☞ Welcome to the CerebroMatic Toolbox Manual (V1.5) ☜

☞ Installation and installation requirements ☜

You ***will*** need the following pieces of software in order to work with this toolbox:

- the CerebroMatic toolbox itself (free, see https://www.medizin.uni-tuebingen.de/en-de/das-klinikum/einrichtungen/kliniken/kinderklinik/kinderheilkunde-iii/forschung-iii/software)
- the ARESLab toolbox from Gints Jekabsons, Institute of Applied Computer Systems, Riga Technical University (free, see www.cs.rtu.lv/jekabsons/)
- SPM12 from the Wellcome Department of Imaging Neuroscience, University College London (free, see www.fil.ion.ucl.ac.uk/spm/software/spm12/)
- Matlab ≥ Version 7.3, from The Mathworks Inc. (not free, see www.mathworks.com)

You ***may*** need the following pieces of software in order to work with this toolbox:

- the regression parameters (free, see https://www.medizin.uni-tuebingen.de/en-de/das-klinikum/einrichtungen/kliniken/kinderklinik/kinderheilkunde-iii/forschung-iii/software)
- cat_sanlm (free, see http://dbm.neuro.uni-jena.de/cat12)

To install, unpack all of the toolbox files into spm12's toolbox directory (e.g., to whatever/spm12/toolbox/com). After a restart, the toolbox functionality should be available via the usual spm12 interfaces, via the toolbox button as well as via the batch manager. The ARESlab toolbox and the (very large!) regression parameter files each can go into a directory of your choice (but ARESlab [and potentially cat_sanlm] must be in the Matlab path). If the above installation requirements are not met, the toolbox will provide you with a (hopefully informative) error message.

☞ Credits, Disclaimer & Licensing ☜

Thanks to Gints Jekabsons, the SPM team, and Christian Gaser for making available their software to the scientific community. The toolbox contains the 3D-median filter developed by Damien Garcia (http://www.biomecardio.com), included with kind permission by the author. Also thanks to Jan Gläscher, whose nanmean I have now included.

The general approach and the toolbox itself have been described in

> Wilke M, Altaye M, Holland SK; CMIND Authorship Consortium, 2017. *CerebroMatic: A Versatile Toolbox for Spline-Based MRI Template Creation.* Front Comput Neurosci (11) 5, freely available at https://www.frontiersin.org/articles/10.3389/fncom.2017.00005/full

The DARTEL approach and dataset have been described in

> Wilke M, 2018. *A spline-based regression parameter set for creating customized DARTEL MRI brain templates from infancy to old age.* Data in Brief (16), 959–966, freely available at https://doi.org/10.1016/j.dib.2017.12.001

☞ Introduction ☜

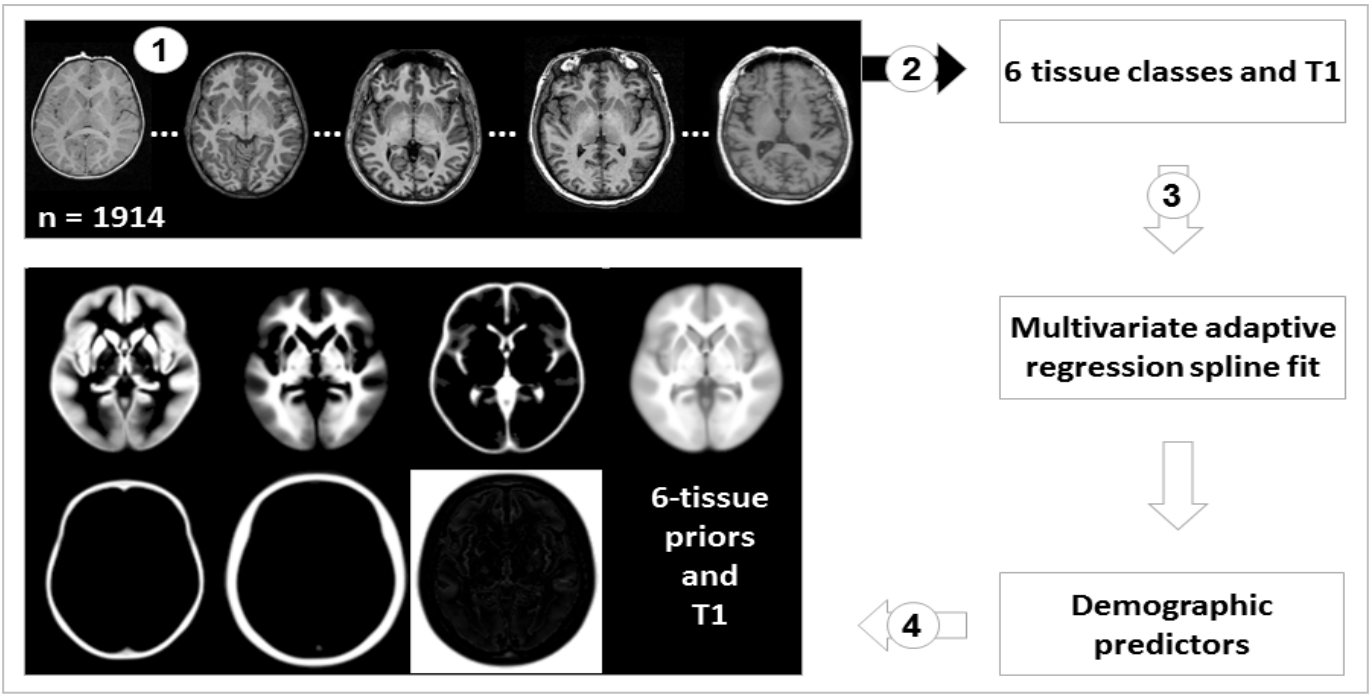
This is the manual for the CerebroMatic toolbox, an integrated software package allowing for the generation of MRI templates for tissue segmentation and spatial normalization, in the form of (currently) 6-class tissue priors and (optionally) a T1-weighted whole brain image. It is also ready to generate priors for use within the DARTEL framework. It is made available free of charge to the scientific community to generate custom templates for specific, and potentially unusual, populations.

It was developed and is intended for use within spm12 (or later) by Marko Wilke (Department of Pediatric Neurology and Developmental Medicine, Children's Hospital and Experimental Pediatric Neuroimaging group, Children's Hospital & Dept. of Neuroradiology, University of Tübingen, Germany), Mekibib Altaye (Pediatric Neuroimaging Research Consortium, Cincinnati Children's Research Foundation and Department of Pediatrics, Division of Biostatistics and Epidemiology, University of Cincinnati College of Medicine, Cincinnati, OH, USA) and Scott Holland (Pediatric Neuroimaging Research Consortium, Cincinnati Children's Research Foundation and Department of Radiology, University of Cincinnati College of Medicine, Cincinnati, OH, USA), in cooperation with the CMIND authorship consortium. For version information, see the respective files.

This collection of files is not officially supported since we do not have the resources to provide this service on an acceptable and ongoing basis. Should there be the need for a public announcement, it will be posted to the SPM-mailing list (currently at https://www.jiscmail.ac.uk/cgi-bin/webadmin?A0=spm). Also, possible new versions or updates may be sent out to registered users. We do, however, as always appreciate any comments or bug reports and are open to suggestions. Please contact us at Marko.Wilke@med.uni-tuebingen.de

Cheers,
Marko

☞ General Approach ☜



The idea behind the toolbox, as illustrated above, is simple: step 1, compile a large MRI dataset which is characterized by important demographic and technical variables (in our published study with n = 1914 and age, gender, fieldstrength at which the data was acquired, and data quality). Step 2, partition the MRI data into tissue classes (typically gray matter, white matter, cerebrospinal fluid, and three background classes). Step 3, analyze the resulting dataset with regard to the influence of the demographic and technical variables, and store these results. To this effect, we employed a multivariate adaptive regression splines approach. Step 4, based on these stored parameters, we can now generate synthetic tissue priors using the demographic information of a (new) sample of interest.

As you can see, there are two main aspects to the core endeavor: first, generate regression parameters from a (reference) population (step 3), and second, use previously-estimated parameters to generate tissue priors for a (new) population under study (step 4). Hence, there are two main functions:

- `mw_com_est` estimate regression parameters => see page 6
  - ⇨ to work with this function, you need to have partitioned MRI data from a reference population (*not* provided)
- `mw_com_gen` generate new templates => see page 12
  - ⇨ to work with this function, you need to have already-estimated regression parameters from a reference population (either generated by using `mw_com_est` or as provided at our website)

☞ Using the toolbox via the Toolbox dropdown menu ☜

If you want to use either of the toolbox's functions (described below in more detail) in an interactive fashion, the easiest way is to click on the toolbox dropdown menu in SPM's menu window and select `com` (this menu item may only appear after you restarted SPM following installation).

If you do this,

- an interactive menu *will* appear, asking: '*Do what…?*
  - ⇨ you will now have to select one of three options: '*Estimate regression parameters*', *'Generate templates'*, or *'Prepare for batch mode'.* You can also choose to *'Abort'* if you have taken a wrong turn
  - ⇨ to remind you of the meaning of the different options, some explanatory text will be shown in the command window: 'the toolbox allows for three options: -estimate regression parameters based on your own reference population; - use regression parameters to generate templates for your population; - prepare to generate templates for your population within the SPM batch mode;'
  - ⇨ depending on your selection, the corresponding function will start (with option 1 calling `mw_com_est`, option 2 calling `mw_com_gen`, and option 3 calling `mw_com_predictors`)
  - ⇨ if you selected *'Abort'*, the toolbox will exit

Note again that **most users** will probably want to use the *use regression parameters to generate templates for your population* option only; to do this, you should download these parameters from our website (see page 1) and skip ahead to the "**Using mw_com_gen to generate custom priors"** section, further below on page 12.

☞ Using `mw_com_est` to estimate regression parameters ☜

Use `mw_com_est` to generate your own regression parameters. This step may be optional: if you "only" want to generate custom priors based on already provided regression parameters (the more likely scenario for most users), see below for instructions on how to use `mw_com_gen` on page 12.

This function requires you to provide a set of partitioned (segmented) MRI datasets, which ideally number in the hundreds or above. One set of gray matter images (one per subject) is expected, the others are selected automatically (if the naming convention is met; see assumptions, below). For each subject, you should be able to provide demographic predictors (such as age and gender) and ideally also some technical information (such as field strength and data quality).

Assumptions 1: the toolbox will try to determine the demographic information from the filename, provided the following naming convention is followed:

- if the filename contains 4 numbers prior to first '_', this is assumed to designate the subject's age [in months]
- if the filename contains '_F_' or '_M_', this is assumed to designate the subject's gender [female/male]; please note that in my nomenclature, male sex = 1 and female sex = 0
- if the filename contains '_30_' or '_15_', this is assumed to designate the field strength the data was acquired on [3.0 or 1.5 Tesla]

For example, a file called 'wp10060_M_30_whatever.nii' will be interpreted to be from a 60 months old boy acquired at 3 Tesla. If you have data following another naming convention, the function will interactively prompt you for the inputs (see below) instead.

Assumptions 2: based on the name of the provided first tissue class, it will try to determine the corresponding next tissue classes (white matter, dura etc.). To this effect, you can supply segmentation results from SPM12 (prefix 'wc1') or the cat12-toolbox (prefix 'wp1' or 'wrp1', or affine-only images with an '_affine' postfix), which the toolbox will then analyze and use to find the other tissue classes, as well as accompanying T1 images (prefix 'wm' or 'wrm'). If you have tissue classes following another naming convention, this behavior can be changed in the code of `mw_com_est` (at about line 485).

Assumptions 3: the files supplied should not have been modulated with the Jacobian determinant. Also, they need to be in the same orientation, they should be normalized to the same voxel size (ideally

1.5 mm) and dimensions (121 × 145 × 121). If you have data not conforming to this convention, these latter settings may have to be adapted in `mw_com_gen` (see below).

Assumptions 4: if you want to create your own DARTEL or SHOOT template, you need to provide either 12 or 14 tissue classes, 6 (or 7) each for GM and WM. These must be provided in the correct order, i.e., GM(1), WM(1), GM(2), WM(2), GM(3), WM(3), and so on. When then using these results in the template creation step (see below), this order will automatically be recognized and a set of DARTEL or SHOOT templates will be generated.

******* Interactive use *******

Interactive use means that you call the function (by typing its name [`mw_com_est`] in the Matlab command window) or via the toolbox dropdown integration (see above) and see what happens. This is the recommended approach for pretty much all users. It is a good idea to create a dedicated output directory, cd into it and call the function from there, as by default, the current work directory will be the output directory (but see below for more options).

The following prompts *will* or *may* appear, either as new file selector windows or in SPM's interactive window. Note that the toolbox will also inform you about its progress via prompts in the Matlab command window.

- a file selector window *will* ask you to *'Please select GM images to analyze'*
  - ⇨ you should now select the gray matter images you want to analyze
  - ⇨ note that at this stage, the file name conventions mentioned above are already prespecified in the file selector, such that only images conforming to '^w*r*[cp]1.*.' are shown (see assumptions, above). To see/select other images, you could disable this filter (but realize that potential other tissue classes may not be found automatically anymore)
  - ⇨ when selecting no images, the toolbox will return
  - ⇨ when selecting images, the toolbox will acknowledge receipt, detailing the number of images and the path from which they were selected
  - ⇨ the filenames of the selected images are then analyzed w.r.t. the expected naming conventions; if this pattern (see assumptions, above) is found, the toolbox will attempt the automated determination of demographic variables. If this fails, manually entering these variables is required (see next).

- an interactive prompt *may* appear: '*Please enter n × nimgs predictors (e.g., age, gender...)*'
    - ⇨ you should now enter the demographic variables of your subjects in the form of a n × X matrix (number of predictors by number of subjects); while age, gender, and field strength are recommended, there is no limit to the kind, or number of predictors you can enter here (with the exception that X should be >>> n)
    - ⇨ if the dimensions do not match (e.g., X = 200 images but only n × 199 predictors are entered), the toolbox will not accept the input and ask you to respecify
    - ⇨ note that this will only appear if the automated determination of these demographic variables from the filenames did not succeed

- successive interactive prompts *may* appear: '*Please enter identifier for predictor [1:n]* '
    - ⇨ you should now enter an identifier for each demographic variable you provided (e.g., 'age', 'gender', etc.)
    - ⇨ a unique identifier must be provided for each variable; this name will be stored and will be used later-on to identify this variable
    - ⇨ note that this will only appear if the automated determination of these demographic variables from the filenames did not succeed

- an interactive menu *will* appear: '*Supply data quality weights?*
    - ⇨ you will now have to select one of three options: '*Yes (enter now)*', or' *Yes (determine from cat12 reports)*', or '*No*'
    - ⇨ if you selected '*Yes (enter now)*', you will have to provide one data quality measure for each input image; please note that for AresLab, smaller values indicate BETTER quality, they cannot be all zero, and negative values are disallowed
    - ⇨ if you selected '*Yes (*determine from cat12 reports*)*', the algorithm will try to read in the cat12-generated report mat-file and will use that measure later-on; note that this will only work if the files are in a 'report' subfolder of the folder the corresponding image is in, or in a folder on the same level (e.g., if your images are in bla\bla\images, then the report files can be either in bla\bla\report or in bla\bla\images\report)
    - ⇨ if you selected '*No*', no measure of image quality will be used

- successive non-interactive prompts re: your inputs *will* appear in the Matlab window
    - ⇨ these provide you with information regarding other tissue classes (*'... found X further tissue classes with a __ prefix, please wait...'* or *'... found NO further tissue classes, please wait...*'), T1 images (*' ... found accompanying T1 images with a __ prefix, please wait...*' or '*... found NO accompanying T1 images, please wait...*') and tissue quality indicators (*'... reading image quality indicators, please wait...*' or '*... reading image quality FAILED, proceeding without... '*). In case the messages do not reflect what you wanted (e.g., you did have accompanying T1 images but they were not found), see the section on assumptions, above. If all goes well, the toolbox will then proceed to calculations, which is indicated by further status reports in the Matlab window.

- a final interactive prompt *may* appear: *'Shut down computer when done?*'
    - ⇨ if you select the *'Yes'* button, the function will try to shut down the computer when processing is complete
    - ⇨ note that this prompt will only appear on a PC workstation

Please note that once processing has started, this may take a **very** long time (up to days, in the case of a large population and several tissue classes + T1s) and may require a lot of memory and CPU (and ultimately, disk space). The function will try to open a Matlabpool (or will use an existing one) as this speeds up processing significantly (to avoid this, see processing options, below). The progress reports in the Matlab window will be updated from time to time, but there is no indicator on how long it will take as there is no way to tell beforehand (as each tissue class is processed successively and independently, and processing one class may take longer than processing another). So be patient. Once it is done, it will let you know that it is done and will also inform you where the resulting data files are store. These will then need to be passed to `mw_com_gen` (see below).

******* Scripted use *******

Scripted use means that you call the function from the command line (or another script) with one or more input arguments, allowing you to finetune the toolbox's processing behavior or to automate data processing. This is only recommended for advanced users. Input arguments are as follows:

```
mw_com_est(imgs, predicts, predicts_nm, weights, mask, trainParams, twostep, out, modelOld, shutdown);
```

For details regarding each input option, please check the file where each parameter is (I hope) described in enough detail. You can get an overview by typing `help mw_com_est` at the Matlab command prompt.

******* Batch mode *******

As this function will only have to be used rather infrequently and as the interactive mode is so self-explaining ;) there is no SPM batch mode interface for this function.

******* Options *******

There are a number of further processing options that can be set/changed in the file itself. Note that the function was used by us to generate regression parameters from a very large reference population, so you may want to check if these options are the best fit for your dataset. They can be found (and changed) in `mw_com_est` as of about line 65 and include options regarding the spline fitting options, the twostep approach, mask creation options, whether or not to include accompanying T1 images, naming conventions, and how to include the weights.

There are also some debugging and display options that you can change as of about line 74, including options on how to deal with already existing older results, on whether a (Windows) computer should be shut down after processing, whether you want to use a Matlabpool, whether you want to show all intermediate results and whether you want to save results to additional graphics files. Also, you can define shortcuts here for debugging purposes.

Finally, if you want to change further options for multivariate adaptive regression spline fitting, you could have a look at about lines 325. All options are (I think) exhaustively documented in the code.

******* Outputs *******

The function will save the results in two ways: first, general information about the analysis will be stored in a `mw_com_info.mat` file which contains information about the number of subjects, the demographic information, processing options etc. (on how to access this information, see below). This file is usually rather small. It is this file that needs to be selected by / passed to `mw_com_gen` (see below) for actually generating synthetic tissue priors.

Further, there will be a `mw_com_class_??.mat` data file for each tissue class you analyzed (with ?? usually being 1 = gray matter, 2 = white matter, 3 = cerebrospinal fluid, 4:6 = different background classes, 7 = T1 images [but note that the naming convention is based solely on the order of the selected files]). These files may be rather large. Note that there may or may not be 7 classes as this depends on the initial analyses (i.e., if there was no accompanying T1, there will only be 6 classes, if you provided more background classes, there may be more etc.). The function will also inform you about the end of processing in the command window, reminding you where results were stored.

☞ Using mw_com_gen to generate custom priors ☜

Use mw_com_gen **to generate your own tissue priors. If you "only" (or initially) want to generate custom regression parameters from your own sample (the less likely scenario for most users), see above for instructions on how to use** mw_com_est **on page 6.**

This function requires you to provide a set of regression parameters as generated by mw_com_est, either generated by yourself or downloaded from (e.g.) our website. These parameters consist of a "general information file" (mw_com_info.mat) as well as of one large datafile for each tissue class analyzed. You will only need to select / pass the general information file, the other files will be found and used automatically (if they are in the same directory).

The idea is to generate statistically defined tissue priors that match your population 1:1. To this effect, the toolbox employs a "matched pairs" approach: for each of your subjects, it will internally create a matching set of tissue classes, and these will be averaged in the end. Hence, for each of your subject, you should be able to provide the same demographic and/or technical predictors (such as age, gender, and field strength) as was used when generating the regression parameters. An exception here is data quality: if this was used during the estimation process, it will automatically be set to the best-available value during the prior generation step (so you don't have to provide it here).

If you do not know what information you need to have available, you can either read it up in the accompanying text file or type mw_com_gen(spm_select, 0) at the Matlab prompt. When the file selector window comes up, select the mw_com_info.mat file you are interested in using. The algorithm will then print what information was used to generate the regression parameters, including (for example for age) the ranges available. This is important as you cannot generate tissue priors for data outside of the range of the originally analyzed population (e.g., if only subjects up to 900 months were included, you cannot generate tissue priors matching subjects of 901 months of age), and if only 1.5 & 3 Tesla imaging data was assessed, you cannot match 7 Tesla data. Also, it will inform you if a demographic predictor is binary (e.g., male/female) or not (e.g., age).

******* Interactive use *******

Interactive use means that you call the function (by typing its name [`mw_com_gen`] in the Matlab command window) or via the toolbox dropdown integration (see above) and see what happens. This is the recommended approach for experienced users. Again, it is a good idea to create a dedicated output directory, cd into it and call the function from there, as by default, the current work directory will be the output directory (but see below for more options).

The following prompts *will* or *may* appear, either as new file selector windows or in SPM's interactive window. Note that the toolbox will also inform you about its progress via prompts in the Matlab command window.

- a file selector window *will* ask you to *' Please select CerebroMatic (or predictor) file to use* '
  - ⇨ you should now select the `mw_com_info.mat` file you wish to analyze OR the predictor file generated by `mw_com_predictors`
  - ⇨ when selecting no file, the toolbox will return
  - ⇨ upon correct file selection, the toolbox will print details about the selected file

- one to several interactive prompts *may* appear: '*… enter values for [age, gender, etc.]*'
  - ⇨ you should now enter the demographic variables of your subjects in the form of a 1 × n matrix (one value per subject); e.g., for 5 subjects aged 200, 300, 400, 500, and 600 months, you would enter '200 300 400 500 600' here (hint: for larger collections it may be best to copy and paste from a spreadsheet, or use Matlab variables)
  - ⇨ note that this is only necessary upon selecting the original `mw_com_info.mat` file; if you pass a `mw_com_predictors` file, this has already been done and will be skipped here
  - ⇨ to remind you what you need to enter, the toolbox prints information about the current predictor in the command window; for example, it may say '*... enter values for age in the range of 13 - 900; this is NOT a binary variable'*
  - ⇨ if the numbers of one predictor do not match the numbers of another one, the toolbox will throw an error and bail out
  - ⇨ if a binary variable (such as gender) was used, you also need to provide a maximum of two different values; if you provide more (e.g., 0, 1, and 2) the toolbox will throw an error and bail out
  - ⇨ this procedure is repeated for each variable used during the estimation step

For most cases, that's it. The toolbox will start processing and will ultimately write the generated tissue priors to the current directory. This process will not take as long as the estimation part, but may still take hours to complete, depending on how many subjects in your population need to be matched. Again, the function will try to open a Matlabpool (or will use an existing one) as this speeds up processing significantly (to avoid this, see processing options, below), and progress reports in the Matlab window will be updated from time to time. Once it is done, it will let you know and will also inform you where the resulting data files are store. These can then be used as tissue priors to process your population.

******* Scripted use *******

Scripted use means that you call the function from the command line (or another script) with one or more input arguments, allowing you to finetune the toolbox's processing behavior or to automate data processing. This is only recommended for advanced users. Input arguments are as follows:

```
[TPM1, TPM2, TPM3, TPM4, TPM5, TPM6, T1] = mw_com_gen(mat, predicts_n, domean, smo, sanlm, mrf, out, outfile);
```

For details regarding each input and output option, please check the file where each parameter is (I hope) described in enough detail. You can get an overview by typing `help mw_com_gen` at the Matlab command prompt.

******* Batch mode *******

As this function may be used more frequently, there is a batch mode interface for it. However, there is a caveat: the exact input to the function will depend on the specific data file used. Hence, simply selecting a regression parameter dataset and hoping that the provided values will match it is asking for trouble. Hence, the (regrettably slightly circuitous) way around this is to use `mw_com_predictors` (see below) first to generate a file that contains all the information the algorithm requires. This file can then be selected and passed to the algorithm during batch mode. You can also select some more processing options (including smoothing options and where to create the output files) in the batch mode. All options are documented in the batch mode help texts. The output of the batch system (the dependency) can then be used as tissue priors for later use in the, e.g., unified segmentation or DARTEL approach or in cat12. Beware to only specify output options which actually are available with the selected file: if the regression parameters did not contain information about a T1, specifying a T1 output

will lead to an error. Similarly, do not specify DARTEL or SHOOT output without being sure that the regression parameters can actually deliver.

******* Options *******

There are a number of further processing options that can be set/changed in the file itself. They can be found in `mw_com_gen` as of about line 84 and include options regarding the generation of a mean image, the default smoothing applied to the data, the approach used for this, whether the previously-generated figures should be shown, and whether a Matlabpool should be used. There are also several experimental options and some concerning finer points of data processing. None of these should have to be changed in a usual setting. The most pertinent ones are available for modification in batch mode.

******* Outputs *******

The function will by default save your results in the form of a single 4D nifti file, containing all available tissue classes, in the present work directory (note that you can pass another output directory in scripted mode). In a typical setting, this will contain 6 tissue classes (usually with 1 = gray matter, 2 = white matter, 3 = cerebrospinal fluid, and 4:6 = different background classes). If you specified a project name in `mw_com_predictors`, this will be appended (to yield, e.g., *'mw_com_prior_myproject.nii'*); if you did not, the naming convention of the output files will reflect the dominant regression parameter (the one entered first, usually age) and the mean of the supplied values (e.g., if you provided values from 20 of your subjects with a mean age of 515 months, the file will be called *'mw_com_prior_age_0515.nii'*). , 7 = T1 images). If T1 images were included in the original regression parameters, such an image will also be generated and saved. In the above example, the name would be *'mw_com_T1_ age_0515.nii'.*

If either 12 or 14 tissue classes were included in the original regression parameters, the algorithm will assume that this represents a set of 6 or 7 iterations for GM and WM, i.e., the output of a DARTEL processing approach passed to the estimation function in the form of GM(1), WM(1), GM(2), WM(2), GM(3), WM(3), and so on. Consequently, the algorithm will generate a different output, namely *'mw_com_Template_1_age_0515.nii'* up to *'mw_com_Template_6_age_0515.nii'* (or _7_). Again, if you specified a project name, this will be used here, too. Note that this will bundle classes 1 & 2 and 3 & 4 and 4 & 5 etc. into a 2-volume nifti-volume, which is why it is important to pay attention to the file selection order and to double-check following completion. This behavior can be changed or disabled in the code. If you specified an output for the SHOOT algorithm, there will be a final tissue class in each volume, reflecting 1-(sum-of-all-other-classes), as required by the SHOOT algorithm.

All filenames (TPM and T1 [or Template_1 to 6], including the full path) will also be provided as function outputs, such that the output can then be passed to later processing steps (e.g., as priors for segmentation and/or spatial normalization).

☞ Using mw_com_predictors to generate predictor files ☜

Use mw_com_predictors **to save your demographic/other predictors to a dedicated file that can be passed to** mw_com_gen**. This is to avoid later interaction and to allow for using the SPM batch mode.**

This is a very simple function that only allows for

******* Interactive use *******

Interactive use means that you call the function (by typing its name [mw_com_predictors] in the Matlab command window) or via the toolbox dropdown integration (see above) and see what happens. This is the recommended (and indeed, only) approach for all users. Please note that this function only collects information required by the respective algorithms. If you want to play with the input options and/or modify processing settings, see above and/or the file itself. Again, the reason why this file is necessary is that the input for mw_com_gen depends on the exact regression parameter file you select. Without analyzing this file, the algorithm cannot ensure that the correct values are provided. Hence, if you want to use the batch mode, the two steps (defining your inputs and passing to the function for actual calculation) must be separated.

- a file selector window *will* appear, asking: *'Please select CerebroMatic mat file to use'*
  - ⇨ you should now select the regression parameter file (mw_com_info.mat) that you intend to use; this file will then be read and analyzed
  - ⇨ e.g., if you want to generate matching priors for your study population of 25 images, you would select a CerebroMatic file that covers the age range of your population
  - ⇨ note that, to avoid inconsistencies, the selected CerebroMatic file (including its full path) will be stored in the resulting predictor file; hence, if you move the regression parameters, an error will result when the actual calculation should take place

- successive interactive prompts *will* appear, asking: *'Predictors for X?'*
  - ⇨ you should now provide the values for each of these predictors in succession, based on your group's demographics

- ⇨ the name (e.g., '*Age'*) and range (e.g., '*13-900'*) of the predictor will be printed in the Matlab command window (and will also be spelled out instead of 'X' in the interactive window); it will also state whether this is a binary variable or not
- ⇨ in our example, this would require 25 values for age [hit '*enter*'], then 25 values for gender ['*enter*' again], and then 25 values for fieldstrength [final '*enter'*]
- ⇨ if you enter values outside of the originally-assessed range (e.g., 901 months), or enter more than two values for a binary variable [e.g., 1.5, 3, and 7 for fieldstrength), or enter inconsistent numbers for different demographic predictors (e.g., 25 for age, but only 24 for fieldstrength), an error will result which ideally will be caught gracefully
- ⇨ note that, if image weight is detected as a predictor, it will automatically be set to the best-available value

- an interactive prompt *will* appear, asking: *'Please enter output identifier name'*
  - ⇨ you should now enter an identifier for the project you just specified
  - ⇨ while *'myproject'* is prespecified, you can obviously enter whatever name you like, for example the project title; this will then be appended to the output file name, resulting in (e.g.) `mw_com_predictors_myproject.mat`
  - ⇨ this project title will also be appended to the generated priors

- a file selector window will come up, asking: *'Please select output directory'*
  - ⇨ you should now select the output directory
  - ⇨ this defaults to the present work directory, but again, you are free to select whichever directory you like best (and have write permissions to)
  - ⇨ if you later pass this file to `mw_com_gen`, the resulting files will also be saved in this directory

The function will then save the file with the specified name in the specified directory, informing you to this effect if all went well (and throwing an error message if it did not). That's it. You can now pass this file to `mw_com_gen`, as described above, in scripted, interactive or batch mode.