**SUPPLEMENTARY MATERIALS**

Below is a list of all functions present in the toolbox along with a brief description. In addition to the toolbox functions, the GitHub repository also contains brief examples on how to use the toolbox, more specifically a Python notebook which shows how the manuscript figures were generated.

1. Artifact Functions
    a. **add_time_domain_noise()**: adds complex normally distributed noise with standard deviation specified by the user to time domain free induction decays.
    b. **add_spur_echo_artifact()**: adds a spurious echo artifact to time domain data with function specific user parameters amp (amplitude), cs (chemical shift), phase, tstart (start time of artifact in ms), tfinish (end time of artifact in ms).
    c. **add_eddy_current_artifact()**: adds an eddy current artifact to time domain data with function specific user parameters amp (amplitude), tc (time constant in ms).
    d. **add_linebroad()**: line broadens spectral data by providing time domain data with function specific user parameters amp (amplitude), damp (degree of line broadening).
    e. **add_nuisance_peak()**: adds a peak to time domain data by providing a peak profile which specifies the peak type (shape), amp (amplitude), width (FWHM width in ppm), res_freqs (resonant frequency), edited (boolean indicating an edited sequence has been passed).
    f. **add_baseline()**: adds a baseline contaminant to spectral data by providing time domain data with a baseline profile which specifies the base type (shape), num_bases (number of sub-bases to combine), amp_bases (amplitude of each sub-base), comp_bases (cycle frequency of each sub-base), slope_bases (slope of each sub-base), spline_fitted (boolean indicating whether to fit a spline to the combined sub-bases).
    g. **add_freq_drift_linear()**: adds a linear frequency drift over multiple transients by providing time domain data with function specific user parameters freq_offset_var (variance at each transient from the linear drift), freq_shift (frequency shift over the entire drift), start_trans (first transient in sequence to

experience drift), num_trans (number of transients over which the frequency shift is experienced).

h. **add_freq_shift()**: adds a frequency shift to individual transients by providing time domain data with function specific user parameters freq_var (+/- frequency shift from which samples are uniformly sampled from).

i. **add_zero_order_phase_shift()**: adds a zero order phase shift to individual transients by providing time domain data with function specific user parameters phase_var (+/- phase shift from which samples are uniformly sampled from).

j. **add_first_order_phase_shift()**: adds a first order phase shift to individual transients by providing time domain data with function specific user parameters phase_var (+/- phase shift from which samples are uniformly sampled from).

2. Applied Functions

a. **add_progressive_motion_artifact()**: uses *add_linear_freq_shift()* to simulate the motion of head drift during a scan. Time domain data should be provided but no function specific user parameters are required.

b. **add_subtle_motion_artifact()**: uses *add_freq_shift()* and *add_zero_order_phase_shift()* to simulate small motion or small stochastic scanner effects throughout the scan. Time domain data should be provided but no function specific user parameters are required.

c. **add_disruptive_motion_artifact()**: uses *add_linebroad()* and *add_baseline()* to simulate single large motion such as a cough. Time domain data should be provided but no function specific user parameters are required.

d. **add_lipid_artifact()**: uses *add_nuisance_peak()* to simulate a lipid peak contaminant at 1.5 ppm. Time domain data should be provided but no function specific user parameters are required.

3. IO Functions

a. **get_FIDA_mat_data():** loads time domain data (FIDs) from an existing MATLAB .mat FID-A struct.

b. **return_FIDA_mat_data()**: returns time domain data (FIDs) to an existing MATLAB .mat FID-A struct.

c. **load_nifti_mrs_data()**: loads time domain data (FIDs) from an existing nifti-MRS file.
d. **return_nifti_mrs_data()**: returns time domain data (FIDs) to an existing nifti-MRS file.

4. Support Functions
    a. **to_fids()**: converts from the frequency domain (spectrum) to the time domain (free induction decay) via the inverse Fourier transform.
    b. **to_specs()**: converts from the time domain (free induction decay) to the frequency domain (spectrum) via the Fourier transform.
    c. **interleave()**: combines edit-ON and edit-OFF transient matrices into a single matrix where subspectra have been interleaved (1-0-1-0...).
    d. **undo_interleave()**: takes an interleaved matrix with pattern seen in *interleave()* and returns distinct edit-ON and edit-OFF matrices.
    e. **scale()**: scales data by the maximum absolute value of each spectrum and returns the scaled data and associated scale factor required to achieve this scaling. Recommended to apply in the spectral domain.
    f. **undo_scale()**: provided with the scaled data and scale factor from *scale()*, the function reverses the operation and returns the data in its original scale.