Parametric Analysis of Temporal Structure (PATS)

This project includes a MATLAB implementation of PATS, and the scripts for simulations, analyses, and plotting Figures reported in the article titled “Extracting evidence for neural temporal structure from behavioral measurements.” To ensure the reproducibility of the simulation results, some scripts are provided with a random seed used for simulations reported in the article. You will need MATLAB R2016b or later with Optimization Toolbox to run the scripts.

For a quick tutorial on how to use PATS with your own data, please refer to the tutorial scripts.

Few scripts rely on bayesFactor Toolbox (https://klabhub.github.io/bayesFactor/, see the “lib/bayesFactor-2.2.0/LICENSE” file for license information) which requires Statistics and Machine Learning Toolbox.

# Directories and Files

data-fitted/ directory – stores simulation results and analyzed data.

data-step5/ directory – stores open data published by Ruzzoli et al. (2019). A zip file containing the data will be downloaded and unpacked to this directory (Step 5A).

fig-images/ directory – stores image files for Figures 2 through 6.

lib/ directory – contains simple functions used to implement PATS and simulations. Please refer to the help text in the beginning of each function file.

lib/bayesFactor-2.2.0/ directory – contains bayesFactor Toolbox version 2.2.0.

fitRhythms() function – implements PATS. Please refer to the help text in the beginning of the file.

fftRhythms() function –implements the spectral analysis employed in Cha & Blake (2019). Please refer to the help text in the beginning of the file.

simRandomWalk() function – generates reaction times using a modified random walk model simulation. Please refer to the help text in the beginning of the file.

sampleFromCDF() function – samples reaction times from a given CDF using the inverse transform sampling method. Please refer to the help text in the beginning of the file.

batchFitRhythms() and batchFFTRhythms() functions – run a batch of PATS and spectral analysis, respectively. The functions are used in Steps 3A and 4B.

CommonVars\_CDF.m and CommonVars\_Figure.m scripts – define common variables (initial guesses for CDF parameters and rhythmic transformation parameters, font size and line thickness for Figures, etc.).

# Tutorial Scripts

Tutorial\_FitRhythms\_LognormCDF.m script – shows a simple example of how to analyze a single set of duration data with PATS.

Tutorial\_FitRhythms\_GammaCDF.m script – shows an example of how to analyze a single set of duration data with PATS using a gamma CDF (or any other CDF of one’s choice).

# Scripts Whose Name Starts with “Step”

Step1 – plots example rCDFs varying in rhythmic transformation parameters (Figure 2).

Step2 – runs simulations for Example 1 reported in the article. Step 2A generates response time data using a modified random walk model simulation, analyzes the data with PATS. Step 2B generates another set of response time data, analyzes the data with spectral analysis. Step 2C plots PATS and spectral analysis results (Figure 3).

Step3 – runs simulations for the first part of Example 2. Step 3A generates lots of data sets using a modified random walk model simulation, analyze all data sets with both PATS and spectral analysis. Step 3B plots PATS and spectral analysis results against the ground-truth values (Figure 4) Step 3C performs statistical tests for correlations and partial correlations shown in Figure 4.

Step4 – runs simulations for the second part of Example 2. Step 4A builds a data pool by sampling from rCDFs. Step 4B repeats four runs of PATS and spectral analyses using varying number of data points sampled from the pool. Step 4C plots PATS and spectral analysis results for varying numbers of data points. Step 4D performs statistical tests for Figure 5.

Step5 – runs simulations for Example 3. Step 5A fetches open data from an OSF repository. Step 5B analyzes the data with PATS. Step 5C analyzes the same data with spectral analysis. Step 5D plots PATS and spectral analysis results (Figure 6). Step 5E performs statistical tests for Figure 6.

Step6 – calculates the goodness of fit measures (adjusted *R*2) of lognormal CDF fits estimated in Examples 1 and 3 (reported in Footnote 1).