2) generate m multiple imputed datasets
   Enter 1 or 2 to indicate your choice: **2**

INPUT: textfile that contains additional predictor variables. …
Enter predictors filename: …
Predictor_List.txt
Input number of imputations to perform (def M=5): **5**
Performing multiple-imputed statistics on voxels with missing subjects ...

A table from van Buuren and Groothuis-Oudshoorn (2011) is shown, which lists variable types (and imputation methods) that can be specified for the dependent and predictor variables in the imputation model. The user is prompted for the METHOD name (not the variable type) for each variable. For continuous variables, we recommend norm to implement Bayesian linear regression; binary variables, specify logreg; and for multi-level categorical, specify polyreg. These should be specified even if the predictor variables do not contain missing values, because these affect the distribution types used by the imputation model for each variable. Correctly specified models perform better.

Enter METHOD for missing contrast values: **norm**
... for predictor variable, neighborhood average: **norm**
... for predictor variable, Age: **norm**
... for predictor variable, Gender: **logreg**
... for predictor variable, MotT: **norm**
... for predictor variable, MotR: **norm**

- **Note:** If you were not prompted to specify method/distribution for any predictor variables, then those predictors may not have been included in the model! The predictor file must contain a column with **SID** heading (*see example predictor file above*), used to match data to predictors on the basis of that column name.

Output Files: Step 2 (Each step 2 option may take ~ 1 hour to complete.)

Option 1: **Sensitivity_analysis_results.txt** is a text file that contains summary statistics about the extent to which specified predictor variables related to which subjects were missing in the voxels with missing data. The output file is not formatted as nicely as the table below, but formatting can be done simply in Excel or spreadsheet software.

	Std_Est	LCI95pct	UCI95pct	Zscore	Zpval	Csep	Isep
Age	0.51	0.46	0.56	20.18	< 0.0001	27.59%	0.01%
Gender2	-2.91	-2.99	-2.82	-68.89	< 0.0001	4.20%	24.26%
MotT	3.03	2.91	3.14	52.49	< 0.0001	27.65%	0.02%
MotR	-0.28	-0.31	-0.24	-16.24	< 0.0001	27.29%	0.00%
Total						27.82%	24.29%

The results of the sensitivity analysis indicate that the imputed voxels did not contain data missing completely at random (MCAR). Since each tested predictor was significant and was included in the imputation model (step 2/option 3), multiple imputation is performed under the MAR assumption (predictable missingness) and not MNAR (systematic, unpredicted missingness).

The table of results gives the standard estimate pooled across missing data voxels for each predictor in the logistic regression, along with confidence intervals and Z-scores. The last two columns gives the percentage of voxels that are perfectly predicted by a combination of factors (CSep) or the individual factor alone (ISep). Perfect prediction or separation means that missing subjects were predictable without error (for example, a voxel with missing data only from older

males). Voxels that are perfectly predicted result in warning messages and are not included in the pooled statistics or Z-scores, as their estimates are large (approach infinity before regression is halted). The "Total" row gives the proportion of voxels that were perfectly predicted by combinations of predictors (27.82%) or single predictors (24.29%). Unlike ISep, the summed rows of CSep are greater than the total, because various combinations formed perfect predictions in 27.82% of the missing data voxels. Since ISep + CSep = 52.11%, more than half of the voxels were perfectly predicted in the results of logistic regression. The other statistics (i.e., standard estimates, Z-scores, etc.) include the remaining 47.99% voxels.

Option 2: `multiple_imputed_datasets_m5.txt` is an output text file that contains a column of subject ID numbers, voxel ID numbers, original contrast values including missing entries, and *m* columns of contrast values with missing data replaced using multiple imputation.

```
346   2082  23.247578  23.247578  23.247578  23.247578  23.247578  23.247578
653   2082  52.333275  52.333275  52.333275  52.333275  52.333275  52.333275
6023  2082  22.783566  22.783566  22.783566  22.783566  22.783566  22.783566
8340  2082  8.475183   8.475183   8.475183   8.475183   8.475183   8.475183
6947  2082  NA         -36.65195  -57.08805  -20.81025  -1.971663  -25.86682
3117  2082  NA         82.873536  65.606403  44.244048  97.912897  41.032558
…     …     …          …          …          …          …          …
```

Remember: options 1 and 2 should be performed before proceeding to Step 3.

Instructions – Step 3: Export statistic results into brain maps in Matlab

Each of these steps gives an example of what I type at the command line in **courier** font:

1. Start a matlab session:
   > **matlab**
2. Change current directory to the location where you saved Step3_MICE_export.m script:
   > **cd /images/vaden/imputation/batchscripts**
3. Execute Step3_MICE_export.m
   > **Step3_MICE_export**
4. Enter the working directory where the data are stored, in response to the prompt.
   Specify working directory: …
   > **/images/vaden/imputation/Analysis1**

Output Files: Step 3

Step 3 uses SPM functions to produce *m* temporary copies of the contrast images, each with a difference set of imputed replacement values for missing voxels, then performs t-tests on each, which are averaged to form a resultant t-statistic map **MIresult/spmT_0001.img**. The script also produces a t-statistic map without multiple imputation **NOMIresult/spmT_0001.img**, and a map of the difference: **MIresult/spmT_multimputed_voxels.nii**. Each statistic map can be overlaid on anatomical images (Fig 3ABC, below), and opened in SPM to assess significance and cluster sizes. Degrees of freedom = N-1 for all results, whether or not multiple imputation was used.

Fig 3A: Step 3 output – t-scores from voxels only with complete data (NOMIresult/spmT_0001.img)

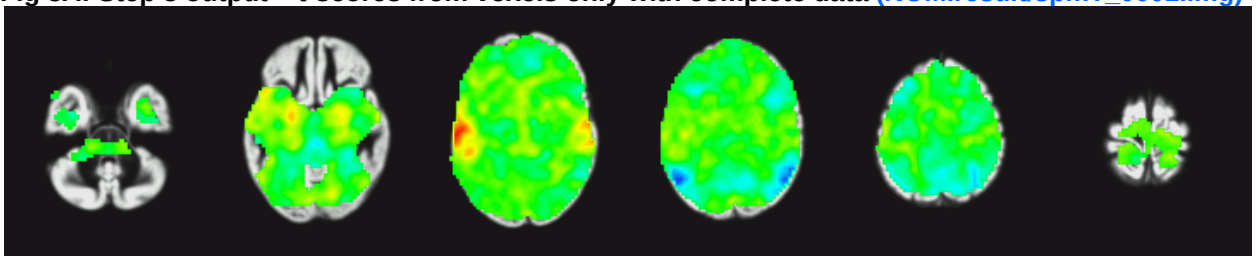


Fig 3B: Step 3 output – t-map of the missing data (MIresult/spmT_multimputed_voxels.nii)

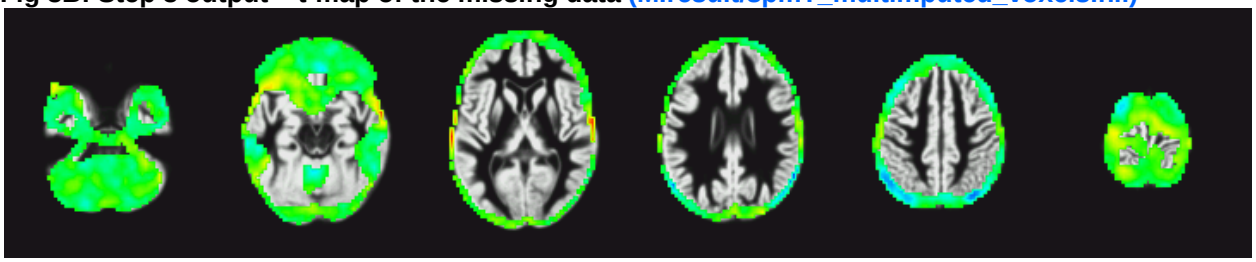


Fig 3C: Step 3 output – t-scores based on all voxels combined (MIresult/spmT_0001.img)

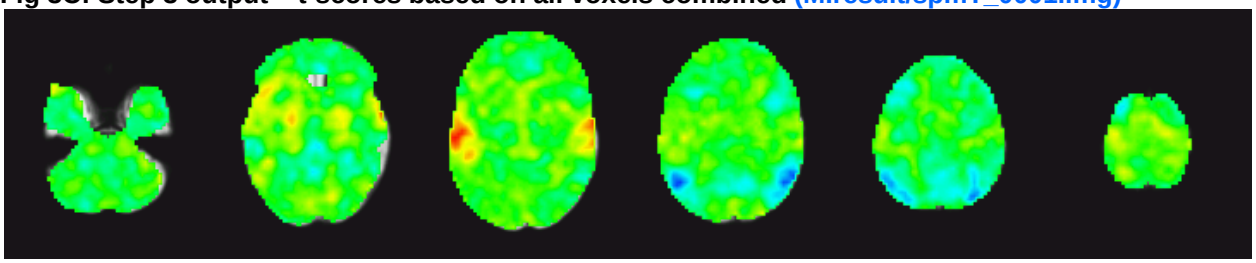


Fig 3. The original t-statistic map based only on voxels with no missing data is shown in A. Many voxels along the cortical boundary were missing fewer than 30% subjects so there was sufficient data to perform multiple imputation and calculate pooled t-statistics in those voxels. There are bright red and dark blue spots in that cortical ribbon (B), suggesting that significant

cluster sizes are underestimated in A, and will be correctly estimated in C. Spatial coverage of by the group statistic map in this example dataset was expanded as a result of multiple imputation, to include an additional 14,927 voxels (B, C).

Instructions – Step 4: Result Tables from Statistic Maps in Matlab

Two non-temporary folders are created during Step 3:

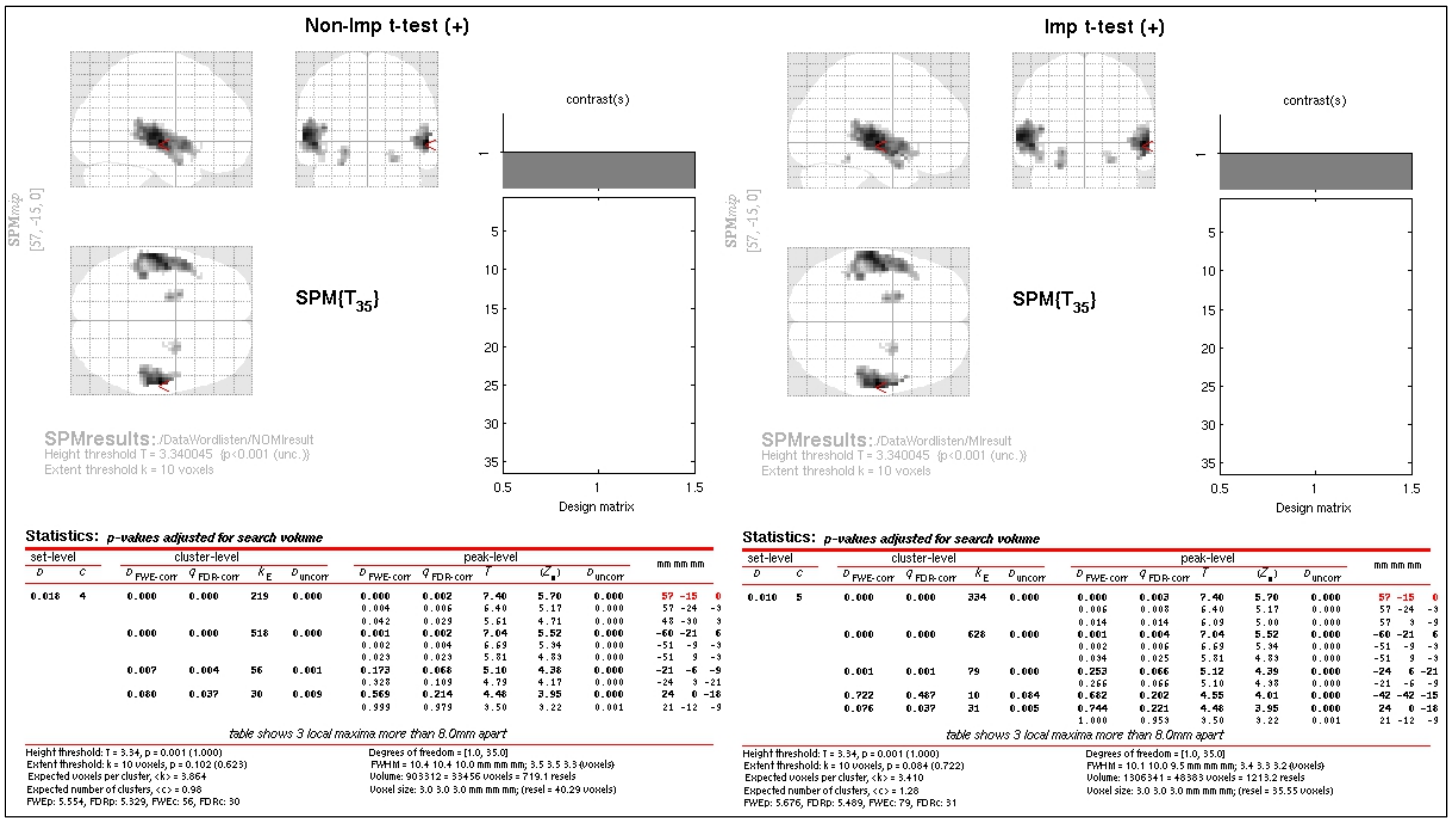
- **MIresult** - Multiple imputation results.
- **NOMIresult** - Results calculated without multiple imputation.

Each folder contents are slightly different, but the most important files are:

- **SPM.mat** - File that is loaded in Matlab to view results.
- **spmT_0001.img** - Statistic parametric map, positive t-test results.
- **spmT_0002.img** - Statistic parametric map, negative t-test results.

The **MIresult** folder also contains a series of **miCON_<SID>.nii** contrast image files (one for each subject) that are placeholders for SPM8. Those miCON images do not contain any imputed values (the imputed contrast values are replaced with empty voxels), so by default contrast value plots based on significant peaks or clusters do not include multiple imputation results.

In order to generate statistic tables, open SPM5 or SPM8 within Matlab (see SPM Manual) and load the SPM.mat file that is located in the **MIresult** folder that was automatically created, within the working directory. For comparison, the SPM.mat file within the **NOMIresult** folder was calculated without multiple imputation and pooled t-tests. In practice, opening these should be no different from viewing the results of a normal SPM analysis.



Works Cited

Gebregziabher, M., DeSantis, S. (2010). A latent class based multiple imputation approach for missing categorical data. *J Stat Plan Inference*, *140*, 3252-3262.

Ibrahim, J.G., Molenberghs, G. (2009). Missing data methods in longitudinal studies: a review. Test, 18, 1-43.

Little, R.J.A., Rubin, D.B. (2002). Statistical analysis with missing data (2nd ed.) J. Wiley & Sons, New York, NY.

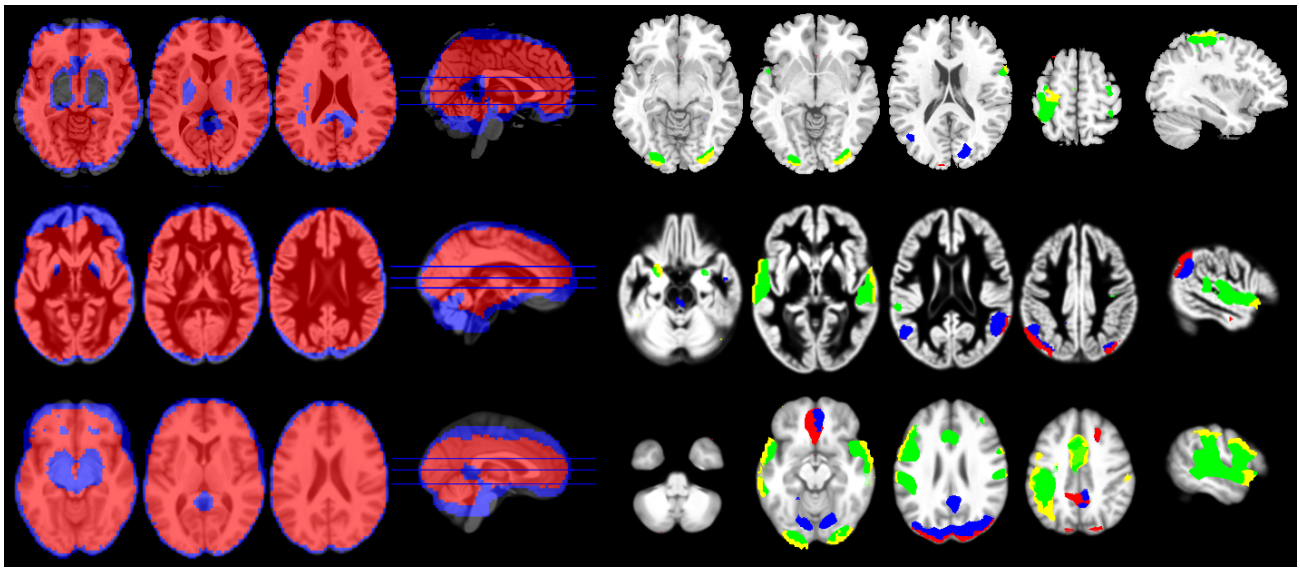
Rubin, D.B. (1973). Inference and missing data. *Biometrika, 63(3),* 581-592.

Schafer, J.L. (1999). Multiple imputation: a primer. Stat. Methods Med. Res., 8, 3–15.

Vaden, K.I., Gebregziabher, M., Kuchinsky, S.E., Eckert, M.A. (2012). Multiple imputation of missing fMRI data in whole brain analysis. *NeuroImage, 60(3),* 1843-1855. *Retrieved from:* http://www.sciencedirect.com/science/article/pii/S1053811912001504

van Buuren and Groothis-Oudshoorn (2011): mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software, 45(3)*, ISSN 1548-7660. *Retrieved from http:/www.stefvanbuuren.nl.*

Appendix I. Example Multiple Imputation Results from Three fMRI Datasets



Left: red voxels contained complete data, blue voxels show where multiple imputation allowed statistics to be performed with ≤ 30% missing subjects. **Right:** significant results (p=0.001 unc.) with and without multiple imputation; effects in voxels with complete data: green = positive and blue = negative; effects in voxels with missing data: yellow = positive and red = negative.