Canonical Response Parameterization

Description of data and code resources

Kai J. Miller, *Mayo Clinic Neurosurgery*
miller.kai@mayo.edu

Summary

This document describes the code written in MATLAB and the sample data to reproduce all of the steps and illustrations contained in the manuscript "*Canonical Response Parameterization: Quantifying the structure of responses to single-pulse intracranial electrical brain stimulation*", by Kai J. Miller, Klaus-Robert Müller, Gabriela Ojeda Valencia, Harvey Huang, Nicholas M. Gregg, Gregory A. Worrell, and Dora Hermes. All accompanying files are available at **https://osf.io/tx3yq** and **https://github.com/kaijmiller/crp_scripts**. Code are freely usable under the GNU public license.

Sample datafiles in folder "**sampledata**"

The folder "sampledata" contains datafiles to reproduce each illustration in the manuscript. These datafiles are enumerated in **Table 1**. Each file is a MATLAB data file with the illustrative example data. These data were recorded from stereoEEG (sEEG) electrodes placed during monitoring to localize seizure onset zone for patients with epilepsy. Each patient or their adult representative was consented research conducted under Mayo Clinic IRB 15-006530, which also authorizes sharing of the data. Each file contains the following variables:

data: stimulation-evoked voltage matrix. The dimensions of **data** are $T \times K$, with T total timepoints and K total stimulation events (trials).

t: vector of timepoints corresponding to indices in **data**, with respect to stimulation time. The dimensions of **t** are $1 \times T$.

MATLAB function "**CRP_illustration.m**": Methodological illustration script

This MATLAB script performs a series of steps to recreate the illustrations of the manuscript. Calling the script from the command line will execute all steps and reproduce the key illustrations. It would be beneficial to the read through the comments at each step of the cell-based sections within the .m-file to understand each step (can execute each cell by using ctrl-enter (PC) or cmd-enter (Mac)). These reference back to the manuscript extensively. The most important purpose of this illustration is to show how the function "CRP_method.m" is called and how to package data to send to it. Users may either use the sample data by commenting the appropriate filename in (step 2) or send their own **.mat** file (in the function call), using a string corresponding to the path & filename.

This function performs the following steps, with each indicated in the code with cell-based sections each headed by "%%":

1. **Set options for illustration** - This has manually entered options, including starting and ending times for analysis (t_1 and t_2), whether to save generated figures, and whether to remove artifactual outliers.
2. **Load sample data** if user hasn't sent their own datafile into the function call. This is done by manually commenting a line with the desired filename in.
3. **Convert data** from original format into format required for CRP function.
4. **Identify and remove potentially artifactual trials** if this option has been selected.
5. **Call CRP function**.
6. **Calculate an estimate of uncertainty**. In the example way used for the manuscript, this estimate of uncertainty in τ_R is chosen for when projection magnitude exceeds 98% of maximum.
7. **Plot traces & time-resolved projection magnitudes**.
8. **Plot projection magnitudes of full epoch input and at response duration (**τ_R**)**.
9. **Plot single-trial traces, parameterization, and residuals**.
10. **Display parameters** to MATLAB command window.

Sub-functions contained in this function (at end of script) are:

trial_tests - This function identifies outliers in individual trials.

kjm_printfig - This function prints figures in .eps and .png formats.

MATLAB function "CRP_method.m": Canonical Response Parameterization code

This function performs canonical response parameterization. The file is extensively commented - please read through, using the manuscript to understand each step. This code uses the "code section approach" to delineate each analysis step (i.e. sections started with "%%"), and it allows one to evaluate each section (can execute each cell by using ctrl-enter (PC) or cmd-enter (Mac)) in sequence to understand each step.

CRP_method takes an input matrix of individual trials and projects them into one another using a semi-normalized dot product. The length of timepoints from stimulation included in the dot product is varied to obtain a temporal profile, and the peak of the profile uniquely identifies the duration of the response. Using linear kernel PCA, a canonical response shape is obtained over this duration, and then single-trial traces are parameterized as a projection of this canonical shape with a residual term. This parameterization also quantifies response duration, projection magnitudes and amplitudes, signal-to-noise ratios, explained variance, and statistical significance. Artifactual trials can be automatically identified by outliers in sub-distributions of projection magnitude, and rejected prior to parameterization.

The function is called as: **[crp_parms, crp_projs]=CRP_method(V,t_win);**

The **input variables** are:

> **V**: stimulation-evoked voltage matrix $V_k(t)$, where t denotes the time from the kth electrical stimulation, τ_k: $(\tau_k + t_1) \leq t \leq (\tau_k + t_2)$. The dimensions of **V** are $T \times K$, with K total stimulation events (trials) by T total timepoints (over the interval $t_1 \leq t \leq t_2$).
>
> **t_win**: timepoints, with dimensions $1 \times T$ (over the interval $t_1 \leq t \leq t_2$).

The **output variables** are:

> **crp_projs**: a structure that contains the results of projection, including response duration profiles. It contains the following fields:
>> **.proj_tpts**: time points that projections were calculated at
>> **.S_all**: full set of projection magnitudes at all durations
>> **.mean_proj_profile**: mean projection magnitude profile
>> **.var_proj_profile**: variance in projection magnitude profile
>> **.tR_index**: index into proj_tpts (projection magnitude times) corresponding to response duration
>> **.t_value**: t-statistic for projection magnitudes at response duration (τ_R)
>> **.p_value**: p-value at response duration, τ_R (extraction significance) by t-test

> **crp_parms**: a structure that contains the full canonical response parameterization of the input matrix. It contains the following fields:
>> **.V_tR**: Reduced length voltage matrix (to response duration)
>> **.al**: alpha coefficient weights for C into V
>> **.C**: canonical shape, C(t)
>> **.ep**: residual epsilon after removal of form of CCEP
>> **.tR**: response duration in units seconds
>> **.parms_times**: times for parameterized data
>> **.avg_trace_tR**: simple average trace, truncated to response duration
>> **.al_p**: alpha-prime - alpha scaled to sqrt(number of samples),
>> **.epep_root**: scalar summarizing residual for each trial
>> **.Vsnr**: "signal to noise" for each trial
>> **.expl_var**: explained variance by parameterization for each trial

Sub-functions contained in this function (at end of script) are:
  ccep_proj - This function calculates the semi-normalized projections.
  kt_pca - This function performs linear kernel PCA.

Filename	Figure	Notes on file
Fig1_sampledata.mat	1	Example of significant data to show **V** matrix
Fig2_sampledata.mat	2	Large response with long τ_R for projection illustration
Fig3_sampledata.mat	3	Short τ_R to illustrate how duration is calculated
Fig4_sampledata.mat	4	Polyphasic illustrative data for parameterization
Fig5_sampledata_1.mat	5	Very large significant response
Fig5_sampledata_2.mat	5	Simple example of significant response
Fig5_sampledata_3.mat	5	Simple example of significant response
Fig5_sampledata_4.mat	5	Interesting polyphasic response, possibly artifactual
Fig5_sampledata_5.mat	5	Example of no significant response
Fig6_a_significant_example.mat	6A	Simple example of significant response
Fig6_b_insignificant_example.mat	6B	Simple example of no response
Fig6_c_earlycomponent.mat	6C	Dwindle in stimulation artifact giving an early significant component
Fig6_e_20muV_offset.mat	6E	False significance resulting from 20baselining error
Fig6_f_offset_corrected.mat	6F	Data from Fig. 6e, but baseline corrected - significance is lost
Fig7_artificial_box.mat	7A	Synthetic data - 100ms, 100μV response
Fig7_artificial_box_short.mat	7B	Synthetic data - 50ms, 100μV response
Fig7_artificial_box_split.mat	7C	Synthetic data - two separated 50ms, 100μV responses
Fig7_artificial_noise_0p5.mat	7D	Synthetic data - 100ms, 100μV response, low noise added
Fig7_artificial_noise_2p5.mat	7E	Synthetic data - 100ms, 100μV response, intermediate noise added
Fig7_artificial_noise_7p5.mat	7F	Synthetic data - 100ms, 100μV response, high noise added
Fig7_artificial_ramp.mat	7G	Synthetic data - linearly ascending ramp response to 100μV over 100ms
Fig7_artificial_ramp_descend.mat	7H	Synthetic data - linearly descending ramp from 100μV to 0μV over 100ms
Fig7_artificial_ramp_dual.mat	7I	Synthetic data - linear ascent+descent to 100μV peak over 100ms
Fig7_artificial_sine.mat	7J	Synthetic data - complete sine wave over 100ms, from $+100\mu$V to -100μV
Fig7_artificial_sine_inverted.mat	7K	Synthetic data - inverted sine wave over 100ms, from $+100\mu$V to -100μV
Fig7_artificial_sine_abs.mat	7L	Synthetic data - absolute value of a sine wave over 100ms, max $+100\mu$V
Fig8_sampledata.mat	8	Example of noisy but significant response
Fig9_eeg_data_zeropad.mat	9	Example of scalp EEG data in response to intracranial sEEG stimulation

Table 1: Sample data files to recreate the images from the manuscript

Exported figures in folder "figures"

If the option to save generated figures is chosen in the "CRP_illustration.m" function, then rendered figures are exported into the folder "**figures**". Running "CRP_illustration.m" for data in file "[dataname]" saves the following image files in both **.eps** and **.png** formats:

- **[dataname]_projmags_tR_and_input**, is an image of sorted projection magnitudes at response duration τ_R and the full interval of input sent.
- **[dataname]_cross_projections_by_trial**, is an image of cross-projections generated by the step that identifies potentially artifactual trials, highlighting projections of each trial.
- **[dataname]_traces_projections**, is an image of individual and average trials, with the epoch to τ_R highlighted. It also shows projection magnitudes as a function of time, with τ_R circled in red.
- **[dataname]_single_trial_parmsfig**, is an image of the parameterization of each trial k, showing the initial trace ($V_k(t)$, in black), the scaled canonical response shape ($\alpha_k C(t)$, in red), and the residual ($\varepsilon_k(t)$, in green).